Kernel Density Estimation Techniques for Monte Carlo Reactor Analysis

by

Timothy P. Burke

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Nuclear Engineering and Radiological Sciences and Scientific Computing)
in the University of Michigan
2016

Doctoral Committee:

Professor William R. Martin, Co-Chair
Assistant Professor Brian C. Kiedrowski, Co-Chair
Professor Thomas J. Downar
Professor Robert M. Ziff

ACKNOWLEDGEMENTS

First and foremost, I would like to thank the people that have made this work possible:

- My co-advisors, Professors Brian Kiedrowski and Bill Martin, for giving me the freedom to pursue my research as I saw fit and for providing invaluable guidance along the path towards graduation. Brian's ever-present encouragement and insightful research discussions made graduating that much easier, and for that I am forever grateful.
- My committee, for their suggestions and comments on the potential applications of my research.
- Los Alamos National Lab and the people there (Forrest Brown) and formerly there (Brian Kiedrowski) for hosting me for four summers and providing me with the computational resources that enabled me to create an entire chapter in this dissertation.

I would also like to thank my friends, for making this work and my time in Ann Arbor and Los Alamos enjoyable, especially:

- Molly McCulloch, for always believing in me, supporting me, and encouraging me.
- Matt Marcath, for honest feedback, long walks, and car advice.
- Brad Stark, for bicycles, brews, and board games.
- Sean O'Neal, Ed Harvey, and David "Bearclaw" Collins, for living with me in whole or in part of the four years of my Ph.D.
- My friends in the desert, Simon Bolding and Rian Bahran, for basketball lessons.

Additionally, I would like to thank those who have aided me in various steps of my research:

- Mitch Young, for maintaining the basement cluster, and shop talk.
- Kaushik Banerjee, for having a well-written thesis and for discussions on the implementation of KDEs during the early stages of this work.
- Professor Martin's former students, for convincing me that being a graduate student wasn't so bad.

I would also like to acknowledge my financial support:

- This material is based upon work supported in part by the National Science Foundation Graduate Research Fellowship under Grant No. DGE 1256260 and by the US DOE/NNSA Advanced Scientific Computing program.

And last but not least, my family:

- My parents, for shaping who I am.
- My brother and sister, for being some of the best people I know.

TABLE OF CONTENTS

Acknowledgments . . . . . **ii**

List of Figures . . . . . **v**

List of Tables . . . . . **ix**

Chapter

1 Introduction **1**

1.1 Background 1
1.2 The Monte Carlo Method 4
1.3 Kernel Density Estimators 7
1.3.1 Estimator Definition 7
1.3.2 Optimal Bandwidth 11
1.3.3 Choice of Kernel 14
1.3.4 Boundary Treatment 15
1.3.5 Considerations for Eigenvalue Problems 17
1.3.6 Remaining Challenges in KDEs Applied to Reactor Analysis . . . 19
1.4 High Performance Computing 20
1.5 Multi-physics Coupling 21
1.6 Thesis Outline 22

2 Kernel Density Estimators for Capturing Reaction Rates in Reactor Physics Problems **24**

2.1 Estimating Reaction Rates With KDEs 24
2.1.1 Verification in 1-D Reactor Physics Problems 25
2.2 1-D MFP KDE 34
2.2.1 1-D MFP KDE Derivation 34
2.2.2 Analytic Solution in 1-D Slab Geometry 35
2.2.3 Verification in 1-D Reactor Physics Problems 37
2.3 2-D MFP KDE with Univariate Kernels 40
2.3.1 Derivation of 2-D MFP KDE with Univariate Kernels 40
2.3.2 Verification in One-Group Reactor Physics Problems 42
2.4 Multivariate MFP KDE 44
2.4.1 Derivation of Multivariate MFP KDE 44
2.4.2 Verification of Multivariate MFP KDE 47
2.5 Minimum Bandwidth and Fractional MFP KDE 53

2.6 Approximate MFP KDE . . . 56
2.6.1 1-D Slab With Thin, Strong Absorber . . . 58
2.7 Volume-Average KDE . . . 61
2.7.1 Volume-Average Approximate MFP KDE KDE Results . . . 65
2.8 Cylindrical KDE . . . 70
2.8.1 Motivation . . . 70
2.8.2 Derivation of Cylindrical KDE . . . 70
2.8.3 Bias and Variance of Cylindrical KDE . . . 72
2.9 Cylindrical MFP KDE . . . 74
2.9.1 Derivation . . . 74
2.9.2 Verification of Cylindrical MFP KDE . . . 78

3 GPU Acceleration of KDEs and Algorithmic Considerations . . . 83

3.1 Motivation for GPUs . . . 83
3.2 CPU Algorithm . . . 84
3.2.1 Nearby Neighbor List . . . 85
3.2.2 Maximum Bandwidth . . . 86
3.2.3 Local Cross Section Lookup . . . 86
3.2.4 Minimum Bandwidth . . . 87
3.2.5 Cylindrical MFP KDE Ray Tracing . . . 88
3.3 GPU Architecture and Bottlenecks . . . 88
3.4 GPU Algorithm . . . 90
3.5 GPU Optimization . . . 92
3.6 GPU Acceleration and Performance Comparisons . . . 94
3.6.1 Overview . . . 94
3.6.2 2-D Boxcell . . . 95
3.6.3 2-D Pincell . . . 98
3.6.4 Assembly of Pincells . . . 102

4 Applications of the Cylindrical MFP KDE to Reactor Physics Problems . . . 108

4.1 Overview . . . 108
4.2 IFBA Pincell . . . 109
4.3 Quarter Assembly . . . 112
4.4 Depleted Fuel Pin . . . 117
4.5 Unstructured Mesh Pincell . . . 120

5 Summary & Future Work . . . 123

5.1 Mean Free Path KDEs For Estimating Reaction Rates . . . 123
5.2 GPU Acceleration of KDEs and Algorithmic Considerations . . . 127
5.3 Applications of the Cylindrical MFP KDE to Reactor Physics Problems . 129
5.4 Future Work . . . 130

Bibliography . . . 134

LIST OF FIGURES

2.1 Graphic depicting analog (right) and non-analog (left) view of fission reaction rate estimation using KDEs. $X_{i,c}$ indicates a fission or collision site with the dotted line representing the kernel associated with that event. 25
2.2 Graphic depicting geometry of 1-D alternating slab problem with material labels 1, 2, and 3 corresponding to moderator, fuel, and absorber, respectively. . 26
2.3 Comparison of flux (left) and fission reaction rate densities (right) obtained from a reference histogram and KDE tally for the one-group, 1-D array of pincells problem. 27
2.4 Comparison of flux (left) and fission reaction rate densities (right) obtained from histogram and KDE tallies for the continuous energy problem. Estimated distributions are shown on top with C/E (KDE/histogram) values shown on the bottom. 28
2.5 Comparison of flux (left) and fission reaction rate densities (right) obtained from histogram and KDE tallies for the continuous energy problem. Estimated distributions are shown on top with C/E (KDE/histogram) values shown on the bottom. 28
2.6 Comparison of thermal flux (top) and thermal absorption reaction rate densities (bottom) obtained from histogram and KDE tallies for the 1-D alternating slab problem in continuous energy . Estimated distributions are shown on top of each figure with C/E (KDE/histogram) values shown on the bottom. 30
2.7 Comparison of epithermal flux, fission, and absorption distributions obtained from histogram and KDE tallies for the 1-D alternating slab problem in continuous energy. 31
2.8 Comparison of fast flux, fission, and absorption distributions obtained from histogram and KDE tallies for the 1-D alternating slab problem in continuous energy. 32
2.9 Microscopic total cross section for U-238 and microscopic fission cross section for U-235 obtained from data library ENDF/B-VII.0 using JANIS. 33
2.10 Graphic depicting geometry of analytic 1-D slab problem with material labels 1, 2, and 3 corresponding to materials with $\Sigma_{t,1} = 1$, $\Sigma_{t,2} = 100$, $\Sigma_{t,3} = 0.5$, respectively. 36
2.11 Comparison of KDE methods and histogram to analytical solution for a 1-D, 3-material problem with a thin, strong absorber. 36
2.12 Flux distribution comparisons between distance-based KDE (left) and MFP KDE (right) and reference results for a 1-D array of alternating slabs with a central absorber. 38

2.13 Absorption density comparisons between distance-based KDE (left) and MFP KDE (right) and reference results for a 1-D array of alternating slabs with a central absorber. 38
2.14 Comparison of MFP KDE and reference histogram absorption reaction rate densities with two different resolutions in the right edge of a fuel slab for the alternating slabs problem. Markers are left off of the KDE tallies for clarity. . . 39
2.15 Comparison between the 2-D KDEs and reference results for a one-group 1-D array of pincells with a central absorber. 43
2.16 Comparison between the 2-D collision MFP KDE using univariate kernels and reference histogram flux results for the one-group boxcell problem. 44
2.17 Flux distribution obtained from multivariate MFP KDE in Eq. (2.31) (left) and C/E comparison to reference histogram (right) for the one-group pincell problem. 47
2.18 C/E comparison of flux distribution obtained from multivariate MFP KDE with optimal bandwidths computed using the number of samples in the previous batch to reference histogram for the one-group pincell problem. 48
2.19 Ratio of the flux distribution obtained using the multivariate MFP KDE in Eq. (2.31) to the flux distribtuion obtained using the 2-D MFP KDE using product of univariate kernels in Eq. (2.20). 49
2.20 Fission and absorption distributions in a 2-D pincell problem in continuous energy obtained from the multivariate MFP KDE. 51
2.21 Comparison of 2-D MFP KDE and reference histogram absorption distributions for the 1-D slab representation of a fuel pin with varying amounts of total active histories. 52
2.22 Comparison of absorption reaction rate densities from the 2-D MFP KDEs with minimum bandwidths of 0.004 cm and 0.001 cm and reference histogram distributions for the 1-D slab representation of a fuel pin. A close-up of the distributions at the fuel-water interface on the right edge of the fuel is shown on the right. 54
2.23 Comparison of absorption reaction rate densities from the 2-D fractional MFP KDE and reference histogram distributions for the 1-D slab representation of a fuel pin. A close-up of the distributions at the fuel-water interface on the right edge of the fuel is shown on the right. 55
2.24 Relative uncertainties in the estimates of the absorption reaction rate densities from the 2-D fractional MFP KDE and the MFP KDE using a minimum bandwidth of $h_{\min} = 0.004$ cm for the 1-D slab representation of a fuel pin. . . 56
2.25 Graphic depicting problem with a thin, strong absorber. Regions labeled 1, 2, and 3 contain water, fuel, and strong absorber, respectively. The slab widths are shown underneath the geometry. 58
2.26 Flux comparison between aMFP KDE, MFP KDE, and reference histogram near the strong absorber of the thin, strong absorber problem. 59
2.27 Comparison between track-length MFP KDE, track-length aMFP KDE, and histogram reference solution for the thin, strong absorber problem near the material interface. 59

2.28 Flux comparison between aMFP collision and track-length KDE using the kernel described in Eq. (2.37) and the reference histogram for the thin, strong absorber problem. 60
2.29 Graphic depicting how integration components are defined for a 1-D mesh with bin width Δx for the case where the kernel support is smaller than the bin width (left) and where the kernel support is greater than the bin width (right). 62
2.30 Graphic depicting the regions defining the 9 integral components of the volume-average track-length KDE for a single mesh element. An example particle track is shown crossing 5 regions. 64
2.31 Fission and absorption distribution comparisons between the volume-average track-length aMFP KDE and track-length histogram across the fuel in the 1-D pincell problem. 65
2.32 Fission and absorption distribution comparisons between the volume-average track-length aMFP KDE and track-length histogram with a fine mesh at the right edge of the fuel slab in the 1-D pincell problem. The difference in scale between the fission and absorption distributions (top) should be noted. 66
2.33 Fission reaction rate distribution for collision estimators for the 160×160 mesh. 68
2.34 Absorption reaction rate distribution for collision estimators for the 160x160 mesh. 69
2.35 Depiction of a cylindrical kernel centered at tally point $\mathbf{r}$ on a quarter-view of a pincell with a collision site at $\mathbf{r}'$. The distance between the tally point and collision site in cylindrical coordinates is marked by Δr and $\Delta\theta$. Material regions marked 1, 2, and 3 indicate fuel, cladding, and water, respectively. . . . 71
2.36 Comparison between distance-based cylindrical KDE and reference histogram solution for the absorption distribution along the 22.5° azimuth in the rim region of the fuel. 79
2.37 Flux distribution comparison between the cylindrical MFP KDE and histogram for the pincell problem along the 22.5° azimuth. 80
2.38 Fission distribution comparison between the cylindrical MFP KDE and histogram for the pincell problem along the 22.5° azimuth. 80
2.39 Absorption distribution comparison between the cylindrical MFP KDE and histogram for the pincell problem along the 22.5° azimuth. 81

3.1 Fission reaction rate densities in the 2-D boxcell problem with 60×60 bins and tally points obtained using the fractional aMFP KDE (left) and the collision histogram (right). 96
3.2 Ratio of relative uncertainties in the fractional aMFP KDE to the collision histogram for the absorption distribution in the 2-D boxcell problem with 60×60 bins and tally points. 97
3.3 Relative uncertainties in the flux and absorption distributions along the 22.5° azimuth for the five estimators detailed in Table 3.3 for the 2-D pincell problem. 99
3.4 Relative uncertainty in the absorption distribution along the 22.5° azimuth at the fuel-cladding interface for the five estimators detailed in Table 3.3 for the 2-D pincell problem. 99

3.5 Relative uncertainties in the absorption and fission distributions along the 22.5° azimuth for the cylindrical aMFP KDE with and without the boundary kernel method at material interfaces for the 2-D pincell problem. 102
3.6 Depiction of the assembly of pincells. 103
3.7 Flux distribution from the fractional aMFP KDE and C/E comparison to collision histogram for the assembly of pincells. 103
3.8 Fission distribution estimated from fractional aMFP KDE for the assembly of pincells. 104

4.1 Flux and fission distribution comparisons between the cylindrical MFP KDE and histogram for the pincell problem along the 22.5° azimuth. 110
4.2 Absorption distribution comparison between the cylindrical MFP KDE and histogram for the IFBA pincell problem along the 22.5° azimuth. 110
4.3 Comparison of fission distribution in rim region (left) and absorption distribution in IFBA (right) between the cylindrical aMFP KDE and the reference histogram. 111
4.4 Depiction of the assembly of pincells. 113
4.5 Comparison of flux distributions estimated using the cylindrical MFP KDE and a collision histogram on a cylindrical mesh across pincell 1 and within the rim region in pincell 1 along the $337.5°$ azimuth. 114
4.6 Comparison of absorption reaction rate densities in pincell 1 estimated using the cylindrical MFP KDE and a collision histogram along the $337.5°$ azimuth. . 114
4.7 Cylindrical MFP KDE flux distributions in pincells 2 and 3. 115
4.8 Cylindrical MFP KDE fission distributions in pincells 2 and 3. 115
4.9 Comparison of fission reaction rate densities in pincell 2 estimated using the cylindrical MFP KDE and a collision histogram along the $22.5°$ azimuth. . . . 116
4.10 Comparison of absorption reaction rate densities in pincell 2 estimated using the cylindrical MFP KDE and a collision histogram along the $22.5°$ azimuth. . 116
4.11 Comparison of fission reaction rate densities in the fully depleted pincell estimated using the cylindrical MFP KDE and a collision histogram along the $22.5°$ azimuth. 118
4.12 Comparison of absorption reaction rate densities in the fully depleted pincell estimated using the cylindrical MFP KDE and a collision histogram along the $22.5°$ azimuth. 119
4.13 Unstructured mesh of the 2-D pincell with cladding. 120
4.14 Flux obtained on the unstructured mesh of the 2-D pincell with cladding using the cylindrical MFP KDE. 122
4.15 Fission and absorption reaction rate densities within the fuel obtained on the unstructured mesh of the 2-D pincell with cladding using the cylindrical MFP KDE. 122

LIST OF TABLES

2.1 One-group cross sections for first set of test problems. 26
2.2 Cross sections for one-group problems. 42
2.3 Flux FOM for the volume-average collision and track-length aMFP KDE and histogram for a range of mesh sizes. 67
2.4 Fission rate FOM for the volume-average collision and track-length aMFP KDE and histogram for a range of mesh sizes. 68
2.5 Absorption rate FOM for the volume-average collision and track-length aMFP KDE and histogram for a range of mesh sizes. 68

3.1 KDE GPU kernel compute times for 2-D boxcell problem with 120×120 tally points. 93
3.2 Runtimes and FOM for the fractional aMFP KDE, collision histogram and track-length histogram on the the 2-D Boxcell problem with 60×60 and 120×120 bins and tally points. 96
3.3 Active tally CPU runtimes and FOM ratios relative to the histogram for 2-D pincell benchmark problem. 98
3.4 History calculation rates and speedups from the single-precision and double-precision GPU tallies compared to the CPU-only tallies for the 2-D pincell assembly problem with 120×120 tally points for 1 to 16 MPI processes. 105
3.5 History calculation rates and speedups from the single-precision and double-precision GPU tallies compared to the CPU-only tallies for the 2-D pincell assembly problem with 240×240 tally points for 1 to 16 MPI processes. 106

CHAPTER 1

Introduction

1.1 Background

The study of computational radiation transport is a diverse field with many applications including nuclear reactor design, radiation shielding and detection, radiography, and medical physics, among others. In nuclear reactor design, computational radiation transport is used to calculate important quantities such as heat generation rates inside the reactor to determine temperatures in the reactor, absorption reaction rates for determining burn-up in nuclear reactor fuel, energy deposition in materials to determine radiation effects on structural components, and radiation levels in the shielding and beyond for calculating dose outside of the reactor. The calculation of these quantities is based upon the transport of radiation in space, which can be determined using the linear Boltzman transport equation [1].

The transport of radiation, viewed as discrete particles in the Boltzman transport equation, through a medium is a stochastic process. The life of a particle begins with it being born at a source, either an external source like a prescribed beam or an internal source like fission. During the lifetime of a particle it undergoes collisions with the problem medium or it escapes by leaking out of the external boundary of the problem. When the particle collides it can either scatter, changing energy and direction, or it can be absorbed. While the behavior of any individual particle in a nuclear system is random, the average behavior of the radiation in the system is not. For neutrons, this average particle behavior is modeled using the neutron transport equation: the neutron counterpart of the linear Boltzman equation [1]. The neutron transport equation is defined as

$$\frac{1}{v}\frac{\partial\psi(\mathbf{x},\mathbf{\Omega},E,t)}{\partial t}+\mathbf{\Omega}\cdot\nabla\psi(\mathbf{x},\mathbf{\Omega},E,t)+\Sigma_t(\mathbf{x},E)\psi(\mathbf{x},\mathbf{\Omega},E,t)= \int_0^\infty\int_{4\pi}\Sigma_s(\mathbf{x},E'\rightarrow E,\Omega'\rightarrow\Omega)\psi(\mathbf{x},\mathbf{\Omega'},E',t)\mathrm{d}E'\mathrm{d}\Omega'+Q(\mathbf{x},\mathbf{\Omega},E,t), \tag{1.1}$$

where v is the neutrons' speed, Σ_t is the macroscopic total cross section, Σ_s is the macroscopic scattering cross section, Q is the neutron source, and ψ is the angular flux defined as

$$\psi(\mathbf{x}, \mathbf{\Omega}, E, t) = vN(\mathbf{x}, \mathbf{\Omega}, E, t), \tag{1.2}$$

where $N(\mathbf{x}, \mathbf{\Omega}, E, t)$ is the neutron density at location $\mathbf{x}$ with direction $\mathbf{\Omega}$ and energy E at time t. The scalar flux is defined as

$$\phi(\mathbf{x}, E, t) = \int_{4\pi} \psi(\mathbf{x}, \mathbf{\Omega}, E, t) \mathrm{d}\Omega. \tag{1.3}$$

The scalar flux is of interest in nuclear engineering calculations due to its use in calculating reaction rates via

$$f_r(\mathbf{x}, t) = \int_0^\infty \Sigma_r(\mathbf{x}, E)\phi(\mathbf{x}, E, t) \mathrm{d}E. \tag{1.4}$$

Analytic solutions to Eq. (1.1) can only be obtained in simple problems with limiting assumptions. As such, numerical methods are used to solve the neutron transport equation. Typically, these methods fall into one of two groups: deterministic or stochastic. Deterministic methods discretize Eq. (1.1) and solve the resulting system of equations [2]. Rather than solve for the average behavior of the radiation in the system directly, stochastic, or Monte Carlo, methods simulate the random behavior of radiation by transporting individual particles through the system. Monte Carlo methods obtain an estimate of the average behavior of particles, and thus an estimate to the solution of the neutron transport equation, by simulating millions of particles from their birth to their death in the system.

Monte Carlo methods offer several benefits over deterministic methods. Deterministic methods incur truncation error in their discretization of space and angle in the transport equation, and often times they must make approximations in order to model the problem geometry. Monte Carlo methods do not require discretization of the equations, and they are capable of modeling complex geometries without approximation. Furthermore, Monte Carlo methods provide solutions in continuous energy; that is, they do not require the discretization of energy space in order to solve the transport equation. Deterministic methods discretize energy space into G groups with $E_g < E_{g-1}$. This discretization requires group-average

cross sections, defined as

$$\Sigma_{tg}(\mathbf{x}) = \frac{\int_{E_g}^{E_{g-1}} \Sigma_t(E)\phi(\mathbf{x}, E)\mathrm{d}E}{\int_{E_g}^{E_{g-1}} \phi(\mathbf{x}, E)\mathrm{d}E}. \tag{1.5}$$

Calculating the group-average cross section in Eq. (1.5) requires already knowing the quantity of interest, the scalar flux. Thus, approximations must be made in order for the deterministic calculation to proceed, usually by including additional assumptions, relying on integral data obtained from experiments, and approximating the scalar flux with simpler calculations [3]. These discretizations can impose truncation errors, and they require expert knowledge in order to avoid the pitfalls associated with computing group-average cross sections. Continuous energy Monte Carlo simulations removes the need for these discretizations, both increasing the fidelity of nuclear reactor analysis methods and simplifying the workflow, reducing the chance of human error.

Even so, Monte Carlo methods have historically been prohibitively expensive for studying large and complex problems. For example, simulating a full 3-D PWR core with Monte Carlo methods would require estimating on the order of 6 billion quantities, posing a problem for both computing time and memory [4]. As the performance and capability of computers increases, so too does the complexity of tasks that engineers ask them to do. While high fidelity Monte Carlo methods have been of limited use for studying large systems like nuclear power reactors, their use has steadily increased in the past several decades as algorithms have improved and computer power has increased. As such, Monte Carlo methods are now being investigated for use in studying full-core commercial power systems and to conduct comprehensive multi-physics coupling.

This dissertation continues the work of improving Monte Carlo methods for their use in studying nuclear systems, particularly nuclear power reactors. This work focuses on improving the density estimation techniques for capturing spatial distributions within nuclear reactors, with applications for multi-physics coupling. Previous work on applying an Kernel Density Estimators (KDEs) as an alternative to histograms for estimating spatial distributions is expanded, with emphasis on accurately capturing distributions at material interfaces with continuous energy cross sections. The KDEs developed in this paper are applied in a High Performance Computing (HPC) environment, and their applications for reactor analysis and multi-physics coupling problems are highlighted. The remainder of this chapter introduces the Monte Carlo particle transport method in Section 1.2, the specifics of the KDE and previous work in applying KDEs to Monte Carlo particle transport methods in Section 1.3, the role of HPC for Monte Carlo simulations in Section 1.4, and the problem of multi-physics

coupling in Section 1.5.

1.2 The Monte Carlo Method

Monte Carlo methods are used to solve systems involving random processes, with applications ranging from finance [5], analyzing baseball games [6], and radiation transport [7]. The Monte Carlo particle transport method solves the linear Boltzman transport equation by simulating individual neutron histories. A particle's behavior is random in nature, and Monte Carlo methods use this by simulating millions of particles in order to discern the average behavior of a particular system. A particle is simulated from its birth, or source point, to its death by either absorption, leakage out of the problem, or elimination through variance reduction techniques. During the particle's life it undergoes collisions, streams through cells, and crosses surfaces. Each of these events can be used as a sample to estimate distributions within the system. For spatially-distributed quantities of interest, statistical estimators are used to determine estimates of these distributions, described later in this thesis.

In a Monte Carlo code, the distance a particle travels between collisions is determined by randomly sampling from the Probability Density Function (PDF) describing the probability that a particle with energy E will collide in $\mathrm{d}x$ about x:

$$p(x, E)\mathrm{d}x = \Sigma_t(x, E)e^{-\int_0^x \Sigma_t(x', E)\mathrm{d}x'}\mathrm{d}x, \tag{1.6}$$

where $\Sigma_t(x)$ is the macroscopic total cross section. The probability of a reaction type r occurring given a collision occurred is determined by

$$\mathrm{P}_r(\mathbf{x}, E) = \frac{\Sigma_r(\mathbf{x}, E)}{\Sigma_t(\mathbf{x}, E)}. \tag{1.7}$$

The macroscopic cross section of type r for a given material is calculated via

$$\Sigma_r(\mathbf{x}, E) = N\sigma_r(\mathbf{x}, E) \tag{1.8}$$

where N is the number density of the material and $\sigma_r(\mathbf{x}, E)$ is the microscopic cross section of the material at energy E and position $\mathbf{x}$. The microscopic cross sections are obtained from measurements and calculations from theory. For most energies of interest in reactor physics problems, from 1 meV to 10 MeV, microscopic cross sections are measured and

tabulated in data tables like ENDF/B [8] and JENDL [9], among others.

Geometries in Monte Carlo radiation transport simulations are typically created using Constructive Solid Geometry (CSG). In CSG cells are defined using bounding surfaces or through the union or intersection with other cells. This allows complex geometries to be created using only basic surfaces like planes, cylinders, cones, and spheres. Alternatively, geometries can be described via unstructured meshes. Arbitrary geometries can be constructed using an unstructured mesh, with the Stanford Bunny being a prime example [10, 11]. Allowing particle tracking and density estimation on unstructured meshes has applications for multi-physics coupling, as well as simplifying the nuclear engineer's design process by allowing existing geometry models, like those developed using CAD (Computer Aided Design) or CAE (Computer Aided Engineering) software, to be used directly by the transport software rather than having to replicate the geometry using CSG.

In Monte Carlo neutron transport two types of problems may exist: eigenvalue and fixed-source problems. Fixed-source problems are equivalent to solving Eq. (1.1) by transporting particles that originate from user-defined sources through the problem geometry. These types of problems are often used for shielding calculations and dose-deposition calculations. Steady state eigenvalue, or criticality, problems break the source term in Eq. (1.1) into fixed sources and the fission source:

$$Q(\mathbf{x}, \mathbf{\Omega}, E) = \frac{1}{k_{\text{eff}}} \chi(\mathbf{x}, E) \int_0^\infty \int_{4\pi} \nu\Sigma_f(\mathbf{x}, E')\psi(\mathbf{x}, \mathbf{\Omega}', E')\mathrm{d}E'\mathrm{d}\Omega' + Q'(\mathbf{x}, \mathbf{\Omega}, E), \quad (1.9)$$

where k_{eff} is the eigenvalue, χ is the Watt fission spectrum, and Q' encompasses all additional sources. Since the fission source includes the unknown angular flux, the eigenvalue problem must be solved iteratively. For Monte Carlo tallies this means breaking the simulation up into batches of neutrons with a number of inactive and active batches specified at the start of the simulation. During the inactive batches the fission source is being converged and quantities of interest are not tallied. For each batch, neutrons are simulated from their initial source site through the problem, and any collision that results in a fission is recorded as a fission source site. In the next batch, neutrons are born at the fission source sites and transported through the problem with any resulting fission source sites saved for the next batch. This process is repeated until a pre-specified number of inactive batches are simulated, ideally when the eigenvalue k and the eigenfunction ψ have converged. After that, the user-defined tallies become active and a number of active batches are run using the same process.

Results are extracted from Monte Carlo simulations using tallies. Since the simulation is stochastic, results are presented through a mean and a variance or relative uncertainty.

Confidence intervals about the mean are drawn using the central limit theorem. The central limit theorem states that the mean of a large number of independent and identically distributed random variables will be approximately normally distributed. This enables the user of a Monte Carlo code to determine the precision of the simulation results by drawing a confidence interval about the estimated mean. For example, a user would be 95% confident that the true value of the quantity of interest lies within 1.96 standard deviations of the estimate obtained from the simulation.

Traditionally, spatially-dependent quantities like the scalar flux and reaction rate densities are obtained using collision or track-length histograms. The phase space of interest is broken up into histogram bins, typically via a structured mesh, and quantities of interest are tallied in each bin. The collision density per source neutron in the j^{th} histogram bin is estimated via

$$\frac{1}{V_j}\int_0^\infty \int_{V_j} \Sigma_t(\mathbf{x}, E)\phi(\mathbf{x}, E)\mathrm{d}V\,\mathrm{d}E = \Sigma_t\phi_j = \frac{1}{V_j N}\sum_{i=1}^{N}\sum_{c=1}^{c_i} w_{i,c}, \tag{1.10}$$

where N is the total number of histories, c_i is the number of collisions in history i, and $w_{i,c}$ is the weight of the particle undergoing the c^{th} collision in history i. The average scalar flux in the j^{th} histogram bin is estimated using the collision histogram via

$$\phi_j = \frac{1}{V_j N}\sum_{i=1}^{N}\sum_{c=1}^{c_i} \frac{w_{i,c}}{\Sigma_t(\mathbf{X}_{i,c}, E)}. \tag{1.11}$$

The scalar flux can be thought of as the path length per unit volume per unit time generated by neutrons at a specific location. This leads to the definition of the track-length histogram for the average scalar flux in the j^{th} bin:

$$\phi_j = \frac{1}{V_j N}\sum_{i=1}^{N}\sum_{c=1}^{c_i} w_{i,c} d_{i,c}, \tag{1.12}$$

where $d_{i,c}$ is the length of the particle track generated by the particle within the volume of histogram bin j as it traveled to the c^{th} collision during history i. The track-length histogram is often preferred over the collision histogram due to its lower variance in problems with large streaming and its ability to natively capture flux distributions in voids and low-density materials. However, in reactor physics problems where regions of interest are highly scattering the use of either estimator is acceptable.

Another estimator commonly used in nuclear engineering problems is the next-event estimator, also known as the point-detector tally (PDT) [7]. The PDT allows the estimation of

the scalar flux or reaction rate density at a point. The PDT is mentioned here as it obtains the same type of information - quantities at a point - as the estimator introduced in the next section, the KDE, but in a fundamentally different manner. Probabilistically, the point detector estimator can be defined as

$$\phi(\mathbf{x}) = \sum_{i=1}^{N} \left[\mathrm{P}(\mathbf{\Omega} \cdot \mathbf{\Omega}')\mathrm{P}(\alpha) + \sum_{c=1}^{c_i} \mathrm{P}(c)\mathrm{P}(\mathbf{\Omega} \cdot \mathbf{\Omega}')\mathrm{P}(\alpha) \right], \tag{1.13}$$

where $\mathrm{P}(\mathbf{\Omega} \cdot \mathbf{\Omega}')$ is the probability of scattering from the current direction of travel into the angle subtended by the tally point $\mathbf{x}$, $\mathrm{P}(\alpha)$ is the probability a particle travels from $\mathbf{x}_{i,c}$ to $\mathbf{x}$ without undergoing a collision, and $\mathrm{P}(c)$ is the probability a particle will scatter given it underwent a collision. For isotropic scattering the PDT becomes

$$\phi(\mathbf{x}) = \sum_{i=1}^{N} \left[\frac{e^{-\alpha(\mathbf{x},\mathbf{x}_{i,c})}}{4\pi|\mathbf{x}-\mathbf{x}_{i,c}|^2} + \sum_{c=1}^{c_i} \left(\frac{\Sigma_s(\mathbf{x}_{i,c})}{\Sigma_t(\mathbf{x}_{i,c})} \right) \frac{e^{-\alpha(\mathbf{x},\mathbf{x}_{i,c})}}{4\pi|\mathbf{x}-\mathbf{x}_{i,c}|^2} \right]. \tag{1.14}$$

where $\alpha(\mathbf{x}, \mathbf{x}_{i,c})$ is the number of MFP between $\mathbf{x}$ and $\mathbf{x}_{i,c}$:

$$\alpha(\mathbf{x}, \mathbf{x}_{i,c}) = \left| \int_{\mathbf{x}}^{\mathbf{x}_{i,c}} \Sigma_t(s) \mathrm{d}S \right|. \tag{1.15}$$

While the mean of the PDT is bounded and unbiased, the variance of the PDT is infinite when collisions are able to occur at or near the tally point. The infinite variance is due to the $|\mathbf{x} - \mathbf{x}_{i,c}|^{-2}$ term in Eq. (1.14). One practical method for creating a PDT with finite variance is to specify an average flux region in a sphere around the tally point and score the average flux in that sphere to the tally point rather than computing the score from the PDT [12]. While this produces a bounded variance with a convergence rate of $N^{-2/3}$ [7], it also introduces a bias to the estimate of the flux at the point.

1.3 Kernel Density Estimators

1.3.1 Estimator Definition

Density estimation methods are statistical tools used to reconstruct an unknown PDF from a set of samples drawn from the PDF. A PDF describes the distribution of a random variable in phase space. In 1-D, the probability that a random variable X drawn from the PDF $f(x)$

lies between a and b is given by

$$\mathrm{P}(a < X < b) = \int_a^b f(x)\mathrm{d}x. \tag{1.16}$$

In practice the PDF is unknown but samples can be drawn from it through observations or through simulations of the physical processes that govern the PDF. These samples are used to reconstruct or estimate the PDF using either nonparametric or parametric estimators. A nonparametric estimator does not use *a priori* assumptions about the underlying distribution in order to obtain estimates about that distribution. In comparison, a parametric estimator assumes the functional form of the underlying PDF and uses the samples or observations to obtain estimates of the parameters that define the shape of the PDF. An example of a parametric estimator would be to assume the shape of the PDF being estimated is Gaussian and use the samples to obtain estimates of the mean and standard deviation of the Gaussian distribution. Nonparametric estimators make no assumption about the shape of the distribution, and thus are more flexible tools for capturing distributions where insufficient information is known *a priori*.

Several types of nonparametric estimators have been applied to Monte Carlo radiation transport simulations in recent years. Traditionally, histograms are used to obtain spatially distributed flux and reaction rate densities. While the use of histograms does not require any information about the shape of the distribution, the user must decide how many histogram bins to use (the volume of each bin) and where to place the bin edges. These decisions generally require careful consideration of system and the quantities being estimated and are subject to the user's judgement.

One alternative to the histogram is the Functional Expansion Technique (FET), studied in detail by Griesheimer [13]. The FET uses a series expansion to represent the quantity of interest using an orthogonal set of basis functions. While the FET worked well in some 1-D and 2-D problems, it was shown to have difficulties resolving material discontinuities in eigenvalue problems when using Legendre polynomials as the basis functions. It was also shown to have difficulties extending to general 3-D problems [13, 14]. Recent work by Ellis suggests that using a set of basis functions that more closely represents the problem geometry has potential applications for capturing fission reaction rate densities in light water reactors [15].

Another alternative to the histogram and the main focus of this dissertation is the Kernel Density Estimator (KDE). KDEs obtain estimates of the underlying density at discrete points in phase space, compared to the volume-average quantities obtained by a histogram and

the functional representation obtained by the FET. KDEs are nonparametric estimators first referenced by name by Parzen [16] in 1962, with earlier known references to estimators with properties similar to KDEs first by Fix and Hodges [17] in 1952 and by Rosenblatt [18] in 1956. The first multivariate KDE was introduced by Cacoullos [19] in 1966. Since then KDEs have been widely used as statistical tools to estimate unknown distributions in numerous fields with recent applications to nuclear engineering.

KDEs were first applied to the field of radiation transport in nuclear engineering by Du and Martin to re-sample electron trajectories [20] and by Tyagi et al. to create photon sources by re-sampling the phase space in the accelerator portion of a cancer treatment system [21]. Banerjee created a collision and a track-length KDE for Monte Carlo tallies in radiation transport, used KDEs to eliminate singularities in the next-event estimator and the surface-crossing flux estimator, and used KDEs to resample fission source sites in an attempt to accelerate fission source convergence [14]. Since Banerjee's work on KDEs they have been used in an increasing number of published works for nuclear engineering problems. Dunn followed on Banerjee's work by developing an integral track-length estimator as well as studying applications for KDEs on unstructured meshes in fixed-source problems in depth [22]. Yamamoto used KDEs in a weight-cancelling technique to obtain higher-order k-eigenfunctions [23]. Nease applied KDEs to residual Monte Carlo in 1-D to circumvent the need to discretize phase space [24]. Holgado et al. applied KDEs to obtain estimates of energy deposition in thermal radiative transfer problems [25]. Even though KDEs are cited with increasing frequency in the literature, histograms are still the most commonly used tool for density estimation in nuclear engineering problems.

The univariate KDE is defined as follows. Given a set of samples $X_1, \ldots, X_N$ from an underlying PDF $f(x)$, the KDE $\hat{f}(x)$ is defined by

$$\hat{f}(x) = \frac{1}{Nh} \sum_{i=1}^{N} k\left(\frac{x - X_i}{h}\right), \tag{1.17}$$

where h is the bandwidth, or smoothing parameter, and k is the univariate kernel function. The bandwidth will be discussed in more detail later in this paper. The kernel function k is a user-defined function with the following properties:

$$\int k(u)\mathrm{d}u = 1, \quad \int uk(u)\mathrm{d}u = 0, \quad \text{and} \quad \int u^2 k(u)\mathrm{d}u = k_2 \neq 0. \tag{1.18}$$

The univariate KDE can be extended to estimate multivariate densities two different ways

[26]. One method is to use a product of univariate kernels via

$$\hat{f}(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N}\prod_{l=1}^{d}\frac{1}{h_l}k\left(\frac{x_l - X_{l,i}}{h_l}\right), \tag{1.19}$$

where d is the number of dimensions, and h_l is the bandwidth in dimension l and $x_l - X_{l,i}$ is the difference between the point $\mathbf{x}$ and the sample in dimension l. Alternatively, a general multivariate kernel can be used to form a multivariate KDE:

$$\hat{f}(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{|\mathbf{H}|}K\left(\mathbf{H}^{-1}\left[\mathbf{x} - \mathbf{X}_{i,c}\right]\right), \tag{1.20}$$

where $|\mathbf{H}|$ is the determinant of the $d \times d$ symmetric positive definite bandwidth matrix. In practice the product of univariate kernels is preferred [26].

Previously, KDEs have been applied to Monte Carlo neutron transport simulations resulting in the collision and track-length KDEs [14] and the integral track-length KDE [27]. The multivariate collision KDE for scalar flux is defined as

$$\hat{\phi}(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{c_i}\frac{w_{i,c}}{\Sigma_t(\mathbf{X}_{i,c}, E)}\prod_{l=1}^{d}\frac{1}{h_l}k\left(\frac{x_l - X_{l,i,c}}{h_l}\right), \tag{1.21}$$

where N is the number of histories, c_i is the number of collisions in history i, $x_l - X_{l,i,c}$ is the distance between the tally point at location $\mathbf{x}$ and the location of collision c from history i in dimension l, Σ_t is the total macroscopic cross section, and h_l is the bandwidth in dimension l. A track-length KDE was developed based on choosing collision sites along the particle track, with the number of collision sites sampled being a trade-off between bias in the result and computational efficiency. To remove this need to decide the number of collision sites along the particle track, the integral track-length KDE was created [27]. The multivariate integral track-length KDE is defined as

$$\hat{\phi}(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{c_i} w_{i,c}\int_0^{d_{i,c}} K_s dS, \tag{1.22}$$

where K_s is a 3-D kernel function defined along the path length S in terms of the starting point of the track (X_o, Y_o, Z_o) and its unit direction vector (u, v, w):

$$K_s = \frac{1}{h_x}k\left(\frac{x - X_o - uS}{h_x}\right)\frac{1}{h_y}k\left(\frac{y - Y_o - vS}{h_y}\right)\frac{1}{h_z}k\left(\frac{z - Z_o - wS}{h_z}\right). \tag{1.23}$$

The collision KDE in Eq. (1.21) will be referred to as the distance-based collision KDE throughout this thesis, and the integral track-length estimator in Eq. (1.22) will be referred to as the distance-based track-length KDE (TL KDE) throughout this thesis. In practice Dunn used a 4-point Gaussian quadrature scheme to compute the integral of the kernel in the track-length KDE in Eq. (1.22), and this same methodology is applied in this thesis.

The multivariate KDEs in Eqs. (1.21) and (1.22) can be used to obtain an estimate of the scalar flux at a discrete point in phase space, with a single collision or particle track segment contributing to the score at multiple points. This behavior is distinct from traditional density estimation methods in Monte Carlo simulations, namely the histogram. Since a particle event, either a collision or a particle track segment, can contribute to the score at multiple tally points this estimator has the potential to produce estimates with reduced variance compared to histograms. Furthermore, the point-wise solution of the KDE offers advantages compared to the volume-average nature of the histogram, as will be discussed more in Section 1.5.

Similar to the PDT described in Section 1.2, KDEs obtain estimates of the underlying density at discrete points. While the type of information produced by the two estimators is the same, this is where the similarities between the two estimators end. The PDT is a next-event estimator where every source location contributes to the score at every tally point. The KDE uses collision sites or particle tracks to contribute to scores at tally points, and only to tally points in the local vicinity of the particle event when using kernels with finite support. Furthermore, the PDT requires ray-tracing between each source location and each tally point, where the KDE does not necessarily require such ray-tracing. Another major difference between the two estimators is the convergence rate. For an unmodified PDT in a scattering medium the estimator has an infinite variance and does not converge, even though the estimator is unbiased. When using a sphere of average flux to remove the singularity, the PDT's converge rate becomes $N^{-2/3}$ [7]. This convergence rate is comparable to the KDE's convergence rate of $N^{-4/7}$ in 3-D. Even though the convergence rate of the PDT is better than that of the KDE in 2-D, the PDT's requirement of ray tracing between each collision site and tally point makes the KDE the more computationally efficient estimator for problems with a large number of tally points in a scattering medium.

1.3.2 Optimal Bandwidth

The performance of the KDEs is heavily dependent upon the bandwidth. An optimal bandwidth for general KDEs is discussed in depth by Silverman and is defined as the bandwidth

that minimizes the Mean Integrated Square Error (MISE), the sum of the integrated square bias and the integrated variance [28]. The bias introduced by the kernel approximation in Eq. (1.17) is

$$\begin{aligned}
\text{bias}_h(x) &= E\left[\hat{f}(x)\right] - f(x) \\
&= E\left[\frac{1}{Nh}\sum_{i=1}^{N} k\left(\frac{x - X_i}{h}\right)\right] - f(x) \\
&= \frac{1}{Nh}\sum_{i=1}^{N} E\left[k\left(\frac{x - X_i}{h}\right)\right] - f(x) \\
&= \int \frac{1}{h} k\left(\frac{x-y}{h}\right) f(y)dy - f(x)
\end{aligned} \tag{1.24}$$

Using the substitution $u = (x - y)/h$, Eq. (1.24) becomes

$$\text{bias}_h(x) = \int k\left(u\right) f(x - uh)du - f(x). \tag{1.25}$$

Applying a Taylor series expansion of $f(x - uh)$ about x results in

$$\text{bias}_h(x) = -f(x) + f(x)\int k(u)du - hf'(x)\int uk(u)du + \frac{1}{2}h^2 f''(x)\int u^2 k(u)du + \ldots \tag{1.26}$$

Using the kernel properties in Eq. (1.18), the bias reduces to

$$\text{bias}_h(x) = \frac{1}{2}h^2 f''(x)k_2 + O(h^3). \tag{1.27}$$

The variance of the KDE is defined as

$$\text{var}\left[\hat{f}(x)\right] = \frac{1}{N}\int \frac{1}{h^2} k\left(\frac{x-y}{h}\right)^2 f(y)\mathrm{d}y - \frac{1}{N}\left[\int \frac{1}{h} k\left(\frac{x-y}{h}\right) f(y)\mathrm{d}y\right]^2. \tag{1.28}$$

Using the definition of the bias in Eq. (1.24) and the bias of the KDE in Eq. (1.27) results in

$$\begin{aligned}
\text{var}\left[\hat{f}(x)\right] &= \frac{1}{N}\int \frac{1}{h^2} k\left(\frac{x-y}{h}\right)^2 f(y)\mathrm{d}y - \frac{1}{N}\left[f(x) + \text{bias}_h(x)\right]^2 \\
&= \frac{1}{N}\int \frac{1}{h^2} k\left(\frac{x-y}{h}\right)^2 f(y)\mathrm{d}y - \frac{1}{N}\left[f(x) + O(h^2)\right]^2.
\end{aligned} \tag{1.29}$$

Using the substitution $u = (x - y)/h$, Eq. (1.29) becomes

$$\text{var}\left[\hat{f}(x)\right] = \frac{1}{Nh}\int k\left(u\right)^2 f(x - uh)\mathrm{d}u - \frac{1}{N}\left[f(x) + O(h^2)\right]^2. \tag{1.30}$$

Applying a Taylor series expansion of $f(x - uh)$ about x results in

$$\text{var}\left[\hat{f}(x)\right] = \frac{1}{Nh}\int k\left(u\right)^2 \left[f(x)\int k(u)du - hf'(x)\int uk(u)du + \right. \tag{1.31}$$

$$\left.\frac{1}{2}h^2 f''(x)\int u^2 k(u)du + \ldots\right]\mathrm{d}u - \frac{1}{N}\left[f(x) + O(h^2)\right]^2. \tag{1.32}$$

Assuming h is small and N is large and using the kernel properties in Eq. (1.18) reduces Eq. (1.32) to

$$\text{var}\left[\hat{f}(x)\right] = \frac{1}{Nh}f(x)\int k\left(u\right)^2 \mathrm{d}u + O\left(\frac{1}{N}\right). \tag{1.33}$$

This yields the approximate variance:

$$\text{var}\left[\hat{f}(x)\right] \approx \frac{1}{Nh}f(x)\int k\left(u\right)^2 \mathrm{d}u. \tag{1.34}$$

The MISE is just the sum of Eq. (1.27) squared and Eq. (1.34) integrated over all space, using the fact that $f(x)$ is a PDF, and is given by

$$\text{MISE} \approx \frac{1}{4}h^4 k_2^2 \int f''(x)^2 dx + \frac{1}{Nh}\int k(u)^2 du, \tag{1.35}$$

with the assumption that $f''(x)$ is continuous and square integrable. The MISE in Eq. (1.35) indicates a core issue in density estimation techniques: increasing the bandwidth to reduce variance results in an increased bias and vice-versa. Histograms suffer from the same issue: increasing the resolution of the histogram by reducing the volume of each histogram bin increases the variance within each histogram bin. This is a core feature of the KDE, and the reason why choosing the appropriate bandwidth is so important.

