# A New Method for Coupling 2D and 3D Deterministic and Stochastic Radiation Transport Calculations

A Thesis Presented for

The Master of Science

Degree

The University of Tennessee, Knoxville

Joel Aaron Kulesza

August 2011

© by Joel Aaron Kulesza, 2011
All Rights Reserved.

*For my parents.*

# Acknowledgements

I would like to recognize Westinghouse Electric Company, generally, and John Carlson, Patricia Paesano, and Abdul Dulloo, specifically, for supporting me in the completion of this body of work and the accompanying degree. I would also like to thank Gianluca Longoni for his assistance and guidance throughout the time spent generating this body of work. Finally, I would like to thank Stan Anderson and Arnold Fero for their continued encouragement and counseling.

*Most software today is very much like an Egyptian pyramid with millions of bricks piled on top of each other, with no structural integrity, but just done by brute force and thousands of slaves.*

*Alan Kay, 2004*

# Abstract

The objective of this body of work was to produce a code system capable of processing boundary angular flux data from discrete ordinates calculations in 2D and 3D Cartesian and cylindrical geometries into cumulative probability density functions that can be used with a Monte Carlo radiation transport code to define neutron and photon initial positions, directions, and energies. In order to accomplish this goal, the DISCO (DetermInistic-Stochastic Coupling Operation) code was created to interface between the DORT and TORT deterministic radiation transport codes and the MCNP stochastic radiation transport code. DISCO introduces new methods to use the boundary angular flux data, along with information regarding the deterministic quadrature sets and spatial mesh structure, to create cumulative probability density functions that are passed to MCNP for sampling within the source.F90 subroutine that was also generated as part of this work. Operating in concert, DISCO and the MCNP source.F90 subroutine create a source term according to the discrete ordinates angular flux information. In order to validate the work described herein, 24 test cases were created to exercise the different geometries and execution modes available. The results of these test cases confirm that the methodology and corresponding implementation is appropriate and functioning correctly. Furthermore, this work incorporates several novel features such as compatibility with all 2D and 3D Cartesian and cylindrical geometries, an angular and spatial indexing scheme to reduce random sampling operations, a streamlining of process execution, and the ability for the resulting Monte Carlo code to operate in either serial and parallel mode.

# Contents

**1 Introduction** **1**

1.1 History . . . . . 1

1.2 Motivation . . . . . 3

1.3 Objective . . . . . 5

**2 Methodology** **7**

2.1 Previous Work in DOMINO . . . . . 7

2.1.1 Generating CDFs with DOMINO . . . . . 7

2.1.2 Generating Starting Particles in MCNP . . . . . 12

2.2 CDF Implementation in DISCO . . . . . 13

2.2.1 Indexing of Discrete Directions . . . . . 13

2.2.2 CDF Generation in DISCO for $r$-$z$ Cases . . . . . 13

2.2.3 CDF Generation in DISCO for $r$-$\theta$ Cases . . . . . 14

2.2.4 CDF Generation in DISCO for $x$-$y$ Cases . . . . . 15

2.2.5 Considerations in the Transition from 2D to 3D . . . . . 17

2.2.6 CDF Generation in DISCO for $x$-$y$-$z$ Cases . . . . . 18

2.2.7 CDF Generation in DISCO for $r$-$\theta$-$z$ Cases . . . . . 21

2.2.8 Summary of CDF Associations . . . . . 22

2.3 Process for Randomly Sampling CDFs . . . . . 22

2.4 Starting Monte Carlo Particles . . . . . 24

2.4.1 Particle Initial Energy . . . . . 24

2.4.2 Particle Initial Position . . . . . 25
2.4.3 Particle Initial Direction . . . . . 26

**3 Numerical Implementation** **30**
3.1 DISCO Code Structure . . . . . 30
3.1.1 DISCO Subroutine: `main.f90` . . . . . 31
3.1.2 DISCO Subroutine: `read_input.f90` . . . . . 31
3.1.3 DISCO Subroutine: `read_bndrys.f90` . . . . . 33
3.1.4 DISCO Subroutine: `read_dirflx.f90` . . . . . 33
3.1.5 DISCO Subroutine: `calc_domino_source.f90` . . . . . 34
3.1.6 DISCO Subroutine: `calc_rz_source.f90` . . . . . 34
3.1.7 DISCO Subroutine: `calc_rt_source.f90` . . . . . 34
3.1.8 DISCO Subroutine: `calc_xy_source.f90` . . . . . 35
3.1.9 DISCO Subroutine: `calc_xyz_source.f90` . . . . . 35
3.1.10 DISCO Subroutine: `calc_rtz_source.f90` . . . . . 35
3.1.11 DISCO FIDO Subroutines . . . . . 35
3.1.12 DISCO LAPACK Sorting Subroutine: `slasrt.f90` . . . . . 36
3.2 DISCO Error Checking . . . . . 36
3.2.1 Error Checks in `calc_domino_source.f90` . . . . . 37
3.2.2 Error Checks in `read_bndrys.f90` . . . . . 37
3.2.3 Error Checks in `read_dirflx.f90` . . . . . 38
3.2.4 Error Checks in `read_input.f90` . . . . . 39
3.3 MCNP Subroutine: `source.F90` . . . . . 40