Finding a bandwidth h that minimizes the MISE by setting $\frac{\partial \text{MISE}}{\partial h} = 0$ results in a global optimal bandwidth in 1-D:

$$h_{opt} = k_2^{-2/5}\left(\int k(u)^2 du\right)^{1/5}\left(\int f''(x)^2 dx\right)^{-1/5} N^{-1/5}. \tag{1.36}$$

The optimal bandwidth in Eq. (1.36) is global in the sense that it provides a single bandwidth

for the problem's phase space. If more information about the underlying distribution is known or can be calculated beyond integral quantities then the Mean Square Error (MSE), the sum of the square bias and variance at a point in the PDF, can be minimized to obtain a MSE-optimal bandwidth

$$h_{MSE}(x) = k_2^{-2/5} \left(\int k(u)^2 du \right)^{1/5} f(x)^{1/5} f''(x)^{-2/5} N^{-1/5}. \tag{1.37}$$

This equation would allow for spatially-dependent optimal bandwidths and will be discussed further in future work in Section 5.4.

The optimal bandwidth equations in Eqs. (1.36-1.37) indicate another issue with KDEs: the optimal choice of bandwidth depends on the distribution being estimated. This issue is central to the work in this thesis, and will be revisited in Section 2.1. Without an *a priori* estimate of $f(x)$, an assumption must be made about the distribution of $f(x)$ in order to approximate $f''(x)$. The most common assumption is to approximate $f(x)$ as the normal distribution and use the moments of the estimated distribution as parameters in the optimal bandwidth [28]. Applying the assumption that $f(x)$ is normal, MISE-optimal bandwidths can be computed using

$$h_{\mathrm{MISE},l} = \left(\frac{4}{3N} \right)^{1/5} \sigma. \tag{1.38}$$

This optimal bandwidth can be extended to the multivariate KDE, resulting in the MISE-optimal multivariate bandwidth [28]:

$$h_{\mathrm{MISE},l} = \left(\frac{4}{(2+d)N} \right)^{1/(4+d)} \sigma_l, \tag{1.39}$$

where d is the number of dimensions used in the KDE tally, N is the number of samples, and σ_l is the standard deviation of the distribution of data in dimension l.

1.3.3 Choice of Kernel

The choice of the user-defined kernel function k and its multivariate counterpart K is governed by the kernel properties defined in Eq. (1.18). It has been found that as long as these kernel properties are satisfied then the actual choice of kernel has a minimal impact on the estimate of the underlying density compared to the choice of the bandwidth [28]. The optimal kernel that minimizes the MISE given the kernel properties in Eq. (1.18) is known

as the Epanechnikov kernel [29], defined as

$$k(u) = \frac{3}{4\sqrt{5}}\left(1 - \frac{u^2}{5}\right), |u| \leq \sqrt{5}. \tag{1.40}$$

This kernel minimizes the value of k_2 and $\int k(u)^2 \mathrm{d}u$, thus minimizing the MISE. The Epanechnikov kernel has the added benefit of having a finite support, which is a desirable property from a computational efficiency standpoint. This kernel is used throughout this thesis.

It is possible to use higher-order kernels that have the additional property that $k_2 = 0$, which produces a bias of $O(h^4)$ rather than $O(h^2)$. These higher-order kernels no longer satisfy the conditions for a PDF as they necessitate having negative values. Dunn investigated using higher order kernels for fixed-source problems, with the conclusion that the reduced bias came at a trade-off of increased variance and a reduction in computational efficiency [22]. This thesis uses the Epanechnikov kernel for all KDEs due to its efficiency from a MISE and computational efficiency standpoint along with its prolific use in the literature.

1.3.4 Boundary Treatment

The KDE suffers from additional bias at boundary locations due to the loss of the symmetric kernel property and normality. On a finite domain, the univariate kernel properties become

$$\int_V k(x_l)\mathrm{d}V = 1, \quad \int_V x_l k(x_l)\mathrm{d}V = 0 \tag{1.41}$$

At the boundary, the kernel function becomes clipped and these kernel properties no longer hold. This motivates the use of a boundary kernel that maintains these properties. Banerjee investigated several methods that reduce this bias at the boundary and found that the boundary kernel method was sufficiently accurate for nuclear engineering problems [14]. The boundary kernel method, developed by Jones, uses an asymmetric kernel whose first moment is zero to maintain these kernel properties [30]. The boundary kernel method uses a new kernel that is a linear multiple of the kernel function k that is used only at tally points whose support region extends beyond the domain of the problem. The univariate boundary kernel in the x dimension for minimum and maximum boundaries x_{min} and x_{max}, respectively, is defined as

$$k_B(u) = \frac{\left(a_2(p_1,p_2) - a_1(p_1,p_2)\right) uk(u)}{a_0(p_1,p_2)a_2(p_1,p_2) - a_1(p_1,p_2)^2}, \quad \text{for} \quad p_1 < u < p_2 \tag{1.42}$$

where $p_1 = \max(-\sqrt{5}, (x - x_{max})/h)$, $p_2 = \min(\sqrt{5}, (x - x_{min})/h)$, and

$$a_i(p_1, p_2) = \int_{p_1}^{p_2} u^i k(u) du. \tag{1.43}$$

In the case of axis-aligned boundaries, the multivariate boundary kernel is just the product of these boundary kernels

$$K_B(\mathbf{u}) = \prod_{l=1}^{3} k_B(u_l). \tag{1.44}$$

While this method works well for axis-aligned boundaries, it must be extended to handle curved surfaces that can exist in neutron transport problems. The multivariate boundary kernel method for arbitrarily shaped boundaries is

$$K_B(\mathbf{x}) = c_0 K(\mathbf{u}) + c_1 u K(\mathbf{u}) + c_2 v K(\mathbf{u}) + c_3 w K(\mathbf{u}). \tag{1.45}$$

where the coefficients c_l are unknown and are solved for via the constraints of the kernel properties:

$$\int\int\int \left[c_0 K(\mathbf{u}) + c_1 u K(\mathbf{u}) + c_2 v K(\mathbf{u}) + c_3 w K(\mathbf{u})\right] du dv dw = 1, \tag{1.46}$$

$$\int\int\int u \left[c_0 K(\mathbf{u}) + c_1 u K(\mathbf{u}) + c_2 v K(\mathbf{u}) + c_3 w K(\mathbf{u})\right] du dv dw = 0, \tag{1.47}$$

$$\int\int\int v \left[c_0 K(\mathbf{u}) + c_1 u K(\mathbf{u}) + c_2 v K(\mathbf{u}) + c_3 w K(\mathbf{u})\right] du dv dw = 0, \tag{1.48}$$

$$\int\int\int w \left[c_0 K(\mathbf{u}) + c_1 u K(\mathbf{u}) + c_2 v K(\mathbf{u}) + c_3 w K(\mathbf{u})\right] du dv dw = 0. \tag{1.49}$$

Here the integration is taken over the support domain. The constants can be solved for using a linear system:

$$\begin{bmatrix} a_{00} & a_{01} & a_{02} & a_{03} \\ a_{10} & a_{11} & a_{12} & a_{13} \\ a_{20} & a_{21} & a_{22} & a_{23} \\ a_{30} & a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{1.50}$$

where

$$a_{i,j} = \int\int\int x_i x_j K(\mathbf{u}) du dv dw, \tag{1.51}$$

with $x_0 = 1$, $x_1 = u$, $x_2 = v$, $x_3 = w$. This boundary kernel method for arbitrary surfaces is not used in this due to the computationally expensive integrals required in Eq. (1.51), described in more detail in Section 2.3. Furthermore, the problems studied in this thesis are

all pincell problems with axis-aligned boundaries with reflecting boundary conditions, and thus a product of univariate boundary kernels can be used at external boundaries with no loss in accuracy. The reflection method [28, 30] could also be used for these problems since reflecting boundaries are used for all problems, however it was found that the boundary kernel method is sufficient for problems studied in this thesis.

1.3.5 Considerations for Eigenvalue Problems

Routines for obtaining the optimal bandwidth for eigenvalue problems have been discussed by Banerjee in significant detail [14]. It was shown that the optimal bandwidth can be calculated on a per-batch basis in an eigenvalue problem, with the optimal bandwidth for a batch being calculated using the data from the previous batch via Eq. (1.39) with N being the number of collisions or track-lengths in the previous batch and σ_l being the standard deviation of the distribution being estimated in dimension l. For flux distributions, σ_l can be estimated using a collision estimator via

$$\sigma_l = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\frac{w_{i,l}}{\Sigma_t(X_{i,l})}\left(X_{i,l}\right)^2 - \left(\frac{1}{N}\sum_{i=1}^{N}X_{i,l}\frac{w_{i,l}}{\Sigma_t(X_{i,l})}\right)^2}. \tag{1.52}$$

This allows optimal bandwidths to be calculated using the simulation data without the need to store collision or track information.

However, using batch statistics for the optimal bandwidth can lead to inconsistencies in Eigenvalue problems. The number of particles per batch and the number of active batches in an eigenvalue problem are problem-dependent quantities that are chosen by the user. If the fission source is converged and the number of particles per batch is large enough to mitigate bias due to small batch sizes, then adjusting the number of particles per batch and total number of active batches while maintaining the same total number of active particles should not affect the mean of the tally. Results from a histogram tally produce the same mean within the statistical variation of the simulation when adjusting the ratio of the number of particles per batch to the number of batches while the KDE will converge to a different mean when this ratio is adjusted. Alternatively, the estimated total number of samples in the tally can be used to compute optimal bandwidths via

$$h_l = \left\{\frac{4}{(d+2)N_{samples}\times N_{batches}}\right\}^{\frac{1}{d+4}}\sigma_l. \tag{1.53}$$

Optimal bandwidths obtained using Eq. (1.53) with different ratios of number of particles per batch to number of active batches while maintaining the same total number of active particles are now equivalent within statistical variation. Thus, KDE tallies with optimal bandwidths using Eq. (1.53) will have means that are invariant to different ratios of number of particles to number of active batches where as bandwidths calculated using Eq. (1.36) will result in a bias that is dependent upon the number of particles per batch regardless of the total number of batches. Results showing the difference between using the estimated total number of samples in a KDE region versus the number of samples in the previous batch are produced later in this thesis in Section 2.4.2.

Furthermore, it should be noted that since the bandwidth is proportional to $N^{-(1/4+d)}$ and the variance is proportional to $(hN)^{-1}$, the KDEs using an optimal bandwidth converge at a rate proportional to $N^{-(4)/(4+d)}$ rather than the N^{-1} of histogram tallies. This slightly lower convergence rate compared to histogram tallies is expected to be balanced by the ability of KDEs to allow a particle event to contribute scores to multiple tally points.

Additionally, it has been shown that the accuracy of KDE methods can suffer when estimating multi-modal densities. A suggested solution is to segment the multimodal problem into several unimodal problems [31]. Banerjee employed this technique to obtain locally-adaptive region-based bandwidths by defining unique KDE regions over collections of cells in the Monte Carlo simulation and using the statistics in each region to define bandwidths unique to each region. Using region-based bandwidths was shown to be a more accurate method for estimating quantities in heterogeneous problems than a KDE that utilizes a global bandwidth [14].

For the purposes of comparing results from histograms and KDEs, this thesis directly compares volume-average histogram bin estimates with point-wise estimates of KDEs evaluated at the center of the histogram bin for the majority of test problems. This method instills a degree of bias between the comparison of the two estimators as they are estimating fundamentally different quantities. If the underlying density estimate within the bin is non-linear, then it is likely that the average value of the density within the bin will disagree with the value of the density at the midpoint of the bin. Dunn investigated several methods for quantitatively comparing histograms and KDEs using either a "node-to-cell" method or a "cell-to-node" method [22], however both of these methods create additional bias as well. Section 2.7 introduces a volume-average KDE for directly comparing results obtained using KDEs and histograms without introducing bias into the comparison. Even so, the direct comparison of KDEs to histograms on a high-resolution mesh is sufficient in this thesis to determine the degree of accuracy of the KDE methods developed here compared

to the previous state-of-the-art in KDEs for reactor physics problems. This thesis uses the difference between results in units of number of standard deviations as one method of comparing the performance of various KDE methods with the standard deviation in the results calculated using error propagation via

$$\sigma_j = \sqrt{\sigma_{j,\text{Hist.}}^2 + \sigma_{j,\text{KDE}}^2}, \tag{1.54}$$

where σ_j is the uncertainty in the difference between the histogram estimate in the j^{th} bin and the KDE estimate at the j^{th} tally point.

1.3.6 Remaining Challenges in KDEs Applied to Reactor Analysis

Prior to the start of this work, the KDE methods developed by Banerjee and Dunn were not applied to analyze reactor physics problems in continuous energy. Banerjee's work developed the initial collision KDE for neutron transport problems and used it to model 1-D and 2-D one-group reactor physics problems where the macroscopic total cross sections of the fuel, water, and control rods were set to the same value. Dunn's work focused on the development of the track-length KDE and its application to fixed source shielding applications. Thus, one of the remaining challenges is to apply KDEs to estimate distributions in more-realistic reactor physics problems in continuous energy.

Furthermore, prior to the start of the work detailed in this thesis the estimators developed by Dunn and Banerjee were only used to estimate scalar flux; no work had been done on estimating reaction rates. While Banerjee did use KDEs to estimate the fission source distribution at the end of an eigenvalue batch, it was done using an analog estimator with the samples being the fission sites generated during the batch. Thus, another remaining challenge in applying KDEs to analyze reactor physics problems is to create a non-analog KDE to estimate reaction rates and verify the new estimator for a set of realistic problems. This thesis focuses on the application of KDEs to Pressurized Water Reactor (PWR) analysis, with application to reactors with large streaming paths left for future work. This challenge area encompasses the majority of the work detailed in this thesis, as described in Section 1.6.

Additionally, Dunn studied methods to quantitatively compare results obtained from KDEs to those obtained via histograms on an unstructured mesh. However, both the methods introduced, discussed in more detail in Section 2.7, introduce bias in the comparison. Therefore another remaining challenge was to obtain a comparison method between the

point-wise KDE results and the volume-average histogram quantities that did not introduce bias.

1.4 High Performance Computing

Monte Carlo radiation transport calculations are often run on computer clusters due to their large computational cost. Computer clusters are composed of a large number of compute nodes, each with one or more processors each containing several independent processing units called cores and one or more Graphics Processing Units (GPUs). GPUs are often included in a compute node as a means of accelerating performance for benchmarks and vectorizable algorithms without adding additional compute nodes. For example, each node of the HPC cluster used in this thesis contains two 8-core processors and two Graphics Processing Units (GPUs).

Monte Carlo radiation transport codes easily take advantage of the resources provided by HPC clusters due to the independent nature of particle simulation and are often considered to be embarrassingly parallel. Each particle is independent from the simulation of any other particle, thus it is easy for multiple cores to simulate different particle histories simultaneously. This is done by replicating the problem across cores via MPI or by using threads. The simulation is parallelized by dividing the total number of particle histories being simulated among each thread in each MPI process.

While this method of parallelization works well for clusters of multi-core CPUs, Monte Carlo radiation transport calculations are not well suited for parallelization on GPUs. GPUs are composed of hundreds to thousands of threads, with the GPU architecture requiring groups of 32 threads to either all perform the same instruction or sit idle while the other threads in the group execute the instruction. This means that the acceleration of traditional Monte Carlo radiation transport codes on GPUs loses efficiency from the use of logic statements as well as the random access of memory when looking-up and computing continuous energy cross sections. While a large body of research has been generated for creating Monte Carlo algorithms that can run on GPUs, e.g. [32], there is still no production-level Monte Carlo neutron transport code capable of using GPUs.

Rather than use GPUs for the main Monte Carlo particle transport algorithm, GPUs can be used to accelerate compute-intensive vectorizable portions of the code. Since the tally routines are independent of the transport routines and the KDE scoring process for each tally point and particle event are independent from one another it is possible to export the KDE

tally routines to the GPU. Particle events and the relevant information for computing scores to tally points for each particle event are sent in batches to the GPU for the calculation of scores while the CPU continues the transport process. This enables the GPU to hide the majority of the cost of the KDE.

1.5 Multi-physics Coupling

When designing nuclear reactor systems, studying the various physical phenomena occurring inside the reactor separately is often insufficient even for steady-state calculations. The physical phenomena are linked: neutron and photon transport determines the heat generation rates in the system and thermal-hydraulics and heat conduction use these heat generation rates to determine temperatures inside the reactor. The transport equations are affected by temperatures through the changing density of the materials inside the system as well as through the Doppler broadening of microscopic cross sections. Thus, coupled multi-physics simulations are required in order to obtain high-fidelity solutions of reactor physics problems.

Traditionally, the high cost of Monte Carlo methods prevented their use in coupled multi-physics simulations. However, with the increasing computer power accompanied with algorithmic developments Monte Carlo methods are now being investigated for use in coupled multi-physics simulations. Recent efforts in Monte Carlo methods development include particle tracking and tallying on unstructured meshes for use in multi-physics problems as well as for complex geometry representation, for example in MCNP6 [33], and Serpent [34]. KDEs have shown potential for use in coupled multi-physics solutions for two main reasons: their ability to obtain estimates at points without requiring an underlying mesh and their ability to obtain high-resolution estimates of spatially-resolved quantities without necessarily having an increased variance compared to a low-resolution estimate.

Unstructured meshes are a common feature of codes used to model physics that are coupled to neutronics solvers, with CFD being a prime example. The meshes generated for solution of coupled physics phenomena are generally different than those designed to capture neutronics distributions within a transport code. For example, when coupling neutronics and CFD the unstructured mesh generated for the thermal-hydraulics is generally much finer than is required to capture the desired power generation distribution from the neutronics code. Using different meshes for the various physics solvers requires mapping the results from one mesh to another. This remapping is a non-trivial exercise and has had a significant body

of research dedicated to solving the problem of transferring information from one mesh to another [35, 36, 37].

The difficulties of remapping can be reduced or marginalized by using either the same mesh or meshes as similar as possible for the various physics solvers. Thus, it is desirable to use a mesh that is finer than required in the neutronics solutions. However, using a finer mesh than required has undesirable effects on the efficiency of the Monte Carlo neutronics solution when using histogram tallies, often requiring a prohibitive number of particles to reduce the uncertainties in each mesh element to an acceptable level. Moreover, this may not eliminate the issue with transferring information between meshes as histograms obtain volume-average results and finite element codes require point-wise values at the nodes of the unstructured mesh. KDEs are capable of solving both of these issues as they natively obtain results at points; placing tally points on the nodes of an unstructured mesh naturally lends itself to building finite element representations of the estimated quantity. Furthermore, KDEs use the bandwidth to determine the spatial scale upon which underlying features are resolved; the density of tally points does not affect the variance of the density estimates although it will impact the runtime of the simulation.

These potential benefits were the driver for the study, composed by Dunn, of KDEs for use in unstructured meshes [22]. This work continues that effort by demonstrating that the MFP KDE developed in Chapter 2 can accurately capture reaction rate distributions in reactor physics problems on an unstructured mesh. While this work does not apply the KDEs developed here to real multi-physics coupling problems due to the effort required in properly constructing a non-trivial problem, it does prove that they can now be used effectively on such problems.

1.6 Thesis Outline

- Chapter 2 introduces new methods for computing reaction rate densities in reactor physics problems using KDEs. A new estimator, the Mean Free Path (MFP) KDE, is introduced in order to accurately estimate densities at material interfaces. Univariate and multivariate MFP KDEs are derived and verified using one-group reactor physics problems. Approximations to the MFP KDE are introduced in order to improve computational efficiency at the cost of potential bias in the results. The volume-average KDE is developed for the collision and track-length KDEs as a method for quantitatively comparing results obtained from KDEs and histograms, as well

as obtaining volume-average quantities using KDEs. The bias introduced by the approximation is estimated using the volume-average KDE in 1-D and 2-D reactor physics problems in continuous energy. The fractional MFP KDE and minimum bandwidth are introduced in order to mitigate computational efficiency issues in the multivariate MFP KDE. A new KDE formulated for cylindrical coordinates is introduced as a means for capturing distributions in cylindrical geometries common in reactor physics problems. The variance and bias of the cylindrical KDE is derived. A cylindrical MFP KDE is derived and verified using a reactor physics pincell problem in continuous energy.

- Chapter 3 describes the KDE algorithms in detail and introduces a method to accelerate the KDE tally using GPUs. The CPU algorithm is detailed along with routines to improve the efficiency of the algorithm. The GPU algorithm is introduced, with performance comparisons given for various optimizations of the GPU algorithm. The speedup (relative increase in performance) obtained by using the GPU algorithm is analyzed for several problems and performance comparisons are made between the KDE methods developed in this thesis and histogram tallies on equivalent meshes.
- Chapter 4 shows applications of the cylindrical MFP KDE to important problems in reactor physics pertaining to the modeling of PWRs. While multi-physics coupling was not conducted in this thesis, the application of the cylindrical MFP KDE to estimate distributions on an unstructured mesh is shown in this section as a proof-of-principle for their future application to simulations with coupled multi-physics.
- Chapter 5 summarizes the main accomplishments of this thesis and discusses areas of future work.

CHAPTER 2

Kernel Density Estimators for Capturing Reaction Rates in Reactor Physics Problems

2.1 Estimating Reaction Rates With KDEs

In order to apply KDEs to reactor physics problems, they need to be able to accurately estimate reaction rates. KDEs take a collection of samples from an unknown density function and place a kernel function around each sample to obtain a smooth estimate of the underlying density function. While this smoothing is desirable throughout most of the distribution, it has several drawbacks for reactor physics problems and in particular the estimation of reaction rate densities.

KDEs obtain an estimate of a distribution by taking samples from that distribution and distributing the scores from those samples through phase space via the kernel function. An analog view of applying KDEs to estimate reaction rates would be to apply a kernel function to every reaction site and summing those kernel functions to obtain a reaction rate density at a given point. However, this analog view would allow a fission site near the fuel-cladding interface in a fuel pin to spread the fission density into the cladding, yielding a non-zero fission density outside of the fuel and depressing the score inside of the fuel.

A non-analog alternative of this would be to view reaction rate estimation through the use of a modifier to the scalar flux. At each collision site the distance-based collision KDE spreads out that collision site's contribution to the scalar flux at the energy of the particle that caused the collision. Thus, estimates of the reaction rate at a tally point can be obtained by multiplying every collision's contribution to the scalar flux by the cross section of the reaction of interest at that point evaluated at the energy of the particle prior to the collision. A depiction of this non-analog interpretation is shown in Figure 2.1b. This non-analog

(a) Analog (b) Non-analog

Figure 2.1: Graphic depicting analog (right) and non-analog (left) view of fission reaction rate estimation using KDEs. $X_{i,c}$ indicates a fission or collision site with the dotted line representing the kernel associated with that event.

interpretation results in a collision KDE for reaction rates:

$$\hat{f}(\mathbf{x}) = \frac{1}{Nh} \sum_{i=1}^{N} \sum_{c=1}^{c_i} \frac{w_{i,c} \Sigma_r(\mathbf{x}, E)}{\Sigma_t(\mathbf{X}_{i,c}, E)} k\left(\frac{\mathbf{x} - \mathbf{X}_{i,c}}{h}\right), \tag{2.1}$$

where $\hat{f}(\mathbf{x})$ is the estimate of the reaction rate density at tally point $\mathbf{x}$ and Σ_r is the cross section for the reaction rate of interest. Similarly, this idea can be applied to Eq. (1.22) to obtain the track-length KDE for reaction rates:

$$\hat{f}(\mathbf{x}) = \frac{1}{N} \sum_{i=1}^{N} \sum_{c=1}^{c_i} w_{i,c} \int_0^{d_{i,c}} \Sigma_r(\mathbf{x}, E) K_s dS, \tag{2.2}$$

2.1.1 Verification in 1-D Reactor Physics Problems

In order to verify that the reaction rate KDEs are working properly, Banerjee's simple 1-D, one-group representation of a fuel lattice containing a strong absorber was modeled [14] with KDE results compared to a reference histogram tally. The KDEs described in this thesis are implemented in a modified version of OpenMC [38], an open-source Monte Carlo code, and all reference histogram results throughout this thesis are obtained using OpenMC. The 1-D fuel lattice is depicted in Figure 2.2 and consists of alternating slabs of fuel and moderator 1 cm thick with a 1 cm thick central slab of absorber material and slabs of moderator 0.5 cm thick at the two boundaries. The one-group cross sections are given in Table 2.1. It should be noted that the macroscopic total cross sections given in Table 2.1 are identical for each material.

The reference histogram consists of a uniform mesh of 1,680 bins over 14 cm with bin-edges coinciding with material interfaces. KDE tally points are placed at the center of each

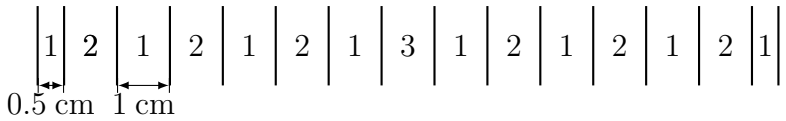


Figure 2.2: Graphic depicting geometry of 1-D alternating slab problem with material labels 1, 2, and 3 corresponding to moderator, fuel, and absorber, respectively.

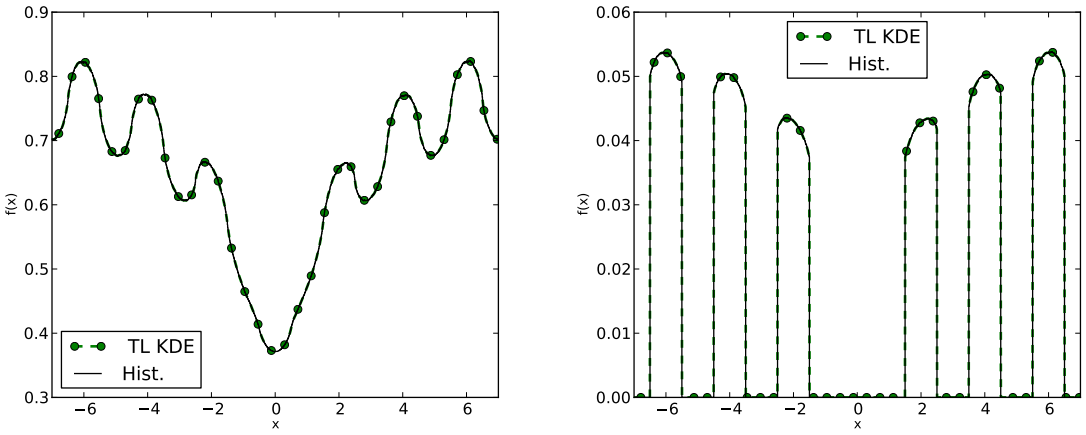
Table 2.1: One-group cross sections for first set of test problems.

	Σ_t	Σ_a	Σ_s	Σ_f	ν
Moderator	0.3264	0.097920	0.228480	0	0
Fuel	0.3264	0.078336	0.248064	0.06528	2.7
Absorber	0.3264	0.321760	0.004640	0	0

histogram bin. The KDEs treat each cell as a unique region, thus making 15 regions with unique bandwidths. Results are obtained using 100,000 histories per batch with 50 inactive batches and 100 total batches. The optimal bandwidths calculated using Eq. (1.36) range between 0.07 and 0.11 cm for the two boundary moderator cells, with the bandwidths in the other fuel and water cells ranging between 0.13 and 0.2 cm and the bandwidth in the central absorber ranging between 0.16 cm and 0.21 cm. Figure 2.3 shows comparisons between the flux and fission reaction rate density estimated from the histogram tally and the track-length KDE tally. The collision KDE and track-length KDE results agree within three standard deviations with maximum relative uncertainties of 0.6% and 0.25% for the collision KDE and track-length KDE respectively, and as such the collision KDE results have been omitted for clarity.

Figure 2.3 shows that the distance-based KDE accurately captures the flux throughout the one-group problem where each material has the same macroscopic total cross section. Furthermore, Figure 2.3 shows that the distance-based KDE produces the correct behavior when estimating reaction rates, with the discontinuity in the fission reaction rate between the moderator and the fuel being accurately captured. The maximum percent difference between the track-length KDE results and the histogram results is less than 0.3% for all distributions, with every bin and tally point agreeing within one standard deviation with maximum relative uncertainties of 0.25% for both estimators. Thus, the methodology for capturing reaction rates using KDEs is proven to be effective in one-group problems and can now be extended to problems in continuous energy.

The continuous energy test problem uses the same geometry as the one-group problem, depicted in Figure 2.2, with the materials at 300K consisting of light water in place of

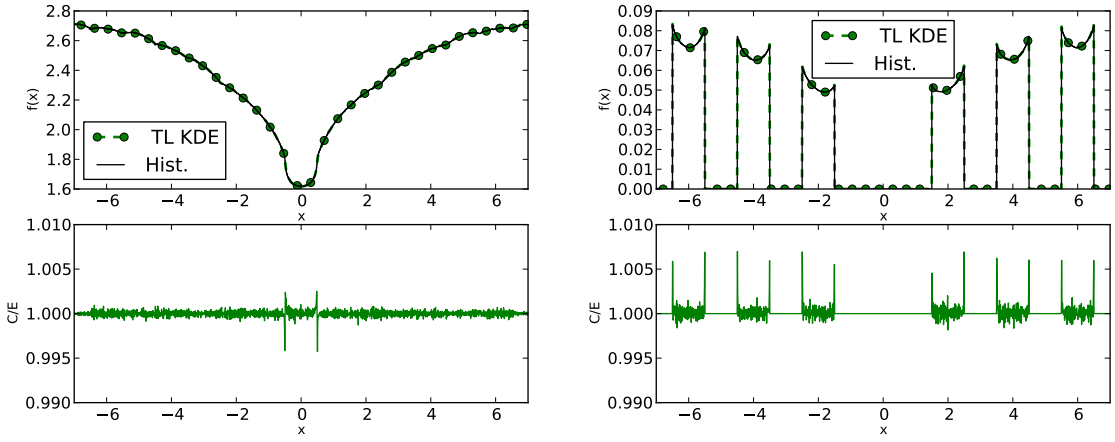


(a) Flux (b) Fission

Figure 2.3: Comparison of flux (left) and fission reaction rate densities (right) obtained from a reference histogram and KDE tally for the one-group, 1-D array of pincells problem.

the moderator, 3% enriched UO_2 in place of the fuel, and a central slab of B_4C in place of the absorber. All continuous energy cross sections in this thesis are obtained from the ENDF/B-VII.0 library [39]. The run-time parameters are the same as in the one-group problem with 1,680 bins and tally points, one KDE region defined for each cell, and 100,000 particles per batch with 50 inactive batches and 100 total batches. Figures 2.4 and 2.5 show the comparisons between the flux and the fission reaction rate density and the absorption reaction rate density, respectively, obtained by the track-length KDE and the reference histogram for the continuous energy problem. The different C/E (calculated to reference value) scales between the flux and fission distributions and the absorption distribution should be noted. The distributions are shown on top with every 50th KDE tally point marked, and the C/E values (KDE/histogram) for each bin and tally point are shown on the bottom.

From Figure 2.4a it is apparent that the KDE is capable of accurately capturing flux profiles and reaction rate densities in the 1-D array of alternating slabs in continuous energy, with the maximum difference between the KDE and histogram being less than 0.1% throughout most of the distribution. However, there is a noticable bias of approximately 0.4% at the interface between the control rod and the water corresponding to a disagreement of 4-σ. A similar picture is seen in Figure 2.4b, with the fission reaction rate density agreeing between the two estimators within 0.1% throughout most of the distribution, with a noticable increase in disagreement due to bias in the KDE estimate at the material interface of up to 0.7%. The absorption reaction rate density in Figure 2.5 shows worse agreement, with up to 20%



(a) Flux (b) Fission

Figure 2.4: Comparison of flux (left) and fission reaction rate densities (right) obtained from histogram and KDE tallies for the continuous energy problem. Estimated distributions are shown on top with C/E (KDE/histogram) values shown on the bottom.

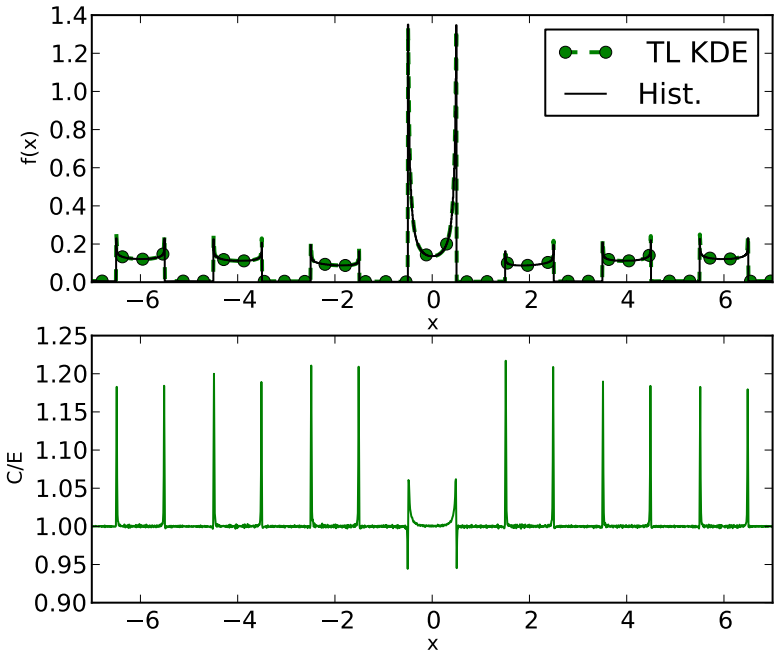


Figure 2.5: Comparison of flux (left) and fission reaction rate densities (right) obtained from histogram and KDE tallies for the continuous energy problem. Estimated distributions are shown on top with C/E (KDE/histogram) values shown on the bottom.

error at the material interface between the fuel and the water and approximately 5% error at the absorber-water interface. This discrepancy is not due to statistical uncertainty, as the maximum relative error of the two distributions is 0.14% and the disagreement between the

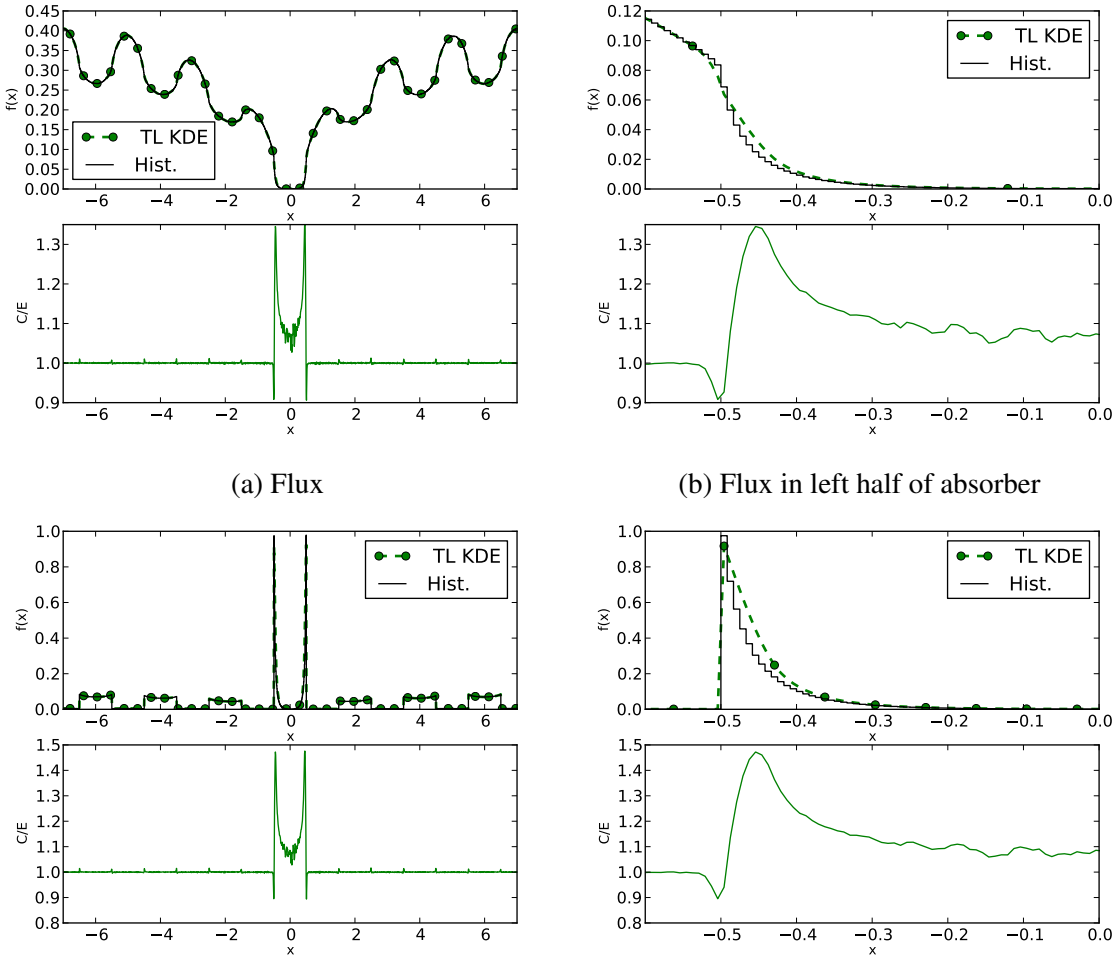
two distributions exceeds 30-σ.

The source of this bias is two-fold: in part due to using bandwidths that are too large for the distribution the KDE is trying to estimate, and in part due to using kernels that do not have the appropriate shape. While the region-based bandwidths provide a coarse adaptive bandwidth, they are insufficient for capturing the drastic changes that can occur due to material heterogeneity. To better demonstrate the cause of the problem, the same 1-D array of alternating slabs was run with the tally divided into three energy ranges: a thermal range from 0-0.5 eV, an epithermal range from 0.5-1 keV, and a fast range for energies greater than 1 keV. The thermal flux and thermal absorption distributions are shown in Figure 2.6, with the distribution across the array shown on the left and the distribution within the left edge of the absorber shown on the right. The epithermal flux, fission, and absorption distributions along with a close-up of the fuel-water interface are shown in Figure 2.7 and the fast flux, fission, and absorption distributions are shown in Figure 2.8.

The first thing to note from Figures 2.6-2.8 is that the flux, fission, and absorption distributions change significantly with energy. This is especially evident in the central absorber, where the thermal absorption distribution is overestimated by as much as 45% while the fast absorption distribution is accurately estimated to within 1%. The absorption distributions within the fuel also change significantly with energy; the thermal and fast absorption distributions exhibit low curvature while the epithermal distributions are highly peaked at the material interface due to resonance absorption.

Additionally, the KDE fails to capture the discontinuity in the first derivative of the flux. The transition between the water and the absorber, shown in Figure 2.6d, yields an abrupt change in the derivative of the flux. The histogram captures this feature while the KDE smoothes it out, creating a smooth transition between the flux in the water and the flux in the absorber. This feature was not present in the one-group version of the problem due to the macroscopic total cross sections of the various materials being equivalent.

The cause of the inaccuracies in the KDE results is two-fold: the distance-based KDE uses one bandwidth for all energy ranges and it does not account for changing material properties across material interfaces. A first approach to remedying this issue would be to use different bandwidths for each energy range. However, this approach does not solve the entire problem as distributions within an energy range also change significantly in space. The epithermal absorption distribution seen in Figures 2.7c-2.7d has significant curvature at the fuel-water interface, however it becomes comparatively flat in the majority of the fuel. Using a single bandwidth to capture this distribution, even with an epithermal-specific bandwidth, would



(a) Flux

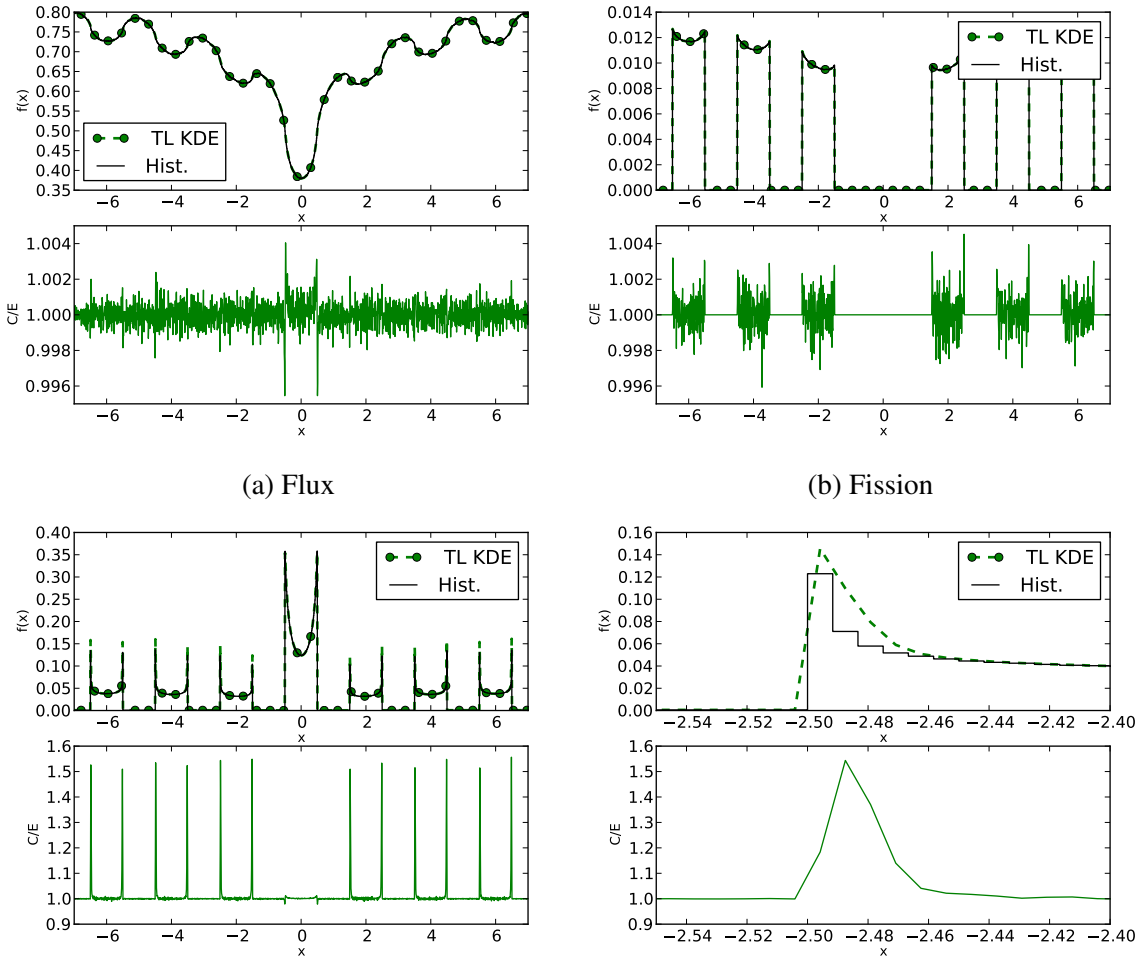
(b) Flux in left half of absorber

(c) Absorption

(d) Absorption in left half of absorber

Figure 2.6: Comparison of thermal flux (top) and thermal absorption reaction rate densities (bottom) obtained from histogram and KDE tallies for the 1-D alternating slab problem in continuous energy . Estimated distributions are shown on top of each figure with C/E (KDE/histogram) values shown on the bottom.

either result in a bandwidth that's too large, producing a bias at the material interface, or produce a bandwidth that's too small, resulting in a large variance through the majority of the fuel.



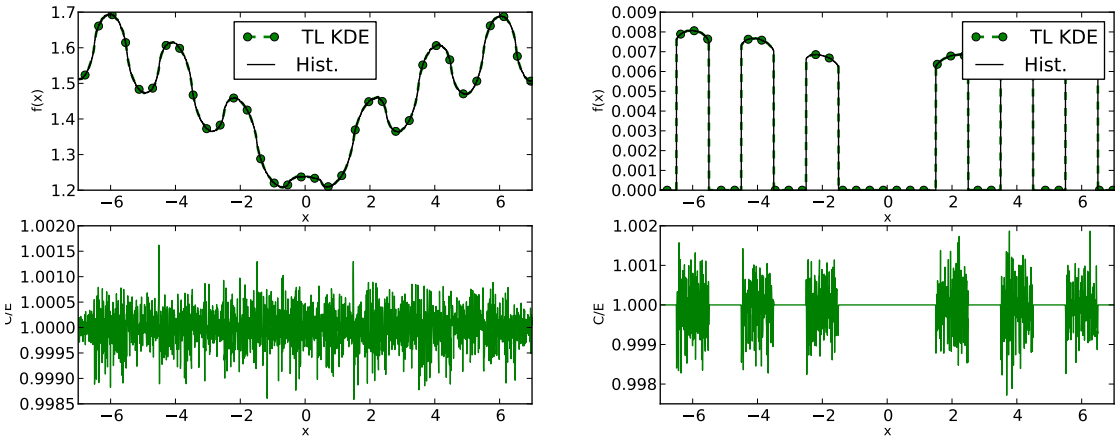
(a) Flux

(b) Fission

(c) Absorption

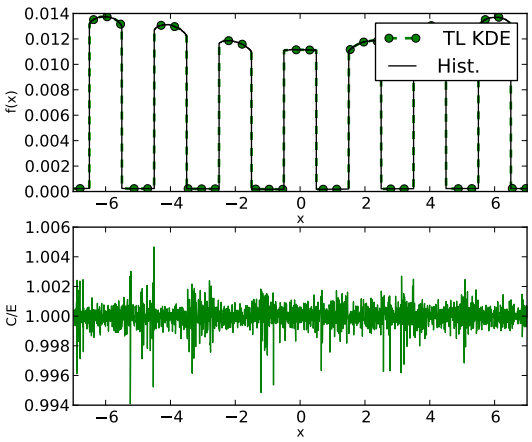
(d) Absorption at left fuel-water interface.