**4 Test Cases** **42**
4.1 Description of Test Cases . . . . . 42
4.1.1 DORT Geometry for $r$-$z$ Test Cases . . . . . 44
4.1.2 DORT Geometry for $r$-$\theta$ Test Cases . . . . . 46
4.1.3 DORT Geometry for $x$-$y$ Test Cases . . . . . 46
4.1.4 TORT Geometry for $x$-$y$-$z$ Test Cases . . . . . 46

4.1.5 TORT Geometry for $r$-$\theta$-$z$ Test Cases . . . . . 49
4.2 DISCO Test Case Results Summary . . . . . 49
4.2.1 Results for $r$-$z$ Test Cases . . . . . 49
4.2.2 Results for $r$-$\theta$ Test Cases . . . . . 51
4.2.3 Results for $x$-$y$ Test Cases . . . . . 53
4.2.4 Results for $x$-$y$-$z$ Test Cases . . . . . 53
4.2.5 Results for $r$-$\theta$-$z$ Test Cases . . . . . 54
4.3 MCNP Test Case Results Summary . . . . . 54

**5 Summary & Future Work** **58**
5.1 Summary . . . . . 58
5.2 Future Work . . . . . 59

**Bibliography** **61**

**A DISCO Input File Specification** **66**
A.1 Example $r$-$z$ DISCO, DOMINO-Compatible, Input File . . . . . 71
A.2 Example $r$-$\theta$ DISCO Input File . . . . . 73
A.3 Example $x$-$y$-$z$ DISCO Input File . . . . . 75

**B MCNP Test Case Initial Particle Behavior** **77**
B.1 Test Case 5 . . . . . 78
B.2 Test Case 6 . . . . . 80
B.3 Test Case 7 . . . . . 82
B.4 Test Case 8 . . . . . 84
B.5 Test Case 9 . . . . . 86
B.6 Test Case 10 . . . . . 88
B.7 Test Case 11 . . . . . 90
B.8 Test Case 12 . . . . . 92
B.9 Test Case 13 . . . . . 94

B.10 Test Case 14 . . . . . 96
B.11 Test Case 15 . . . . . 98
B.12 Test Case 16 . . . . . 100
B.13 Test Case 17 . . . . . 102
B.14 Test Case 18 . . . . . 104
B.15 Test Case 19 . . . . . 106
B.16 Test Case 20 . . . . . 108
B.17 Test Case 21 . . . . . 110
B.18 Test Case 22 . . . . . 112
B.19 Test Case 23 . . . . . 114
B.20 Test Case 24 . . . . . 116

**Vita** **118**

# List of Tables


2.1 Relationship of Boundary Face Indicies $n$ to 3D Coordinate Systems . 18

2.2 Summary of DOMINO and DISCO CDF Associations . . . . . 22

3.1 DISCO Source Types and Directions that Correspond to Each Geometry 40

4.1 DISCO Test Case Summary . . . . . 43

4.2 DISCO/DOMINO $r$-$z$ Test Case Results . . . . . 51

4.3 DISCO/DORT $r$-$z$ Test Case Results . . . . . 51

4.4 DISCO/DORT $r$-$\theta$ Test Case Results . . . . . 53

4.5 DISCO/DORT $x$-$y$ Test Case Results . . . . . 53

4.6 DISCO/MCNP Source Neutron Fraction Comparison . . . . . 56

4.7 DISCO/MCNP Source Photon Fraction Comparison . . . . . 57

A.1 DISCO Geometry, Source, and Direction Input Options . . . . . 67

A.2 DISCO Quadrature Set and Spatial Mesh FIDO Array Associations . 69

# List of Figures


1.1 Current DORT-DOMINO-MCNPBQ-MCNP Process . . . . . 4

2.1 DOMINO Partial Current Options . . . . . 8
2.2 Two-dimensional $S_6$ Quadrature Set Sphere . . . . . 10
2.3 DISCO $r$-$\theta$ Partial Current Options . . . . . 14
2.4 DOMINO $x$-$y$ Partial Current Options . . . . . 16
2.5 DISCO Cartesian Face Example with Indexed (Upper) & Absolute (Lower) Positions . . . . . 19
2.6 Example of Mapping a Uniform Random Variable to an Event with a CDF . . . . . 25
2.7 Global (Monte Carlo Cartesian) and Local (Discrete Ordinates Cylindrical) Frames of Reference with Particle Direction Vector . . . . . 27
2.8 Local Coordinate Systems used to Transform Direction Cosine-based Particle Direction to Global Direction . . . . . 28

3.1 DISCO Program Flow . . . . . 32

4.1 Test Geometry for $r$-$z$ Upward/Downward Source . . . . . 45
4.2 Test Geometry for $r$-$z$ Inward/Outward Source . . . . . 45
4.3 Test Geometry for $r$-$\theta$ Inward/Outward Source . . . . . 47
4.4 Test Geometry for $r$-$\theta$ Upward/Downward Source . . . . . 47
4.5 Test Geometry for $x$-$y$ Inward/Outward Source . . . . . 48
4.6 Test Geometry for $x$-$y$ Upward/Downward Source . . . . . 48