Figure 2.7: Comparison of epithermal flux, fission, and absorption distributions obtained from histogram and KDE tallies for the 1-D alternating slab problem in continuous energy.



(a) Flux

(b) Fission



(c) Absorption

Figure 2.8: Comparison of fast flux, fission, and absorption distributions obtained from histogram and KDE tallies for the 1-D alternating slab problem in continuous energy.

To develop a solution to this problem, it is imperative to consider the source of the issue. The energy dependence of the distributions arises from the energy dependence of the underlying microscopic cross sections of the materials. A plot of the energy dependence of the U-235 microscopic fission and U-238 microscopic total cross sections produced using JANIS with data from ENDF/B-VII.0 is shown in Figure 2.9 [39, 40].

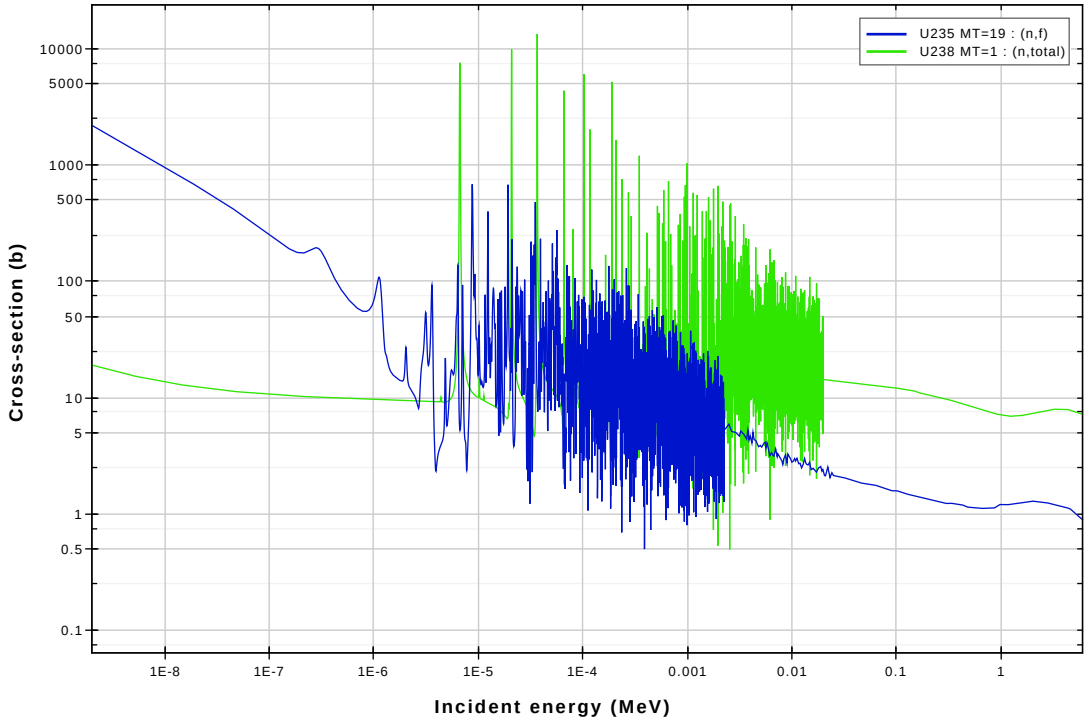


Figure 2.9: Microscopic total cross section for U-238 and microscopic fission cross section for U-235 obtained from data library ENDF/B-VII.0 using JANIS.

The energy dependence of the microscopic cross sections is a result of the neutron wavelength and the shell structure of the compound nucleus formed from any non-elastic collision. The increase in cross sections at low energies is due to the increase of the neutron wavelength, which is proportional to $1/\sqrt{E}$. The large increases in the cross section within small energy ranges are known as resonances and correspond to excited states in the shell structure of the compound nucleus.

These resonances are the cause of the peaks in the absorption reaction rate at the fuel-water interface. Neutrons entering the fuel at an energy corresponding to a resonance in the absorption cross section will be absorbed at the edge of the fuel, leading to the rim effect seen in Figure 2.7d. Therefore, in order for KDEs to accurately capture distributions in reactor physics problems they must take into account their underlying energy dependence.

One way to do this is to base the kernel argument on the number of MFPs between the particle event and the tally point rather than the distance between the event and tally point. This makes physical sense since the distance neutrons travel between collisions is dictated by the macroscopic total cross section for a material, with the average distance being one MFP, or Σ_t^{-1}. This method effectively produces a bandwidth that is continuously adaptive in energy rather than a piece-wise approximation that would result from using unique bandwidths for each energy range. Section 2.2 details the derivation of the MFP KDE in 1-D.

2.2 1-D MFP KDE

2.2.1 1-D MFP KDE Derivation

In one dimension, the MFP KDE can be derived by starting with the unitless kernel $k(u)$ and performing a substitution: Let

$$u = \frac{\int_{X_n}^{x} \Sigma_t(x', E)\mathrm{d}x'}{h_{\mathrm{MFP}}} \quad \text{and} \quad \mathrm{d}u = \frac{\Sigma_t(x, E)}{h_{\mathrm{MFP}}}\mathrm{d}x, \tag{2.3}$$

where X_n is the location of the tally point and h_{MFP} is the bandwidth in MFP for the x direction. Then the kernel function with its normalization coefficient becomes

$$k(u)\mathrm{d}u = \frac{\Sigma_t(x, E)}{h_{\mathrm{MFP}}} k\left(\frac{\int_{X_n}^{x} \Sigma_t(x', E)\mathrm{d}x'}{h_{\mathrm{MFP}}}\right)\mathrm{d}x. \tag{2.4}$$

Using the kernel function described in Eq. (2.4) to construct a KDE yields the 1-D collision MFP KDE:

$$\phi(x) = \frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C_i} \frac{w_{i,c}}{h_{\mathrm{MFP}}} k\left(\frac{\int_{x}^{X_{i,c}} \Sigma_t(x', E)\mathrm{d}x'}{h_{\mathrm{MFP}}}\right). \tag{2.5}$$

It should be noted that the normalization coefficient for the collision MFP KDE described in Eq. (2.5) cancels out the $\Sigma_t(X_{i,c})^{-1}$ term usually present in the collision estimator for the scalar flux. The 1-D track-length MFP KDE can be made to use the same kernel argument, and is defined as

$$\hat{\phi}(x) = \frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{c_i} w_{i,c}\int_0^{d_{i,c}} \frac{\Sigma_t(x, E)}{h_{\mathrm{MFP}}} k\left(\frac{\int_{x}^{X_o+uS} \Sigma_t(x', E)\mathrm{d}x'}{h_{\mathrm{MFP}}}\right) dS. \tag{2.6}$$

The optimal bandwidth for the MFP KDE is calculated using

$$h_{l,\mathrm{MFP}} = \left\{ \frac{4}{(d+2)N_{\mathrm{samples}} \times N_{\mathrm{batches}}} \right\}^{\frac{1}{d+4}} \frac{\sigma_l}{\overline{\Sigma_t}}, \tag{2.7}$$

with

$$\overline{\Sigma_t} = \frac{\int_0^\infty \int_\Gamma \Sigma_t(\mathbf{x}, E)\phi(\mathbf{x}, E)\mathrm{d}V\mathrm{d}E}{\int_0^\infty \int_\Gamma \phi(\mathbf{x}, E)\mathrm{d}V\mathrm{d}E}, \tag{2.8}$$

where Γ denotes the volume contained by the KDE region. This is the same optimal bandwidth formula as for the distance-based KDE in Eq. (1.36) but with the added multiplicative factor of $\overline{\Sigma_t}^{-1}$. The flux-weighted average cross section in Eq. (2.8) is a quantity that is readily calculated using the collision or track-length estimators in Eqs. (1.10-1.12). For eigenvalue problems, the MFP-based bandwidth in Eq. (2.7) can be calculated during the last inactive batch for use in all active batches, or during the previous active batch for use in the current batch.

The MFP KDE is more computationally expensive than the distance-based KDE, as calculating the number of MFP between a tally point and a particle event requires conducting a ray trace for every tally point within the support of the kernel. For the track-length MFP KDE in Eq. (2.6), the integral over particle track can be prohibitively expensive to compute when attempting to contribute the score from a particle track to a tally point across a material interface. Because of this, the track-length MFP KDE does not allow particle tracks in one material to contribute to the scores of tally points outside of the current material. This is accomplished by using the boundary kernel method at material interfaces.

2.2.2 Analytic Solution in 1-D Slab Geometry

While KDEs are often used for their smoothing properties, this can be detrimental when the distribution being estimated has a discontinuous first derivative: a feature commonly found in the scalar flux distribution. To demonstrate the effectiveness of the MFP KDE in a problem with an analytic solution, a simple 1-D purely absorbing slab with a mono-directional mono-energetic beam of neutrons incident on the left face of the slab was modeled. The problem consists of 3 slabs of different materials with macroscopic total cross sections $\Sigma_{t,1} = 1$ cm^{-1}, $\Sigma_{t,2} = 100$ cm^{-1}, and $\Sigma_{t,3} = 0.5$ cm^{-1} with thicknesses 0.5 cm, 0.02 cm, and 1 cm, respectively. Since the underlying distribution is known, the optimal bandwidth from Eq. (1.36) can be computed exactly using the reference solution. For a collision density of

$f(x) = \Sigma_t(x)e^{-\int_0^x \Sigma_t(x')dx'}$, the MISE-optimal bandwidth for the collision density is

$$h = \left(\int_0^\infty \Sigma_t(x)^6 e^{-2\int_0^x \Sigma_t(x')dx'} dx \right)^{-1/5} N^{-1/5} k_2^{-2/5}. \tag{2.9}$$

For the MFP KDE, the substitution

$$u = \int_0^x \Sigma_t(x')\mathrm{d}x' \quad \text{and} \quad \mathrm{d}u = \Sigma_t(x)\mathrm{d}x, \tag{2.10}$$

is used to obtain the underlying distribution $f(u) = e^{-u}$. Using this distribution to obtain the optimal bandwidth for the MFP KDE results in

$$h_{\text{MFP}} = \frac{1}{2} N^{-1/5} k_2^{-2/5}. \tag{2.11}$$

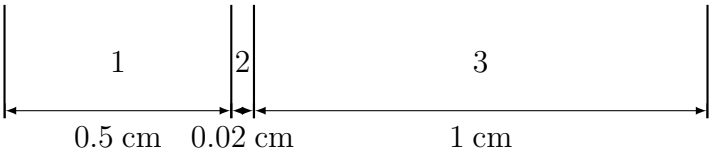
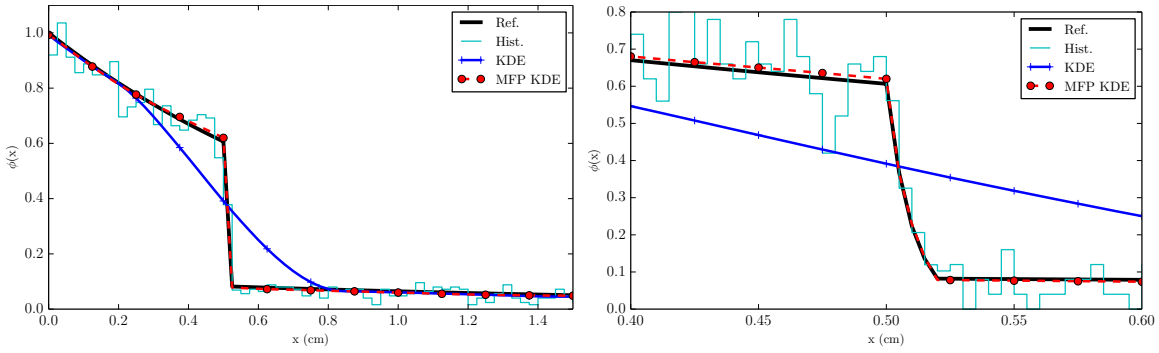


Figure 2.10: Graphic depicting geometry of analytic 1-D slab problem with material labels 1, 2, and 3 corresponding to materials with $\Sigma_{t,1} = 1$, $\Sigma_{t,2} = 100$, $\Sigma_{t,3} = 0.5$, respectively.

This problem was run with 10,000 particles with 80 histogram bins and KDE tally points placed at the center of the histogram bins from 0 cm to 2 cm. Both KDEs use the Epanechnikov kernel defined in Eq. (1.40). The results, as well as a close-up at the material interface with the number of bins and tally points increased to 400, are shown in Figure 2.11.



(a) Whole problem.

(b) Close-up of thin, strong absorber.

Figure 2.11: Comparison of KDE methods and histogram to analytical solution for a 1-D, 3-material problem with a thin, strong absorber.

Figure 2.11 shows that the histogram estimate has significant variance throughout the

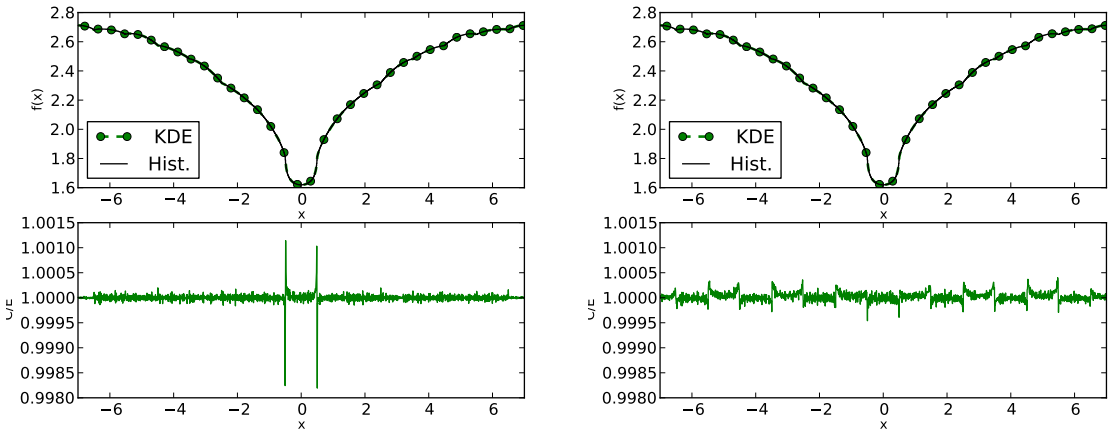
majority of the problem and the distance-based KDE shows substantial bias at the material interface with under-prediction as large as 30% prior to the strong absorber and over-predictions over 400% after the strong absorber. The distance-based KDE spreads the flux past the strong absorber, causing an under-prediction in the flux prior to the strong absorber and an over-prediction after the strong absorber. Conversely, the MFP KDE shows excellent agreement with the reference solution throughout the problem with an $L^2-\text{norm}$ of 0.046 and reduced variance compared to the histogram with an $L^2-\text{norm}$ of 0.38 . Looking closer at the material interface in Figure 2.11b shows that the MFP KDE accurately captures steep flux gradients through thin, strong absorbers and is capable of capturing the discontinuity in the derivative of the flux at material interfaces. Thus, the MFP KDE shows potential for capturing global solutions with discontinuous first derivatives with reduced variances compared to histogram tallies.

2.2.3 Verification in 1-D Reactor Physics Problems

1-D Array of Alternating Slabs with Central Absorber in Continuous Energy

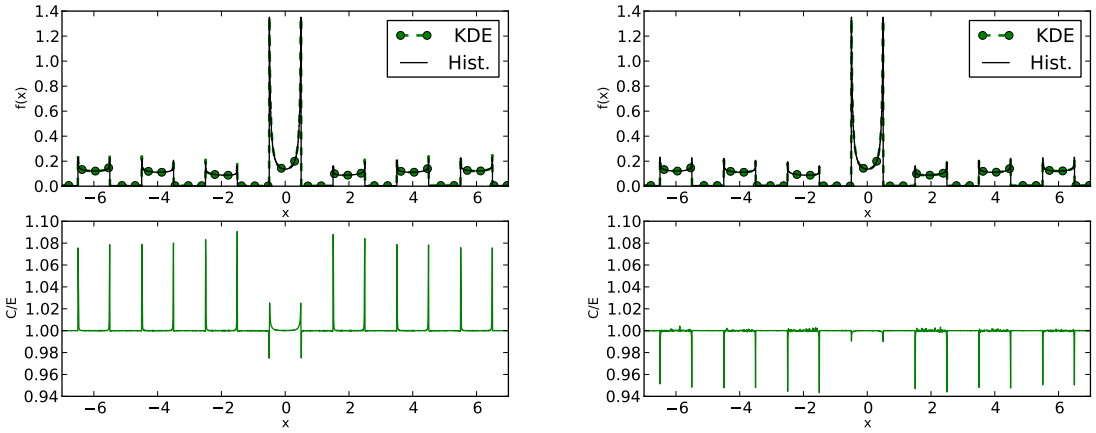
With the ability to capture steep gradients and discontinuous first derivatives in the scalar flux confirmed in the 1-D analytic solution, the MFP KDE was tested on the 1-D array of alternating slabs, detailed in Section 2.1 and depicted in Figure 2.2. The problem was run using 100,000 particles per batch with 100 inactive batches and 1,000 total batches with 1,680 bins and tally points. These run-time parameters were chosen to produce small relative uncertainties, below 0.35% for all distributions, in order to discern any bias produced in the KDE results. Even though using a large number of histories reduces the bias of the KDEs due to the dependence of the optimal bandwidth on the number of samples, these simulations can be used to compare the degree of bias one would expect from the distance-based KDE to that of the MFP KDE. Figures 2.12 and 2.13 show the comparison between the flux and absorption reaction rates, respectively, for the distance-based KDE and the reference result and the MFP KDE and the reference result. Markers are placed on the KDE plots every 50 tally points unless otherwise specified, with KDE data points connected with straight lines. The collision KDE results agree with the track-length KDE results within three standard deviations throughout most of the distributions shown, with maximum disagreements less than 4-σ. Thus, the collision KDE results have been omitted for clarity due to their close agreement with the track-length KDE results.

Figures 2.12a and 2.13a show that the distance-based KDE produces a bias in the scalar flux



(a) Distance KDE (b) MFP KDE

Figure 2.12: Flux distribution comparisons between distance-based KDE (left) and MFP KDE (right) and reference results for a 1-D array of alternating slabs with a central absorber.

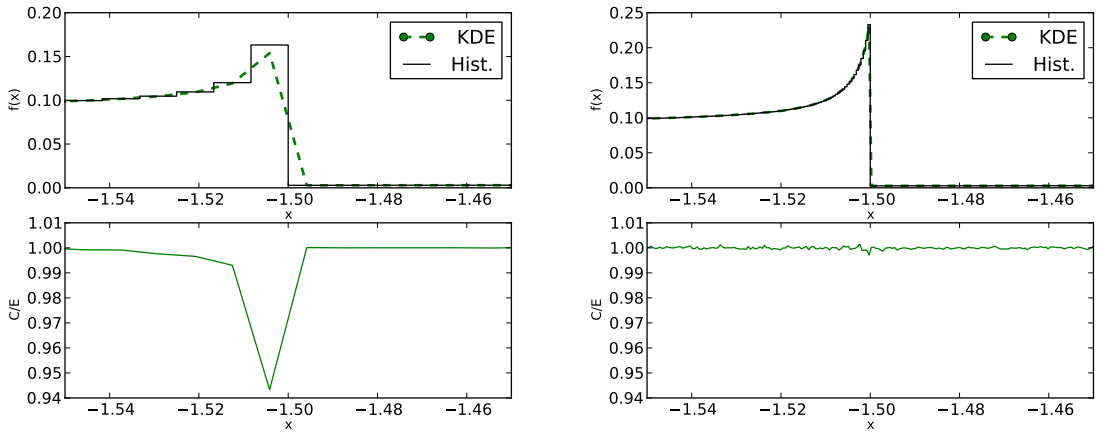


(a) Distance KDE (b) MFP KDE

Figure 2.13: Absorption density comparisons between distance-based KDE (left) and MFP KDE (right) and reference results for a 1-D array of alternating slabs with a central absorber.

and reaction rates in areas of steep gradients near material interfaces. The distance-based KDE shows a 0.17% disagreement in the flux at the water-absorber interface, corresponding to a 6-σ difference compared to the reference histogram, and a 9% disagreement in the absorption reaction rate at the fuel-water interface corresponding to a 40-σ difference. Figures 2.12b and 2.13b show that there is better agreement between the MFP KDE and the reference histogram for both distributions. The pattern in the plot of the C/E for the MFP KDE can be attributed to the use of the boundary kernel method at material interfaces as this does not appear in the collision KDE results. Even so, the maximum disagreement of 0.04% in the scalar flux distribution from the MFP KDE corresponds to a 1.5-σ difference,

indicating very strong agreement. Additionally, the MFP KDE shows better agreement than the distance-based KDE for the absorption distribution. The disagreement of up to 5% at the fuel-water interfaces is caused by the resolution of the reference histogram solution. Using the histogram result as an estimate for the value of the distribution at the center of the histogram bin is a poor approximation when the distribution has large, rapidly changing gradients. This can be seen if the resolution of the tallies is increased by a factor of four. Figure 2.14 shows a close-up of the MFP KDE and reference histogram absorption reaction rate densities at the right edge of a fuel slab with a resolution of 1,680 (left) and 26,880 (right) bins and tally points.



(a) 1,680 bins and tally points

(b) 26,880 bins and tally points

Figure 2.14: Comparison of MFP KDE and reference histogram absorption reaction rate densities with two different resolutions in the right edge of a fuel slab for the alternating slabs problem. Markers are left off of the KDE tallies for clarity.

Figure 2.14 shows that as the resolution of the histogram is increased, the reference results converge towards those obtained using the MFP KDE. Using 26,880 bins and tally points has reduced the discrepancy between the MFP KDE and reference histogram from 5.5% down to less than 0.5%. This is a direct result of comparing the value obtained using a histogram bin to the estimate of the distribution obtained at the center of the bin. As the distribution across a bin becomes more linear, the average value in a bin becomes a better estimate for the result at the center of a bin. As such, increasing the resolution of the histogram bin should increase the agreement with the KDE in regions of steep gradients as long as the KDE is accurately capturing the underlying distribution. This showcases one of the benefits of the KDE, that the resolution of the KDE tally does not affect the tally results.

2.3 2-D MFP KDE with Univariate Kernels

2.3.1 Derivation of 2-D MFP KDE with Univariate Kernels

Extending the MFP KDE to higher spatial dimensions is more complex due to the geometry dependence of the kernel function arguments. Previous applications of KDEs to Monte Carlo radiation transport simulations define the multivariate KDE using products of univariate kernels [14, 22]. This is recommended in practice for the general application of KDEs, with general multivariate kernels typically reserved for theoretical studies [26]. This section derives the multivariate MFP KDE defined using products of univariate kernels and describes the issues with this approach. Section 2.4 derives the multivariate MFP KDE using a general multivariate kernel to circumvent the issues with using a product of univariate kernels.

If the multivariate kernel is expressed as a product of univariate kernels, then the 2-D MFP KDE has the form

$$\phi(x) = \sum_{i=1}^{N} \sum_{c=1}^{C_i} \frac{w_{i,c}}{\Sigma_t(\mathbf{x})} C(x,y) k\left(\frac{\int_{\mathbf{X}_n}^{\mathbf{x}} \Sigma_t(\mathbf{x}) \mathrm{d}S \frac{\Delta X}{\Delta R}}{h_x} \right) k\left(\frac{\int_{\mathbf{X}_n}^{\mathbf{x}} \Sigma_t(\mathbf{x}) \mathrm{d}S \frac{\Delta Y}{\Delta R}}{h_y} \right), \tag{2.12}$$

where $C(x,y)$ is a normalization coefficient that depends on both x and y, $\mathbf{X}_n$ represents the location of a tally point, $\mathbf{x}$ represents the location of a collision, ΔR is the distance between $\mathbf{x}$ and $\mathbf{X}_n$, and ΔX and ΔY are the distances between $\mathbf{x}$ and $\mathbf{X}_n$ in the x- and y-dimensions, respectively. The MFP subscripts have been omitted for clarity, and will be suppressed for the remainder of the thesis. The MFP-based kernel function must satisfy the property

$$\int \int C(x,y) k\left(\frac{\int_{\mathbf{X}_n}^{\mathbf{x}} \Sigma_t(\mathbf{x}) \mathrm{d}S \frac{\Delta X}{\Delta R}}{h_x} \right) k\left(\frac{\int_{\mathbf{X}_n}^{\mathbf{x}} \Sigma_t(\mathbf{x}) \mathrm{d}S \frac{\Delta Y}{\Delta R}}{h_y} \right) \mathrm{d}y \mathrm{d}x = 1. \tag{2.13}$$

The normalization coefficient can be derived for a 2-D MFP KDE in 1-D slab geometry. For clarity, the MFP KDE is derived using a problem comprised of only two materials whose interface lies at $x = x_b$, but the final result is applicable to any number of materials in slab geometry. First, slab geometry allows the kernel function in the x-direction to be independent of y and the argument of y-dependent kernel function to be expressed as

$$\frac{\int_{\mathbf{X}_n}^{\mathbf{x}} \Sigma_t(\mathbf{x}) \mathrm{d}S \frac{\Delta Y}{\Delta R}}{h_y} = \frac{\Sigma_t(\mathbf{x})(x - x_b) + \Sigma_t(\mathbf{X}_n)(x_b - X_n)}{h_y} \frac{\Delta Y}{\Delta X}. \tag{2.14}$$

Inserting Eq. (2.14) into Eq. (2.13) and applying a substitution,

$$u = \frac{y - Y_n}{h_y}\left(\frac{\Sigma_t(\mathbf{x})\left(x - x_b\right) + \Sigma_t\left(\mathbf{X}_n\right)\left(x_b - X_n\right)}{x - X_n}\right), \tag{2.15}$$

and

$$\mathrm{d}u = \frac{1}{h_y}\left(\frac{\Sigma_t\left(\mathbf{x}\right)\left(x - x_b\right) + \Sigma_t\left(\mathbf{X}_n\right)\left(x_b - X_n\right)}{x - X_n}\right)\mathrm{d}y, \tag{2.16}$$

to the resulting equation with some rearranging yields

$$\int k\left(\frac{\int_{\mathbf{X}_n}^{\mathbf{x}} \Sigma_t(\mathbf{x}')\mathrm{d}S\frac{\Delta X}{\Delta R}}{h_x}\right)\int C(x,y)h_y\left(\frac{x - X_n}{\Sigma_t(\mathbf{x})\left(x - x_b\right) + \Sigma_t(\mathbf{X}_n)\left(x_b - X_n\right)}\right)k\left(u\right)\mathrm{d}u\mathrm{d}x. \tag{2.17}$$

In order for Eq. (2.13) to be true, then $C(x,y)$ must be

$$C(x,y) = C'(x)\frac{1}{h_y}\left(\frac{\Sigma_t(\mathbf{x})\left(x - x_b\right) + \Sigma_t(\mathbf{X}_n)\left(x_b - X_n\right)}{x - X_n}\right) = C'(x)\frac{1}{h_y}\frac{\int_{\mathbf{X}_n}^{\mathbf{x}} \Sigma_t(\mathbf{x}')\mathrm{d}S}{\Delta R}. \tag{2.18}$$

Substitution into Eq. (2.12) yields

$$\int C'(x)k\left(\frac{\int_{\mathbf{X}_n}^{\mathbf{x}} \Sigma_t(\mathbf{x}')\mathrm{d}S\frac{\Delta X}{\Delta R}}{h_x}\right)\int k\left(u\right)\mathrm{d}u\mathrm{d}x. \tag{2.19}$$

From the kernel properties in Eq. (1.18), $\int k\left(u\right)\mathrm{d}u = 1$. A similar procedure is then performed to obtain $C'(x) = \Sigma_t(\mathbf{x}){h_x}^{-1}$. Inserting these parameters for $C(x,y)$ into Eq. (2.12) yields the 2-D collision MFP KDE for slab geometry:

$$\phi(x) = \sum_{i=1}^{N}\sum_{c=1}^{C_i}\frac{w_{i,c}}{h_x}k\left(\frac{\int_{\mathbf{x}}^{\mathbf{X}_{i,c}} \Sigma_t(\mathbf{x}')\mathrm{d}S\frac{\Delta X}{\Delta R}}{h_x}\right)\frac{\int_{\mathbf{x}}^{\mathbf{X}_{i,c}} \Sigma_t(\mathbf{x}')\mathrm{d}S}{h_y\Delta R}k\left(\frac{\int_{\mathbf{x}}^{\mathbf{X_{i,c}}} \Sigma_t(\mathbf{x}')\mathrm{d}S\frac{\Delta Y}{\Delta R}}{h_y}\right). \tag{2.20}$$

The track-length MFP KDE uses the same normalization coefficient, but boundary kernels must still be used at internal material interfaces in order to prevent the calculation of prohibitively expensive integrals.

The 2-D MFP KDE described in Eq. (2.20) is only derived for 2-D tallies in 1-D slab geometries, and as such its use is limited in more complicated geometries. It can be used without approximation in a subset of 2-D geometries as long as there are only parallel, axis-aligned surfaces between any collision and tally point. Even so, the 2-D MFP KDE described here can be used in general geometries if boundary kernels are used at material interfaces, preventing particle events from scoring to tally points across material interfaces.

However, the general multivariate boundary kernel method for curved surfaces described in Section 1.3.4 does not work well with the MFP KDE since the support area of the estimator's kernels change at each collision. This means that general multivariate boundary kernel method would require the calculation of 9 integrals (3×3 coefficient matrix) and a matrix inversion for every tally point within range of a material interface and the kernel function centered at each particle event. Since the general multivariate boundary kernel method is too expensive and the boundary kernel method composed of products of univariate kernels does not handle curved surfaces well, the 2-D MFP KDE derived using a product of univariate kernels is only accurate without approximation in geometries with axis-aligned boundaries.

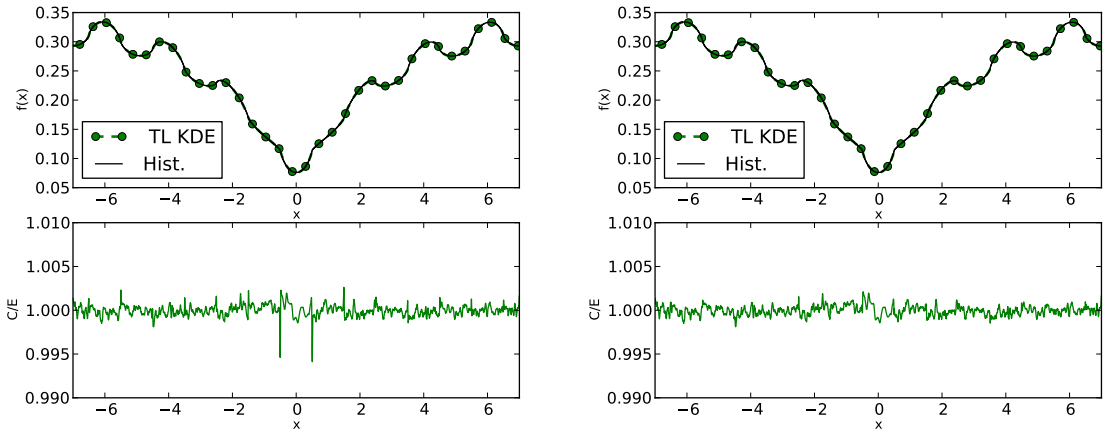
2.3.2 Verification in One-Group Reactor Physics Problems

2-D Tally in 1-D Array of Alternating Slabs

A different one-group version of the 1-D array of alternating slabs problem was used to test the 2-D collision MFP formulation in Eq. (2.20). The geometry is the same as the 1-D array of pincells from Section 2.1 depicted in Figure 2.2, with the slab extending $\pm$ 1 cm in the y-direction. The cross sections used are detailed in Table 2.2; the materials have different macroscopic total cross sections to illustrate the difference between the distance-based KDE and the MFP KDE. The reference histogram results are obtained using 1,680 bins over the range of the problem with tally points placed in the center of each histogram bin. The KDE and reference histogram results are obtained using 100,000 particles per batch with 100 inactive batches and 4,000 total batches. Flux results for the 2-D distance-based track-length KDE and 2-D track-length MFP KDEs are shown in Figure 2.15. The distance-based bandwidths for the fuel, water, and absorber range between 0.01 cm and 0.014 cm in the x dimension and 0.021 cm to 0.026 cm in the y dimension, compared to 1-D bin widths of 0.0083 cm for the reference histogram.

Table 2.2: Cross sections for one-group problems.

	Σ_t	Σ_a	Σ_s	Σ_f	ν
Moderator	0.3264	0.097920	0.228480	0	0
Fuel	0.6528	0.156672	0.496128	0.13056	2.7
Absorber	0.9792	0.965280	0.013920	0	0



(a) distance-based KDE

(b) MFP KDE

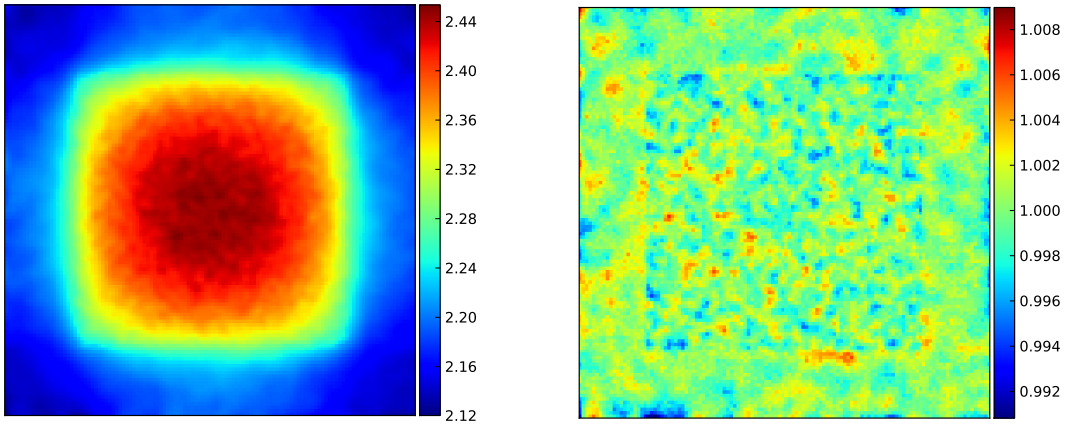
Figure 2.15: Comparison between the 2-D KDEs and reference results for a one-group 1-D array of pincells with a central absorber.

Figure 2.15 shows that the 2-D MFP KDEs are more accurate than the distance-based KDEs for a 1-D one-group problem with different macroscopic cross sections, with maximum disagreement decreasing from 0.6% to 0.2% with corresponding disagreements of 8.6 and 2.3-σ, respecitvely. The relative uncertainties of the distributions are approximately 0.025% and 0.06% for the histogram and KDE tallies, respectively, with the track-length MFP KDE having increased uncertainties, up to 0.08%, at material interfaces due to the use of the boundary kernel method there. The largest disagreements exist at the material interface between the central absorber and the water, which corresponds to a factor of 3 increase in Σ_t between the moderator and absorber.

One-Group Boxcell

A 2-D pincell problem with a square fuel region was created to demonstrate the MFP KDE in 2-D geometries. A square fuel region was used due to the necessity of using the boundary kernel method at material interfaces for the track-length MFP KDE and since the 2-D MFP KDE with univariate kernels is only derived for parallel axis-aligned material interfaces. The problem consists of a lattice pitch of 1.875 cm with a square fuel region with a side length of 1.25 cm. A 120×120 mesh was used for the histogram, with KDE tally points placed at the center of each bin. The same one-group cross sections are used as the previous problem and are detailed in Table 2.2. The simulation was run using 100,000 particles per batch with 100 inactive batches and 1,000 total batches. The flux distribution for the 2-D collision MFP KDE tally and its C/E comparison with the histogram reference solution is

shown in Figure 2.16.



(a) Flux (b) C/E

Figure 2.16: Comparison between the 2-D collision MFP KDE using univariate kernels and reference histogram flux results for the one-group boxcell problem.

Figure 2.16 shows good agreement between the 2-D univariate MFP KDE and the reference histogram, with the maximum difference between the two distributions being less than 1%. Additionally, 99.7% of the KDE tally points agree with their corresponding reference histogram bins within 3-σ, with no discernible pattern evident in the C/E distribution seen in Figure 2.16b. Thus, the 2-D MFP KDE composed of univariate kernels is capable of accurately estimating distributions in one-group problems in axis-aligned geometries. Continuous energy problems are investigated later in Section 2.4.2.

2.4 Multivariate MFP KDE

2.4.1 Derivation of Multivariate MFP KDE

The 2-D MFP KDE defined in Eq. (2.20) does not account for curved surfaces exactly. Thus, another form of the MFP KDE using a general multivariate kernel was derived to account for curved surfaces. The distance-based multivariate Epanechnikov kernel [28] is defined as

$$K(\mathbf{x}) = \begin{cases} \frac{1}{2\sqrt{5}^d} C_d^{-1}(d+2)(1-\frac{1}{5}\mathbf{x}^{\mathrm{T}}\mathbf{x}), & \mathbf{x}^{\mathrm{T}}\mathbf{x} < 5 \\ 0, & \text{otherwise} \end{cases} \tag{2.21}$$

where $\mathbf{x} = (x_1, \ldots, x_d)^{\mathrm{T}}$ and C_d is the volume of the unit sphere in d-dimensions: $C_1 = 2$, $C_2 = \pi$, $C_3 = 4\pi/3$, etc. The multivariate Epanechnikov kernel cited in literature does not

normally contain extra factors of 5, $1/5$ and $\sqrt{5}$, but they are included here so the multivariate Epanechnikov kernel is equivalent to the univariate version in 1-D. The multivariate KDE in Eq. (1.20) applied to Monte Carlo collision tallies results in the multivariate collision KDE

$$\hat{f}(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{c_i}\frac{w_{i,c}\Sigma_r(\mathbf{x},E)}{\Sigma_t(\mathbf{X}_{i,c},E)}\frac{1}{|\mathbf{H}|}K\left(\mathbf{H}^{-1}\left[\mathbf{x}-\mathbf{X}_{i,c}\right]\right), \tag{2.22}$$

where $|\mathbf{H}|$ is the determinant of the $d \times d$ symmetric positive definite bandwidth matrix. For uncorrelated data, $\mathbf{H}$ is a diagonal matrix comprised of the elements $h_1, \ldots, h_d$. While the distributions estimated in this thesis are not uncorrelated, in practice using a diagonal matrix mirrors the effect of using a product of univariate kernels and produces acceptably accurate results. The multivariate kernel must still satisfy the basic kernel properties:

$$\int K(\mathbf{u})\mathrm{d}\mathbf{u} = 1 \quad \text{and} \quad \int \mathbf{u}K(\mathbf{u})\mathrm{d}\mathbf{u} = 0. \tag{2.23}$$

In order to have the kernel argument be a function of the number of MFPs between the collision site and tally point the normalization coefficient $C(\mathbf{x})$ needs to be determined such that the kernel satisfies the properties in Eq. (2.23),

$$\int C(\mathbf{x})K\left(\mathbf{H}^{-1}\left[\mathbf{\Omega}\int_{\mathbf{X}_n}^{\mathbf{x}}\Sigma_t(\mathbf{x}')\mathrm{d}S\right]\right)\mathrm{d}\mathbf{x} = 1, \tag{2.24}$$

where $\mathbf{X}_n$ represents the location of the tally point, $\mathbf{x}$ represents the collision location, and $\mathbf{\Omega}$ is a unit vector depicting the angle between the collision site and tally point projected onto the Cartesian axes. The quantity in brackets, $\mathbf{\Omega}\int_{\mathbf{X}_n}^{\mathbf{x}}\Sigma_t(\mathbf{x}')\mathrm{d}S$, is a vector describing the number of MFPs between the tally point and the collision site in x, y, and z. Finding a coefficient $C(\mathbf{x})$ such that Eq. (2.24) is satisfied begins with examining the first kernel property in units of space:

$$\int \frac{1}{|\mathbf{H}|}K(\mathbf{H}^{-1}\left[\mathbf{x}-\mathbf{X}_n\right])\mathrm{d}\mathbf{x} = 1, \tag{2.25}$$

Changing this integral to spherical coordinates, shifting the system so it is centered about r_n, and applying a change of variables $v = r$ yields

$$\int_{-1}^{1}\int_{0}^{2\pi}\int_{0}^{\infty}\frac{1}{|\mathbf{H}|}K(\mathbf{H}^{-1}\left[v\mathbf{\Omega}\right])v^{d-1}\mathrm{d}v\mathrm{d}\theta\mathrm{d}\mu = 1. \tag{2.26}$$

Another substitution is performed to change the kernel argument to be the number of MFPs between the collision site and tally location: let

$$v = \int_0^r \Sigma_t(r'\mathbf{\Omega})\mathrm{d}r \quad \text{and} \quad \mathrm{d}v = \Sigma_t(r\mathbf{\Omega})\mathrm{d}r. \tag{2.27}$$

Inserting this substitution into Eq. (2.26) produces

$$\int_{-1}^{1}\int_0^{2\pi}\int_0^{\infty} \frac{\Sigma_t(r\mathbf{\Omega})}{|\mathbf{H}|}\left(\int_0^r \Sigma_t(r'\mathbf{\Omega})\mathrm{d}r\right)^{d-1} K\left(\mathbf{H}^{-1}\left[\mathbf{\Omega}\int_0^r \Sigma_t(r'\mathbf{\Omega})\mathrm{d}r\right]\right)\mathrm{d}r\mathrm{d}\theta\mathrm{d}\mu = 1. \tag{2.28}$$

Equation (2.28) provides insight into producing normalization coefficients for the multivariate MFP KDE. The normalization coefficient $C(\mathbf{x})$ is found by equating the integrands of Eqs. (2.24) and (2.28) in spherical coordinates, thus yielding

$$C(r,\mathbf{\Omega}) = \frac{\Sigma_t(r\mathbf{\Omega})}{|\mathbf{H}|}\frac{\left(\int_0^r \Sigma_t(r'\mathbf{\Omega})\mathrm{d}r\right)^{d-1}}{r^{d-1}}. \tag{2.29}$$

The quantity $\left(\int_0^r \Sigma_t(r'\mathbf{\Omega})\mathrm{d}r\right)/r$ is the average cross section between the tally point and collision site, and does not cause instabilities when it is evaluated near $r = 0$. Switching back to Cartesian coordinates, we get the d-dimensional normalization coefficient

$$C(\mathbf{x}) = \frac{\Sigma_t(\mathbf{x})}{|\mathbf{H}|}\frac{\left(\int_{\mathbf{x}}^{\mathbf{X}_{i,c}} \Sigma_t(\mathbf{x}')\mathrm{d}S\right)^{d-1}}{||\mathbf{X}_{i,c}-\mathbf{x}||^{d-1}}. \tag{2.30}$$

Using this normalization coefficient, the multivariate MFP KDE is defined as

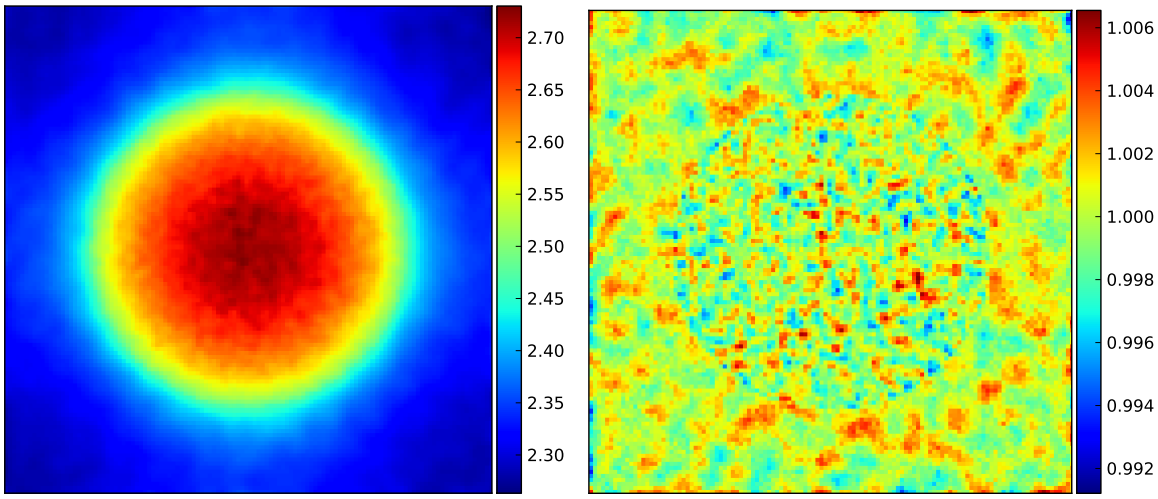
$$\hat{f}(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{c_i}\frac{w_{i,c}\Sigma_r(\mathbf{x})}{|\mathbf{H}|}\frac{\left(\int_{\mathbf{x}}^{\mathbf{X}_{i,c}} \Sigma_t(\mathbf{x}')\mathrm{d}S\right)^{d-1}}{||\mathbf{X}_{i,c}-\mathbf{x}||^{d-1}} K\left(\mathbf{H}^{-1}\left[\mathbf{\Omega}\int_{\mathbf{x}}^{\mathbf{X}_{i,c}} \Sigma_t(\mathbf{x}')\mathrm{d}S\right]\right) \tag{2.31}$$

The normalization coefficient defined for the multivariate MFP KDE in Eq. (2.31) is identical to that obtained using the MFP KDE defined using a product of univariate kernels in Eq. (2.20). The only difference between the two equations is the shape of the kernel function. In 2-D, the kernel function in Eq. (2.31) has a support region defined by an ellipse while the support region of the kernel function in Eq. (2.20) is defined by a rectangle.

2.4.2 Verification of Multivariate MFP KDE

One-Group Pincell

The multivariate MFP KDE is tested in a 2-D one-group pincell problem and is compared against a reference histogram and the 2-D MFP KDE defined using a product of univariate kernels. The pincell is comprised of a cylinder of UO_2 with radius 0.603 cm surrounded by water with a lattice pitch of 1.875 cm and reflecting boundary conditions. One-group cross sections used in this problem are shown in Table 2.2. The fuel and water are defined as separate KDE regions. The boundary kernel method is used at tally points whose kernels overlap with the external boundaries. The multivariate MFP KDE reverts to using a product of univariate kernels in order to use the boundary kernel method for tally points within one support region of the problem boundary. This does not affect the accuracy of the multivariate MFP KDE results as there are no material interfaces in that region. Flux results are collected on a structured grid with 120 histogram bins in each dimension with KDE tally points placed at the center of the histogram bins. The simulations are run with 1,000 batches, 100 inactive batches with 100,000 particles per batch. The results obtained using the 2-D multivariate MFP KDE in Eq. (2.31) are compared to the reference histogram results in Figure 2.17 with the flux distribution obtained from the multivariate MFP KDE shown on the left and the C/E values (KDE/histogram) shown on the right.



(a) Flux distribution (b) C/E

Figure 2.17: Flux distribution obtained from multivariate MFP KDE in Eq. (2.31) (left) and C/E comparison to reference histogram (right) for the one-group pincell problem.

Figure 2.17 shows that the multivariate MFP KDE accurately estimates the flux distribution

in problems containing curved surfaces, with all tally points agreeing with the reference histogram results within 1%. While the cylindrical fuel region is distinguishable from the surrounding water in the plot of the C/E in Figure 2.17b, there is no pattern in the difference between the reference histogram and KDE tally results. Furthermore, 99.87% of the KDE tally points agree with the reference histogram within 3-σ. The fuel region is discernible from the surrounding water due to the difference in bandwidths between the two areas: the water region uses a bandwidth of 0.031 cm while the fuel region uses a bandwidth of 0.015 cm and therefore the fuel region has a larger variance.

The KDE results of the 2-D pincell problem are strongly affected by the choice of optimal bandwidth formula. Using the estimated total number of samples in each region to produce optimal bandwidths via Eq. (1.53) has shown to produce undetectable bias in the 2-D problems studied thus far. To determine the effect of using the number of samples in each region from the previous batch to calculate optimal bandwidth rather than the total expected number of samples, the 2-D pincell problem was re-run using these larger optimal bandwidths for the KDE. The C/E comparison of the multivariate MFP KDE using bandwidths computed only using the number of samples in the previous batch to the reference histogram is shown in Figure 2.18.

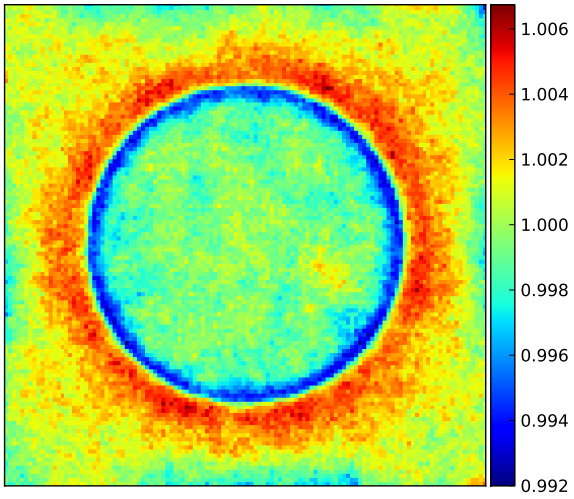


Figure 2.18: C/E comparison of flux distribution obtained from multivariate MFP KDE with optimal bandwidths computed using the number of samples in the previous batch to reference histogram for the one-group pincell problem.

Using the number of samples in the previous batch to compute optimal bandwidths for the 2-D pincell problem results in bandwidths of 0.043 cm and 0.09 cm for the fuel and water regions, respectively. These bandwidths are approximately a factor of 3 larger than those

obtained using the estimated total number of scores in each KDE region. As a result, the KDE produces a non-negligible amount of bias of approximately 1% in the estimate of the flux, seen in Figure 2.18. The larger bandwidths cause the flux at the fuel-water interface to spread out, resulting in an under-prediction of the flux within the fuel and an over-prediction in the water near the fuel-water interface. While increasing the number of particles per batch would decrease this bias, reducing the bias by a factor of 2 would require 800,000 particles per batch rather than 100,000 particles per batch due to the bias being proportional to h^2 and $h \propto N^{-1/6}$. Thus, using the number of samples in each region in the previous batch to compute optimal bandwidths rather than the estimated total number of samples in each region over the course of the simulation produces a non-negligible bias for reactor physics problems considered in this thesis.

Additionally, the multivariate MFP KDE was compared to the 2-D MFP KDE comprised of univariate kernels. Since the 2-D MFP KDE comprised of univariate kernels is not derived to handle curved surfaces, it should produce a different results near the fuel-water interface in the 2-D pincell problem compared to the multivariate MFP KDE. To accentuate the difference between the two estimators, larger optimal bandwidths computed using the number of samples in the previous batch of the eigenvalue calculation are used rather than the estimated total number of samples in the simulation. The ratio of the flux distribution obtained using the multivariate MFP KDE in Eq. (2.31) to the flux distribution obtained using the 2-D MFP KDE in Eq. (2.20) is shown in Figure 2.19.

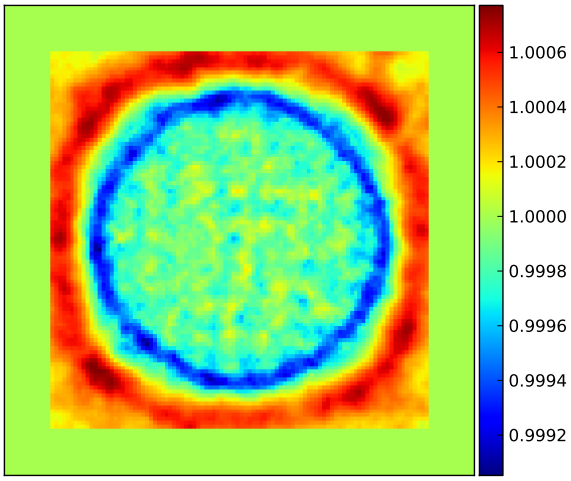


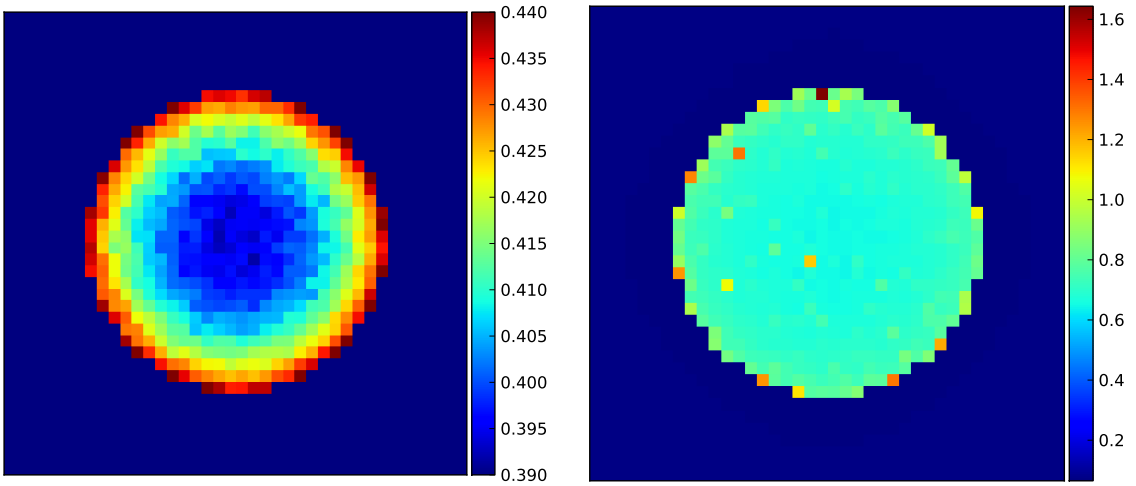
Figure 2.19: Ratio of the flux distribution obtained using the multivariate MFP KDE in Eq. (2.31) to the flux distribtuion obtained using the 2-D MFP KDE using product of univariate kernels in Eq. (2.20).

The first thing to note in Figure 2.19 is that the two estimators agree exactly at tally points within the kernel support range of the external boundary. This is due to both estimators using the boundary kernel method with the multivariate kernel expressed as a product of univariate kernels in each dimension for tally points in that area. Additionally, Figure 2.19 shows that there is little difference between the two versions of the 2-D MFP KDE, with the maximum difference being less than 0.1%. This is expected, since the normalization coefficient for the kernel function used in Eq. (2.20) is the same as that of the kernel function used in Eq. (2.31). Even so, there is a pattern in the difference between the two results, with the two estimators disagreeing by 1-σ around the material interface. The kernel functions have slightly different shapes and support volumes, with the multivariate MFP KDE in Eq. (2.31) having an ellipse-shaped support area in 2-D while the support area of the 2-D MFP KDE in Eq. (2.20) is rectangular. As such, the MFP KDE in Eq. (2.20) does not integrate to one in geometries with curved surfaces, causing this pattern of error. Even so, the close agreement between the two estimators indicates that using a product of univariate kernels for the MFP KDE does not significantly detract from the accuracy of the estimator in problems that contain curved surfaces.

2-D Pincell in Continuous Energy

With the multivariate MFP KDE verified for one-group problems in 2-D, the next step is to apply them to reactor physics problems in continuous energy. The 2-D test problem consists of a cylinder of 3% enriched UO_2 with radius 0.603 cm surrounded by water with reflecting boundary conditions and a lattice pitch of 1.86 cm. The problem was run with 100,000 particles per batch with 100 inactive batches and 1,000 total batches with KDE tally points placed on a structured 40×40 grid over the pincell. The fission and absorption reaction rate distributions for the multivariate collision MFP KDE are depicted in Figure 2.20.

Figure 2.20 shows that the multivariate MFP KDE is capable of capturing the fission distribution for the pincell problem. However, large spikes appear in the absorption distribution in Figure 2.20b. These spikes are due to the presence of resonances in the microscopic absorption cross sections, seen in Figure 2.9, combined with the bandwidth being inversely proportional to the cross section. The bandwidth shrinks for particle events at resonance energies resulting in a large contribution to the score in a localized area surrounding the particle event. A large resonance of approximately 7000 barns occurs in the U-238 microscopic total cross section at 6.67 eV. With the 2-D MFP KDE, a neutron undergoing a collision at 6.67 eV in U-238 will result in a bandwidth that is approximately 700 times smaller



(a) Fission (b) Absorption

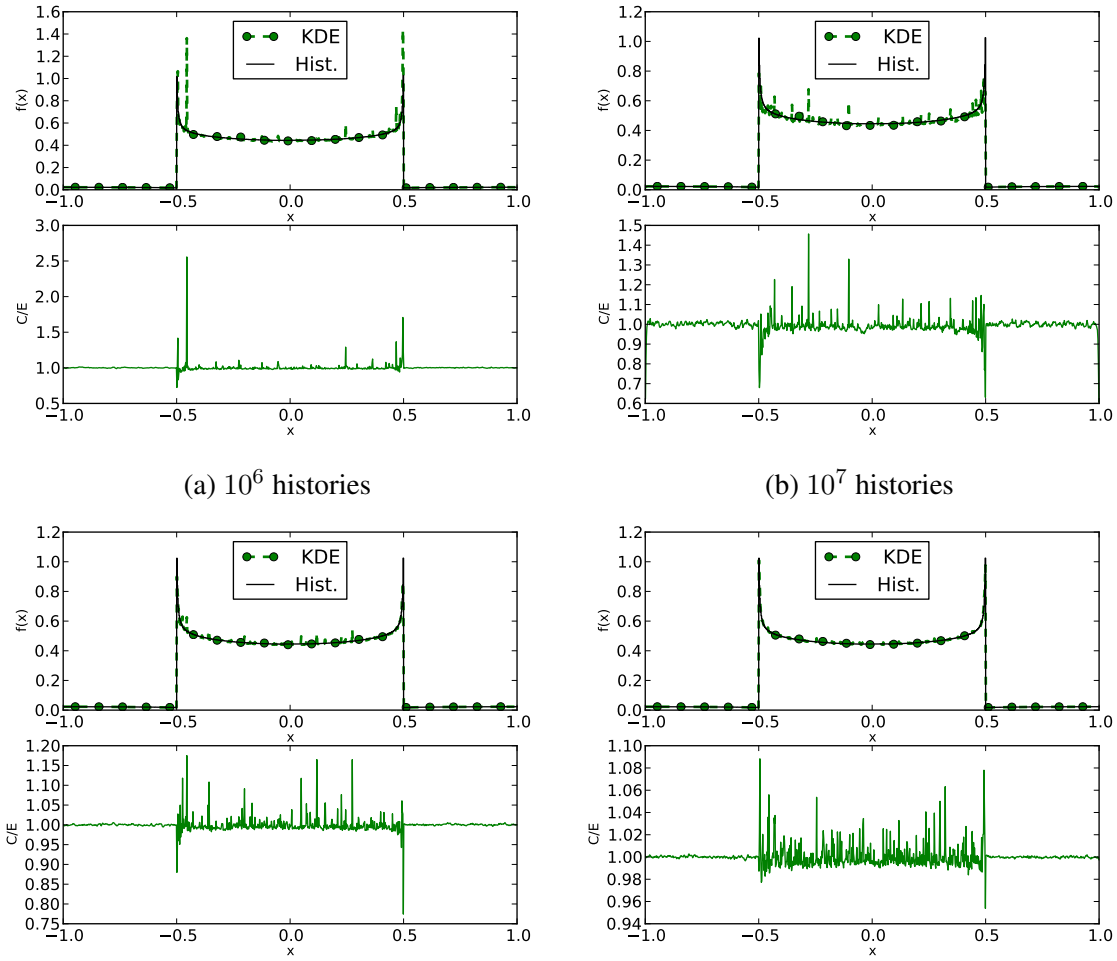
Figure 2.20: Fission and absorption distributions in a 2-D pincell problem in continuous energy obtained from the multivariate MFP KDE.

compared to a neutron undergoing a collision at 1 eV in U-238, resulting in a KDE score that is 4.9×10^5 larger compared to the score from a collision at 1 eV. Even so, spikes are not apparent in the fission distribution even though the U-235 microscopic fission cross section contains resonances. This is due to the U-235 fission resonances being approximately one order of magnitude smaller than the U-238 absorption resonances combined with the fuel being comprised of 3% U-235 and 97% U-238. This causes the macroscopic total cross section to be small enough to not cause spikes in the fission distribution when a neutron encounters a fission resonance in 3% enriched UO_2.

2-D Tally on a 1-D Slab in Continuous Energy

To better study and demonstrate the spikes, a series of simulations were done using 2-D MFP KDEs in 1-D slab geometry consisting of a 1 cm slab of 3% enriched UO_2 surrounded by 0.5 cm slabs of water with reflecting boundary conditions. For the 2-D tallies the slabs are modeled to extend $\pm$ 1 cm in the y-dimension. The simulations are run using 100,000 particles per batch with 100 inactive batches and 10, 100, 400, and 1,600 active batches. The 1-D MFP KDE was shown in Figures **??** and 2.13b to accurately estimate distributions in 1-D reactor physics problems without producing spikes. However, using the 2-D MFP KDE on this same problem produces spikes in the results. Figure 2.21 shows the absorption reaction rate densities obtained from the reference histogram and the 2-D track-length MFP KDE using products of univariate kernels as well as the C/E (KDE/histogram) values from

four simulations with different total active histories.



(a) 10^6 histories

(b) 10^7 histories

(c) 4×10^7 histories

(d) 1.6×10^9 histories

Figure 2.21: Comparison of 2-D MFP KDE and reference histogram absorption distributions for the 1-D slab representation of a fuel pin with varying amounts of total active histories.

All four simulations produce spikes in the absorption reaction rates, seen in Figure 2.21. While the magnitude of the spikes decrease with increasing number of histories, from a maximum disagreement of 250% with 250,000 active histories down to 9% disagreement with 16 million active histories, their presence when using more than 10 million histories in a simple pincell problem is disconcerting. Even though the absorption distribution is converging to the correct solution and the spikes are damping, the spikes are still present when an excessive number of histories is used for the simple 1-D pincell problem. Thus, modifications to the multivariate MFP KDE are necessary in order to suppress the formation of spikes in distributions that contain large resonances. The next section introduces additional methods and approximations in order to suppress these spikes.

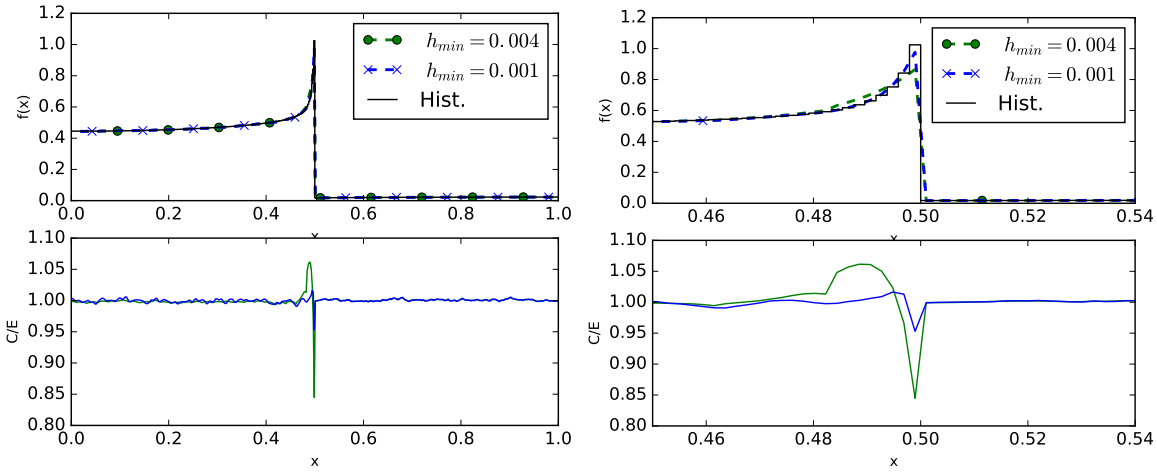
2.5 Minimum Bandwidth and Fractional MFP KDE

The presence of spikes in the multivariate MFP KDE results is due to the severe reduction in bandwidth when a collision or particle track occurs at an energy corresponding to a resonance in the cross section. Therefore, methods that either decrease the reduction in bandwidth when a neutron encounters a resonance or place a limit on the amount of reduction will suppress the spikes in the tallies. One method for limiting the reduction in bandwidth is to specify a minimum bandwidth throughout the problem. Since the MFP KDE can be thought of as an adjustment to the distance-based KDE, specifying a minimum bandwidth is equivalent to specifying a maximum cross section allowed for a region. For a specified minimum bandwidth of $h_{\min}$, the maximum cross section in a region that can be used to adjust the bandwidth is determined by

$$\Sigma_{t,\max}(\mathbf{x}) = h_l \overline{\Sigma_t} / h_{\min}. \tag{2.32}$$

In practice, this maximum cross section is used when the material between a collision and a tally point has a cross section above $\Sigma_{t,\max}(\mathbf{x})$. When this occurs, $\Sigma_{t,\max}(\mathbf{x})$ simply replaces $\Sigma_t(\mathbf{x})$ when calculating the number of MFP between two points or calculating the normalization coefficient in the MFP KDEs. Two multivariate MFP KDEs with a minimum bandwidth of 0.004 cm and 0.001 cm are used to estimate the absorption distribution of the 1-D slab problem detailed in Section 2.4.2, with results shown in Figure 2.22. The absorption distribution across the right half of the problem is shown on the left, with a close-up of the fuel-water interface at the right edge of the fuel shown on the right. The simulation was run using 100,000 particles per batch, 100 inactive batches with 500 total batches.

Figure 2.22 shows that the use of the minimum bandwidth removes the spikes seen in Figure 2.21c, albeit at the cost of introducing bias at the material interface. This is not surprising, as the minimum bandwidth reverts the MFP KDE to the distance-based KDE with a bandwidth equal to the minimum bandwidth, in this case 0.004 cm or 0.001 cm, whenever a large resonance is encountered. Thus, the inaccuracies in the distance-based KDE at the fuel water interfaces, seen in Figure 2.5, are present when using the minimum bandwidth. Reducing the minimum bandwidth results in a reduction in the bias, from 15% to 5%, at the cost of an increase in the variance, with the relative uncertainty in the fuel for a minimum bandwidth of 0.001 cm being on average a factor of 2 larger than the relative uncertainties obtained when using a minimum bandwidth of 0.004 cm. Overall, while the minimum bandwidth works at quelling spikes it does not accurately capture the steep gradients and



(a) Right half of problem

(b) Close-up of fuel-water interface

Figure 2.22: Comparison of absorption reaction rate densities from the 2-D MFP KDEs with minimum bandwidths of 0.004 cm and 0.001 cm and reference histogram distributions for the 1-D slab representation of a fuel pin. A close-up of the distributions at the fuel-water interface on the right edge of the fuel is shown on the right.

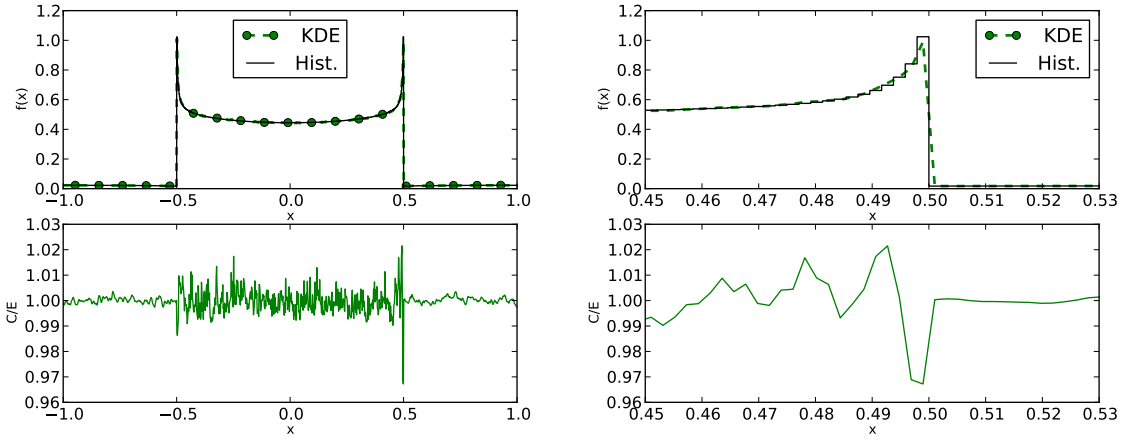
large curvature present at material interfaces in reactor physics problems.

Alternatively, the rate at which the bandwidth is reduced in space with increasing cross section can be altered in order to prevent the spikes from occurring. Since the MFP KDE was proven to work well in 1-D problems with materials that contain resonances and strong absorbers and the normalization coefficient in 1-D only has one factor of Σ_t, an alternative form of the MFP KDE is proposed whose normalization coefficient is proportional to Σ_t rather than Σ_t^2 in 2-D and Σ_t^3 in 3-D. Using fractional powers of the cross section such that the normalization coefficient of the multivariate MFP KDE has one factor of Σ_t results in the fractional MFP KDE:

$$\hat{f}(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{c_i} \frac{w_{i,c}\Sigma_r(\mathbf{x})}{|\mathbf{H}|} \frac{\left(\int_{\mathbf{x}}^{\mathbf{X}_{i,c}} \Sigma_t(\mathbf{x}')^{1/d} \mathrm{d}S\right)^{d-1}}{||\mathbf{X}_{i,c} - \mathbf{x}||^{d-1}} K\left(\mathbf{H}^{-1}\left[\mathbf{\Omega}\int_{\mathbf{x}}^{\mathbf{X}_{i,c}} \Sigma_t(\mathbf{x}')^{1/d}\mathrm{d}S\right]\right) \tag{2.33}$$

While fractional powers of the cross section do not often appear in the transport equation, it is important to note that this alternate form of adjusting the bandwidth does not affect the physics of the simulation; it is just an alternative form of defining an adaptive bandwidth. The fractional MFP KDE was used to estimate the absorption reaction rate density in the 1-D slab problem, with results shown in Figure 2.23. The distribution across the slab is shown on the left, with a close-up of the distribution at the fuel-water interface on the right edge of the slab shown on the right. The simulation was run using 100,000 particles per

batch, 100 inactive batches with 500 total batches.



(a) Fractional MFP KDE

(b) Close-up of fuel-water interface

Figure 2.23: Comparison of absorption reaction rate densities from the 2-D fractional MFP KDE and reference histogram distributions for the 1-D slab representation of a fuel pin. A close-up of the distributions at the fuel-water interface on the right edge of the fuel is shown on the right.

As seen in Figure 2.23, the fractional MFP KDE effectively suppresses the spikes seen in the 2-D MFP KDE in Figure 2.21c. However, the fractional MFP KDE disagrees with reference solution by more than 3% at the material interface, seen in the close-up of the fuel-water interface in Figure 2.23b. This disagreement is largely due to the variance in the fractional MFP KDE. The relative uncertainties of the 2-D MFP KDE with a minimum bandwidth and 2-D fractional MFP KDE for the results shown in Figures 2.22 and 2.23 are plotted in Figure 2.24.

Figure 2.24 shows that the fractional MFP KDE produces significantly larger variance throughout the problem compared to the MFP KDE with a minimum bandwidth, with the relative uncertainty being a factor of 3 larger on average and as much as 9 times larger at individual tally points. The disagreement of 3% seen at the edge of the fuel for the fractional MFP KDE still agrees with the reference solution with 3-σ due to the increase in uncertainty at the material interface. This large variance is again due to the resonance interactions; resonances still significantly reduce the bandwidth thus resulting in large variances Thus, while the fractional MFP KDE effectively handles the spikes seen in 2-D estimates of the absorption reaction rate, the large variances incurred produce additional complications. This issue is revisited later in this thesis in Section 2.9.2 when the fractional MFP KDE is compared to the cylindrical MFP KDE, another method capable of suppressing spikes.

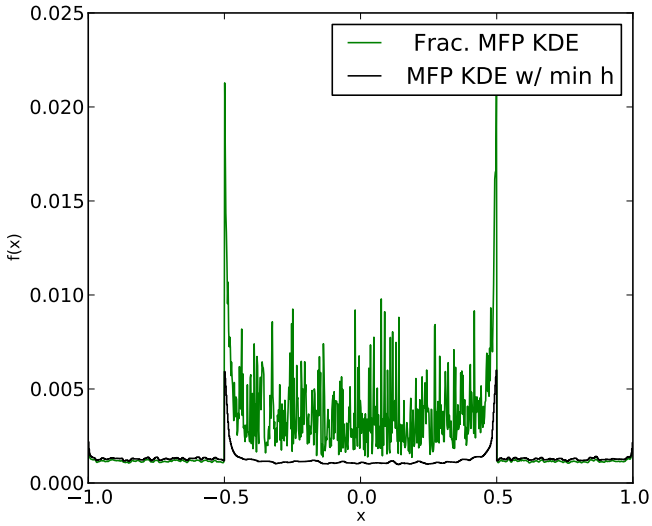


Figure 2.24: Relative uncertainties in the estimates of the absorption reaction rate densities from the 2-D fractional MFP KDE and the MFP KDE using a minimum bandwidth of $h_{\min} = 0.004$ cm for the 1-D slab representation of a fuel pin.

Overall, two solutions have been proposed to enable the multivariate MFP KDEs to capture distributions in reactor physics problems without producing spikes in the estimates. While the minimum bandwidth is successful at eliminating spikes within the majority of distributions, it incurs a bias at material interfaces where the distribution shows substantial curvature. The fractional MFP KDE also suppresses the spikes seen in the 2-D MFP KDE, however it produces a significant amount of variance due to resonance interactions, particularly at the fuel-water interface. While a combination of the fractional MFP KDE and the minimum bandwidth is likely to produce reasonably accurate distributions with reduced variance, a more robust method is desired. This issue will be revisited later in this thesis when another method for suppressing spikes is introduced in Section 2.9.

2.6 Approximate MFP KDE

Thus far the multivariate MFP KDE has two issues: spikes appearing in estimates of reaction rates that contain large resonances in their underlying cross sections, and difficulties associated with the requirement of ray tracing between tally points and particle events. This section focuses on removing the need to conduct ray tracing when using the MFP KDE. The need to conduct ray tracing precludes the use of the track-length MFP KDE as computationally expensive integrals would need to be computed in order to contribute

scores across material interfaces. This was avoided previously by using the boundary kernel method at material interfaces, preventing particle tracks from contributing to scores at tally points across material interfaces. However, the use of the boundary kernel method also requires the use of axis-aligned geometries or a series of expensive integrals after every particle track, significantly limiting the range of application of the track-length MFP KDE. Furthermore, requiring ray tracing between tally points and collision sites would potentially make the collision MFP KDE prohibitively expensive in complex geometries.

An approximation to the MFP KDE can be made to remove the need to conduct ray tracing by assuming that the material at each tally point extends through the support region of the kernel centered at the tally point. This is equivalent to performing a zeroth order Taylor expansion about $x = a$ on the cross section for all space

$$\Sigma(x, E) = \Sigma(a, E) + \frac{\partial \Sigma(x, E)}{\partial x}(x - a) + \dots \tag{2.34}$$

$$\Sigma(x, E) \approx \Sigma(a, E). \tag{2.35}$$

This will eliminate the geometry dependence of the MFP KDE, allowing the track-length MFP KDE to be used across material interfaces. This approximation is exact at internal points where the kernel function does not overlap any material interfaces. With this approximation the collision MFP KDE becomes the collision approximate MFP KDE (aMFP KDE):

$$\hat{f}(\mathbf{x}) = \frac{1}{N} \sum_{i=1}^{N} \sum_{c=1}^{c_i} \frac{w_{i,c} \Sigma_r(\mathbf{x}, E)}{\Sigma_t(\mathbf{X}_{i,c}, E)} \prod_{l=1}^{d} \frac{\Sigma_t(\mathbf{x}, E)}{h_l} k\left(\frac{\Sigma_t(\mathbf{x}, E)\,(X_{l,i,c} - x_l)}{h_l} \right). \tag{2.36}$$

This same approximation can be made to the fractional MFP KDE to obtain the fractional aMFP KDE. Note that the material cross section at the collision site, $\Sigma_t(\mathbf{X}_{i,c}, E)$, is still used in the collision estimator because the KDE still requires an estimate of the flux from the collision at $\mathbf{X}_{i,c}$.

The speedup produced by using the aMFP KDE is problem-dependent with the major factors being the geometry complexity and average number of surfaces within a kernel support region. For the 2-D pincell problem described in Section 2.4.2 a speedup of 1.5 is obtained by using the 2-D fractional aMFP KDE rather than the 2-D fractional MFP KDE. While the aMFP KDE offers several performance improvements on the MFP KDE, it also introduces additional bias at material interfaces.

The degree of this bias cannot be deduced by comparing the KDEs alone due to the existence

of their natural bias component. Ideally the bias could be determined by comparing results from the aMFP KDE to an unbiased estimate of the underlying density, like a histogram bin. However, the KDE and the histogram produce fundamentally different quantities: point-wise estimates are obtained from the KDE and volume-average estimates are obtained from the histogram. Directly comparing these two estimators would include the bias from the KDE's estimate at the point and the bias from comparing point-wise and volume-average quantities. Section 2.7 introduces a variant of the KDE that produces volume-average quantities, enabling a direct comparison between the aMFP KDE and histogram in order to determine the bias in the aMFP KDE. While a volume-average comparison is necessary to determine the bias introduced by the aMFP KDE for the majority of problems, the aMFP KDE produces a significant amount of bias in a subset of problems with geometrically thin, strong absorbers, detailed in Section 2.6.1.

2.6.1 1-D Slab With Thin, Strong Absorber

The aMFP KDE is shown to produce acceptable bias in 1-D and 2-D reactor physics problems in continuous energy in Section 2.7.1, it is capable of producing significantly biased estimates in problems that contain geometrically thin, strong absorbers. To demonstrate this issue a 1-D problem was created that consists of two slabs of 1 cm thick fuel with 1 cm thick slabs of water between them and a 0.1 cm thick strong absorber separated from the fuel by 10 cm of water. The fuel is 3% enriched UO_2 and the absorber is comprised of B_4C with a fictitious density of 120 g/cm^3. The density of the absorber was artificially increased in order to clearly demonstrate the potential issues that could arise when using the aMFP KDE. The simulation was run with 100,000 particles per batch, 1,000 batches with 60 inactive

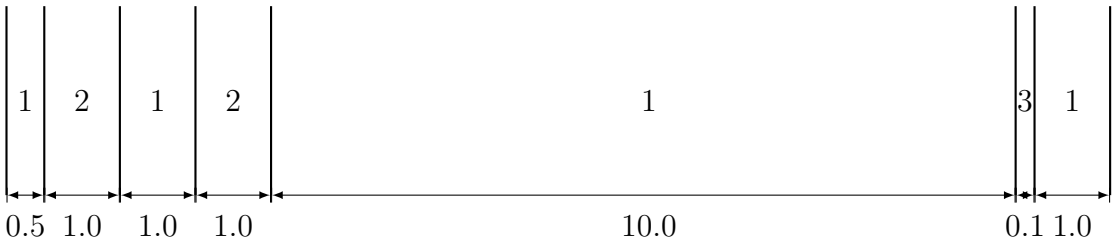


Figure 2.25: Graphic depicting problem with a thin, strong absorber. Regions labeled 1, 2, and 3 contain water, fuel, and strong absorber, respectively. The slab widths are shown underneath the geometry.

batches. Results are collected at 10,000 histogram bins placed uniformly from 13 cm to 14 cm with KDE tally points placed at the center of the bins. Results for the track-length MFP

KDE, track-length aMFP KDE, and the reference histogram are shown in Figure 2.26. The results for the collision KDE agree within 3-σ of the track-length KDE and are omitted for clarity. Close-ups of the left edge of the strong absorber and the right edge of the strong absorber are shown in Figure 2.27.

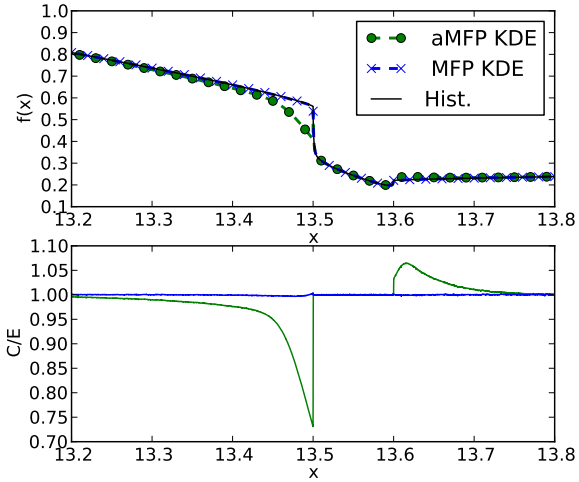
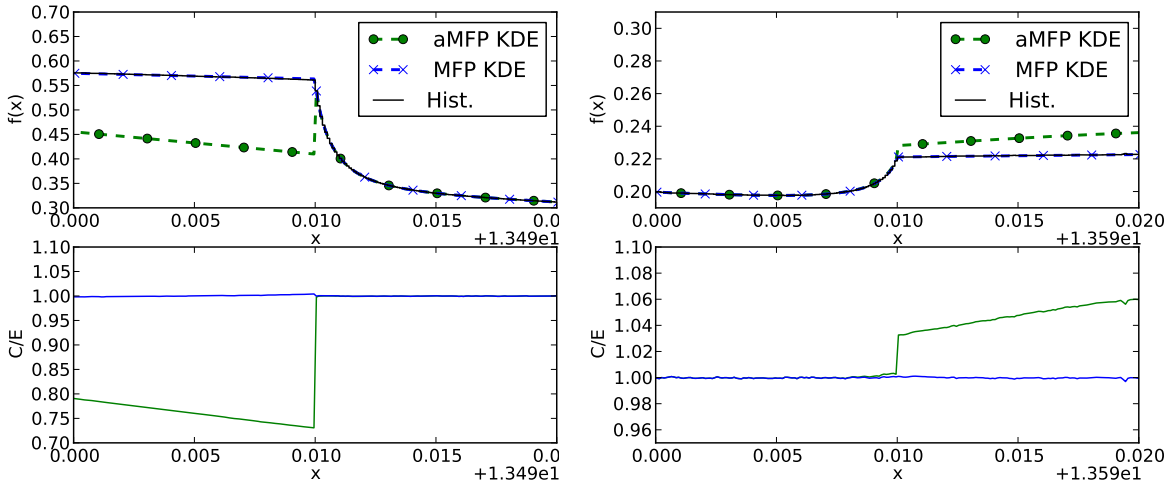


Figure 2.26: Flux comparison between aMFP KDE, MFP KDE, and reference histogram near the strong absorber of the thin, strong absorber problem.



(a) Left side of strong absorber

(b) Right side of strong absorber

Figure 2.27: Comparison between track-length MFP KDE, track-length aMFP KDE, and histogram reference solution for the thin, strong absorber problem near the material interface.

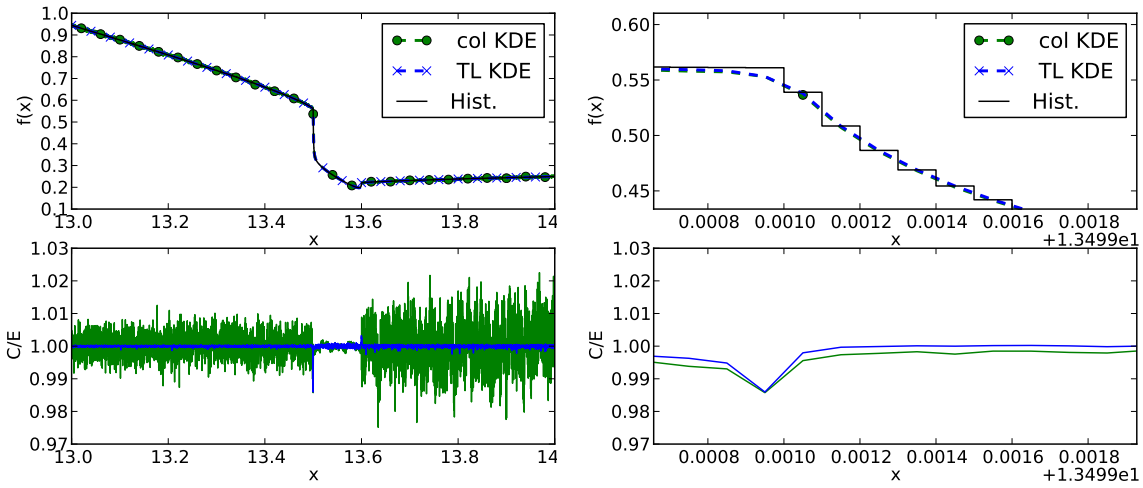
Figures 2.26 and 2.27 show that the aMFP KDE produces a substantial bias near the material interface of the strong absorber with 25% underprediction on the left edge and approximately

6% overprediction on the right edge. This bias is due to the spread of scores across the strong absorber when the aMFP KDE is used, artificially increasing the result on the other side of the strong absorber while decreasing the score locally. Since particles are born on the left side of the strong absorber and are attempting to stream through to the right side of the slab, the aMFP KDE produces a depression in the estimate on the left and an increase of the estimate on the right. The MFP KDE, on the other hand, is capable of capturing the steep gradients without any loss in accuracy with less than 1% difference between the MFP KDE and the reference histogram results.

Since the aMFP KDE accurately captures the estimate within the strong absorber, it is possible to adjust the aMFP KDE such that it is better at capturing the distribution near strong absorbers. A reduction in the bandwidth is necessary to prevent the score from spreading across the strong absorber. This can be done by using the maximum cross section in the problem to calculate the number of MFPs between the collision site and tally point in the water surrounding the strong absorber. This results in a kernel defined as

$$k(u)du = \frac{\max(\Sigma_t(x))}{h} k\left(\frac{\max(\Sigma_t(x))(X-x)}{h}\right) \mathrm{d}x. \tag{2.37}$$

Figure 2.28 shows the results from the problem with the thin, strong absorber using the kernel defined in Eq. (2.37) for the KDE.



(a) Whole problem.

(b) Close-up of left side of strong absorber.

Figure 2.28: Flux comparison between aMFP collision and track-length KDE using the kernel described in Eq. (2.37) and the reference histogram for the thin, strong absorber problem.

Figure 2.28 shows that using the maximum cross section in the problem to compute the

number of MFPs between the particle track or collision site and the tally point produces results with reduced bias near the strong absorber. The maximum bias is approximately 1.5% at the left edge of the strong absorber, pictured in Figure 2.28b. However, this reduction in bias comes at a cost of increased variance. Since the kernel described in Eq. (2.37) is used at every tally point in the problem, the relative uncertainty in the KDE solution in the water increases by a factor of 6 to 9. While it is possible to restrict the use of the alternate kernel to regions within a few bandwidths of the strong absorber, those regions will still have increased variance.

It is important to note that while the aMFP KDE fails to capture the flux at material interfaces in these 1-D heterogeneous problems with strong, thin absorbers, most problems in neutron transport do not contain materials that produce flux gradients as severe as seen in this problem. The problem with thin, strong absorber is shown as a potential limitation of the aMFP KDE and the user should be aware that it exists. Even so, this problem does appear later in this thesis in Section 4.2 when modeling a realistic 2-D pincell problem.

2.7 Volume-Average KDE

Previous methods for directly comparing KDEs to histograms for neutron transport simulations were studied by Dunn and were either a "node-to-cell" method or a "cell-to-node" method [27]. Both of these methods introduce additional error into the comparison by either extrapolating from point-wise to volume-average quantities or vice-versa. These errors can hinder attempts to quantify the bias between histogram results and those obtained using a KDE.

These conversions can be avoided in structured meshes by integrating the KDE over each mesh element to obtain a volume-average KDE. The volume-average collision KDE is obtained by integrating Eq. (1.21) over a volume. While this method is derived to determine the volume-average bias in the aMFP KDE, it can also be used as an estimator for general purposes to obtain volume-average quantities as is done in Section 3.6.3. Assuming the material remains constant in the volume, this results in

$$\hat{f}_j = \frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{c_i}\frac{w_{i,c}}{\Sigma_t(\mathbf{X}_{i,c},E)}\int_{\Gamma}\left(\Sigma_r(\mathbf{x},E)\prod_{l=1}^{d}\frac{1}{h_l}k\left(\frac{X_{l,i,c}-x_l}{h_l}\right)\right)dV, \tag{2.38}$$

where $\hat{f}_j$ is the volume-average reaction rate in mesh bin j and Γ denotes the region contained by mesh bin j. Using the Epanechnikov kernel [Eq. (1.40)] in Eq. (2.38) and integrating

over a mesh element in a 1-D Cartesian mesh results in the 1-D volume-average collision KDE for mesh bin j:

$$\hat{f}_j = \frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{c_i}\frac{w_{i,c}\Sigma_r(\mathbf{x},E)}{\Sigma_t(\mathbf{X}_{i,c},E)}\frac{1}{h}\frac{3}{4\sqrt{5}}\left(x-\frac{h}{15}\left(\frac{x-X_{i,c}}{h}\right)^3\right)\Bigg|_{x_{\min}}^{x_{\max}}, \tag{2.39}$$

where

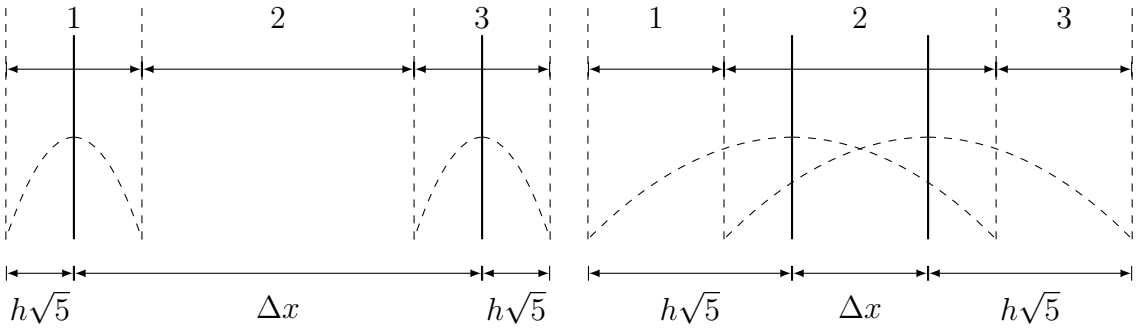
$$x_{\max} = \min(X_{i,c}+h\sqrt{5}, x_{j+1}) \quad \text{and} \quad x_{\min} = \max(X_{i,c}-h\sqrt{5}, x_j),$$

and x_{j+1} and x_j represent the upper and lower bounds of mesh bin j, respectively.

A volume-average track-length KDE can be derived by integrating the track-length KDE in Eq. (1.22) over a mesh element, resulting in

$$\hat{f}_j = \frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{c_i} w_{i,c}\int_\Gamma\int_0^{d_{i,c}}\Sigma_r(\mathbf{x},E)K_s\mathrm{d}S\mathrm{d}V. \tag{2.40}$$

For a 1-D tally, this integral is broken into three components along the particle track, depicted in Figure 2.29. The integral bounds change if the kernel's support is larger than the histogram bin width. Figure 2.29a depicts how the line integral bounds are determined if the kernel's support is smaller than the bin width while Figure 2.29b depicts the bounds for kernel support ranges that exceed the histogram bin width. The regions labeled 1, 2, and 3 in Figure 2.29 correspond to the first, second, and third components of the integral, respectively. The first component is



(a) $h\sqrt{5} > \Delta x$ (b) $h\sqrt{5} < \Delta x$

Figure 2.29: Graphic depicting how integration components are defined for a 1-D mesh with bin width Δx for the case where the kernel support is smaller than the bin width (left) and where the kernel support is greater than the bin width (right).

$$\int_{s_1}^{s_2} \int_{x_j}^{X_o+uS+h\sqrt{5}} k\left(\frac{x - X_o - uS}{h}\right) \mathrm{d}S\mathrm{d}x, \tag{2.41}$$

where

$$\begin{aligned} s_1 &= \max\left(0, \frac{x_j - X_o - h\sqrt{5}}{u}\right), \quad \text{and} \\ s_2 &= \begin{cases} \min\left(d_{i,c}, \frac{x_j - X_o + h\sqrt{5}}{u}\right), & h\sqrt{5} < \Delta x \\ \min\left(d_{i,c}, \frac{x_{j+1} - X_o - h\sqrt{5}}{u}\right), & h\sqrt{5} > \Delta x. \end{cases} \end{aligned} \tag{2.42}$$

The second component is

$$\int_{s_1}^{s_2} \int_{X_o+uS-h\sqrt{5}}^{X_o+uS+h\sqrt{5}} k(x - X_o - uS) \mathrm{d}S\mathrm{d}x, \tag{2.43}$$

where

$$\begin{aligned} s_1 &= \begin{cases} \max\left(0, \frac{x_j - X_o + h\sqrt{5}}{u}\right), & h\sqrt{5} < \Delta x \\ \max\left(0, \frac{x_{j+1} - X_o - h\sqrt{5}}{u}\right), & h\sqrt{5} > \Delta x \end{cases}, \quad \text{and} \\ s_2 &= \begin{cases} \min\left(d_{i,c}, \frac{x_{j+1} - X_o - h\sqrt{5}}{u}\right), & h\sqrt{5} < \Delta x \\ \min\left(d_{i,c}, \frac{x_j - X_o + h\sqrt{5}}{u}\right), & h\sqrt{5} > \Delta x. \end{cases} \end{aligned} \tag{2.44}$$

And the third component is

$$\int_{s_1}^{s_2} \int_{X_o+uS-h\sqrt{5}}^{x_{j+1}} k(x - X_o - uS) \mathrm{d}S\mathrm{d}x, \tag{2.45}$$

where

$$\begin{aligned} s_1 &= \begin{cases} \max\left(0, \frac{x_{j+1} - X_o - h\sqrt{5}}{u}\right), & h\sqrt{5} < \Delta x \\ \max\left(0, \frac{x_j - X_o + h\sqrt{5}}{u}\right), & h\sqrt{5} > \Delta x \end{cases}, \quad \text{and} \\ s_2 &= \min\left(d_{i,c}, \frac{x_{j+1} - X_o + h\sqrt{5}}{u}\right). \end{aligned} \tag{2.46}$$

Thus, for each particle track crossing a mesh element, up to three integrals must be computed in order to get the kernel density estimate of the volume-average flux in that mesh element.

The multivariate volume-average track-length KDE increases this complexity with up to

nine different integrals in 2-D and 27 in 3-D. A depiction of the components along a particle track for the 2-D volume-average track-length KDE is shown in Figure 2.30. Each area defining an integral component is outlined in a dashed line with the mesh element defined by a solid line. For the example particle track shown, five integrals are required to compute the volume-average track-length KDE score for this mesh element. The integral defined in Eq. (2.40) for computing the score due to tracks in the lower-left element of Figure 2.30 is

$$\int_{s_1}^{s_2}\int_{x_j}^{X_o+uS+h_x\sqrt{5}}\int_{y_j}^{Y_o+vS+h_y\sqrt{5}} k\left(\frac{x-X_o-uS}{h_x}\right)k\left(\frac{y-Y_o-vS}{h_y}\right)\mathrm{d}S\mathrm{d}x\mathrm{d}y, \quad (2.47)$$

where

$$s_1 = \max\left(0, \frac{x_j - X_o - h_x\sqrt{5}}{u}, \frac{y_j - Y_o - h_y\sqrt{5}}{v}\right), \quad \text{and}$$
$$s_2 = \min\left(d_{i,c}, \frac{x_j - X_o + h_y\sqrt{5}}{u}, \frac{y_j - Y_o + h_y\sqrt{5}}{v}\right). \quad (2.48)$$

Extension to the other areas is straightforward using the integrals defined for the 1-D volume-average track-length KDE in Eqs. (2.41)-(2.46). These volume-average KDE formulations can be applied to the aMFP KDE in order to determine an estimate of the bias introduced by the aMFP KDE.

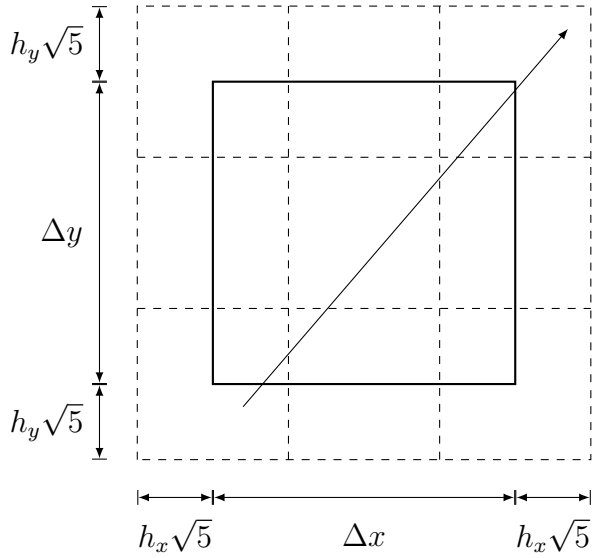


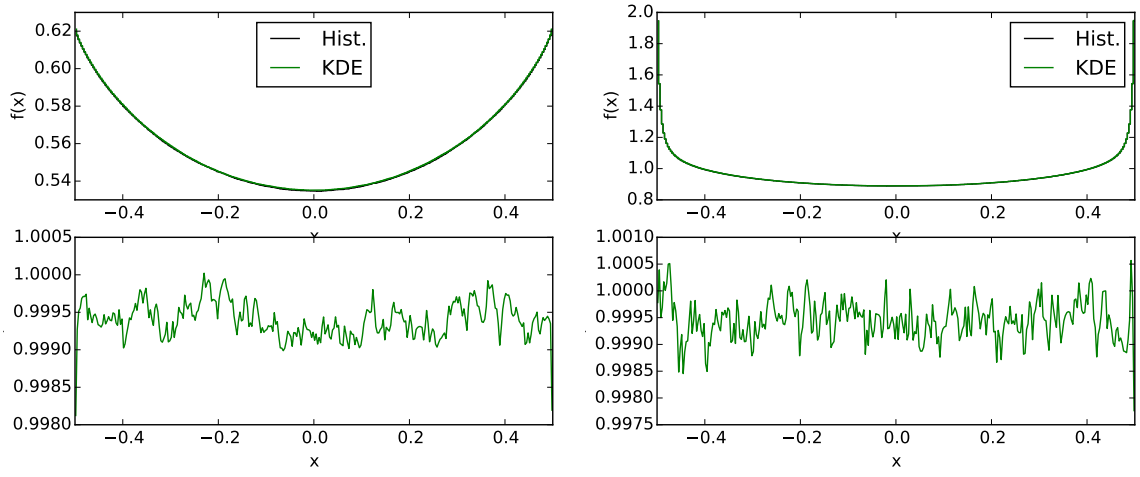
Figure 2.30: Graphic depicting the regions defining the 9 integral components of the volume-average track-length KDE for a single mesh element. An example particle track is shown crossing 5 regions.

2.7.1 Volume-Average Approximate MFP KDE KDE Results

The volume-average collision and track-length aMFP KDEs are tested in 1-D and 2-D continuous energy reactor physics problems. Volume-average KDE results are compared to a reference solution obtained using a histogram tally on the same mesh. The volume-average KDEs are conducted only in the fuel in order to avoid boundaries since the volume-average KDE has not been derived for the boundary kernel method. Axis-aligned boundaries are used in order to conduct reaction rate comparisons between the histogram tally and KDE results.

2.7.1.1 1-D Slab in Continuous Energy

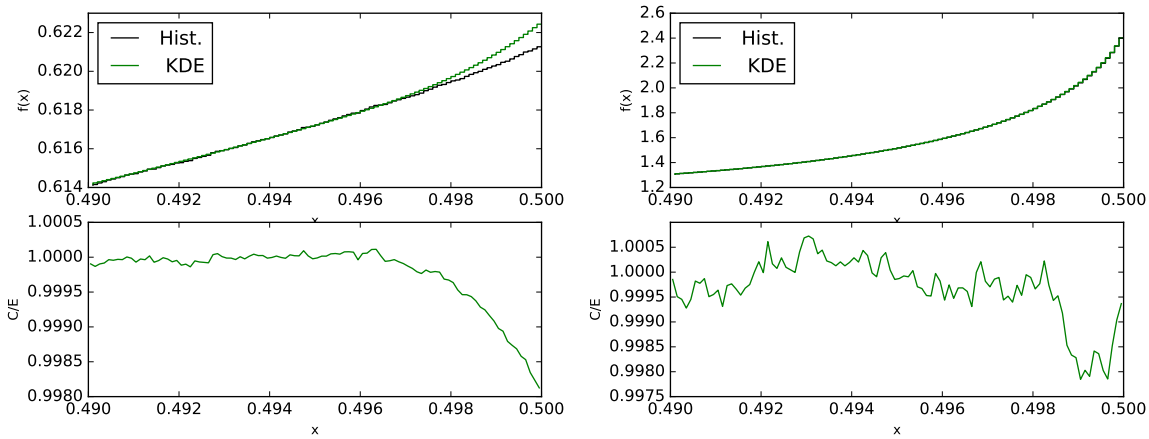
The first problem consists of a single 1 cm wide slab of 3% enriched UO_2 with 0.5 cm thick slabs of water on both sides with reflecting boundary conditions. The simulations are run using 320 bins with 100,000 particles per batch, 100 inactive batches and 500 total batches for the track-length tallies and 2,000 total batches for the collision tallies. The fission and absorption distribution comparisons between the volume-average track-length aMFP KDE and the track-length histogram tally are shown in Figure 2.31. The volume-average collision and track-length aMFP KDE agree with one another within 0.3% and within 3-σ for all distributions and have been omitted for clarity. The flux, fission, and absorption



(a) Fission (b) Absorption

Figure 2.31: Fission and absorption distribution comparisons between the volume-average track-length aMFP KDE and track-length histogram across the fuel in the 1-D pincell problem.

distributions of the volume-average aMFP KDE agree with the reference histogram within 0.2% throughout the fuel pin, with increased discrepancy at the material interface. To better discern the discrepancy at the material interface, the mesh resolution is increased and another set of simulations are run with 100 mesh bins placed from 0.49 to 0.5 cm. A close-up of the right edge of the fuel slab is shown in Figure 2.32 with reaction rate results shown on top and C/E values (histogram/KDE) shown below.



(a) Fission (b) Absorption

Figure 2.32: Fission and absorption distribution comparisons between the volume-average track-length aMFP KDE and track-length histogram with a fine mesh at the right edge of the fuel slab in the 1-D pincell problem. The difference in scale between the fission and absorption distributions (top) should be noted.

The fine tally results in Figure 2.32 show that the volume-average aMFP KDE does exhibit bias at the material boundary by over-predicting the reaction rates at the material interface. However, this error is small, with the relative bias being less than 0.25% for the fission and absorption reaction rate densities. It should be noted that the bin widths used here are much finer than those typically used in sub-pin level calculations. Typically, radial bin widths of 0.5 mm to 1 mm are used for tallies within the fuel. Thus, the bin widths in the fine mesh tallies used here are approximately a factor of 1,000 smaller than those found in typical sub-pin level reactor calculations.

2.7.1.2 2-D Boxcell

The volume-average aMFP KDEs are also tested in a 2-D problem consisting of a 1.3 cm wide square of 3% enriched UO_2 surrounded by water with a lattice pitch of 1.875 cm with the tallies conducted on a Cartesian mesh covering the fuel region with 100,000 particles

per batch, 460 batches with 60 inactive batches. Figures Of Merit (FOM) for the flux, fission, and absorption calculations for each method are given in Tables 2.3-2.5, respectively, for mesh sizes ranging from 5×5 to 160×160. The FOM for the tallies are computed using

$$\text{FOM} = \left(\frac{1}{N} \sum_{i=1}^{N} \left(\frac{\sigma_i}{\phi_i} \right)^2 T \right)^{-1}, \tag{2.49}$$

where N is the number of mesh bins or points in the tally and T is the time spent in active tallies during the simulation. To improve the FOM for the aMFP KDE tallies, a maximum bandwidth was used as described later in this thesis in Section 3.2.2. The maximum bandwidth does not allow the equivalent spatial bandwidth obtained from the MFP KDE to become larger than the distance-based bandwidth in that region. This improves the neighborhood lookup algorithm discussed in Section 3.2.1 and reduces the overall number of computations required by the KDE algorithms, thus improving the FOM for the MFP KDEs. The distance-based and MFP-based bandwidths for this problem are 0.014 cm and 0.006, respectively, and are the same in both dimensions. The fission distributions for the collision histogram and volume-average aMFP KDE are shown in Figure 2.33 for the 160×160 mesh.

Table 2.3: Flux FOM for the volume-average collision and track-length aMFP KDE and histogram for a range of mesh sizes.

Mesh Size	Hist. Collision	KDE Collision	Ratio	Hist. TL	KDE TL	Ratio
5	5.07E+04	4.36E+04	0.86	2.15E+05	9.95E+04	0.46
10	1.35E+04	1.33E+04	0.98	1.16E+05	4.08E+04	0.35
20	3.50E+03	4.60E+03	1.31	5.66E+04	1.36E+04	0.24
40	8.90E+02	2.20E+03	2.47	2.42E+04	3.74E+03	0.15
80	2.24E+02	1.35E+03	6.04	9.06E+03	1.06E+03	0.12
160	5.58E+01	7.52E+02	13.49	2.69E+03	2.75E+02	0.10

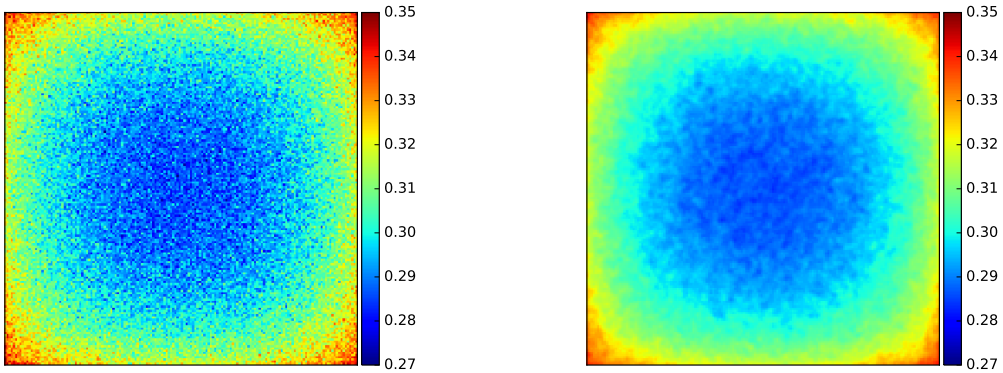
Tables 2.3-2.5 show that the FOM for calculating flux, fission, and absorption reaction rates using the volume-average collision aMFP KDE are better than that of the collision histogram for mesh sizes greater than 10×10, with relative performance of the volume-average aMFP KDE to the histogram increasing with mesh size. The 20×20 mesh size corresponds to 6.7 bandwidths, thus indicating that the volume-average collision aMFP KDE performs well compared to the histogram when the length of a mesh bin is within one order of magnitude of the bandwidth or smaller than the bandwidth. The effectiveness of the KDE can be visualized in Figure 2.33, with the volume-average collision aMFP KDE producing

Table 2.4: Fission rate FOM for the volume-average collision and track-length aMFP KDE and histogram for a range of mesh sizes.

Mesh Size	Hist. Collision	KDE Collision	Ratio	Hist. TL	KDE TL	Ratio
5	1.42E+04	1.21E+04	0.86	3.63E+04	1.68E+04	0.46
10	3.90E+03	3.55E+03	0.91	1.94E+04	6.66E+03	0.34
20	1.02E+03	1.10E+03	1.08	9.34E+03	2.11E+03	0.23
40	2.61E+02	4.04E+02	1.55	3.98E+03	5.16E+02	0.13
80	6.54E+01	1.88E+02	2.87	1.49E+03	1.25E+02	0.08
160	1.63E+01	9.00E+01	5.51	4.43E+02	2.96E+01	0.07

Table 2.5: Absorption rate FOM for the volume-average collision and track-length aMFP KDE and histogram for a range of mesh sizes.

Mesh Size	Hist. Collision	KDE Collision	Ratio	Hist. TL	KDE TL	Ratio
5	1.83E+04	1.53E+04	0.84	2.91E+04	1.33E+04	0.46
10	4.96E+03	4.31E+03	0.87	1.12E+04	3.75E+03	0.34
20	1.30E+03	1.30E+03	1.00	3.91E+03	8.35E+02	0.21
40	3.35E+02	4.34E+02	1.29	1.27E+03	1.39E+02	0.11
80	8.49E+01	1.54E+02	1.81	3.87E+02	2.07E+01	0.05
160	2.13E+01	4.28E+01	2.01	1.00E+02	2.65E+00	0.03

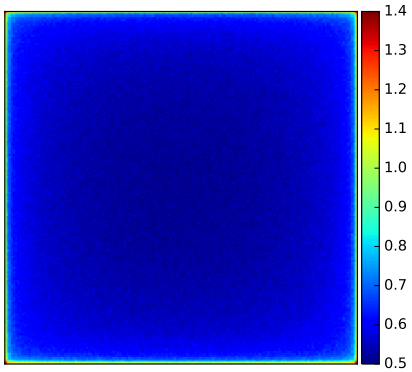


(a) Collision histogram

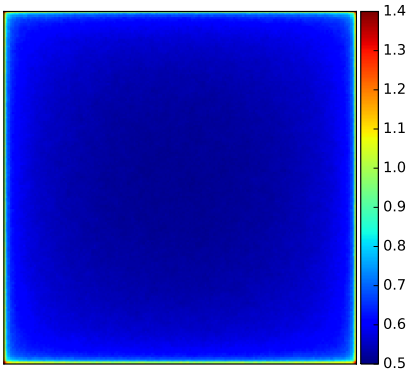
(b) Volume-average collision KDE

Figure 2.33: Fission reaction rate distribution for collision estimators for the 160×160 mesh.

smoother fission distributions with reduced variance compared to the collision histogram. However, the FOM for the volume-average track-length aMFP KDE is worse than the track-length histogram for all mesh sizes for every distribution, with performance relative to the track-length histogram decreasing with increasing mesh size. This is not surprising since several integrals must be computed and several distance-to-boundary routines must be performed for each mesh element that falls within the support range of the particle track.



(a) Collision histogram



(b) Volume-average collision KDE

Figure 2.34: Absorption reaction rate distribution for collision estimators for the 160x160 mesh.

The added benefit of producing smooth distributions and reducing the variance among the mesh elements is negated by the large computational cost associated with integrating the kernel function over the structured mesh.

Furthermore, Tables 2.3-2.5 show that the volume-average aMFP KDE relative performance is worse for estimating fission and absorption reaction rate distributions compared to estimating flux distributions. This decrease in performance is due to the behavior of the bandwidth in the aMFP KDE. The aMFP KDE can be thought of as a distance-based KDE with a bandwidth that is inversely proportional to the total macroscopic cross section. Interactions at resonances contract the bandwidth such that integrating the aMFP kernel function over all volume effectively reduces the volume-average aMFP KDE into histogram tallies. This hinders the effectiveness of the aMFP KDE in tallies whose distributions are dominated by interactions with large cross sections, like the absorption reaction rate in fuel as seen in Figure 2.34. The absorption reaction rate is highly peaked at the fuel-water interface due to resonance interactions and these interactions do not benefit from the smoothing effects of the KDE. Since the KDE has to do additional work to compute scores compared to the histogram tally, this results in a lower FOM for distributions like the absorption and fission reaction rate compared to the flux distribution. This conversion to histogram tallies is also the reason spikes do not appear in the estimates of the absorption reaction rate, seen in Figure 2.34b, as the maximum score that can be contributed to any mesh bin is one for the collision estimator and the particle track-length for the track-length estimator. However, using the volume-average KDE is not a suitable solution to the problem of suppressing spikes for general problems as the histogram bins need to conform to the underlying problem geometry as well as the principal axes on which the KDEs are based.

To determine the bias introduced by the aMFP KDE, the boxcell problem was re-run with 4,800 batches with 100,000 particles per batch with 60 inactive batches. The collision aMFP KDE results are compared to results obtained from a track-length histogram tally using the same parameters. The results indicate that the relative difference between the flux and fission distributions is less than 1% for all mesh bins, and less than 2% for the absorption distribution. Thus, the bias introduced by the aMFP KDE in problems without thin, strong absorbers is sufficiently low to be applied to reactor pincell problems.

2.8 Cylindrical KDE

2.8.1 Motivation

The difficulties associated with accurately estimating reaction rates in reactor physics problems using the MFP KDE, seen in Section 2.4.2, led to the development of an alternate KDE. Using the number of MFP as the kernel argument embeds material information into the estimator, however the MFP KDE is also expected to capture distributions that have a large dependence on geometry as well. As such, a multivariate kernel that takes into account both geometry and material information should produce a better estimate than the multivariate MFP KDE. Since cylindrical geometries are often encountered in nuclear reactors, and the variation in the neutron flux has a strong radial dependence, it is desirable to have a kernel that is designed to capture distributions within a cylinder. This led to the development of a cylindrical KDE whose 2-D support area is wedge-shaped. A depiction of a wedge-shaped kernel centered about a tally point placed on a cylinder is shown in Figure 2.35. However, the kernel must still adapt to the material cross sections in order to accurately capture effects from resonances and strongly absorbing materials without introducing spikes into the results. Applying the MFP KDE formulation only in the radial component of the cylindrical KDE accomplishes this goal and leads to the development of the cylindrical MFP KDE detailed in Section 2.9.

2.8.2 Derivation of Cylindrical KDE

The cylindrical KDE is derived using a multivariate KDE defined using a product of univariate kernels. Rather than using kernels whose arguments are based on the distance away from the tally point in the x and y dimension, we seek a kernel whose arguments are

based on the distance between the tally point and the collision site in cylindrical coordinates (r, θ). This leads to the cylindrical kernel function:

$$K(r,\theta) = C(r,\theta)k\left(\frac{r-r'}{h_r}\right)k\left(\frac{\theta-\theta'}{h_\theta}\right), \tag{2.50}$$

where $\theta - \theta'$ is the difference between the azimuthal position of the tally point and collision site in cylindrical coordinates, $r - r'$ is the difference between the radii of the tally point and collision site in cylindrical coordinates, h_r is the radial bandwidth, h_θ is the azimuthal bandwidth, and $C(r, \theta)$ is the normalization coefficient for the kernel function. The multivariate cylindrical kernel function can be modified to include the z dimension by adding another univariate kernel: $k((z - z')/h_z)$. A depiction of the cylindrical kernel centered at a tally point $\mathbf{r}$ on a quarter section of a cylindrical fuel pin is shown in Figure 2.35 with regions 1, 2, and 3 indicating fuel, cladding, and water, respectively. The normalization coefficient is

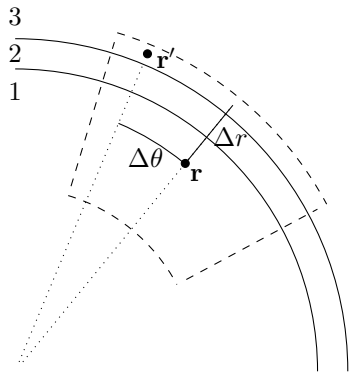


Figure 2.35: Depiction of a cylindrical kernel centered at tally point $\mathbf{r}$ on a quarter-view of a pincell with a collision site at $\mathbf{r}'$. The distance between the tally point and collision site in cylindrical coordinates is marked by Δr and $\Delta\theta$. Material regions marked 1, 2, and 3 indicate fuel, cladding, and water, respectively.

defined by the kernel property

$$\int\int C(r,\theta)k\left(\frac{r-r'}{h_r}\right)k\left(\frac{\theta-\theta'}{h_\theta}\right)r'\mathrm{d}r'\mathrm{d}\theta' = 1. \tag{2.51}$$

Using the substitution $u = \frac{\theta-\theta'}{h_\theta}$ and $v = \frac{r-r'}{h_r}$ yields

$$\int\int C(r,\theta)h_r h_\theta k\left(v\right)k\left(u\right)\left(r - vh_r\right)\mathrm{d}v\mathrm{d}u = 1. \tag{2.52}$$

Using the kernel properties in Eq. (1.18), this reduces to

$$\int \int C(r,\theta) r h_r h_\theta k\left(v\right) k\left(u\right) \mathrm{d}v\mathrm{d}u = 1. \tag{2.53}$$

Thus, the normalization coefficient

$$C(r,\theta) = (r h_r h_\theta)^{-1}, \tag{2.54}$$

allows Eq. (2.53) to hold.

This cylindrical kernel function simply replaces the kernel function in Eq. (2.31) to create the cylindrical collision KDE:

$$\hat{f}(\mathbf{x}) = \frac{1}{N} \sum_{i=1}^{N} \sum_{c=1}^{c_i} \frac{w_{i,c} \Sigma_r(\mathbf{x}, E)}{\Sigma_t(\mathbf{X}_{i,c}, E)} \frac{1}{r h_r h_\theta} k\left(\frac{r - r_{i,c}}{h_r}\right) k\left(\frac{\theta - \theta_{i,c}}{h_\theta}\right) \tag{2.55}$$

The cylindrical track-length KDE is created the same way by replacing the kernel function in the track-length KDE in Eq. (1.22) with the cylindrical kernel function in Eq. (2.50).

2.8.3 Bias and Variance of Cylindrical KDE

The approximate bias and variance in the cylindrical KDE can be derived in a similar fashion to the approximate bias and variance of the univariate kernel function in Section 1.3.2. The bias is defined as

$$\begin{aligned} \mathrm{bias}_h\left(\mathbf{x}\right) &= E\left[\hat{f}(\mathbf{x})\right] - f(\mathbf{x}) \\ &= \int \int \frac{1}{h_r h_\theta r} k\left(\frac{r - r'}{h_r}\right) k\left(\frac{\theta - \theta'}{h_\theta}\right) f(r', \theta') r' \mathrm{d}r' \mathrm{d}\theta' - f(r,\theta). \end{aligned} \tag{2.56}$$

Using the substitution $u = \frac{\theta - \theta'}{h_\theta}$ and $v = \frac{r - r'}{h_r}$ yields

$$\mathrm{bias}_h\left(\mathbf{x}\right) = \int \int \frac{1}{r} k\left(v\right) k\left(u\right) f(r - v h_r, \theta - u h_\theta)(r - v h_r) \mathrm{d}v\mathrm{d}u - f(r,\theta). \tag{2.57}$$

Since $\int \int \frac{1}{r} k\left(v\right) k\left(u\right) (r - v h_r) \mathrm{d}v\mathrm{d}u = 1$, the bias becomes

$$\mathrm{bias}_h\left(\mathbf{x}\right) = \int \int \frac{1}{r} k\left(v\right) k\left(u\right) \left[f(r - v h_r, \theta - u h_\theta) - f(r,\theta)\right] (r - v h_r) \mathrm{d}v\mathrm{d}u. \tag{2.58}$$

Assuming separation of variables, $f(r,\theta) = g(r)w(\theta)$, and applying a Taylor expansion on $w(\theta - uh_\theta)$ about $w(\theta)$ yields

$$\begin{aligned}\text{bias}_h\left(\mathbf{x}\right) = \int\int \frac{1}{r}k\left(v\right)k\left(u\right)\Bigg[g(r-vh_r)\left(w(\theta) - uh_\theta w'(\theta) + \frac{1}{2}u^2h_\theta^2 w''(\theta) + \dots\right) \\ - g(r)w(\theta)\Bigg]\left(r - vh_r\right)\mathrm{d}v\mathrm{d}u. \end{aligned} \tag{2.59}$$

Applying $\int uk(u)\mathrm{d}u = 0$ and performing the same Taylor expansion on $g(r - vh_r)$ yields

$$\begin{aligned}\text{bias}_h\left(\mathbf{x}\right) &= \int\int \frac{1}{r}k\left(v\right)k\left(u\right)\Bigg[\left(g(r) - vh_r g'(r) + \frac{1}{2}v^2h_r^2 g''(r) + \dots\right)\times \\ &\qquad \left(w(\theta) + \frac{1}{2}u^2h_\theta^2 w''(\theta) + \dots\right) - g(r)w(\theta)\Bigg]\left(r - vh_r\right)\mathrm{d}v\mathrm{d}u \\ &\approx \int\int \frac{1}{r}k\left(v\right)k\left(u\right)\Bigg[\left(-vh_r g'(r) + \frac{1}{2}v^2h_r^2 g''(r)\right)\left(w(\theta) + \frac{1}{2}u^2h_\theta^2 w''(\theta)\right) \\ &\qquad + \frac{1}{2}u^2h_\theta^2 w''(\theta)g(r)\Bigg]\left(r - vh_r\right)\mathrm{d}v\mathrm{d}u. \end{aligned} \tag{2.60}$$

Applying the kernel properties in Eq. (1.18) results in

$$\text{bias}_h\left(\mathbf{x}\right) \approx \left(-\frac{1}{r}k_2h_r^2 g'(r) + \frac{1}{2}k_2h_r^2 g''(r)\right)\left(w(\theta) + \frac{1}{2}k_2h_\theta^2 w''(\theta)\right) + \frac{1}{2}k_2h_\theta^2 w''(\theta)g(r). \tag{2.61}$$

Assuming $h_r \propto h_\theta$ and $O(h_\theta) \approx O(h_r)$ results in an approximate bias of

$$\text{bias}_h\left(\mathbf{x}\right) \approx -\frac{1}{r}k_2h_r^2 g'(r)w(\theta) + \frac{1}{2}k_2h_r^2 g''(r)w(\theta) + \frac{1}{2}k_2h_\theta^2 w''(\theta)g(r). \tag{2.62}$$

Thus, the cylindrical KDE has the same $O(h^2)$ bias as the standard multivariate KDE.

The variance is defined as

$$\begin{aligned}\text{var}\left[\hat{f}(\mathbf{x})\right] &= E\left[\hat{f}(\mathbf{x})^2\right] - E\left[\hat{f}(\mathbf{x})\right]^2 \\ &= \frac{1}{n}\int\int\left(\frac{1}{r^2h_r^2h_\theta^2}k\left(\frac{r-r'}{h_r}\right)^2 k\left(\frac{\theta-\theta'}{h_\theta}\right)^2 f(r',\theta')r'\mathrm{d}r'\mathrm{d}\theta'\right) - \\ &\qquad \frac{1}{n}\left(f(r,\theta) + O(h^2)\right)^2. \end{aligned} \tag{2.63}$$

Applying the substitution $u = \frac{\theta - \theta'}{h_\theta}$ and $v = \frac{r - r'}{h_r}$ yields

$$\begin{aligned}\text{var}\left[\hat{f}(\mathbf{x})\right] =& \frac{1}{n}\int\int\left(\frac{1}{r^2 h_r h_\theta} k\left(v\right)^2 k\left(u\right)^2 f(r - vh_r, \theta - uh_\theta)(r - vh_r)\mathrm{d}v\mathrm{d}u\right) - \\ & \frac{1}{n}\left(f(r,\theta) + O(h^2)\right)^2. \end{aligned} \tag{2.64}$$

Assuming separation of variables, $f(r,\theta) = g(r)w(\theta)$, and applying a Taylor expansion on $w(\theta - uh_\theta)$ about $w(\theta)$ and $g(r - vh_r)$ about $g(r)$ yields

$$\begin{aligned}\text{var}\left[\hat{f}(\mathbf{x})\right] = \frac{1}{n}\int\int\Bigg(\frac{1}{r^2 h_r h_\theta} k\left(v\right)^2 k\left(u\right)^2 \left[g(r)w(\theta) - uh_\theta g(r)w'(\theta) - \right. \\ \left. vh_r g'(r)w(\theta) + \ldots\right](r - vh_r)\mathrm{d}v\mathrm{d}u\Bigg) + O\left(n^{-1}\right). \end{aligned} \tag{2.65}$$

Assuming that h_θ and h_r are small and n is large gives an approximation to the variance

$$\text{var}\left[\hat{f}(\mathbf{x})\right] \approx \frac{1}{nh_r h_\theta r}\int\int k\left(v\right)^2 k\left(u\right)^2 f(r,\theta)\mathrm{d}v\mathrm{d}u. \tag{2.66}$$

Thus, the variance of the cylindrical KDE follows the same asymptotic behavior as the usual multivariate KDE with the variance being inversely proportional to the product of the bandwidths in each dimension.

2.9 Cylindrical MFP KDE

2.9.1 Derivation

The MFP KDE can also be applied to the cylindrical KDE. The Cartesian multivariate MFP KDE results in Section 2.4.2 showed that normalization coefficients containing more than one factor of the macroscopic total cross section produced spikes in estimates of the absorption reaction rate density. The fractional MFP KDE was developed in Section 2.5 to mitigate spikes by obtaining a normalization coefficient proportional to $\Sigma_t(\mathbf{x}, E)$ through the use of fractional powers of the cross section in the arguments of the kernel functions. A normalization coefficient with the same behavior as the fractional MFP KDE can be obtained by the cylindrical MFP KDE without using fractional powers of the cross section by using a different approach. Since the distributions of interest in reactor physics problems studied in this thesis vary slowly in the azimuthal dimension relative to the radial dimension, the MFP

KDE is only applied to radial dimension. This yields an estimator with a normalization coefficient proportional to $\Sigma_t(\mathbf{x}, E)$ rather than $\Sigma_t(\mathbf{x}, E)^2$ and results in a suppression of the spikes seen in the 2-D multivariate MFP KDE results.

Applying the MFP KDE to the radial dimension results in the cylindrical MFP kernel function

$$K(r,\theta) = C(r,\theta)k\left(\frac{\int_{r'}^{r}\Sigma_t(r'')dr''}{h_{r,\mathrm{MFP}}}\right)k\left(\frac{\theta-\theta'}{h_\theta}\right). \tag{2.67}$$

The term $\int_{r'}^{r}\Sigma_t(r'')dr''$ in Eq. (2.67) represents the number of MFP between the collision site and tally point in the radial dimension only - not along the ray traced between the two points. For example, $\int_{r'}^{r}\Sigma_t(r'')dr''$ is the number of MFP along the line marked Δr in Figure 2.35. The normalization coefficient, $C(r,\theta)$, for the cylindrical MFP KDE is derived via the kernel property

$$\int\int C(r,\theta)k\left(\frac{\int_{r}^{r'}\Sigma_t(r'')\mathrm{d}r''}{h_{r,\mathrm{MFP}}}\right)k\left(\frac{\theta-\theta'}{h_\theta}\right)r'\mathrm{d}r'\mathrm{d}\theta' = 1. \tag{2.68}$$

The azimuthal dimension is handled in the same fashion as in Eq. (2.56). However, applying the substitution

$$v = h_{r,\mathrm{MFP}}^{-1}\int_{r}^{r'}\Sigma_t(r'')\mathrm{d}r'' \quad \text{and} \quad \mathrm{d}v = h_{r,\mathrm{MFP}}^{-1}\Sigma_t(r')\mathrm{d}r, \tag{2.69}$$

to Eq. (2.68) to yield

$$\int\int C(r,\theta)k\left(v\right)k\left(\frac{\theta-\theta'}{h_\theta}\right)r'\frac{h_{r,\mathrm{MFP}}}{\Sigma_t(r')}\mathrm{d}v\mathrm{d}\theta' = 1, \tag{2.70}$$

does not provide a convenient method for handing the r' term. Using the functional form of the macroscopic cross section with n concentric cylinders with radii $r_1, r_2, \ldots, r_n$ and macroscopic cross sections $\Sigma_{t,1}, \Sigma_{t,2}, \ldots, \Sigma_{t,n}$ existing between the collision site and the tally point the integral can be written as

$$\int_{r}^{r'}\Sigma_t(r'')\mathrm{d}r'' = \Sigma_t(r')(r'-r_1) + \Sigma_{t,1}(r_1-r_2) + \cdots + \Sigma_t(r)(r_n-r). \tag{2.71}$$

Solving for r' yields

$$r' = \frac{1}{\Sigma_t(r')}\left(\int_r^{r'} \Sigma_t(r'')\mathrm{d}r'' - (-\Sigma_t(r')r_1 + \Sigma_{t,1}(r_1 - r_2) + \cdots + \Sigma_t(r)(r_n - r))\right). \tag{2.72}$$

Substituting Eqs. (2.69) and (2.71) into Eq. (2.72) yields

$$r' = \frac{vh_{r,\mathrm{MFP}}}{\Sigma_t(r')} + \frac{1}{\Sigma_t(r')}\left(r'\Sigma_t(r') - \int_r^{r'} \Sigma_t(r'')\mathrm{d}r''\right). \tag{2.73}$$

Substituting Eq. (2.73) into to Eq. (2.70), integrating over θ', and noting the $vk(v)$ term integrates to 0 produces

$$\int C(r,\theta)k\left(v\right)h_{r,\mathrm{MFP}}h_\theta \frac{1}{\Sigma_t(r')^2}\left(r'\Sigma_t(r') - \int_r^{r'} \Sigma_t(r'')\mathrm{d}r''\right)\mathrm{d}r' = 1. \tag{2.74}$$

Thus, the normalization coefficient

$$C(r,\theta) = \frac{1}{h_{r,\mathrm{MFP}}h_\theta}\frac{\Sigma_t(r')^2}{\left(r'\Sigma_t(r') - \int_r^{r'} \Sigma_t(r'')\mathrm{d}r''\right)}, \tag{2.75}$$

allows Eq. (2.68) to hold. It should be noted that Eq. (2.75) is only dependent on r' through the macroscopic total cross section $\Sigma_t(r')$; no factors of r' exist in the normalization coefficient after integrating $\Sigma_t(r'')$ and subtracting the result from $r'\Sigma_t(r')$. Furthermore, if no material interfaces exist between r and r' then the normalization coefficient becomes $C(r,\theta) = (h_{r,\mathrm{MFP}}h_\theta r)^{-1}\Sigma_t(r')$. Thus, there is effectively only one factor of the cross section in the normalization coefficient in Eq. (2.75).

The same approximation can be made to the cylindrical MFP KDE as to the Cartesian MFP KDE to obtain the cylindrical aMFP KDE, with the cylindrical aMFP kernel function defined as

$$K(r,\theta) = \frac{\Sigma_t(r)}{rh_{r,\mathrm{MFP}}h_\theta}k\left(\frac{\Sigma_t(r)\left(r - r'\right)}{h_{r,\mathrm{MFP}}}\right)k\left(\frac{\theta - \theta'}{h_\theta}\right). \tag{2.76}$$

Furthermore, the volume-average KDE can be applied to this estimator, resulting in the volume-average cylindrical aMFP KDE

$$\hat{f}_j = \frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{c_i}\frac{w_{i,c}\Sigma_r(\mathbf{x},E)}{\Sigma_t(\mathbf{X}_{i,c},E)}\int_{\theta_{min}}^{\theta_{max}}\int_{r_{min}}^{r_{max}} K(r,\theta)r\mathrm{d}r\mathrm{d}\theta, \tag{2.77}$$

where

$$r_{\max} = \min(r' + h\sqrt{5}, r_{j+1}) \quad \text{and} \quad r_{\min} = \max(r' - h\sqrt{5}, r_j),$$

and

$$\theta_{\max} = \min(\theta' + h, \theta_{j+1}) \quad \text{and} \quad \theta_{\min} = \max(\theta' - h, \theta_j),$$

for an Epanechnikov kernel in the radial dimension and a uniform kernel in the azimuthal dimension. The final derivation for the volume-average cylindrical aMFP KDE is not shown here as it can be derived using the same process as the volume-average KDE in Section 2.7.

The optimal bandwidths for the cylindrical KDE are obtained in a similar fashion to the Cartesian KDE. Even though the coefficients on the asymptotic bias and variance differ between the Cartesian and cylindrical KDEs, the same optimal bandwidth formula,

$$h_r = \left(\frac{4}{(2+d)N}\right)^{1/(4+d)} \sigma_r, \tag{2.78}$$

is used for the cylindrical KDE with

$$\sigma_r = \sqrt{\left(\sum_{i=1}^{N} \frac{w_{i,l}}{\Sigma_t(\mathbf{X}_i)}\right)^{-1} \sum_{i=1}^{N} \frac{w_{i,l}}{\Sigma_t(\mathbf{X}_i)} r_i^2 - \left(\sum_{i=1}^{N} \frac{w_{i,l}}{\Sigma_t(\mathbf{X}_i)}\right)^{-1} \left(\sum_{i=1}^{N} r_i \frac{w_{i,l}}{\Sigma_t(\mathbf{X}_i)}\right)^2}. \tag{2.79}$$

This is justified since the estimators have the same asymptotic behavior with regards to the bandwidth and the number of samples. Again, for the cylindrical MFP KDE the radial bandwidth is computed via

$$h_{r,\mathrm{MFP}} = h_r \overline{\Sigma}_t. \tag{2.80}$$

For the reactor physics pincell problems studied in this thesis, distributions change much more rapidly in the radial dimension compared to the azimuthal dimension. As such, coarse histogram binning is generally used in the azimuthal dimension with four to eight bins used per radial ring. To better compare cylinder KDE results to reference histogram results, a uniform kernel with a fixed bandwidth is used in the azimuthal dimension. This effectively removes any bias that would occur from comparing point-wise quantities to volume-average quantities due to azimuthal variation. If a problem does have significant azimuthal variation,

then an optimal bandwidth could be obtained in a fashion similar to the radial bandwidth using Eqs. (1.36)-(1.52).

2.9.2 Verification of Cylindrical MFP KDE

2.9.2.1 2-D Pincell

The cylindrical KDEs and fractional MFP KDE are tested using a CASL 2-D pincell benchmark problem [41] comprised of a cylinder of 3.1% enriched UO_2 with radius 0.4096 cm surrounded by Zircaloy-4 cladding and 1300 ppm borated water with a lattice pitch of 1.26 cm and reflecting boundary conditions. The cladding is blended in with the gap and extends from 0.4096 cm to 0.475 cm, with the density of the Zircaloy adjusted to 5.77 g/cm^3 to preserve the amount of Zircaloy in the problem when the gap is explicitly modeled. Histogram tallies are conducted using a collision estimator on a cylindrical structured mesh with 8 bins in the azimuthal dimension and several regions of different bin widths in the radial dimension. The mesh in the radial dimension is constructed using 20 bins from 0 to 0.3896 cm, 200 bins from 0.3896 to 0.4096 cm, 5 bins from 0.4096 to 0.475 cm, and 20 bins from 0.475 to 0.63 cm. KDE tally points are placed at the center of histogram bins. Epanechnikov kernels are used in the radial dimension and uniform kernels with a bandwidth of π/16 are used in the azimuthal dimension. For the test problems shown here, the cylindrical KDE is used at tally points located at $r > 2\sqrt{5}h_r$. This is due to the $1/r$ dependence of the cylindrical KDE normalization coefficient. For tally points closer to the center of the cylinder the fractional aMFP KDE is used in place of the cylindrical MFP KDE and the Cartesian distance-based KDE is used in place of the cylindrical KDE. The simulations are run with 200,000 particles per batch, 20,000 batches with 100 inactive batches. Figure 2.36 shows the 1-D comparison between the distance-based cylindrical KDE and the reference histogram for the absorption distribution in the pincell's rim region, 200 μm into the edge of the fuel, along the 22.5° azimuth, with estimated distributions shown on top and C/E (KDE/histogram) comparisons below. Flux, fission, and absorption distributions for the cylindrical MFP KDE and histogram along with C/E values are plotted in 1-D along the 22.5° azimuth in Figures 2.37 - 2.39 with the distribution across the radius of the entire pincell shown on the left and a close-up of the distributions in the rim region, 200 μm within the edge of the fuel, on the right. The results are obtained using one KDE region defined for the entire pincell, resulting in distance-based radial bandwidth of 3.2×10^{-3} cm and a MFP-based radial bandwidth of 1.9×10^{-3}.

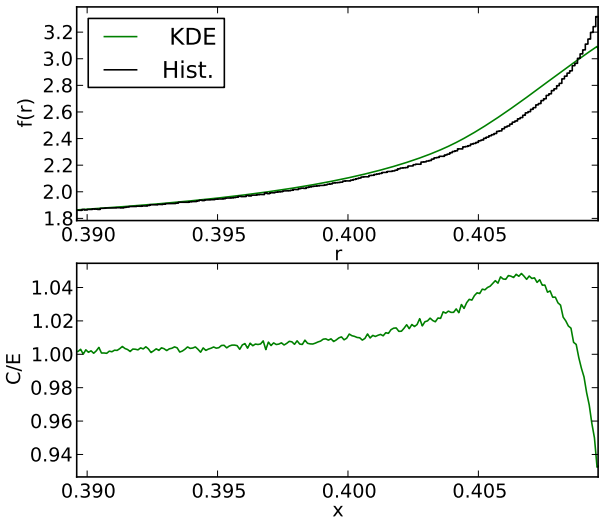
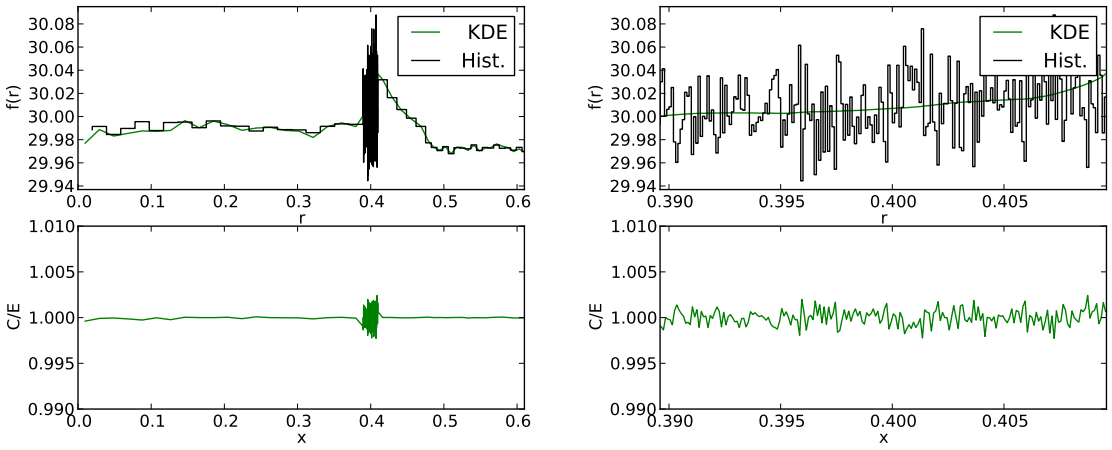


Figure 2.36: Comparison between distance-based cylindrical KDE and reference histogram solution for the absorption distribution along the 22.5° azimuth in the rim region of the fuel.

The distance-based cylindrical KDE accurately estimates the flux and fission reaction rates, with maximum disagreements of 0.6% due to the variance in the histogram results through the rim region. However, the distance-based cylindrical KDE produces a significant bias in the absoprtion distribution, seen in Figure 2.36, with underpredictions of up to 7% at the material interface. This bias becomes worse with fewer samples as the bandwidth increases and further smoothes this region of high curvature. Thus, while the distance-based cylindrical KDE can accurately capture distributions with low degrees of curvature, the MFP KDE is still required to capture distributions with large curvature.

Figures 2.37-2.39 show that the cylindrical MFP KDE accurately estimates the flux, fission, and absorption distributions in the 2-D pincell problem with less than 0.5% disagreement in all distributions. Furthermore, the flux, fission, and absorption distributions agree within three standard deviations for 99.8, 99.8, and 99.4% of the tally points, respectively. All tally points agree within 4 standard deviations with the exception of the tally points in the absorption distribution immediately adjacent to the rim region. This disagreement is due to comparing point-wise KDE results with volume-average histogram results. Increasing the high-resolution rim region to 0.35 cm would eliminate this discrepancy. Additionally, the spikes seen in the 2-D Cartesian MFP KDE do not appear in the cylindrical MFP KDE due to the normalization coefficient for the cylindrical MFP KDE being proportional to Σ_t rather than Σ_t^2.

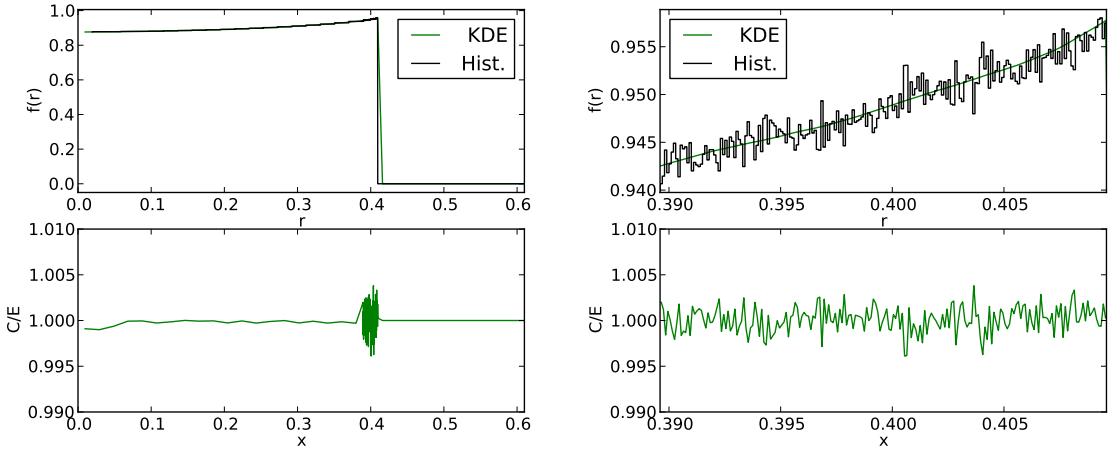
The fractional aMFP KDE was also run on the 2-D pincell problem, with results similar to those generated by the cylindrical MFP KDE and as such graphical representations of



(a) Flux across pincell

(b) Flux in rim region

Figure 2.37: Flux distribution comparison between the cylindrical MFP KDE and histogram for the pincell problem along the 22.5° azimuth.

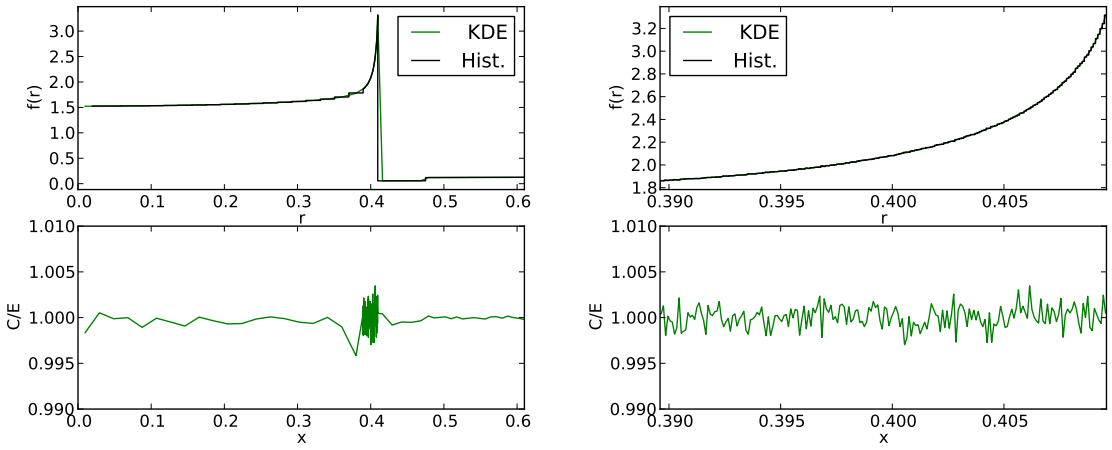


(a) Fission across pincell

(b) Fission in rim region

Figure 2.38: Fission distribution comparison between the cylindrical MFP KDE and histogram for the pincell problem along the 22.5° azimuth.

the results are omitted. The fractional aMFP KDE agrees with the reference histogram within 0.5% at all points in the flux and fission distributions, however it disagrees by as much as 11% in the absorption distribution. While the fractional aMFP KDE successfully removes the spikes seen in the 2-D pincell problem due to the normalization coefficient being proportional to Σ_t rather than Σ_t^2, it still presents issues at the fuel-cladding interface in the absorption distribution due to resonance interactions. Again, this is due to the increased variance in the estimator from resonance interactions due to the bandwidth being inversely proportional to the square root of the cross section. The fractional aMFP KDE disagrees



(a) Absorption across pincell

(b) Absorption in rim region

Figure 2.39: Absorption distribution comparison between the cylindrical MFP KDE and histogram for the pincell problem along the 22.5° azimuth.

with the reference result by overpredicting the reference result by up to 4.5% along one azimuth and underpredicting the absorption density by 11% on a different azimuth. This range of disagreement is due to a sharp increase in the variance at the material interface with relative uncertainties as high as 5.5%. This large uncertainty means the fractional aMFP KDE still agrees with the reference result within 3 standard deviations at all points in the distribution even while disagreeing by 11%. Even though the fractional aMFP KDE does not behave well at the material interface, it effectively suppresses spikes in the majority of the material without introducing significant bias or variance. As such, the fractional aMFP KDE can be used in conjunction with the cylindrical MFP KDE to produce density estimates at any point in the pincell.

The cylindrical aMFP KDE was also tested on the 2-D pincell problem. Using the same parameters as the MFP KDE, the cylindrical aMFP KDE produces identical results to the MFP KDE at all points more than one kernel support length away from a material interface. Therefore, a graphical representation of the cylindrical aMFP KDE results for this problem has been omitted. The fission and absorption distributions from the aMFP KDE and MFP KDE agree within 3-σ at all points with maximum disagreements of 0.1% and 0.3%, respectively. This discrepancy between the aMFP KDE and MFP KDE is not random, and indicates that the aMFP KDE produces a slight bias of approximately 0.1% and 0.3% at the fuel-cladding interface for the fission and absorption distributions, respectively. This is in agreement with the bias seen with the volume-average aMFP KDE seen in Figure 2.32. Even so, a 0.3% bias is acceptable and as such the cylindrical aMFP KDE can be used to

accurately estimate distributions in 2-D pincell problems.

CHAPTER 3

GPU Acceleration of KDEs and Algorithmic Considerations

3.1 Motivation for GPUs

GPUs are commonly found on HPC machines as a means of increasing performance without adding additional compute nodes. While it is difficult for existing Monte Carlo neutron transport codes to take advantage of GPUs without re-writing large portions of the code, it is possible to leverage GPUs through heterogeneous computing. Rather than use GPUs for the majority of the transport routine, it is possible to export the compute-intensive portions of the Monte Carlo algorithm onto GPUs. When using KDEs, it is not unlikely for the active tally runtime to increase by an order of magnitude compared to using a comparable histogram tally. Significantly more floating-point operations are required for KDEs for each collision or particle track than a histogram tally since a single event can contribute to the scores of multiple tally points. For example, the aMFP collision and track-length KDEs are factors of 2.5 and 9.4 slower, respectively, than their histogram counterparts on a 40×40 mesh for the 2-D Pincell problem in continuous energy shown in Section 2.4.2. This problem is exacerbated with increasing node density with the collision and track-length KDEs a factor of 12.7 and 40 slower, respectively, than their histogram counterparts on a 120×120 mesh for the same problem.

Since the KDE routines are independent of the particle transport process and the scores at each tally point are calculated independently from one another, the GPU can be used to efficiently compute scores from particle events to tally points. While the CPU performs the particle transport processes the relevant tally information can be transferred to the GPU and the KDE scoring routines can be run on the GPU simultaneously, potentially hiding the majority of the computational cost of the KDE. Thus, exporting the KDE routines onto the currently unused GPUs can have significant improvements in run times.

3.2 CPU Algorithm

To view the difference between the GPU and CPU algorithms, it is necessary to discuss the KDE CPU algorithm. In this chapter the KDE will be described from the perspective of the collision MFP KDE as the track-length MFP KDE tally has not yet been implemented on the GPU. Even so, the track-length MFP KDE is a close analog to the collision KDE with only two main distinctions: the routines for calculating the score from a particle event to the tally point differ as well as the neighborhood lookup routines and thus the algorithms described in this section translate well to the track-length MFP KDE on the CPU.

The MFP KDE algorithm begins after a collision has occurred, similar to a collision histogram tally. For the distance-based KDE, the first step is to determine what tally points to score to. The most naive approach is to loop over all tally points in the problem and determine if the tally point falls within the support range of the kernel and if so, then compute and contribute the score to that tally point. However, both determining if the tally point falls within the support range of the kernel and evaluating the kernel function at a tally point require knowing the cross sections between the tally point and the collision site when using the MFP KDE. This means that each material's macroscopic total cross section must be computed after each collision unless geometry-specific information is taken to account. Thus, for the MFP KDE the first step is to compute the relevant cross sections: either all cross sections in the problem or a subset of those based on *a priori* knowledge of the problem geometry.

Once the relevant cross sections are known, the second step is to determine which tally points to contribute scores to. Again, the naive approach is to loop over all tally points within the problem. However, this requires computing the number of MFP between each tally point and the collision site and thus is prohibitively expensive. Therefore, a neighborhood algorithm is used, called a Nearby Neighbor List (NNL) in this thesis and is detailed in Section 3.2.1. Once a tally point is selected, the third and final step is to evaluate the kernel function and increment the score at the tally point. In addition to the NNL, several other minor adjustments can be made to the MFP KDE to improve computational performance, namely using a maximum bandwidth as detailed in Section 3.2.2, local cross section lookups as detailed in Section 3.2.3, or a minimum bandwidth introduced in Section 2.5, and rediscussed in Section 3.2.4. The maximum bandwidth improves efficiency by reducing runtimes while the minimum bandwidth can reduce relative uncertainties in the MFP KDE results.

3.2.1 Nearby Neighbor List

The NNL works as a structured mesh of containers that list the tally points or collision sites that fall within the bounds of the container. Dunn describes the use of a neighborhood for the distance-based track-length KDE [22], and the implementation here is similar with adjustments necessary for the MFP KDE. The size of each mesh element is defined by the largest kernel support length, a bandwidth-dependent quantity, among all the KDE regions in the problem for each dimension. The ability to use a NNL is the reason why kernels with finite support are preferred for Monte Carlo radiation transport simulations. For the CPU algorithm, each element of the NNL stores a list of the tally points that fall within the bounds of the mesh element. After a collision, the location of the collision within the NNL is computed and each tally point within the mesh elements adjacent to and including the collision's mesh element are looped over and tallied to. In the case of a 2-D tally, a 3×3 area of elements is searched for potential tally points that a given collision will contribute a non-zero score to.

The complexity of this algorithm increases with the use of the MFP KDE. The MFP KDE either expands or contracts the bandwidth in space, and thus the maximum kernel support length in each dimension becomes an energy-dependent, or a collision-dependent, quantity. Even so, efficiency is still gained by using a NNL. The NNL for the MFP KDE still uses the distance-based bandwidth to determine the size of each mesh element, however the maximum support length in each dimension is determined after each collision. The maximum support length in space is calculated by first determining the maximum bandwidth in units of MFP among all regions in each dimension and then dividing this bandwidth vector by the minimum cross section among all materials in the problem. This maximum spatial bandwidth is now used to determine the number of bins to search through after each collision and as such the number of tally point containers to loop over becomes a collision-dependent quantity. This issue of an energy-dependent maximum support length is circumvented with the use of a maximum bandwidth as described in Section 3.2.2. Using the NNL can have drastic performance improvements, with a speedup of over 800 obtained for the fractional aMFP KDE on the 2-D boxcell problem in Section 2.7.1.2 with a grid of 120×120 tally points.

3.2.2 Maximum Bandwidth

To improve performance of the NNL, a maximum support region was defined for the aMFP KDE. The effective spatial bandwidth for the MFP KDE away from material interfaces is the bandwidth of the distance-based KDE multiplied by the average cross section in that region divided by the cross section at the tally point. Thus, if the cross section at a tally point is smaller than the average cross section in that region then the bandwidth becomes larger. This has a large impact on the performance of the MFP KDE since the neighborhood list must now be searched beyond one bin in each dimension.

However, it is possible to manipulate the bandwidth such that no more than one distance-based kernel support length is searched in each dimension. This is done by limiting the value of the cross section in the argument of the kernel function and the normalization coefficient, denoted as $\Sigma_{t,k}(x)$ in the following equation, such that

$$\Sigma_{t,k}(x) = \begin{cases} \Sigma_t(x) & \Sigma_t(x) > \overline{\Sigma}_t \\ \overline{\Sigma}_t & \text{otherwise} \end{cases}, \tag{3.1}$$

where $\overline{\Sigma}_t$ is the flux-weighted average cross section for a KDE-region defined in Eq. (2.8). Using Eq. (3.1) ensures that if the cross section is below the average cross section for a region, the MFP kernel reverts to the distance-based kernel. This approximation does not adversely affect the accuracy of the simulation, since it only limits the size of the bandwidth. Using a smaller bandwidth rather than a larger bandwidth will reduce the bias in the simulation while increasing the variance in the results. Thus, using Eq. (3.1) will increase the FOM for the aMFP KDE since the decrease in run times is accompanied by only a slight increase in the variance of the solution. Using a maximum cross section on the 2-D bare pincell problem in continuous energy in Section 2.5 with the fractional aMFP KDE results in a speedup of 1.4. Furthermore, limiting the size of the kernel support via the maximum bandwidth also allows for local cross-section lookup, detailed in Section 3.2.3.

3.2.3 Local Cross Section Lookup

Both the distance-based KDE and the MFP KDE require calculating cross sections in materials beyond where the collision or particle track occurred. In the case of the distance-based KDE on the CPU or the MFP KDE with a maximum bandwidth on the CPU these additional cross sections can be calculated on-the-fly when a collision attempts to score to a tally point in a different material within the support range of the kernel. However, for the

GPU algorithm any additional cross sections must be computed prior to sending data to the GPU.

Rather than calculating the cross sections for all materials in the problem after every collision, the use of a maximum bandwidth limits the range of the kernel support such that only materials within range of the distance-based KDE need to be considered. With a constant maximum kernel support and *a priori* knowledge of the problem geometry the materials that fall within range of the kernel centered about a given collision can be determined and only those material cross sections need to be computed. For problems considered in this thesis, the MFP KDE only needs to know the cross section of the collided material for the majority of collisions. Thus, local cross section lookup has the potential to provide significant speedup. In order to prevent unnecessary cross section lookups, either the NNL also needs to keep track of the materials found in each bin or knowledge about the problem geometry can be used to limit the amount of excess calculations. Since the problems of interest in this thesis are primarily lattices of concentric cylinders, this *a priori* knowledge of problem geometry is used to determine which cross sections to calculate. The speedup achieved from this method is problem-dependent with complexity depending on the number of disparate materials and the number of nuclides in each material, but a speedup of 1.3 was obtained for the cylindrical MFP KDE on the 2-D pincell problem with cladding found in Section 2.9.2.1.

3.2.4 Minimum Bandwidth

The minimum bandwidth, originally introduced in Section 2.5 to mitigate spikes in the absorption reaction rate, can also be used to enhance the computational performance of the MFP KDE by reducing variance. The use of a minimum bandwidth creates an upper limit to the modification of the distance-based bandwidth, causing the scores to be more spread out and thus decreasing the variance. While the use of the minimum bandwidth can introduce bias into the problem as seen in Section 2.5, if the minimum bandwidth is only applied away from material interfaces then the variance will be reduced throughout the majority of the distribution without introducing excessive bias. This method is applied in Section 3.6.3 to improve the FOM in the 2-D pincell problem.

3.2.5 Cylindrical MFP KDE Ray Tracing

Since the cylindrical MFP KDE was developed specifically to handle cylindrical geometries in PWR pincell problems this *a priori* knowledge of the problem geometry can be used to remove the need to conduct ray tracing for the cylindrical MFP KDE. Since the geometry is assumed to be a series of concentric cylinders surrounded by water in a lattice configuration, this information can be used to determine the number of MFP between a collision site and tally point in the radial dimension. This is more efficient than conducting a ray trace designed to handle general geometries between the two points and allows for the cylindrical MFP KDE to be conducted on the GPU without introducing an unnecessary amount of warp divergence (discussed in Section 3.3).

3.3 GPU Architecture and Bottlenecks

In order to understand the GPU algorithm a brief background in GPU architecture is required. Since the algorithm described in this thesis uses NVIDIA GPUs the architecture described here uses terms that may be specific to NVIDIA GPUs and the CUDA application programming interface [42]. GPUs are based on Single Instruction Multiple Thread (SIMT) architecture with the number of threads ranging from several hundred to a few thousand per GPU. Threads are organized into groups of 32 threads called warps, with each thread in a warp performing the same instruction or sitting idle. The GPU-specific portions of code are called kernels and are initiated by specifying a number of threads per block and the total number of blocks that kernel will execute. The major memory components on a GPU are global memory and shared memory. Global memory can be accessed from all warps on the GPU as well as from the CPU while shared memory is local to each warp. Fetching data from global memory is expensive, taking approximately 400-800 clock cycles per fetch on GPUs with compute capability 2.x, while shared memory access is approximately 100 times faster than global memory. GPUs use processing units called streaming multiprocessors (SMs) to schedule and execute warps, with multiple warps executed simultaneously and independently on one SM. Each active warp on the GPU is called a context, and SMs can switch between contexts at no cost. This ability to switch contexts enables the multiprocessor to hide memory latency, discussed later in this section.

Since each thread on a warp performs the same instruction or sits idle, the GPU program needs to be written specifically to take advantage of vectorization. Even though threads within a warp are free to execute and branch independently, the processor follows each

path of the branch individually with all threads not on the current path sitting idle. Thus, if the GPU kernels have an excessive amount of branching from conditional statements then the performance of the GPU program is significantly hindered. This branching from conditional statements is called warp divergence and is one of the most important factors for programming on a GPU. Warp divergence is one of the main difficulties in running Monte Carlo neutron transport codes on GPUs.

Beyond warp divergence, the next most important factor to consider is how each thread in a warp accesses memory. The memory latency is partially hidden by the SM due to its ability to switch between contexts at no cost. When one warp of threads is waiting for data from global memory, the multiprocessor can switch contexts to a warp that has already loaded the memory to execute commands on that context, thus hiding some amount of the memory latency. This effectively hides the latency, but only if there are a sufficient number of arithmetic operations on other warps. If there are too many global memory reads for the amount of arithmetic required in each warp then the GPU kernel is limited by the time it takes to read data from global memory. Since a load from global memory takes between 400-800 clock cycles and it takes two clock cycles to issue an instruction to a warp, reducing the number of global memory loads can have a significant impact on the performance of the algorithm.

When a thread requires data from global memory, the warp reads either 32, 64, or 128 bytes of memory at once, depending on what the threads in the warp require. For example, if each thread in a warp needs to operate on a single precision floating-point (4 byte) number, then the warp needs to load in 128 bytes of memory. If the data the warp requires is sequentially located in memory, then this data load is coalesced into one read from global memory. However, if the data is randomly dispersed throughout memory, then it could take up to 32 separate reads from global memory. This is another issue for programming Monte Carlo radiation transport simulations on GPUs as particles move randomly throughout the problem phase space. It is possible to reduce global memory loads by initially loading data from global memory and saving it in shared memory. This is useful if multiple threads require the same piece of data. Shared memory is also used for parallel reductions, like the aggregation of scores from multiple collisions to one tally point, as it is not penalized by uncoalesced memory accesses and each thread has access to the entire block of shared memory. Thus, organizing the memory such that loads from global memory are coalesced and leveraging shared memory can have significant impacts on algorithm performance.

Another potential bottleneck for performance is copying memory between the CPU and GPU. Memory can be copied between the CPU and the GPU's global memory, however

each memory copy call has non-negligible overhead. Furthermore, by default no other CPU or GPU operations can take place while memory is being copied between the CPU and GPU. Since each each collision on the CPU generates information that needs to be transferred to the GPU, the algorithm needs to efficiently transfer data between the CPU and GPU. This is done by leveraging CUDA streams and asynchronous communication. A stream is a sequence of commands that execute in order, and different streams can execute concurrently. Thus, CUDA streams enable the concurrent transfer of a piece of data and execution of kernels on data that has already been transferred. Furthermore, the asynchronous memory copy allows the CPU to immediately return to the transport process after sending collision data to the GPU, effectively hiding the cost of transferring data from the CPU.

3.4 GPU Algorithm

The GPU KDE algorithm is implemented using CUDA C and uses C Bindings to link the CUDA C code to the main Fortran program in OpenMC. The algorithm is designed to hide the cost of copying memory from the host (CPU) to the device (GPU) and have the device compute KDE scores while the host continues transporting particles to collect tally information. Rather than tally scores directly after a collision as described in Section 3.2, the host collects information in one of two sets of sample arrays during the transport process. The host calculates all the necessary cross section information and stores it as well as all necessary collision information in the first set of sample arrays. Once the host has stored a pre-set number of samples (50,000 collisions for this thesis), the host copies the data asynchronously to the device. The host then puts the KDE GPU kernels operating on the first set of samples into the same stream as the memory copy so that the kernels will launch as soon as the memory has finished copying from the host to the device. After this, the host immediately returns and begins transporting particles and populating a second set of sample arrays while the device is obtaining the data and computing scores. Once the host has finished filling the second array, it asynchronously sends the data to the device and waits until the first set of sample arrays has been received by the device before re-filling the first set of arrays. This process repeats until the end of the batch, when the partial set of sample arrays is sent to the device for calculation of scores. Once this has finished, the host acquires the tally data from the device and the host uses the normal MPI processes for combining tally data across multiple processors.

Once the device has received the first array the KDE GPU kernels are launched. The SIMT architecture of the GPU requires a different algorithmic approach from that used on the

CPU. The GPU algorithm has a block of 64 threads compute the scores to a tally point by looping over all relevant collisions within the first set of sample arrays for that tally point. This approach was chosen rather than having a block of threads compute the scores to all relevant tally points from a single collision because the density of tally points can change dramatically through the problem, as witnessed in the 2-D Pincell results in Section 2.9.2.1. Thus, it may be possible to have less than 32 tally points that a collision can potentially contribute a score to, resulting in a loss of efficiency. Furthermore, this approach limits the number of atomic operations necessary as each block of threads is incrementing the score of different tally points.

The relevant collisions for a given tally point are determined through the use of a NNL, similar to that described in Section 3.2.1. The use of a NNL on the GPU results in a speedup by a factor of approximately 16 as seen in Table 3.1. The GPU NNL requires the use of the maximum bandwidth as described in Section 3.2.2 in order to minimize warp divergence as each collision has unique cross sections. Each collision in the set of sample arrays is assigned an index based on its initial location within the sample array and a key based on its location in the NNL. These indices are then sorted based on their NNL keys to create a map that converts each sample array in the set to sorted arrays in order to facilitate better memory coalescence.

Once the arrays are sorted the scoring routine begins. Each block of threads loops over the collisions within the relevant NNL elements for that block's tally point and computes the score to the tally point. Each thread keeps its own sum of scores in shared memory to prevent serialization of threads via atomic operations when incrementing the tally point's score. After all collisions within the set are accounted for the threads' scores are reduced using shared memory and the total score from all collisions within the sample array is added to the running sum located in global memory. This process is repeated for each set of sample arrays sent to the device.

The aMFP KDE is only used on the GPU for the Cartesian KDEs in order to avoid ray tracing and thus further reducing warp divergence. The cylindrical MFP and aMFP KDEs are implemented on GPU as they do not require ray tracing since the problem geometry is assumed to be a pincell.

3.5 GPU Optimization

Several improvements were made throughout the design of the GPU KDE compute kernel to reduce compute times. Kernel compute times on a Tesla M2090 for the various optimization iterations for the 2-D boxcell problem with 120×120 tally points and 50,000 collisions per set of sample arrays are shown in Table 3.1. Kernel compute times were determined using the NVIDIA Visual Profiler [43]. The focal points for optimization were driven by the GPU's SIMT architecture. This section discusses the various algorithm design optimizations and their speedup in terms of the reduction in the GPU kernel compute time, or the time it takes the GPU to process the array of 50,000 collisions.

One design decision that is impacted by the coalesced data reading pattern of GPUs is using an array of structures (AoS) versus a Structure of Arrays (SoA). When using an AoS the data of the structure is placed sequentially in memory. If a thread only requires one piece of data from that structure then efficiency is lost based on the size of the structure. If the structure contains enough data it is possible for each thread in the warp to require a separate read from global memory when each thread is accessing a variable from sequentially located structures. For example, the usual particle data required for tallies in a Monte Carlo code includes position, previous position, angle, energy, cell, material, and weight. If a KDE tally only requires the x position of a collision to compute scores, then an extra 10 floating-point values and two integers, or 88 bytes in double precision, exist in memory between one collision's x position and the x position of the collision accessed by the next thread. This results in an extra 16 loads from global memory when 128 bytes are read in at once, hampering the code's performance.

These extra reads from global memory are eliminated by using a SoA. Rather than have an array of collision structures where each collision structure stores position, angle, energy, cell, material, etc., a single structure, called a collision list, is created that contains separate arrays for the collision data. For example, the collision's x, y, and z positions as well as its material and cell numbers are all located in separate arrays. Therefore, a calculation of the distance between a collision site and a tally point in the x direction will only require two reads from global memory if the x positions are stored as double precision floating-point values. Converting the collision data and cross section information into a SoA results in a GPU kernel compute time reduction of 40%.

Additionally, the shared memory on the GPU was leveraged to further reduce GPU kernel compute times. From the NVIDIA Programming Guide [42], "shared memory is expected to be a low-latency memory near each processor core (much like an L1 cache)." Commonly

used variables are copied to shared memory and results for a tally point are collected by each thread in shared memory before being reduced and added to the results array stored in global memory. Using shared memory results in a 20% reduction of kernel compute times.

Another optimization is to change tally variables from double precision to single precision floating-point values. Changing to single precision has two effects. First, it reduces the size of memory required by each thread. Using the previous example where all threads in a block require the x position of different collisions, using single precision data for the collision position allows all of the x positions for the 32 threads in a block to be loaded in one read from global memory rather than the two reads necessary with double precision data. Furthermore, the GPU has a faster clock speed for single precision floating-point operations. For the NVIDIA Tesla M2090, the clock speed for single precision floating-point operations is double that of double precision operations. This discrepancy is even greater in newer graphics cards, with the NVIDIA Tesla K80 having 8.74 Tflops for single precision floating point values versus 2.91 Tflops for double precision floating point values at peak performance [44]. For the KDE GPU algorithm, switching from double precision to single precision floating-point values further reduces the kernel compute time by approximately 30%. This reduction in precision does not come at a cost in accuracy for the problems studied here as the single precision results agree with those obtained using double precision to six significant digits.

Table 3.1: KDE GPU kernel compute times for 2-D boxcell problem with 120×120 tally points.

Design Iteration	Kernel Compute Time (ms)
Array of structures	56.3
Structure of arrays	35.6
Shared memory improvements with SoA	27.8
Single precision with SoA and shared memory	19.6
Shared memory improvements without NNL	815

Optimization Summary and Future Work

Overall, optimization of the KDE GPU algorithm results in substantial speedup. The largest reduction in kernel compute times comes from improving memory coalescence by using a SoA memory layout rather than an AoS, resulting in a speedup of 1.4. Altogether, the GPU-specific algorithm optimizations result in a speedup of 2.9. This allows the GPU to

handle more difficult tallies containing a greater number of tally points or a greater density of collision sites without causing a slowdown in the overall Monte Carlo algorithm.

Future design implementations could improve the speed of both the CPU and GPU tally process. The current GPU algorithm scales linearly with the number of tally points in the problem. This will limit the algorithm's application to larger, more complex problems. The algorithm could be improved so that it also scales with tally point density rather than the number of tally points, similar to the algorithm on the CPU, thus extending its range of application. Additionally, it may be possible to export the track-length KDE to the GPU as well. While it will be more difficult to sort the particle tracks for memory coalescence, each particle track requires more calculations to compute the score to a tally point, thus enabling the SM to potentially hide the extra loads from global memory.

Another area for improvement is to use GPUs with CUDA compute capability 3.5 or higher. These newer GPUs have a Multi Process Service (MPS) which allows commands sent to the device from several MPI processes to be handled simultaneously. In earlier GPUs, including the M2090, each MPI process would have a separate CUDA context. Kernels from one CUDA context cannot run on the GPU at the same time as kernels from another context. Thus, with the current algorithm design, there is no way to have compute kernels from different MPI processes overlap with one another. This inherently reduces efficiency since a kernel is not using all of the GPU's resources towards the end of the GPU kernel calculation. The MPS fixes this problem by having all MPI processes under one context, handled by the MPS. Thus, multiple kernels from different MPI processes could be run at the same time, improving the efficiency of the algorithm.

3.6 GPU Acceleration and Performance Comparisons

3.6.1 Overview

This section details the speedup obtained by using the GPU algorithm compared to the CPU algorithm for the fractional aMFP KDE and the cylindrical MFP KDE for several problems in continuous energy as well as performance comparisons between the KDEs and equivalent histogram tallies for two problems in continuous energy. The MFP KDE algorithm improvements detailed in Section 3.2 are used for speedup calculations and performance comparisons, with the exception of the minimum bandwidth unless specifically noted. The fractional aMFP KDE is examined on the 2-D boxcell problem detailed in

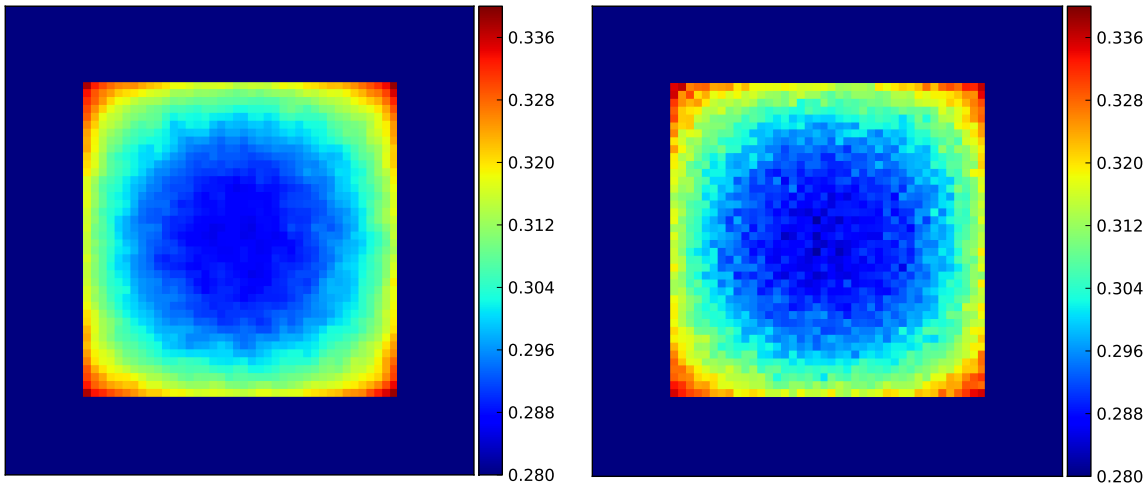
Section 2.7.1.2 with performance comparisons to equivalent histogram tallies. Additionally, the performance of the cylindrical MFP KDE on the 2-D pincell with cladding detailed in Section 2.9.2.1 is examined along with the speedup resulting from the GPU algorithm. Finally, GPU speedup on an assembly of pincells is analyzed for the fractional aMFP KDE. Speedup problems are run using two Tesla M2090 GPUs (compute capability 2.0) with 16 MPI processes on two eight-core Intel Xeon E5-2670 processors.

When analyzing performance comparisons it should be noted that the relative FOM of the KDE to that of the histogram tallies decreases with increasing number of histories. This is due to the bandwidth being proportional to $N^{-1/(d+4)}$ in Eq. (1.39) and the variance of the KDE being inversely proportional to the number of samples times the product of the bandwidths in each dimension as seen in Eqs. (1.34) and (2.66). Therefore, the variance of the 2-D fractional aMFP KDE is proportional to $N^{-2/3}$. Thus, assuming that simulation time is proportional to the number of histories simulated, the FOM for the KDE decreases with increasing number of histories and is proportional to $N^{-1/3}$ for 2-D problems. Nevertheless, KDEs can overcome this issue through its ability to spread the score from one collision to multiple points and through its versatility in applications discussed in Chapter 4.

3.6.2 2-D Boxcell

The first problem analyzed is the 2-D boxcell problem detailed in Section 2.7.1.2. Results are obtained using the collision fractional aMFP KDE, collision histogram, and track-length histogram on a structured grid of 60×60 tally points and bins and 120×120 tally points and bins with 200,000 particles per batch, 200 batches and 100 inactive batches. FOM comparisons are obtained using two Tesla M2090 GPUs with 16 MPI processes on two eight-core Intel Xeon E5-2670 processors for the fractional aMFP KDE results and 16 MPI processes on the CPU for the histogram results. One KDE region is defined for the entire problem. The fission distributions obtained using the fractional aMFP KDE and the collision histogram are shown in Figure 3.1. The runtimes and ratio of the FOM for the fractional aMFP KDE to the histogram estimators for the flux, fission, and absorption reaction rates in the 2-D boxcell problem are given in Table 3.2 with FOM calculated using Eq. (2.49). Using two GPUs and 16 MPI processes on the 60×60 and 120×120 structured grid of tally points results in speedups of 1.9 and 4.1, respectively, compared to using the CPUs alone.

Table 3.2 shows that the fractional aMFP KDE shows a favorable FOM compared to the collision histogram for all distributions on the 120×120 mesh and for the flux and fission distributions on the 60×60 mesh while the track-length histogram has a superior FOM



(a) Fractional aMFP KDE

(b) Collision histogram

Figure 3.1: Fission reaction rate densities in the 2-D boxcell problem with 60×60 bins and tally points obtained using the fractional aMFP KDE (left) and the collision histogram (right).

compared to the collision fractional aMFP KDE for all distributions and mesh sizes. The fission distributions in Figure 3.1 show that the fractional aMFP KDE produces a smoother distribution than the collision histogram tally, a direct result of the lower variance in the KDE results compared to the collision histogram. These reduced variances produce better FOM for the KDE even though the runtime is longer than the collision histogram.

Furthermore, Table 3.2 shows that doubling the number of histogram bins and tally points in each dimension improves the ratio of the FOM by a factor of three to four. This is due to the volume of the histogram bins decreasing by a factor of four, causing an increase

Table 3.2: Runtimes and FOM for the fractional aMFP KDE, collision histogram and track-length histogram on the the 2-D Boxcell problem with 60×60 and 120×120 bins and tally points.

			FOM Ratios (KDE/Histogram)		
Tally Size	Estimator	Time (s)	Flux	Fission	Absorption
60×60	KDE	124	-	-	-
	Col. Histogram	100	3.4	3.4	0.65
	TL Histogram	237	0.17	0.27	0.19
120×120	KDE	131	-	-	-
	Col. Histogram	103	12.8	13.2	2.3
	TL Histogram	375	0.47	0.79	0.59

in the variance of the collision histogram tallies by a factor of four while the variance of the KDE remains constant. Ordinarily, increasing the number of tally points would have a corresponding increase in the active tally runtime when using KDEs, however the use of the GPU enables the algorithm to hide most of the cost of the KDEs, resulting in direct improvements in the relative FOM. The variance in the track-length histogram results only increase by an approximate factor of two when doubling the number of bins in each dimension due to the particles scoring in multiple bins as they stream across the problem geometry. However, the additional bins increase the runtime of the track-length histogram tally thus causing the factor of three increase in the relative FOM.

Table 3.2 also shows that the collision fractional aMFP KDE performs worse at estimating the absorption distribution than it does at estimating the flux and fission distributions compared to the collision histogram, with relative FOM for the flux and fission distributions being a factor of 6 larger than that of the absorption distribution. This is again due to increased uncertainties at the fuel-water interface from resonance interactions. This can be seen in the ratio of the relative uncertainties in the absorption reaction rate density obtained from the KDE to that of the collision histogram depicted in Figure 3.2. As discussed previously in Chapter 2, when a neutron undergoes a collision at a resonance the spatial bandwidth for the MFP KDE decreases. From Eq. (1.34), reducing the bandwidth increases the variance. Even though the fractional MFP KDE eliminates the spikes that would ordinarily result from these decreased bandwidths, it still suffers from an increased variance from resonance interactions thus causing the worse performance for the absorption distribution.

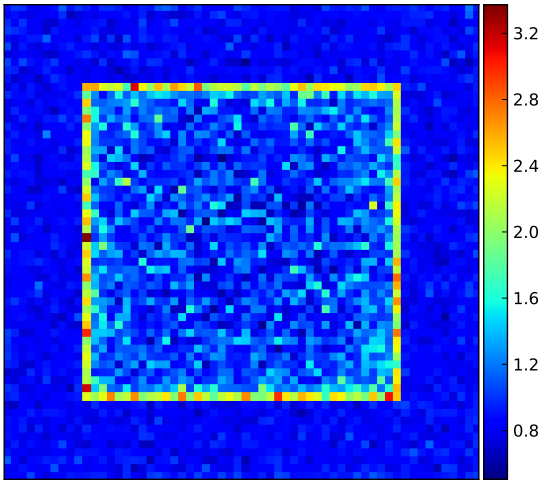


Figure 3.2: Ratio of relative uncertainties in the fractional aMFP KDE to the collision histogram for the absorption distribution in the 2-D boxcell problem with 60×60 bins and tally points.

3.6.3 2-D Pincell

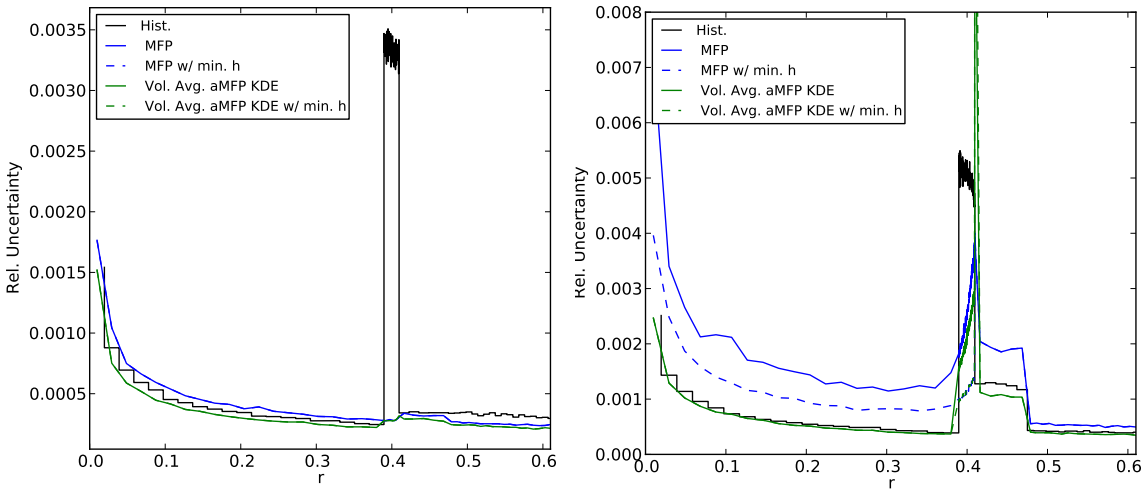
The 2-D pincell problem in Section 2.9.2.1 is analyzed to obtain a performance comparison between four different KDEs and the collision histogram tally on a cylindrical mesh as well as to determine the speedup obtained from the GPU algorithm for the cylindrical KDE on this problem. The four KDE tallies studied here include the cylindrical MFP KDE and cylindrical volume-average aMFP KDE with and without minimum bandwidths of 2 μm. The number of simulated histories is set to 1350 batches, 100 inactive batches and 200,000 particles per batch such that uncertainties in the absorption reaction rate near the rim of the fuel pin are approximately 0.5% for a realistic performance comparison. The performance comparison simulations between the KDEs and the histogram are run using 64 MPI processes across eight Intel Xeon E5-2670 processors. The comparison of the KDEs to the histogram does not use the GPU algorithm, and thus the FOM would be further improved by a factor equal to the speedup obtained from using the GPU. The active tally runtimes as well as ratios (KDE/histogram) of the FOM for the scalar flux, fission, and absorption distributions are given in Table 3.3. It should be noted that the FOM is heavily problem-dependent and mesh-dependent; using a different structured mesh for the histogram or tally point placement would result in a significantly different FOM.

Table 3.3: Active tally CPU runtimes and FOM ratios relative to the histogram for 2-D pincell benchmark problem.

		FOM Ratios (KDE/Histogram)		
Estimator	Time (s)	Flux	Fission	Absorption
Histogram	851	-	-	-
MFP KDE	1187	58.3	45.4	2.2
MFP KDE w/ min. h	1202	58.0	45.2	9.1
Vol. Avg. aMFP KDE	1336	63.0	49.9	2.9
Vol. Avg. aMFP KDE w/ min. h	1336	63.0	49.9	4.2

The FOM results in Table 3.3 show that the KDEs all produce a better FOM than the histogram even though runtimes are longer than that of the histogram tally. These FOM would be improved by using the GPU to conduct the KDE tally, with a speedup of 1.37 obtained from using the GPUs. To better describe the FOM results, plots of the relative uncertainty in the flux and absorption distribution along the 22.5° azimuth for the five estimators are shown in Figure 3.3. Additionally, a close-up of the relative uncertainties in the rim region for the absorption distribution are shown in Figure 3.4. The relative uncertainty in the fission distribution is similar to that of the flux distribution and is not

shown for clarity.



(a) Flux (b) Absorption

Figure 3.3: Relative uncertainties in the flux and absorption distributions along the 22.5° azimuth for the five estimators detailed in Table 3.3 for the 2-D pincell problem.

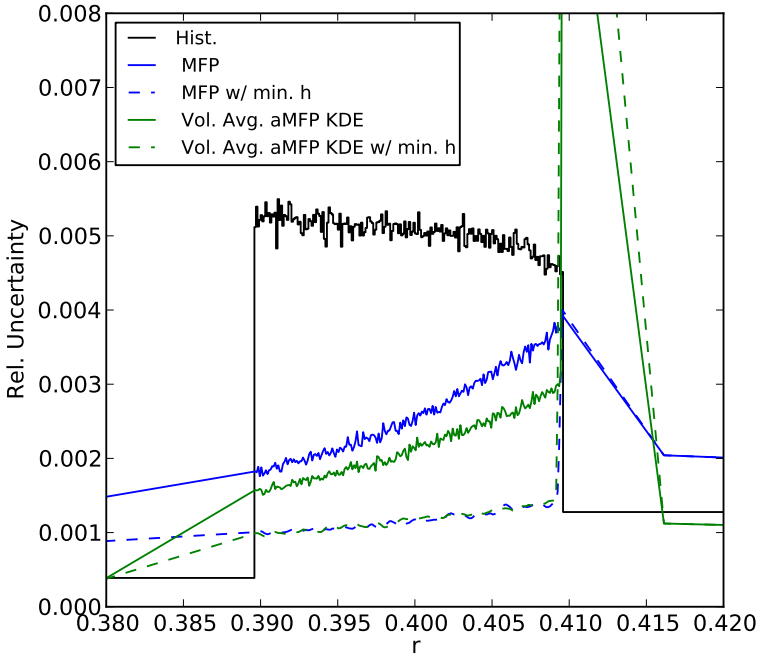


Figure 3.4: Relative uncertainty in the absorption distribution along the 22.5° azimuth at the fuel-cladding interface for the five estimators detailed in Table 3.3 for the 2-D pincell problem.

It is important to note that all tally points and histogram bins carry the same importance when computing the FOM in Eq. (2.49). Thus, the 200 radial bins/tally points in the rim

region of each azimuthal bin carry more significance in determining the FOM than other 45 radial bins/tally points in each azimuthal bin. This is the reason the MFP KDE produces a better FOM than the histogram even though the relative uncertainties seen in Figure 3.3 are larger than those of the histogram throughout the fuel prior to the rim region in the flux distribution and everywhere outside of the rim region in the absorption distribution. The volume-average aMFP KDE reduces this uncertainty throughout most of the problem by combining the smoothing capability of the KDE with the large sampling area provided by the histogram bin. Using the volume-average aMFP KDE ensures a relative uncertainty less than or equal to that of the histogram and MFP KDE in all bins more than one kernel support length away from a material interface, as seen in Figures 3.3 and 3.4.

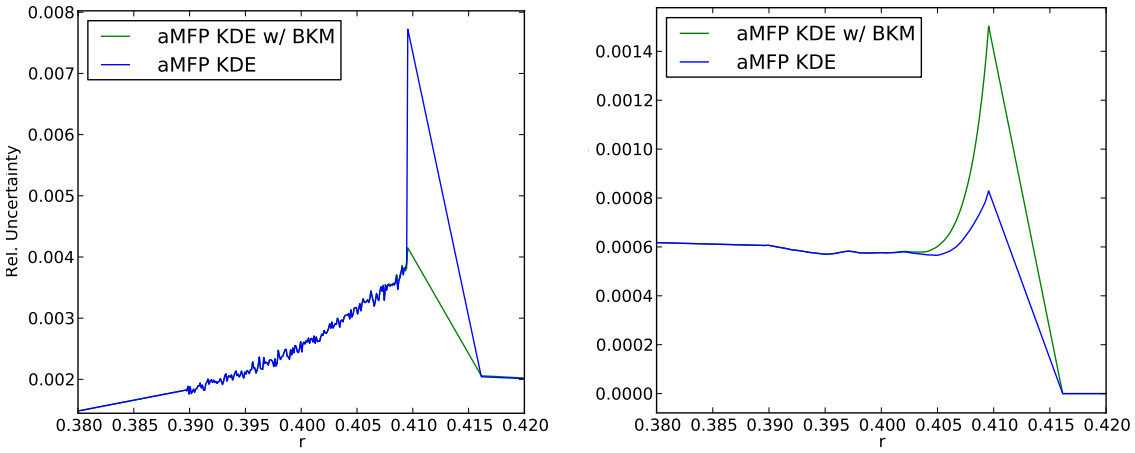
Applying the minimum bandwidth can also reduce uncertainties in distributions affected from collisions at resonances. Limiting the reduction of the size of the bandwidth effectively increases the bandwidth for any interaction at resonance energies, thus reducing the variance. Since the resonance interactions only have a significant impact on the absorption reaction rate density in the 2-D pincell problem, the minimum bandwidth only has a significant effect on the absorption distribution. Furthermore, Figure 3.3 shows that the minimum bandwidth does not significantly affect the volume-average aMFP KDE outside of the rim region. This is due to the relative size of the histogram bin versus the minimum bandwidth; the kernel function is still likely to be entirely contained within the histogram bin where the collision occurred even with the application of the minimum bandwidth. However, inside of the rim region the bin widths are half the size of the minimum bandwidth and thus the minimum bandwidth has the dominant effect, causing the MFP KDE with the minimum bandwidth to have similar uncertainties as the volume-average aMFP KDE with the minimum bandwidth. While this fixed minimum bandwidth of 2 μm has the potential to impose a bias at the fuel-cladding interface of the absorption distribution, in practice the minimum bandwidth can be adjusted for each tally point or it can be turned off at points near the material interface to prevent any appreciable bias from being observed.

Another noticeable feature in the distributions obtained from the KDEs is the increase in relative uncertainty in the fuel-cladding interface with a substantial increase in the uncertainties of the volume-average aMFP KDEs. The plots of the relative uncertainties in the absorption distribution are truncated, with the volume-average aMFP KDE and volume-average aMFP KDE with a minimum bandwidth producing a relative uncertainty of 1.2% and 1.6%, respectively, in the bin in the fuel at the fuel-cladding interface. This increase in uncertainty is due to interactions outside of the fuel scoring to tally points inside of the fuel. Since a collision estimator is being used, the score at a tally point in the fuel

from a collision in the cladding is proportional to the reaction rate in the fuel divided by the total cross section in the cladding, $\Sigma_r(\text{fuel})/\Sigma_t(\text{clad})$. For example, the microscopic absorption reaction rate in U-238, seen in Figure 2.9, ranges between 10 b and 7,000 b at 4 eV and the resonance at 6.67 eV, respectively. On the other hand, the microscopic cross section of Zr-90, the primary component in Zircaloy-4 cladding, is relatively constant at 5.5 b in the eV energy range. Thus, the score to the absorption reaction rate can change nearly three orders of magnitude between collisions at 4 eV and 6.67 eV when a collision in the cladding is being scored to a tally point in the fuel. A histogram tally on the other hand only uses information at the collision site to compute scores, so the score to any bin ranges between 0 and 1 for reaction rate densities. This additional variability in the score to the reaction rate from kernels overlapping material interfaces causes this increase in the estimator uncertainty. This effect is exacerbated by the volume-average aMFP KDE since the aMFP KDE effectively reduces the bandwidth of the MFP KDE in the fuel near the material interface. These increases in the uncertainty of the volume-average aMFP KDE at the fuel-cladding interface cause the estimator to have a worse FOM for the absorption distribution compared to the MFP KDE with a minimum bandwidth.

This increase in uncertainty can be reduced by using the boundary kernel method at the material interface to prevent collisions outside of the fuel from scoring to tally points within the fuel. Since the cylindrical kernel function in Eq. (2.50) is comprised of products of univariate kernels k, the boundary kernel method applied to the Epanechnikov kernel can be used for the radial kernel function at cylindrical boundaries. This allows the boundary kernel method to be used at the fuel-cladding interface. To determine the effects of using the boundary kernel method, the cylindrical aMFP KDE with and without the boundary kernel method is used to estimate the flux, fission, and absorption distributions at the fuel-cladding interface. The simulations are run using the same parameters as the other five estimators with 1350 batches, 100 inactive batches and 200,000 particles per batch. The cylindrical aMFP KDE with the boundary kernel obtains results that agree with the estimator without a boundary kernel within 0.5% and 1-σ at all tally points. Plots of the relative uncertainty in the fission and absorption reaction rate densities near the fuel-cladding material interface are shown in Figure 3.5. The flux results are similar to the fission results and are omitted for clarity.

Figure 3.5 shows that the relative uncertainty in the absorption reaction rate in the fuel at the material interface is reduced to approximately 0.004 from 0.0075 as a result of using the boundary kernel method at the fuel-cladding interface. However, the boundary kernel method also increases the relative uncertainty in the flux and fission distributions. This



(a) Absorption

(b) Fission

Figure 3.5: Relative uncertainties in the absorption and fission distributions along the 22.5° azimuth for the cylindrical aMFP KDE with and without the boundary kernel method at material interfaces for the 2-D pincell problem.

increase in relative uncertainty is due to the reduction in the kernel support area, reducing the number of collisions contributing to the score at that tally point, as well as the increase in the variance of the estimator in Eq. (1.34) since $\int k(u)^2 \mathrm{d}u$ is larger for the boundary kernel method applied to the Epanechnikov kernel than the Epanechnikov kernel itself.

3.6.4 Assembly of Pincells

A quarter-assembly of pincells, depicted in Figure 3.6, is modeled in order to test the speedup of the GPU on a larger problem. The single-precision GPU algorithm is compared to the double-precision CPU and GPU algorithms for two mesh sizes with a varying number of MPI processes. The problem consists of pincells with a lattice pitch of 1 cm containing fuel pins of 3% enriched UO_2 with a diameter of 0.7 cm, depicted in yellow in Figure 3.6, and control rods of B_4C depicted in black. The simulation is run with 200,000 particles per batch, 2,000 total batches with 100 inactive batches. One KDE region is assigned to each pincell. Reference histogram results are collected on a structured grid of 120×120 bins with KDE tally points placed at the center of each bin. The flux obtained using the fractional aMFP KDE and its C/E comparison to a collision histogram tally is shown in Figure 3.7 while the fission and absorption reaction rates are shown in Figure 3.8. Speedups and runtime statistics for the assembly of pincells with 120×120 tally points and 240×240 tally points are given in Tables 3.4 and 3.5, respectively, for 1 to 16 MPI processes. The tables give

the history calculation rate during the inactive and active tally portions of the Monte Carlo algorithm and use these to determine the speedup of the algorithm by comparing the active cycle calculation rates obtained with and without using a GPU for a given number of MPI processes.

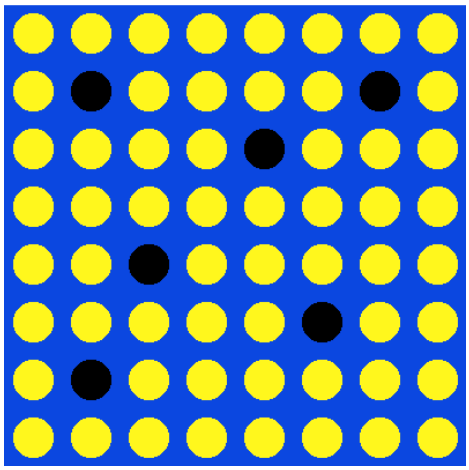
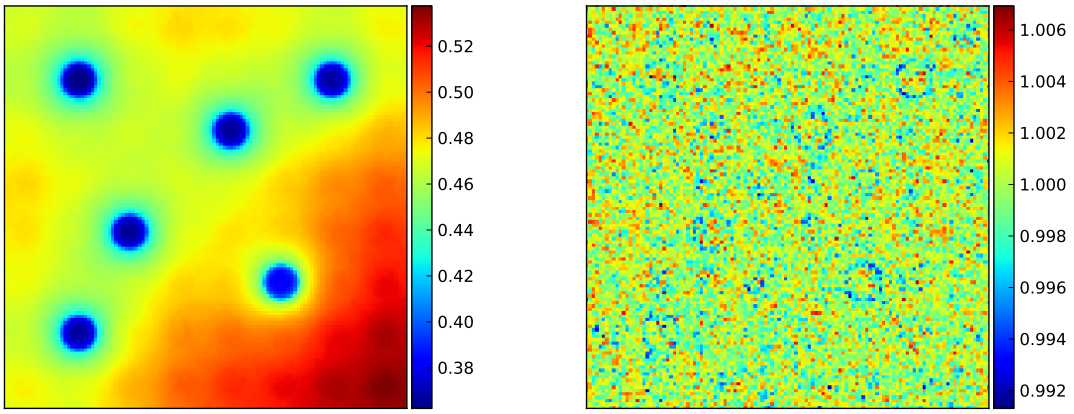


Figure 3.6: Depiction of the assembly of pincells.



(a) Flux

(b) Flux C/E

Figure 3.7: Flux distribution from the fractional aMFP KDE and C/E comparison to collision histogram for the assembly of pincells.

Figure 3.7 shows that the fractional aMFP KDE is capable of accurately capturing the flux distribution with differences of less than 1% for all tally points when compared to the collision histogram and with a FOM that is a factor of 3.3 larger than that of the histogram tally. While noticable differences do exist in the flux around the control rods this is due to

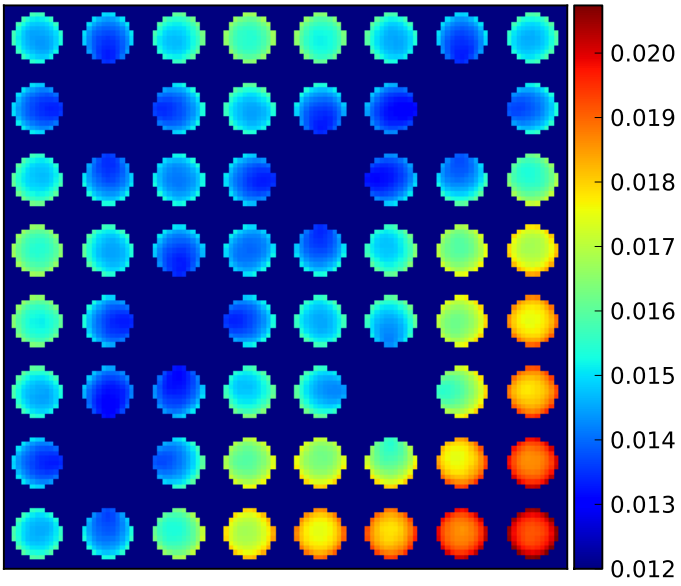


Figure 3.8: Fission distribution estimated from fractional aMFP KDE for the assembly of pincells.

comparing point-wise KDE quantities to volume-average histogram quantities in regions where the gradient of the flux is changing. Furthermore, the histogram bins overlap both the absorber and water, further complicating the comparison to the point-wise KDE results. Thus, these discrepancies are not a fault of the fractional aMFP KDE but rather due to the problems associated with using Cartesian histograms to capture distributions in problems with curved surfaces. Additionally, the depression of the thermal flux in the vicinity of the control rods can be seen via the fission reaction rate in Figure 3.8. The fractional aMFP KDE captures the azimuthal variation in the fission reaction rate in fuel pins adjacent to control rods. Direct comparisons between the reaction rates obtained via the KDE and the histogram are not made here due to the Cartesian histogram bins overlapping the material interfaces.

Table 3.4 shows that the speedup from using the GPU algorithm for 120×120 tally points is relatively constant versus the number of MPI processes, with a final speedup of 1.8 with 16 MPI processes. This speedup is lower than those seen in Table 3.5 when using 240×240 tally points due to the lower tally point density. The maximum speedup achievable by using GPUs occurs when the entire KDE tally can be computed on the GPUs without the CPU having to wait on the GPU. Thus, if the KDE tally is less expensive on the CPU then the GPU speedup will be smaller as long as the GPUs are not already saturated with work. The

Table 3.4: History calculation rates and speedups from the single-precision and double-precision GPU tallies compared to the CPU-only tallies for the 2-D pincell assembly problem with 120×120 tally points for 1 to 16 MPI processes.

	N MPI Proc.	Active (Histories/s)	Inactive (Histories/s)	Speedup vs N MPI Proc.
GPU Single Precision	1	10063	17341	1.64
	2	19410	33513	1.68
	4	37008	63657	1.86
	8	66971	117209	1.98
	16	117615	221870	1.79
GPU Double Precision	1	10015	17304	1.63
	2	19491	33704	1.69
	4	36833	63902	1.85
	8	66918	118339	1.98
	16	116889	221237	1.78
CPU Double Precision	1	6130	17555	-
	2	11538	34157	-
	4	19914	64529	-
	8	33866	119041	-
	16	65805	242964	-

Table 3.5: History calculation rates and speedups from the single-precision and double-precision GPU tallies compared to the CPU-only tallies for the 2-D pincell assembly problem with 240×240 tally points for 1 to 16 MPI processes.

	N MPI Proc.	Active (Histories/s)	Inactive (Histories/s)	Speedup vs N MPI Proc.
GPU Single Precision	1	11591	17560	4.30
	2	22357	34106	4.66
	4	41818	64269	4.73
	8	74568	116962	5.09
	16	98245	227637	3.73
GPU Double Precision	1	11448	17676	4.24
	2	22252	34073	4.64
	4	41442	64897	4.69
	8	73157	120479	4.99
	16	71824	227637	2.73
CPU Double Precision	1	2703	17676	-
	2	4800	34073	-
	4	8844	64897	-
	8	14661	120479	-
	16	26349	227637	-

relatively constant speedup as well as the similar speedups obtained by the single and double precision GPU tallies are an indication that each GPU is able to process all tally arrays sent to it before the CPUs send their next batch of tally arrays to the GPU. In this scenario the KDE tally is essentially free with the exception of looking up additional cross sections after every collision. This lookup of additional cross sections is the source of the majority of the discrepancy between the active and inactive history calculation rates for the GPU tallies in Table 3.4. This discrepancy could be reduced by using a local cross-section lookup like that used for the cylindrical MFP KDE.

Table 3.5 shows a decline in speedup for both single and double-precision tallies when increasing the number of MPI processes from 8 to 16 for the problem with 240×240 tally points. The results also show that the speedup for the algorithm with single-precision is better than that of the double-precision algorithm when using 16 MPI processes. These two factors indicate that the GPUs are saturated with work when using 16 MPI processes with 240×240 tally points. In fact, the active particle calculation rate decreases when doubling the number of CPU cores in use from 8 to 16 for double precision tallies. This is due to the over saturation of the GPU. Assuming that all MPI processes generate sets of sample arrays at similar rates, increasing the number of MPI processes from 8 to 16 means that the two GPUs must process 8 sets of sample arrays each rather than 4 per set of sample arrays generated by an MPI process. Adding more work to the already saturated GPU only increases the amount of time a CPU has to wait before the GPU finishes processing its previous sample arrays and is ready to receive a new set of arrays. Thus, this shows that there is a maximum problem complexity that the GPUs are capable of handling before adding more processors per GPU becomes detrimental.

CHAPTER 4

Applications of the Cylindrical MFP KDE to Reactor Physics Problems

4.1 Overview

This thesis began with the goal of developing a KDE that can accurately estimate reaction rates in reactor physics problems in continuous energy in order to apply KDEs to challenging PWR reactor physics problems and multi-physics coupling problems. The development and verification of the cylindrical MFP KDE in Section 2.9 has demonstrated that a KDE now exists that is capable of accurately capturing distributions in a basic PWR pincell. The cylindrical MFP KDE is well suited for estimating distributions in PWR reactor analysis as it does not require an underlying mesh, enabling the KDE to capture distributions in the square water and cylindrical fuel pin. Furthermore, using the number of MFP in the radial dimension only enables the cylindrical MFP KDE to capture the difficult to resolve rim effect due to resonance absorption interactions at the edge of the fuel pin. While the fractional MFP KDE and the MFP KDE with a minimum bandwidth are also capable of capturing these distributions, Section 2.5 showed that these estimators produce either excessive variance or bias at material interfaces. This chapter applies the cylindrical MFP KDE to more challenging problems in PWR analysis: a pincell with an Integral Fuel Burnable Absorber (IFBA) coating in Section 4.2, a quarter-assembly of pincells with control rods in Section 4.3, a depleted fuel pin at end-of-life in Section 4.4, and finally a pincell with results obtained on an unstructured mesh in Section 4.5.

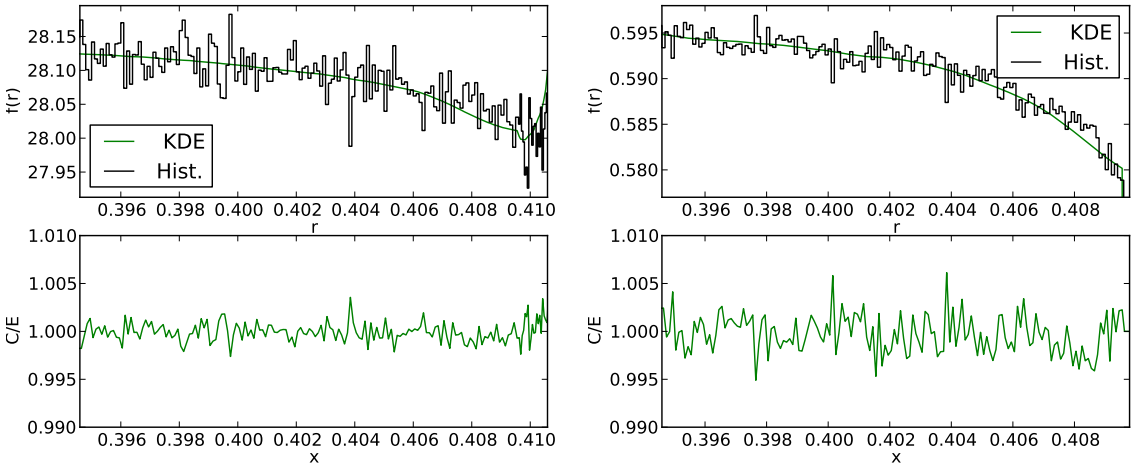
The goal of applying KDEs to multi-physics coupling problems was never realized due to the problem complexity required to see a noticable difference in coupled multi-physics problems conducted using histogram tallies and KDEs. Initially it was believed that a coupled Monte-Carlo neutronics and heat conduction simulation in a PWR pincell would be sufficient to show the advantages of using KDEs over histogram tallies to estimate fission

reaction rates in order to determine the power density within the fuel. To test this, a 1-D pincell sensitivity study was conducted, with results suggesting that the heat conduction equations and fuel centerline temperatures are insensitive to noise present in the Monte Carlo estimated power density profiles. Thus, it was concluded that a more complex problem with proven instabilities was required to demonstrate the advantages of using KDEs over histograms for multi-physics coupling problems. However, this type of problem goes beyond the scope of this thesis. Even so, the application of KDEs to obtain reaction rate densities on an unstructured mesh in Section 4.5 demonstrates that KDEs are now ready to be applied to more complex reactor physics problems with multi-physics coupling.

4.2 IFBA Pincell

The cylindrical MFP KDE used to obtain reaction rate densities on the CASL 2-D pincell benchmark problem with IFBA [41]. IFBA is used as a neutron poison to shield specific fuel rods at the start of life. Over the course of fuel cycle the IFBA is depleted and the fuel that was previously shielded begins to burn, enabling reactor designers to extend fuel cycle length. The problem set up is the same as the 2-D pincell with cladding detailed in Section 2.9.2.1 with the addition of a 10 μm layer of ZrB_2 between the fuel and the cladding from 0.4096 cm to 0.4106 cm. The density of the cladding is not adjusted to compensate for the reduced cladding volume. Reference collision histogram results are obtained on a cylindrical mesh with 8 azimuthal bins and a radial mesh with 20 bins from 0 to 0.3946 cm, 150 bins from 0.3946 cm to 0.4096 cm, 40 bins from 0.4096 cm to 0.4116 cm, 10 bins from 0.4116 cm to 0.475 cm, and 20 bins from 0.475 cm to 0.63 cm. KDE tally points are placed at the center of the histogram bins. Two KDE regions are defined for the pincell with one region encompassing the fuel and IFBA and the other region containing the cladding and water. The distributions outside of the rim region are similar to that of the pincell without IFBA in Figures 2.37 and 2.38 and as such results for the IFBA pincell are only shown in the rim region and IFBA. Flux, fission, and absorption distributions for the cylindrical MFP KDE and reference histogram along with C/E values are plotted in 1-D along the 22.5° azimuth for the IFBA pincell in Figures 4.1 - 4.2. The distributions do not change significantly in the azimuthal dimension, and as such the plots in Figures 4.1 and 4.2 are representative of the distributions along the remaining seven azimuthal bins. The cylindrical aMFP KDE is also used to estimate the flux, fission, and absorption distributions with the fission distribution in the fuel and absorption distribution in the IFBA shown in Figure 4.3. The flux results as well as the fission and absorption results outside of the rim region and

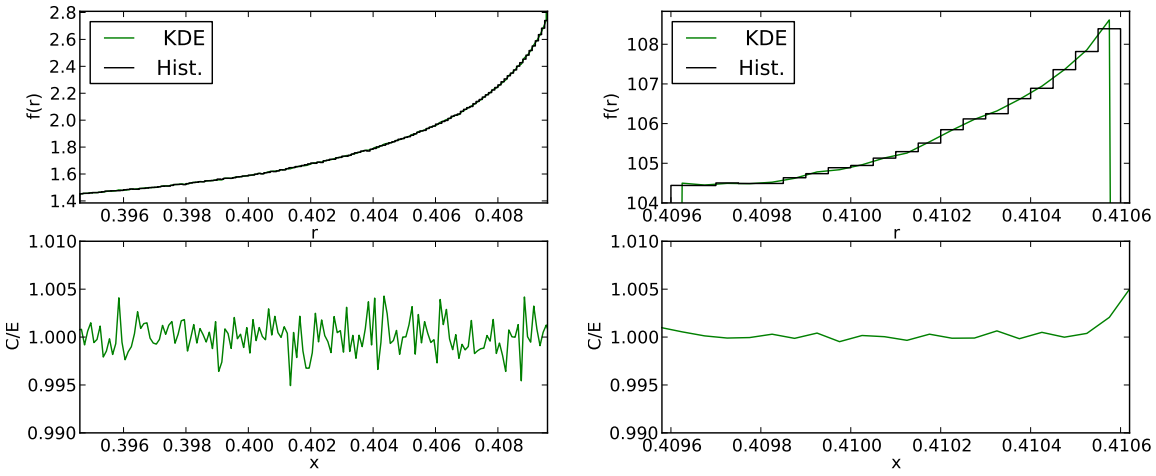
IFBA either match or are similar to those of the cylindrical MFP KDE and as such are omitted for clarity.



(a) Flux in rim region & IFBA

(b) Fission in rim region

Figure 4.1: Flux and fission distribution comparisons between the cylindrical MFP KDE and histogram for the pincell problem along the 22.5° azimuth.



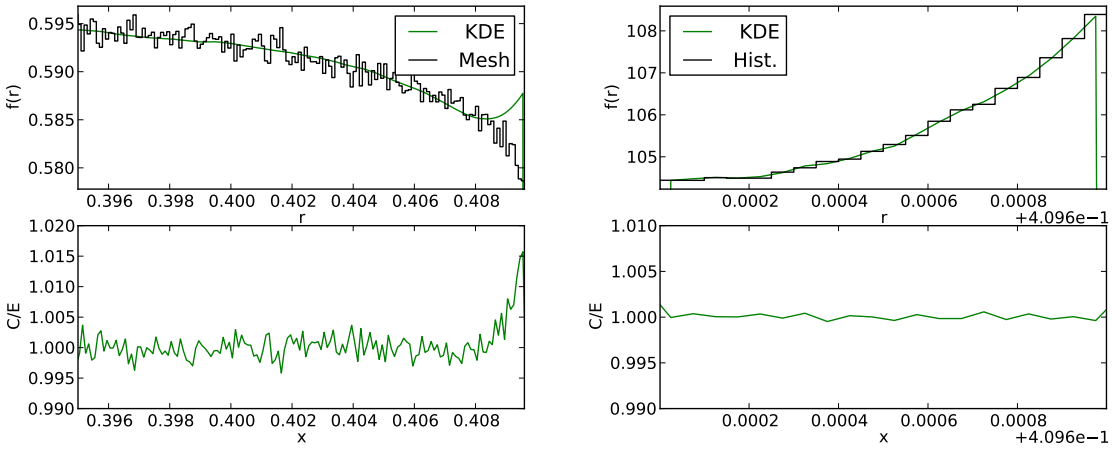
(a) Absorption in rim region

(b) Absorption in IFBA

Figure 4.2: Absorption distribution comparison between the cylindrical MFP KDE and histogram for the IFBA pincell problem along the 22.5° azimuth.

The results indicate that the cylindrical MFP KDE is capable of capturing flux and reaction rate distributions in IFBA problems. The flux results agree within 0.5% at all points, and 0.25% at all points in the fuel. The flux distribution in the rim region in Figure 4.1a demonstrates this agreement, with the cylindrical MFP KDE providing far smoother estimates

than the reference histogram. Similar results are obtained for the fission distribution in Figure 4.1b with the estimators agreeing within 0.5% at most points with a maximum difference of 0.625% and the cylindrical MFP KDE producing smoother results than the reference histogram. For the absorption distribution, the estimators agree within 0.5% in the rim region and within 0.25% in the IFBA. However, there is a slight bias at the edges of the IFBA absorption distribution with a disagreement of up to 5-σ at the outermost bin within the IFBA. Even so, with a maximum disagreement of 0.25% in the IFBA the cylindrical MFP KDE shows good agreement with the reference histogram. It should be noted that the increase in the disagreement at the outer rim of the IFBA is an artifact of the plot; the uncertainty in the cladding is larger due to a fine histogram mesh being used there, causing larger disagreements than those seen in the IFBA. Overall, the cylindrical MFP KDE is capable of accurately estimating flux and reaction rate distributions in pincell problems containing geometrically thin, strong absorbers.



(a) Fission in Rim Region

(b) Absorption in IFBA

Figure 4.3: Comparison of fission distribution in rim region (left) and absorption distribution in IFBA (right) between the cylindrical aMFP KDE and the reference histogram.

Conversely, the cylindrical aMFP KDE fails to accurately predict the fission reaction rate in the rim region for the IFBA problem, seen in Figure 4.3a. From the plots in Figure 4.3 it is clear that the cylindrical aMFP KDE produces a bias in the fission reaction rate density at the material interface between the fuel and IFBA. The fission reaction rate is over-estimated by 1.4%, corresponding to a difference of 8-σ between the results. This over-estimation is due to kernels centered at tally points in the fuel extending either too far into IFBA or through the IFBA and into the cladding. This issue with the aMFP KDE when estimating distributions in problems with thin, strong absorbers was previously demonstrated in Section 2.6.1. The aMFP KDE kernel allows collisions in the far side of the IFBA and the cladding to contribute

to the score at tally points in the fuel without being sufficiently attenuated through the IFBA. The cylindrical MFP KDE does not exhibit this behavior since it attenuates the score through the IFBA by using the number of MFPs between the collision site and the tally point. This issue with the aMFP KDE would be exacerbated by using fewer particle histories as the bandwidth would increase, resulting in an increase in the bias at the edge of the fuel.

Even so, the cylindrical aMFP KDE accurately predicts the absorption reaction rate in the rim region of the fuel and the IFBA, with all points within the rim region of the fuel agreeing within 0.4% of the reference histogram and points within the IFBA agreeing within 3-σ of the reference histogram. The aMFP KDE produces a lower disagreement in units of standard deviations in the IFBA compared to the MFP KDE due to the aMFP KDE producing smaller spatial bandwidths. Since the IFBA has the largest cross section among the materials in the problem, the tally points in the IFBA will have a smaller spatial bandwidth when using the aMFP KDE versus using the MFP KDE. These smaller bandwidths provide a lower bias, and thus the aMFP KDE will have less bias compared to the MFP KDE for tally points within the IFBA. Even so, the cylindrical aMFP KDE disagrees with the reference histogram by as much as 0.35% at the outer edge of the IFBA (along a different azimuth than is seen in Figure 4.3b); a direct result of the increased variance in the aMFP KDE resulting from a smaller bandwidth. Overall, the aMFP KDE is not recommended for use in problems with geometrically thin, strong absorbers to minimize bias in materials surrounding the strong absorber.

4.3 Quarter Assembly

To demonstrate the use of cylinder KDEs in a configuration more representative of a reactor core, a quarter 16×16 assembly with reflecting boundary conditions is modeled with tallies conducted over select pincells. The quarter assembly is depicted in Figure 4.4 and is comprised of pincells defined in the CASL 2-D pincell benchmark problem [41] that were used previously in Section 2.9.2.1. Yellow pins represent UO_2 fuel with Zircaloy-4 cladding and black pins represent B_4C control rods with SS304 cladding. Fuel pins use the same cylindrical mesh as described in Section 2.9.2.1, while tallies over a control rod use 8 azimuthal bins with 15, 150, 5, and 10 bins from 0 to 0.371, 0.386, 0.484, and 0.63 cm, respectively. Designating the bottom left pincell as pincell number (1,1), tallies are conducted over pincells (6,3), (6,4), and (4, 5) referred to as pincells 1, 2, and 3 respectively. The cylindrical KDE tallies use one KDE region for each pincell. The simulations are run using 512 MPI processes distributed across 64 Intel Xeon E5-2670 processors with

2,000,000 particles per batch with 20,000 batches and 200 inactive batches. KDE results are obtained using the cylindrical MFP KDE on the GPU with one NVIDIA M2090 GPU per processor. The flux and absorption reaction rate results along the 337.5° azimuth for pincell 1, the control rod, across the pincell and within the rim region are shown in Figures 4.5 and 4.6 with the estimated distributions shown on top and the C/E (KDE/histogram) comparisons shown below. The 337.5° azimuth is chosen as the KDE and reference solution show the largest disagreement along this azimuth. Pincells 2 and 3 show similar results, with distributions varying in the azimuthal dimension depending on the pincells' proximity to control rods. The 2-D flux distributions of pincells 2 and 3 are shown in Figure 4.7, with their fission distributions shown in Figure 4.8. The corners of Figures 4.7 and 4.8 are comprised of water but are not tallied due to the constraints of the cylindrical mesh. Plots of the fission and absorption reaction rates in pincell 2 along the $22.5°$ azimuth across the radius of the pincell (left) and the rim region (right) are shown in Figures 4.9 and 4.10, respectively. Pincells 2 and 3 present similar results, and as such 1-D radial comparisons for pincell 3 are omitted for clarity.

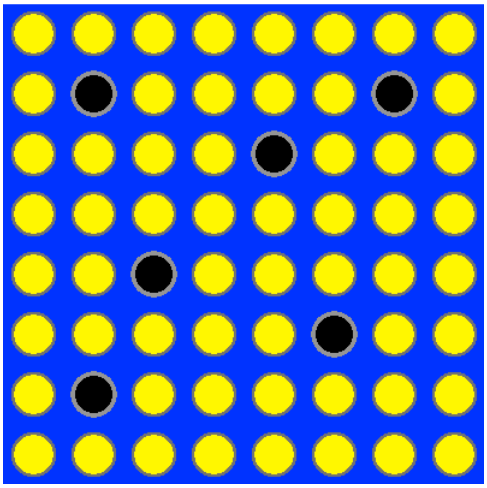
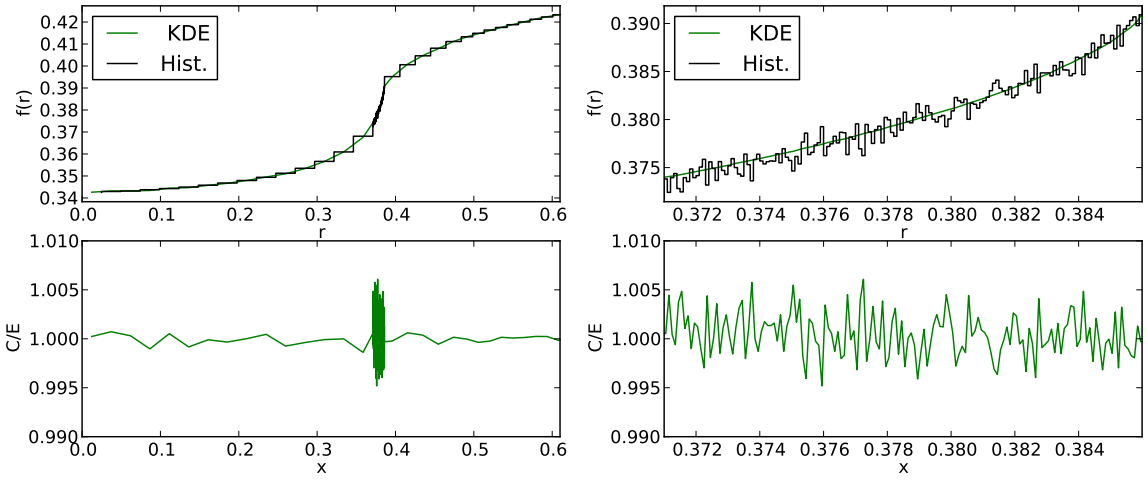


Figure 4.4: Depiction of the assembly of pincells.

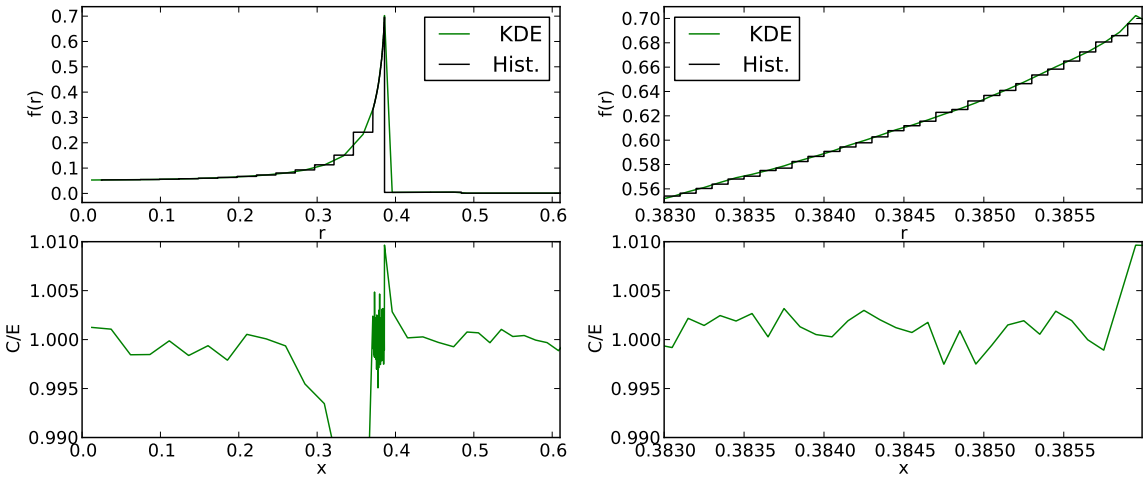
The flux results in Figure 4.5 show that the cylindrical MFP KDE agrees with the reference histogram within 0.5% at most points in the distribution, with disagreement increasing in the rim region. This increase in disagreement is due to the variance of the histogram, as seen in Figure 4.5b. The average relative uncertainty of the cylindrical MFP KDE and histogram in the rim region is approximately 0.04% and 0.4%, respectively, and the two distributions agree within 3-σ at all points within the rim region. Thus, this increased disagreement is due to statistical uncertainty rather than any perceived bias in the KDE results.

The absorption distribution in Figure 4.6 shows similar behavior, with results agreeing within



(a) Across pincell (b) Rim region

Figure 4.5: Comparison of flux distributions estimated using the cylindrical MFP KDE and a collision histogram on a cylindrical mesh across pincell 1 and within the rim region in pincell 1 along the 337.5° azimuth.

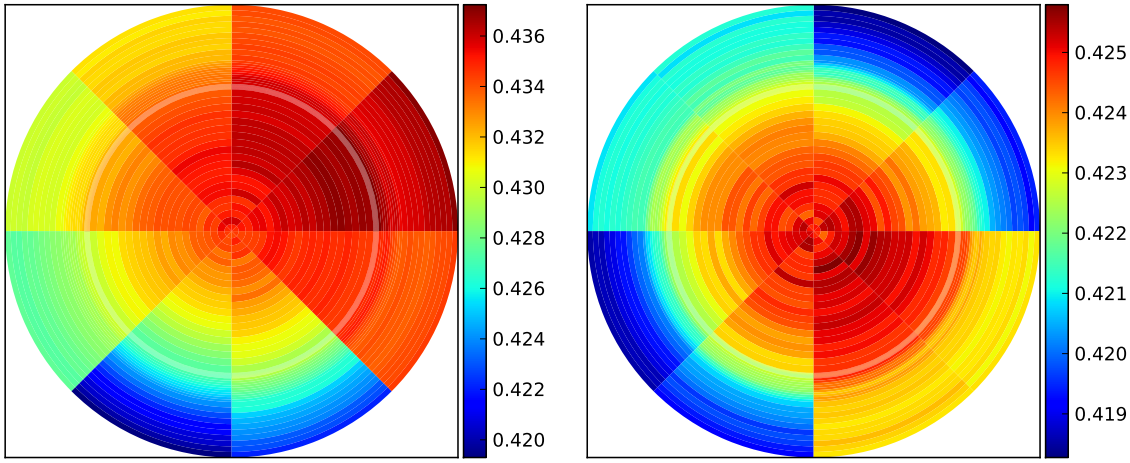


(a) Across pincell (b) Rim region

Figure 4.6: Comparison of absorption reaction rate densities in pincell 1 estimated using the cylindrical MFP KDE and a collision histogram along the 337.5° azimuth.

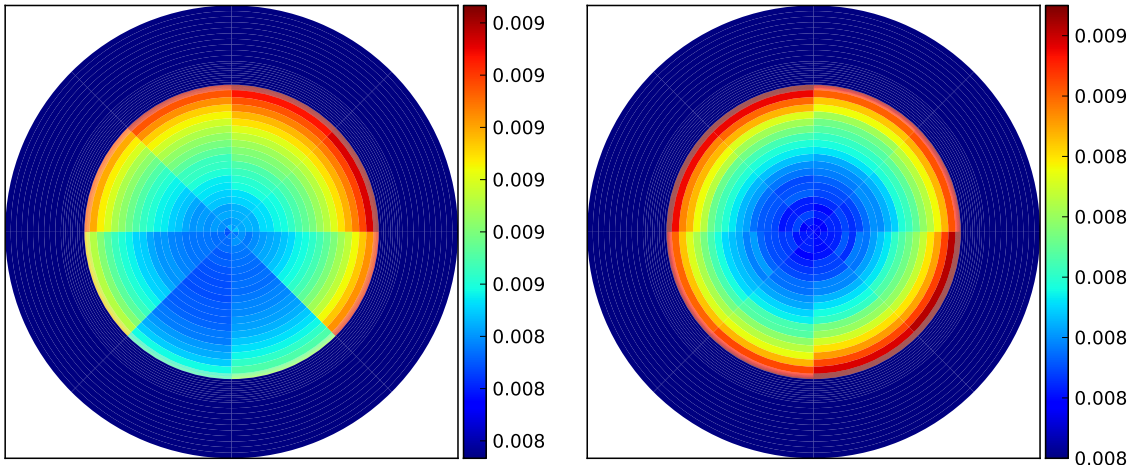
0.5% with the exception of several points at the edge of the control rod and several points prior to the start of the fine mesh in the rim region in the control rod. The disagreement prior to the rim region is due to comparing point-wise quantities from the KDE to volume-average histogram quantities; increasing the resolution in this area would reduce the disagreement. The disagreement of 1% in the few μm prior to the control rod-cladding interface is due to

bias in the KDE. This bias is a result of using the "normal reference rule" with one KDE region defined over the pincell. The bias could be reduced by defining two regions over the pincell, one for the absorber and one for the cladding and water, similar to what is done in Section 4.2. Defining regions this way would reduce the bandwidth, thus reducing the bias at a cost of increased variance throughout the rest of the control rod. Alternatively, a space-dependent bandwidth could be defined for the pincell to reduce bias at the material interface without unnecessarily increasing variance throughout the rest of the problem. Even so, the disagreement extends for less than 2 μm into the control rod. Thus, the cylindrical MFP KDE is capable of accurately capturing flux and reaction rate distributions within a control rod.



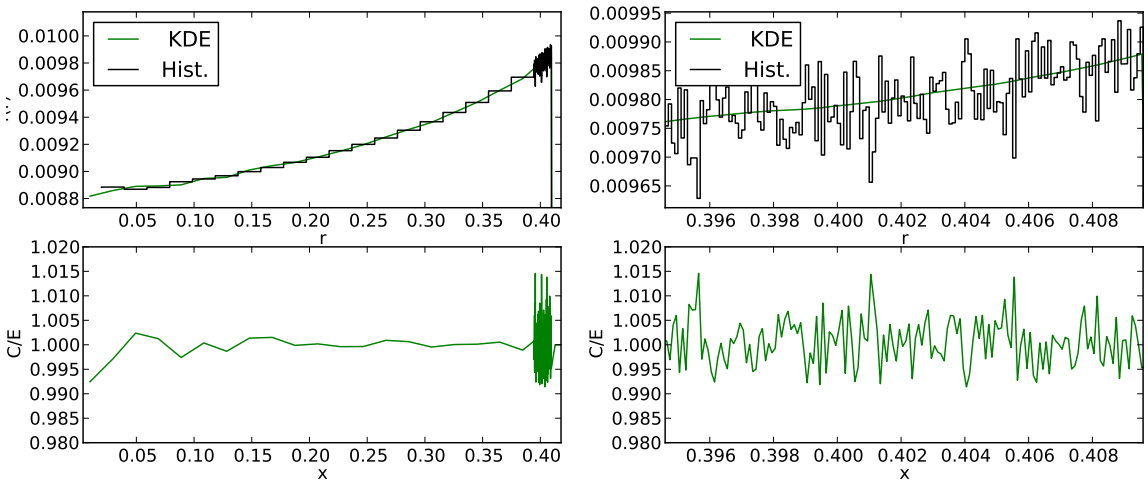
(a) Pincell 2 (b) Pincell 3

Figure 4.7: Cylindrical MFP KDE flux distributions in pincells 2 and 3.



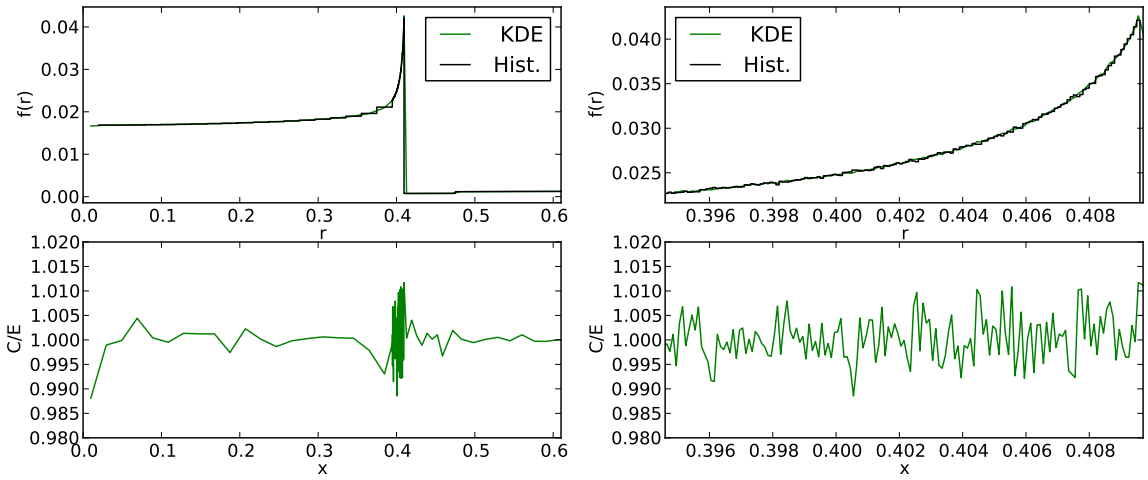
(a) Pincell 2 (b) Pincell 3

Figure 4.8: Cylindrical MFP KDE fission distributions in pincells 2 and 3.



(a) Across pin (b) Rim region

Figure 4.9: Comparison of fission reaction rate densities in pincell 2 estimated using the cylindrical MFP KDE and a collision histogram along the $22.5°$ azimuth.



(a) Across pin (b) Rim region

Figure 4.10: Comparison of absorption reaction rate densities in pincell 2 estimated using the cylindrical MFP KDE and a collision histogram along the $22.5°$ azimuth.

The results for pincells 2 and 3 are similar to those of pincell 1. The flux and fission distributions seen in Figures 4.7 and 4.8 show the azimuthal variation in the distributions due to their proximity to the assembly control rods. The distributions are depressed in the direction of neighboring control rods, as expected from theory. The histogram and KDE radial flux results from pincell 2 agree within 0.25% outside of the rim region and

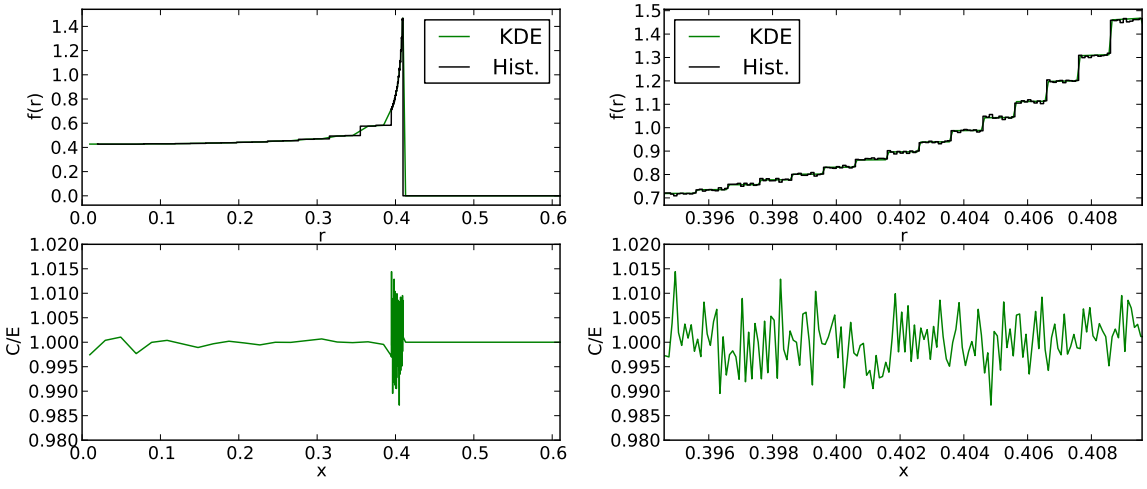
within 1% in the rim region. Fission and absorption results agree within 1% everywhere outside of the rim region and within 2% in the rim region. The few points that do not agree within 1% in the rim region are not concentrated at the material interface, but are randomly distributed through the rim region. Furthermore, 99.5% and 99.6% of the tally points and histogram bins agree within 3-σ for the fission and absorption distributions, respectively, with average relative uncertainties of approximately 0.4% and 0.006% for the histogram and KDE, respectively, in the rim region of the fission distributions and 0.35% and 0.25% respectively for the absorption distribution. Thus, this disagreement in the assembly results can be attributed to statistical uncertainty rather than bias in the KDE results.

With regards to performance the picture is much the same as for the 2-D pincell in Section 3.6.3. The KDE/histogram FOM ratio for the flux, fission, and absorption distributions in pincell 2 is 90, 67, and 2.4, respectively, with the other pincells producing similar results. The active tally runtimes for the histogram tally and KDE are 4.75 and 5.15 hours, respectively. The relative uncertainties in the KDE fission results in the rim region are lower than those of the histogram by factors of 7.5 to 9.5. For the absorption reaction rate, the KDE produces lower uncertainties by a factors of 1-2. Outside of the rim region the KDE produces results with larger relative uncertainties compared to the histogram by factors of 3-5 for the absorption reaction rate and a factor of 1.4 for the fission reaction rate. Since the FOM for the KDE decreases with increasing particle number, running less particles would improve these ratios in favor of the KDE. The uncertainties in the KDE results could be reduced by implementing a minimum bandwidth for these tallies, or by using a volume-average KDE as is demonstrated in Section 3.6.3. Alternatively, a space-dependent bandwidth could be implemented to reduce variance without incurring significant bias by using larger bandwidths in the majority of the fuel where the curvature is low and smaller bandwidths in the rim region of the fuel.

4.4 Depleted Fuel Pin

A depleted fuel pin model was created to explore the rim effect at end-of-life fuel pins. The beginning-of-life pincell model is the same as that of the 2-D pincell with cladding in Section 2.9.2.1. Serpent [45] was used to deplete the fuel pin, with an initial burn up step of 0.15 MWD/kgHM to burn in Xe-135 followed by 15 steps of 5 MWD/kgHM. The depletion mesh is composed of one azimuthal bin with 10 radial bins from 0 to 0.3946 cm, 15 radial bins from 0.3946 cm to 0.4096 cm, and two radial bins in the cladding. These depleted materials are used to create an OpenMC model of the depleted fuel pin using

the same geometry as the depletion model with materials defined using the results of the depletion simulation. All isotopes generated from Serpent with atom densities greater than 10^{-7} atoms per barn-cm as well as all Hf, Sm, and Xe isotopes are included in the material compositions of the OpenMC model. The tallies conducted on the depleted pincell are the same as those detailed in Section 2.9.2.1. The simulations are run using 60 MPI processes across 12 eight-core Intel Xeon E5-2670 processors with 2,000 batches, 100 inactive batches with 200,000 particles per batch. Five out of eight cores per processor are used due to CPU memory constraints. KDE results are obtained using the cylindrical aMFP KDE with one NVIDIA M2090 GPU per processor. The histogram and KDE results are obtained on the same simulation with a total active tally runtime of 4.2 hours. Using the cylindrical MFP KDE algorithm on this problem results in an increase in execution time of approximately 3% compared to using the collision histogram tally alone. The small increase in runtime from using KDEs is due to the large cost of the particle transport process, in part from accessing the data for all of the isotopes in the depleted fuel during the cross section calculation routine as well as the increased cost in the particle tracking routine due to defining each depleted material as a unique cell. The fission and absorption distributions along the 22.5° azimuth obtained using the cylindrical aMFP KDE are compared to histogram results in Figures 4.11 and 4.12 with reaction rate distributions shown on top and C/E (KDE/histogram) comparisons shown below.

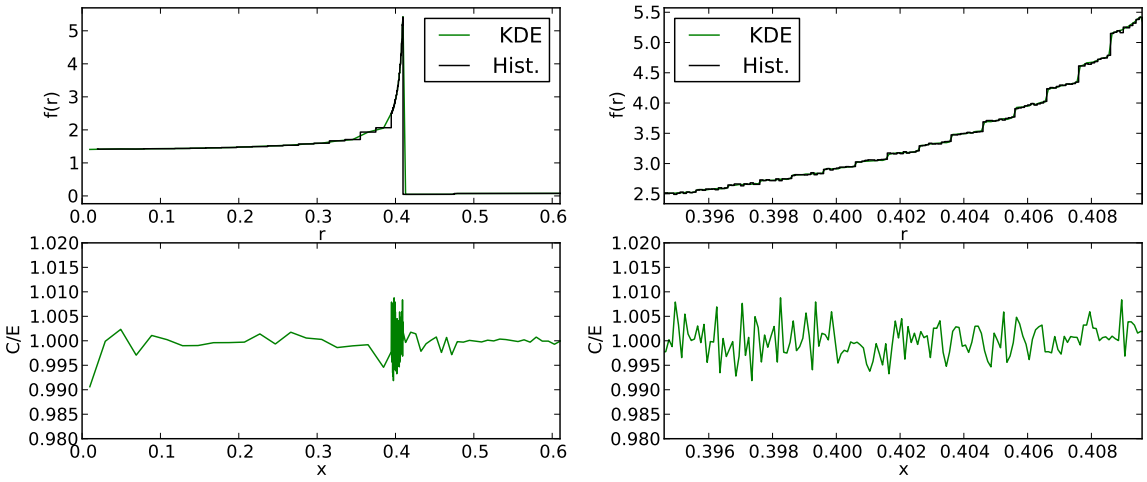


(a) Across pin

(b) Rim region

Figure 4.11: Comparison of fission reaction rate densities in the fully depleted pincell estimated using the cylindrical MFP KDE and a collision histogram along the 22.5° azimuth.

The fission distribution results in Figure 4.11 show that the cylindrical MFP KDE accurately



(a) Across pin

(b) Rim region

Figure 4.12: Comparison of absorption reaction rate densities in the fully depleted pincell estimated using the cylindrical MFP KDE and a collision histogram along the 22.5° azimuth.

captures the rim effect at end-of-life with results agreeing with the histogram within 1% at most points in the distribution with all points agreeing within 2%. The disagreements are due to variance in the histogram results, as there is no bias detectable in the KDE results with all tally points and histogram bins agreeing within 3-σ. The absorption distribution results in Figure 4.12 show similar results but with all points agreeing within 1%. Furthermore, all tally points agree within 3-σ with the exception of the tally points prior to the start of the high resolution binning within the rim region due to comparing volume-average histogram results to point-wise KDE results. Thus, the cylindrical MFP KDE is capable of accurately capturing distributions in depleted materials.

While Figures 4.11 and 4.12 show that the cylindrical MFP KDE accurately captures the discontinuities in the reaction rates at the interfaces of the depletion regions, these discontinuities are not physically real. The discontinuities are created as a direct result of the depletion of materials on a mesh using histogram tallies. Since KDEs capture reaction rates at points, it is possible to deplete materials at tally points and use this information to create a continuous distribution of the isotopes in a material. For example, placing KDE tally points at the nodes of an unstructured mesh would enable a continuous representation of the isotope density via the FEM. However, this would require tracking particles in materials with continuously varying cross sections via either numerical integration [46] or delta-tracking [47, 48]. This topic is discussed further in Section 5.4.

4.5 Unstructured Mesh Pincell

This section demonstrates that the MFP KDE can be used to obtain estimates on an unstructured mesh. While this work does not apply KDEs to conduct multi-physics coupling, this section is a proof-of-principle to show that the MFP KDE can be used to estimate distributions on an unstructured mesh which could then be used to obtain a FEM representation of neutronics distributions for multi-physics coupling applications. An unstructured mesh was created using Abaqus [49] for the 2-D pincell with cladding model described in Section 2.9.2.1 with MFP KDE tally points placed on the nodes of the unstructured mesh. The mesh is comprised of second-order quadrilaterals and triangles and is depicted in Figure 4.13. The water, cladding, and fuel are defined as distinct parts with their own meshes within Abaqus in order to preserve the discontinuity in the reaction rate densities between materials. The tally points are read in to OpenMC from the nodes of the exported Abaqus mesh and are placed into nearby neighbor bins just as they would be if the tally points are defined using a structured mesh.

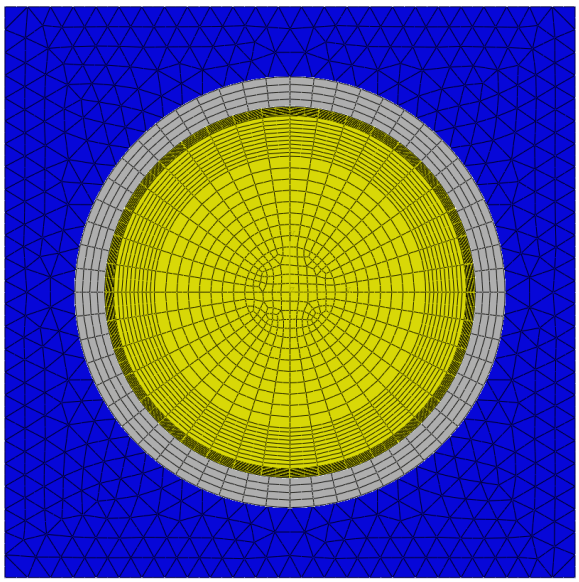


Figure 4.13: Unstructured mesh of the 2-D pincell with cladding.

Current Monte Carlo codes that enable histogram tallying on unstructured meshes also define their geometry in these simulations using the unstructured mesh [33, 34, 50]. In this work, the particle transport routines are conducted using the CSG representation of the model when tallying on the unstructured mesh. This poses an issue since the unstructured mesh representation of a cell will not conform exactly to the CSG representation of that cell. Thus, tally points specified on the nodes of an unstructured mesh representation of

a cell can exist in neighboring cells when placed in the CSG model. For example, tally points defined on the unstructured mesh representation of the UO_2 fuel may exist in the cladding surrounding the fuel in the CSG model. In order to maintain the correct material definitions of the unstructured mesh, the materials at the tally points are defined using the unstructured mesh representation of the model. This has the effect that the tally points are sampling the neutron energy spectrum created from the CSG model using materials defined on the unstructured mesh. Thus, if the mesh does not adequately conform to the underlying geometry then unphysical results will be generated by the KDE. To minimize this effect, second-order elements are used to create the unstructured mesh geometry as they conform better to curved surfaces.

The cylindrical MFP KDE and fractional aMFP KDE are used to estimate the flux distribution and the fission and absorption reaction rate densities on the nodes of the unstructured mesh shown in Figure 4.13. The cylindrical MFP KDE is used for all tally points whose distance-based cylindrical kernels do not overlap the center of the fuel pin or the boundary of the pincell. For all other tally points the fractional aMFP KDE is used. Thus, the fractional aMFP KDE is used for all tally points whose radial positions r meet the conditions $r < \sqrt{5}h_r$ or $r > 0.63 - \sqrt{5}h_r$. One KDE region was defined for the pincell resulting in a radial spatial bandwidth of 0.0041 cm for the cylindrical MFP KDE and spatial bandwidths of 0.0083 cm in the x and y dimensions for the fractional aMFP KDE. The cylindrical MFP KDE is the same as that used in previous sections with a uniform kernel with a bandwidth of $\pi/16$ in the azimuthal dimension and an Epanechnikov kernel in the radial dimension. The simulation was run using 200,000 particles per batch with 1,350 batches and 100 inactive batches. The flux distribution is shown in Figure 4.14 and the fission and absorption reaction date densities are shown in Figure 4.15.

A robust quantitative comparison is not performed here as routines were not written to enable OpenMC to obtain estimates on an unstructured mesh. However, the distributions can be qualitatively compared to those obtained on the structured mesh in Section 2.9.2.1. The flux distribution in Figure 4.14 shows the same trends as those seen in Figure 2.37 with the flux peaking in the cladding adjacent to the fuel with scalar fluxes of approximately 30 cm^{-2}. The variance in the flux is higher at the problem boundaries and the corners of the pincell due to the fractional aMFP KDE being used there. The variance is also higher in the center of the pin due to the fractional aMFP KDE being used there as well as due to the reduced kernel support area of the cylindrical MFP KDE at smaller radii. The fission distribution in Figure 4.15a also shows the correct distribution compared to Figure 2.38, with peaking at the rim of the fuel pin with a fission reaction rate density of 0.96 cm^{-3}.

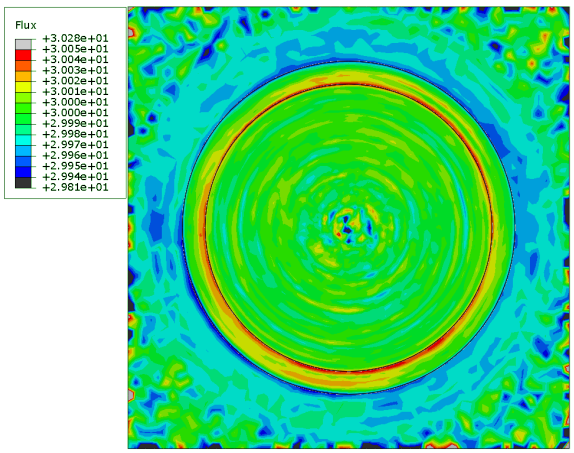
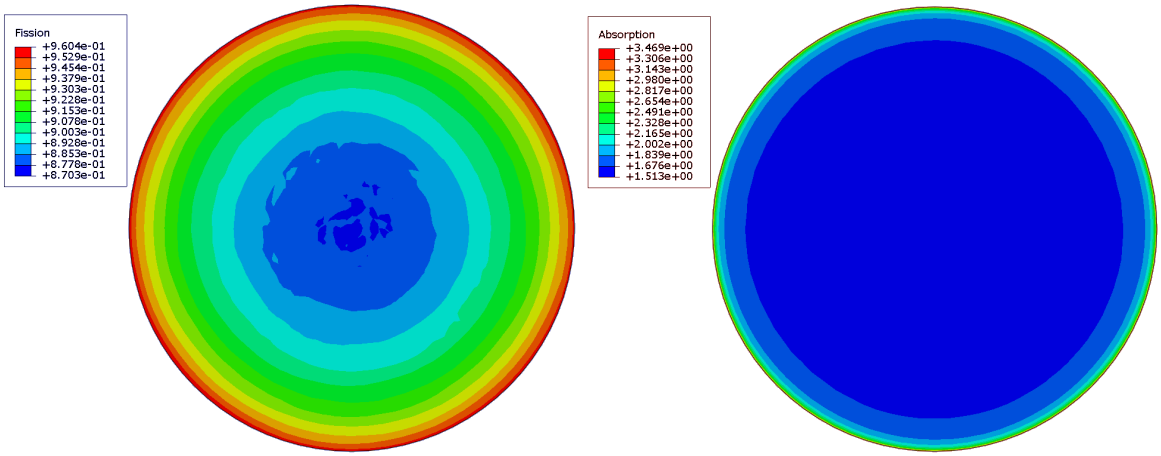


Figure 4.14: Flux obtained on the unstructured mesh of the 2-D pincell with cladding using the cylindrical MFP KDE.



(a) Fission (b) Absorption

Figure 4.15: Fission and absorption reaction rate densities within the fuel obtained on the unstructured mesh of the 2-D pincell with cladding using the cylindrical MFP KDE.

The noise in the fission distribution in the middle of the fuel pin is due to the small relative change in the fission reaction rate across the pin, ranging from 0.87 cm^{-3} at the center of the pin to 0.96 cm^{-3} at the very edge of the pin. Similarly, the absorption distribution on the unstructured mesh in Figure 4.15b has the same behavior as that of the absorption distribution on the structured mesh seen in Figure 2.39 with peaking occurring at the rim with values of approximately 3.4 cm^{-3}. Thus, the cylindrical MFP KDE can accurately estimate distributions on an unstructured mesh for a PWR pincell problem.

CHAPTER 5

Summary & Future Work

5.1 Mean Free Path KDEs For Estimating Reaction Rates

The goal of this work was to develop KDEs capable of accurately estimating reaction rates in reactor physics problems to facilitate their use in simulations with multi-physics coupling. KDEs show potential for obtaining smoother estimates of spatial distributions with lower variance compared to the traditionally used histogram tallies. Furthermore, the point-wise estimates provided by the KDE are independent of the desired resolution of the tally, making the estimator an attractive choice for use on unstructured meshes commonly used in simulations with multi-physics coupling. Prior work applied the distance-based KDEs to estimating scalar flux in 1-D and 2-D one-group reactor physics problems where the macroscopic total cross sections of the fuel, water, and control rods were equivalent, as well as in fixed-source shielding problems. However, these distance-based KDEs were not applied to estimate reaction rates, and they were not tested for reactor physics problems in continuous energy. Chapter 2 of this thesis develops the methodology for applying KDEs to estimate reaction rates in reactor physics problems in continuous energy and introduces new MFP KDEs to reduce inaccuracies at material interfaces generated by the distance-based KDE.

First, a KDE for obtaining reaction rates is introduced in Section 2.1. This KDE was created such that reactions in a material do not spread across material interfaces; estimates of the fission reaction rate do not spread into the cladding and water surrounding a fuel pin. The reaction rate KDE was verified using a 1-D, one-group reactor physics problem, however application to 1-D problems in continuous energy revealed a deficiency in the estimator. Using the current state-of-the-art KDEs for obtaining reaction rates in reactor physics simulations produced significant errors at material interfaces where one of the materials has a large absorption cross section. This inaccuracy is especially seen in materials with large

resonance absorption cross sections.

These errors in the reaction rates motivated the creation of the 1-D MFP KDE in Section 2.2. Rather than use the distance between the tally point and the particle track or collision site in the argument of the kernel function, the number of MFPs, also known as the optical distance, is used instead. The optimal bandwidth is adjusted to account for the change of the kernel argument and a new normalization coefficient is derived to maintain the kernel properties required by the KDE. Verification results for the MFP KDE are shown in a 1-D slab problem with an analytic solution and a 1-D reactor physics problem in continuous energy.

A multivariate MFP KDE is derived in sections 2.3 and 2.4 for univariate kernels in 2-D and for general geometries in 2-D and 3-D. Historically, KDE algorithms for Monte Carlo radiation transport defined the multivariate KDEs by using products of univariate kernels. However, the dependence of the multivariate MFP KDE on the underlying geometry introduces complications into this often-used method of extending the estimator to higher dimensions due to the derivation of the normalization coefficient. Simply stitching two 1-D MFP-based kernel functions together does not preserve the kernel properties for the multivariate MFP KDE and produces inaccurate results, and therefore a new normalization coefficient must be derived. However, this normalization coefficient is dependent upon the underlying geometry, and cannot be derived analytically for general geometries. Even so, the normalization coefficient is derived in 2-D for geometries with axis-aligned geometries. Since this is insufficient to simulate most reactor models, a general multivariate MFP KDE is derived in Section 2.4 using a general multivariate kernel. This multivariate MFP KDE is verified using a 2-D one-group pincell problem. The normalization coefficient yielded from the derivation of the multivariate MFP KDE is the same as that derived for the 2-D MFP KDE using a product of univariate kernels. As such, the two multivariate MFP KDEs produce similar results, with small disagreements of less than 0.1% at material interfaces. These methods were shown to work well for one-group problems, and therefore can be applied to multi-group problems with no loss in accuracy. However, the application of the multivariate MFP KDEs to problems in continuous energy produces spikes in the estimated distributions due to interactions at energies corresponding to resonances in the underlying cross sections. These spikes arise from the normalization coefficient derived for the multivariate MFP KDEs; the bandwidth in each dimension is proportional to Σ_t^{-1} and the normalization coefficient is proportional to Σ_t^2 in 2-D and thus large scores are contributed to localized areas whenever a collision or particle track occurs at a resonance energy.

To combat these spikes produced by the multivariate MFP KDEs, the minimum bandwidth and fractional MFP KDE are created in Section 2.5. Since the MFP KDE does not produce

spikes in 1-D and the normalization coefficient in 1-D only has one factor of Σ_t, it was theorized that alternative formulations of the multivariate MFP KDE that have normalization coefficients proportional to Σ_t rather than Σ_t^2 will not produce spikes. The fractional MFP KDE was created in Section 2.5 in order to maintain the dependence on the optical distance between the particle event and the tally point while only having one factor of Σ_t in its normalization coefficient. The fractional MFP KDE achieves this by using fractional powers of the number of MFP between the tally point and the collision site or particle track. While fractional powers of the cross section do not appear normally in the transport equation, it is important to note that this tally function does not alter the transport process and is simply another means of defining an adaptive bandwidth. The fractional MFP KDE was verified using a 2-D pincell problem and is successful in suppressing spikes in the estimated distributions, however it still produces large variances due to a reduction in the bandwidth at resonance energies, especially at material interfaces like the rim of a fuel pin in a PWR pincell problem.

As an alternative to the fractional MFP KDE, a minimum bandwidth was developed to forcibly increase the bandwidth and thus reduce the normalization coefficient whenever a resonance interaction occurred, effectively eliminating spikes from the multivariate MFP KDE. The MFP KDE can be thought of as a modification to the distance-based KDE where the bandwidth of the distance-based KDE is inversely proportional to Σ_t. A minimum bandwidth can be specified such that if a cross section is produced that would reduce the effective spatial bandwidth of the MFP KDE below some user-defined minimum, then the minimum bandwidth is used instead. While the minimum bandwidth reduces variance throughout the absorption distribution and removes spikes from the distributions, it produces unacceptably large biases of 5% and 15% at the fuel-water interface in a 1-D pincell problem for minimum bandwidths of 0.001 cm and 0.004 cm, respectively.

In addition to the spikes found in the multivariate MFP KDE, the estimator has another issue: performance. Requiring the calculation of the number of MFPs between each collision site and each tally point requires a considerable amount of ray tracing, especially in regions like material interfaces where node densities could be high. This ray tracing hinders the performance of the algorithm when material interfaces exist between the collision site and the tally point and it precludes the use of the track-length MFP KDE due to prohibitively expensive integrals being required for each tally point within the support region of each particle track. An approximation is made to the MFP KDE in Section 2.6 that uses the cross section at the tally point to approximate the number of MFP between the tally point and the collision site. This removes the need to conduct ray tracing, improving active tally

performance by a factor of 1.5 for a one-group 2-D PWR pincell problem without cladding. However, this performance increase is at the cost of additional bias in the estimator. Since histograms and KDEs are estimating fundamentally different quantities (volume-average and point-wise quantities, respectively) they cannot be quantitatively compared directly. A volume-average KDE is developed in Section 2.7 in order to determine the bias introduced by the aMFP KDE. The volume-average KDE is developed for the collision and track-length KDEs with derivations completed using the Epanechnikov kernel. The bias introduced from the aMFP KDE was shown to be less than 0.3% for all distributions in a 1-D PWR pincell problem.

Even though a combination of the fractional MFP KDE and minimum bandwidth is capable of producing sufficiently accurate estimates in 2-D reactor physics problems, a more robust estimation method is desired. Rather than modify the argument of the kernel function directly, an alternative set of kernel functions can be used that more closely match the geometry of the underlying problem. This led to the development of the cylindrical KDE with a wedge-shaped kernel in Section 2.8. The bias and variance of the cylindrical KDE is derived, with its asymptotic behavior found to be the same as that of the Cartesian KDE. The MFP KDE formulation is applied to the cylindrical MFP KDE in the radial dimension only for the purpose of capturing distributions in reactor physics pincell problems, yielding a normalization factor proportional to Σ_t. This enables the cylindrical MFP KDE to estimate distributions in PWR pincell problems without producing spikes. While the estimator is only applied to 2-D problems, it can be extended to 3-D by adding a univariate kernel in the z dimension in the same fashion as is done for the multivariate KDE in Cartesian coordinates. The cylindrical MFP KDE was verified on a PWR pincell with cladding, with disagreements of less than 0.5% in all distributions when compared to a reference collision histogram on a cylindrical mesh.

In summary, state-of-the-art KDEs in Monte Carlo radiation transport prior the start of this work had not been applied to estimate reaction rates in a non-analog fashion, and had yet to be applied to reactor physics problems in continuous energy. With the development and verification of the cylindrical MFP KDE, a KDE now exists that is capable of accurately capturing reaction rate densities in PWR reactor physics problems. This opens the door for the use of KDEs in more complicated reactor physics problems as well as their use in multi-physics coupling problems with Monte Carlo neutron transport.

5.2 GPU Acceleration of KDEs and Algorithmic Considerations

The ability of KDEs to obtain high-resolution estimates of distributions with reduced uncertainty compared to histograms comes at the cost of additional computation. KDEs require computing kernel functions for each tally point that lies within the support area of a kernel function centered on a collision site or particle track. This additional computation has the potential to make KDEs perform worse than histogram tallies, even though they can produce results with lower uncertainty for a set number of particle histories. Chapter 3 discusses the routines used by the KDEs to increase the efficiency of the algorithm on the CPU. The nearby neighbor list, maximum bandwidth, and local cross section lookup methods introduced in this chapter significantly reduce the computation time required to compute the KDE tallies, resulting in combined speedups of over 1,000 for problems studied in this thesis.

Chapter 3 also introduces methods for accelerating the KDE tally process using GPUs. High performance computer clusters often have GPUs on compute nodes as a means of accelerating performance benchmarks and algorithms that are vectorizable. Monte Carlo radiation transport codes are not well suited for acceleration on GPUs due to their use of logic statements leading to warp divergence as well as their random access of memory when looking-up and computing continuous energy cross sections. However, Monte Carlo algorithms are highly parallelizable and are often run on HPC machines that have GPUs attached to the compute nodes. Chapter 3 introduces an algorithm that enables KDE tallies to be computed on these previously idle GPUs while the transport process is performed on the CPU.

The independent nature of computing the contribution of scores from a particle event to tally points lends itself to vectorization, making KDEs well suited for computation on a GPU. Furthermore, the particle transport routines are independent from the tally routines, enabling the tally data to be exported to the GPU and the KDE routines to be executed simultaneously with the transport process on the CPU. GPU algorithms that follow this process are created for the collision MFP KDEs, and the optimization of the algorithms are discussed with emphasis on taking advantage of the underlying SIMT GPU architecture. Using a structure of arrays rather than an array of structures for the tally data, using single-precision calculations, and leveraging GPU shared memory results in a speedup of 2.9 in the GPU KDE kernel compute times. The GPU KDE algorithm produces problem-dependent speedups of 1.4 for the cylindrical MFP KDE and up to 5 for the fractional aMFP KDE

on a quarter-assembly of pincells. For most problems studied in this thesis the GPUs are capable of processing the tally data at a faster rate than the CPUs can generate it, making the KDEs essentially free for most tally sizes in this thesis. However, performance testing shows that there is a maximum problem complexity that GPUs can handle before the KDE GPU algorithm becomes the performance bottleneck.

With the MFP KDE algorithms tuned for performance, FOM comparisons are made between the MFP KDEs and histogram tallies. The collision fractional aMFP KDE is compared to the collision and track-length histograms on a 2-D boxcell problem, with the KDE producing larger FOMs for the flux and fission distributions compared to a collision histogram. However the KDE's performance is reduced when estimating the absorption distribution. This is again due to resonance absorption interactions causing a decrease in the bandwidth of the fractional aMFP KDE, resulting in increased variance in the absorption distribution compared to the estimates of the flux and fission distributions. Thus, the collision fractional aMFP KDE only produces better FOM for the absorption distribution than the collision histogram when a high resolution is desired. While the collision KDE performs well compared to the collision histogram, the track-length histogram produces better FOM for all mesh sizes on the 2-D boxcell problem with KDE/histogram FOM ratios ranging between 0.2 and 0.8. Even so, the runtime for histograms would increase on unstructured meshes, and track-length histograms cannot be used in problems with continuously varying cross sections or simulations that use delta tracking. Therefore, the fractional MFP KDE shows potential for application to reactor physics problems with better performance compared to comparable histogram tallies.

Performance comparisons are also drawn between the cylindrical MFP KDE and the collision histogram on a cylindrical mesh for a 2-D pincell problem. The CPU KDE algorithm produces significantly better FOM than the collision histogram, with KDE/histogram FOM ratios of approximately 60, 50, and 9 for the flux, fission, and absorption distributions, respectively. A GPU speedup of 1.4 was obtained for this problem, which would further improve the KDE FOM by an equivalent factor.

Relative uncertainties in the distribution estimates are compared between the collision histogram, the cylindrical volume-average aMFP KDE, and the cylindrical MFP KDE with and without a minimum bandwidth. The results show that the cylindrical MFP KDE typically produces a larger relative uncertainty compared to the histogram tally where histogram bins are large relative to the size of the distance-based bandwidth. Using the volume-average cylindrical MFP KDE guarantees that the relative uncertainty will be equal to the histogram or smaller than the histogram for all bins more than one kernel support length away from a

material interface. However, uncertainties in the absorption distribution can be large for bins and tally points whose kernels extend into neighboring materials. These uncertainties can be reduced by using the boundary kernel method at the material interface, preventing the kernel from extending into neighboring materials. Even so, the boundary kernel method applied to the Epanechnikov kernel increases uncertainties in the flux and fission distributions due to its larger variance and smaller support area compared to the Epanechnikov kernel. The results also show that the use of a minimum bandwidth further improves the relative uncertainty in regions with a significant amount of resonance interactions, reducing uncertainties by factors of 2 to 4 without introducing a noticable amount of bias in tally points more than one minimum bandwidth away from the material interface.

5.3 Applications of the Cylindrical MFP KDE to Reactor Physics Problems

With the derivation and verification of the cylindrical MFP KDE, a KDE now exists that can accurately capture reaction rates for basic PWR problems. Chapter 4 applies the cylindrical MFP KDE to more advanced problems including an IFBA pincell, a quarter 16×16 assembly, a depleted PWR pincell, and a PWR pincell with an unstructured mesh.

The results showed that the cylindrical MFP KDE is capable of accurately capturing distributions in an IFBA pincell, with results agreeing within 0.5% for the flux and fission distributions and within 0.7% for the fission distribution. However, the cylindrical aMFP KDE produced a bias of approximately 2% at the fuel-IFBA interface due to the kernel functions at tally points in the rim of the fuel spreading through the IFBA and collecting scores from collisions too far into and beyond the IFBA. Even so, the aMFP KDE produced accurate estimates within the IFBA itself. Thus, while the cylindrical MFP KDE is capable of capturing distributions in problems with geometrically thin, strong absorbers, the approximate MFP KDE is not recommended for such applications.

Furthermore, the cylindrical MFP KDE showed that it is capable of estimating distributions in specific pins of a quarter-assembly problem. The cylindrical MFP KDE accurately captures the flux and absorption distributions within strong absorbers, with distributions agreeing within 0.5% throughout most of the distribution. Disagreement increases to 1% at the edge of the rim region, although the estimators agree within 3-σ for all bins and tally points. The relative uncertainties in absorption distribution in the rim region for the cylindrical MFP KDE and the reference histogram are 0.04% and 0.4%, respectively. There-

fore, the cylindrical MFP KDE can accurately capture high-resolution flux and absorption distributions in a control rod with reduced uncertainties compared to a comparable histogram tally.

Similar results are seen for the fuel pincells within the assembly, with results agreeing within 1% everywhere outside of the rim region and within 2% inside of the rim region. The disagreements can be attributed to statistical uncertainty, with 99.5% and 99.6% of the tally points and bins agreeing within 3-σ for the fission and absorption distributions, respectively, and average relative uncertainties in the rim region of the fission distribution of approximately 0.4% and 0.006% for the histogram and KDE, respectively, and average relative uncertainties in the rim region of the absorption distribution of approximately 0.35% and 0.25% for the histogram and KDE, respectively.

Additionally, the cylindrical MFP KDE was shown to accurately capture distributions on a depleted pincell, with most bins and tally points agreeing within 1% or all distributions and all bins and tally points agreeing within 2% and 3-σ. This shows that the cylindrical MFP KDE is capable of being applied to depletion problems, and opens the door for future work in this area.

Finally, the cylindrical MFP KDE was applied to estimate scalar flux and reaction rates on an unstructured mesh. While this work does not apply KDEs to a multi-physics coupling problem as the problem complexity required is beyond the scope of this thesis, estimating solutions on an unstructured mesh is shown as a proof-of-principle of their applicability to such problems. The cylindrical MFP KDE was shown to produce accurate estimates, with the contours of the results on the unstructured mesh agreeing with the expected distribution seen on the structured mesh.

5.4 Future Work

The KDEs developed in this dissertation have shown significant improvement in estimating reaction rates in reactor physics problems compared to the previous state-of-the-art KDEs for Monte Carlo radiation transport simulations, but there is still more to explore in both applications and algorithm development.

Applications to Reactor Analysis

First and foremost, the KDEs developed in this dissertation could be used in the neutronics portion of multi-physics coupling problems. Section 4.5 showed that MFP KDEs could be applied to estimate distributions on an unstructured mesh, however the MFP KDE has yet to be applied to a multi-physics coupling problem. Additionally, the author is aware that while PWRs encompass the majority of power reactors in use today, they are not the only reactors being simulated. KDEs would need to be validated for use in reactors with large streaming paths, like boiling water reactors, high temperature gas reactors, or other varieties of fast reactors. These problems would likely require the use of the track-length MFP KDE.

Furthermore, Section 4.4 showed the unphysical reaction rate distributions that are created as a result of depleting materials on a mesh using a histogram. KDEs can be used to obtain a continuous distribution of materials by depleting the problem at tally points and interpolating the material concentrations between the points. One possible way to do this would be to use a finite element representation of the depleted materials on the nodes of an unstructured mesh.

One difficulty with using KDEs to obtain a continuous distribution of isotope concentrations is the tracking of particles in materials with continuously-varying cross sections. Delta tracking is a popular method for handling the tracking of particles in materials with continuously-varying cross sections and has been implemented for this purpose in the Monte Carlo code Serpent [45]. While the use of continuously-varying cross sections and delta tracking prohibits the use of traditional track-length estimators, the track-length aMFP KDE can be used as it does not require knowledge of material or mesh boundaries and only uses the cross section at the tally point when tallying a particle track. Thus, track-length MFP KDE tallies show potential for use in simulations that use delta tracking or continuously-varying cross sections.

Furthermore, KDEs can be applied to the direction phase space to obtain estimates of angular flux and currents, quantities commonly used in hybrid Monte Carlo-deterministic methods. However, the discontinuous nature of the angular flux should be considered when producing such an estimator.

Algorithm Development

Another area for future work is the advancement of the track-length KDE. The track-length MFP KDE shows potential for application on advanced reactor problems as well as in

simulations that apply delta tracking, however the cylindrical track-length KDEs have not been tested nor have the track-length KDEs been implemented on the GPU. Since the track-length KDE requires looking up and calculating kernels for significantly more tally points per particle event compared to the collision KDE, efficient implementation on both GPUs and CPUs is a remaining challenge.

Furthermore, GPU results in Section 3.6.4 showed that there is a maximum problem complexity that GPUs can handle. While this is heavily dependent upon the complexity of the transport process compared to the tally process as well as the GPU and CPU speeds, the GPU algorithm could be improved to handle problems with large amounts of tally points. Currently, the GPU algorithm scales linearly with the number of tally points while the CPU algorithm scales with tally point density. Thus, altering the GPU algorithm so it does not linearly scale with the number of tally points may increase the problem complexity that the GPUs are capable of handling before they become the bottleneck in the calculation.

Spatially-dependent Bandwidths

One area of future work includes removing unnecessary assumptions regarding the construction of an optimal bandwidth. The optimal bandwidth formula used in this thesis assumes that the underlying distribution is normally distributed: a poor approximation for the vast majority of the distributions estimated in this thesis. Furthermore, one bandwidth is used across a KDE region. While it has been shown that region-based bandwidths decrease bias in multimodal problems, this method still issues a single bandwidth across the entire region. For the MFP KDE, Chapter 2 showed that using a user-defined minimum bandwidth yielded lower variances without significantly increasing the bias in the majority of the fuel. However, the minimum bandwidth does incur a significant bias at the material interface, with a bias of 15% in the 1-D pincell problem results shown in Figure 2.22. Thus, a spatially-dependent optimal bandwidth has the potential to improve the performance of the KDEs.

One potential method for developing a spatially-dependent optimal bandwidth is to minimize the Mean Square Error (MSE) rather than the Mean Integrated Square Error (MISE), producing a MSE-optimal bandwidth that is spatially dependent, previously shown in Eq. (1.37). However, this MSE-optimal bandwidth requires already knowing the distribution being estimated at every point, returning to the issue of requiring assumptions or an *a priori* estimate of the distribution. Thus, one area of future work is to develop a spatially-dependent optimal bandwidth for reactor physics problems, either by using assumptions about the underlying distribution or by obtaining a coarse estimate of the underlying distribution prior

to the active tally portion of the Monte Carlo simulation.

BIBLIOGRAPHY

[1] J. J. Duderstadt and L. J. Hamilton, *Nuclear Reactor Analysis*, John Wiley & Sons (1976).

[2] E. E. Lewis and W. F. Miller, Jr, *Computational Methods of Neutron Transport*, American Nuclear Society, La Grange Park, IL (1993).

[3] G. I. Bell and S. Glasstone, *Nuclear Reactor Theory*, Van Nostrand Reinhold Inc, New York (1970).

[4] J. E. Hoogenboom, W. R. Martin, and B. Petrovic, "Monte Carlo Performance Benchmark for Detailed Power Density Calculation in a Full Size Reactor Core," 2011, https://www.oecd-nea.org/dbprog/documents/MonteCarlobenchmarkguideline_004.pdf.

[5] R. W. Shonkwiler, *Finance With Monte Carlo*, Springer New York, New York (2013).

[6] R. A. Freeze, "An Analysis of Baseball Batting Order by Monte Carlo Simulation," *Operations Research*, **22**, *4*, pp. 728–735 (1974).

[7] I. Lux and L. Koblinger, *Monte Carlo particle transport methods: neutron and photon calculations*, CRC Press, New York (1991).

[8] P. Obložinský et al., "Special Issue on ENDF/B-VII.1 Library ENDF/B-VII.1 Nuclear Data for Science and Technology: Cross Sections, Covariances, Fission Product Yields and Decay Data," *Nuclear Data Sheets*, **112**, *12*, pp. 2887 – 2996 (2011).

[9] K. Shibata et al., "JENDL-4.0: A New Library for Nuclear Science and Engineering," *Journal of Nuclear Science and Technology*, **48**, *1*, pp. 1–30 (2011).

[10] M. Turk, "The Stanford Bunny," 2000, http://www.cc.gatech.edu/ turk/bunny/bunny.html.

[11] J. Leppänen, "Development of a CAD Based Geometry Model in Serpent 2 Monte Carlo Code," *Trans. Am. Nucl. Soc.*, **111**, (2014).

[12] X-5 Monte Carlo Team, "MCNP — A General Monte Carlo N-Particle Transport Code, Version 5 Volume I: Overview and Theory," 2003, Los Alamos National Laboratory, LA-UR-03-1987.

[13] D. P. Griesheimer, *Functional expansion tallies for Monte Carlo simulations*, PhD thesis, University of Michigan, (2005).

[14] K. Banerjee, "Kernel Density Estimator Methods for Monte Carlo Radiation Transport," Ph.D. Thesis, University of Michigan (2010).

[15] M. Ellis, B. Forget, K. Smith, and D. Gaston, "Preliminary coupling of the Monte Carlo code OpenMC and the Multiphysics Object-Oriented Simulation Environment (MOOSE) for analyzing Doppler feedback in Monte Carlo simulations," *M&C*, Nashville, TN, April 19-23, (2015).

[16] E. Parzen, "On Estimation of a Probability Density Function and Mode," *The Annals of Mathematical Statistics*, **33**, *3*, pp. 1065–1076 (1962).

[17] B. W. Silverman and M. C. Jones, "E. Fix and J.L. Hodges (1951): An Important Contribution to Nonparametric Discriminant Analysis and Density Estimation: Commentary on Fix and Hodges (1951)," *International Statistical Review / Revue Internationale de Statistique*, **57**, *3*, pp. 233–238 (1989).

[18] M. Rosenblatt, "Remarks on Some Nonparametric Estimates of a Density Function," *The Annals of Mathematical Statistics*, **27**, *3*, pp. 832–837 (1956).

[19] T. Cacoullos, "Estimation of a multivariate density," *Annals of the Institute of Statistical Mathematics*, **18**, *1*, pp. 179–189.

[20] J. Du and W. R. Martin, "Adaptive kernel density estimation and Monte Carlo sampling," *Trans. Am. Nucl. Soc.*, **74**, (1996).

[21] N. Tyagi, W. R. Martin, J. Du, A. F. Bielajew, and I. J. Chetty, "A Proposed Alternative to Phase-space Recycling Using the Adaptive Kernel Density Estimator Method," *Medical Physics*, **33**, pp. 553–560 (2006).

[22] K. L. Dunn, "Monte Carlo Mesh Tallies based on a Kernel Density Estimator Approach," Ph.D. Thesis, University of Wisconsin–Madison (2014).

[23] T. Yamamoto, "Non-regionwise weight cancellation for Monte Carlo higher order criticality calculations using kernel density estimator," *Annals of Nuclear Energy*, **38**, *11*, pp. 2515 – 2520 (2011).

[24] B. R. Nease, J. D. Densmore, and D. L. Millman, "Residual Monte Carlo Using Kernel Density Estimators," *Trans. Am. Nucl. Soc.*, **111**, (2014).

[25] A. M. Holgado et al., "Kernel density estimation for grey Implicit Monte Carlo radiation transport," *M&C*, Nashville, TN, April 19-23, 2015.

[26] D. W. Scott, *Multivariate Density Estimation: Theory, Practice, and Visualization*, John Wiley and Sons, Inc., Hoboken, NJ (1992).

[27] K. L. Dunn and P. P. H. Wilson, "Monte Carlo Mesh Tallies Based on a Kernel Density Estimator Approach Using Integrated Particle Tracks," *M&C*, Sun Valley, ID, May 5-9, (2013).

[28] B. W. Silverman, *Density Estimation for Statistics and Data Analysis*, Chapman and Hall, London, UK (1986).

[29] V. A. Epanechnikov, "Nonparametric Estimation of a Multidimensional Probability Density," *Theor. Probab. Appl.*, **14**, pp. 153–158 (1969).

[30] M. C. Jones, "Simple Boundary Correction for Kernel Density Estimation," *Statistics and Computing*, **3**, pp. 135–146 (1993).

[31] L. Devroye and L. Györfi, *Nonparametric Density Estimation*, John Wiley & Sons, New York (1985).

[32] T. Liu, N. Wolfe, C. D. Carothers, W. Ji, and X. G. Xu, "Status of Archer — A Monte Carlo Code for the High-performance Heterogeneous Platforms Involving GPU and MIC," *M&C*, Nashville, TN, April 19-23, (2015).

[33] R. L. Martz, "MCNP6 Unstructured Mesh Initial Validation and Performance Results," *Nucl. Tech.*, **180**, pp. 316–335 (2012).

[34] J. Leppänen and M. Aufiero, "Development of an Unstructured Mesh Based Geometry Model in the Serpent 2 Monte Carlo Code.," *PHYSOR*, Kyoto, Japan, Sept. 28 - Oct. 3rd, (2014).

[35] T. J. Tautges and A. Caceres, "Scalable Parallel Solution Coupling for Multi-Physics Reactor Simulation," *Journal of Physics: Conference Series*, 12/2008, 2008.

[36] J. Grandy, "Conservative Remapping and Region Overlays by Intersecting Arbitrary Polyhedra," *Journal of Computational Physics*, **148**, *2*, pp. 433 – 466 (1999).

[37] L. Margolin and M. Shashkov, "Second-order sign-preserving conservative interpolation (remapping) on general grids," *Journal of Computational Physics*, **184**, *1*, pp. 266 – 298 (2003).

[38] P. K. Romano and B. Forget, "The OpenMC Monte Carlo Particle Transport Code," *Ann. Nucl. Energy*, **51**, pp. 274–281 (2013).

[39] P. Obložinský et al., "Evaluated Nuclear Data File ENDF/B-VII.0 ENDF/B-VII.0: Next Generation Evaluated Nuclear Data Library for Nuclear Science and Technology," *Nuclear Data Sheets*, **107**, *12*, pp. 2931 – 3060 (2006).

[40] A. Nouri et al., "JANIS: A New Software for Nuclear Data Services," *J. Nucl. Sci. Techol.*, **39**, pp. 1480–1483 (2002).

[41] "VERA Core Physics Benchmark Progression Problem Specfications," 2014, www.casl.gov/docs/CASL-U-2012-0131-004.pdf.

[42] "CUDA C Programming Guide," 2015, http://docs.nvidia.com/cuda/cuda-c-programming-guide.

[43] "Nvidia Visual Profiler," 2015, http://www.developer.nvidia.com/nvidia-visual-profiler.

[44] "Tesla GPU Accelerators For Servers," 2015, http://www.nvidia.com/object/tesla-servers.html.

[45] J. Leppänen, "A Continuous-energy Monte Carlo Reactor Physics Burnup Calculation Code," *Ann. Nucl. Energy*, **51**, pp. 274–281 (2013).

[46] F. B. Brown and W. R. Martin, "Direct Sampling of Monte Carlo Flight Paths in Media With Continuously Varying Cross-Sections," *Nuclear Mathematical and Computational Sciences*, Gatlinburg, TN, April 6-11, (2003), LA-UR-02-6350.

[47] E. R. Woodcock, et al., "Techniques used in the GEM code for Monte Carlo neutronics calculations in reactors and other systems of complex geometry," 1965.

[48] L. L. Carter, E. D. Cashwell, and W. M. Taylor, "Monte Carlo Sampling with Continuously Varying Cross Sections Along Flight Paths," *Nucl. Sci. Eng.*, **48**, pp. 403–411 (1972).

[49] Dessault Systems Simulia, Inc., "Abaqus User Manuals," 2016.

[50] P. P. Wilson, T. J. Tautges, J. A. Kraftcheck, B. M. Smith, and D. L. Henderson, "Acceleration techniques for the direct use of CAD-based geometry in fusion neutronics analysis," *Fusion Engineering and Design*, **85**, *10–12*, pp. 1759 – 1765 (2010), Proceedings of the Ninth International Symposium on Fusion Nuclear Technology.