4.7 Test Geometry for $x$-$y$-$z$ . . . . . 50
4.8 Test Geometry for $r$-$\theta$-$z$ . . . . . 50
4.9 DOMINO Case 1 / DISCO Case 5 Energy CDF Comparison . . . . . 52
4.10 DOMINO Case 2 / DISCO Case 6 Energy CDF Comparison . . . . . 52
4.11 DOMINO Case 3 / DISCO Case 7 Energy CDF Comparison . . . . . 52
4.12 DOMINO Case 4 / DISCO Case 8 Energy CDF Comparison . . . . . 52
4.13 Test Case 17–23 Surface Leakage CDFs . . . . . 55
4.14 Test Case 17–23 Surface Leakage PDFs . . . . . 55
4.15 DISCO Case 24 Surface Leakage CDF . . . . . 57

B.1 MCNP Test Case 5 2D Projections . . . . . 78
B.2 MCNP Test Case 5 3D Projection . . . . . 79
B.3 MCNP Test Case 6 2D Projections . . . . . 80
B.4 MCNP Test Case 6 3D Projection . . . . . 81
B.5 MCNP Test Case 7 2D Projections . . . . . 82
B.6 MCNP Test Case 7 3D Projection . . . . . 83
B.7 MCNP Test Case 8 2D Projections . . . . . 84
B.8 MCNP Test Case 8 3D Projection . . . . . 85
B.9 MCNP Test Case 9 2D Projections . . . . . 86
B.10 MCNP Test Case 9 3D Projection . . . . . 87
B.11 MCNP Test Case 10 2D Projections . . . . . 88
B.12 MCNP Test Case 10 3D Projection . . . . . 89
B.13 MCNP Test Case 11 2D Projections . . . . . 90
B.14 MCNP Test Case 11 3D Projection . . . . . 91
B.15 MCNP Test Case 12 2D Projections . . . . . 92
B.16 MCNP Test Case 12 3D Projection . . . . . 93
B.17 MCNP Test Case 13 2D Projections . . . . . 94
B.18 MCNP Test Case 13 3D Projection . . . . . 95
B.19 MCNP Test Case 14 2D Projections . . . . . 96

B.20 MCNP Test Case 14 3D Projection . . . . . 97
B.21 MCNP Test Case 15 2D Projections . . . . . 98
B.22 MCNP Test Case 15 3D Projection . . . . . 99
B.23 MCNP Test Case 16 2D Projections . . . . . 100
B.24 MCNP Test Case 16 3D Projection . . . . . 101
B.25 MCNP Test Case 17 2D Projections . . . . . 102
B.26 MCNP Test Case 17 3D Projection . . . . . 103
B.27 MCNP Test Case 18 2D Projections . . . . . 104
B.28 MCNP Test Case 18 3D Projection . . . . . 105
B.29 MCNP Test Case 19 2D Projections . . . . . 106
B.30 MCNP Test Case 19 3D Projection . . . . . 107
B.31 MCNP Test Case 20 2D Projections . . . . . 108
B.32 MCNP Test Case 20 3D Projection . . . . . 109
B.33 MCNP Test Case 21 2D Projections . . . . . 110
B.34 MCNP Test Case 21 3D Projection . . . . . 111
B.35 MCNP Test Case 22 2D Projections . . . . . 112
B.36 MCNP Test Case 22 3D Projection . . . . . 113
B.37 MCNP Test Case 23 2D Projections . . . . . 114
B.38 MCNP Test Case 23 3D Projection . . . . . 115
B.39 MCNP Test Case 24 2D Projections . . . . . 116
B.40 MCNP Test Case 24 3D Projection . . . . . 117

# Nomenclature

| | |
|---|---|
| $\bar{u}$ | Monte Carlo particle direction vector $x$ component |
| $\bar{v}$ | Monte Carlo particle direction vector $y$ component |
| $\bar{w}$ | Monte Carlo particle direction vector $z$ component |
| $\bar{x}$ | Monte Carlo particle $x$ Position |
| $\bar{y}$ | Monte Carlo particle $y$ Position |
| $\bar{z}$ | Monte Carlo particle $z$ Position |
| $\eta$ | Direction cosine corresponding to the $j$-direction (either $y$ or $\theta$, as appropriate) |
| $\hat{\Omega} \cdot \vec{\nabla}$ | Boltzmann transport equation streaming operator |
| $\hat{n}$ | Surface normal vector |
| $\mu$ | Direction cosine corresponding to the $i$-direction (either $x$ or $r$, as appropriate) |
| $\phi(\vec{r}, E)$ | Spatial- and energy-dependent particle flux (scalar flux) |
| $\Pi(x)$ | Cumulative probability density function |
| $\pi(x)$ | Probability density function |
| $\psi(\vec{r}, \hat{\Omega}, E)$ | Spatial-, angle-, and energy-dependent particle flux (angular flux) |

| | |
|---|---|
| $\sigma(\vec{r}, E)$ | Boltzmann transport equation total macroscopic cross-section |
| $\sigma_s(\vec{r}, E' \rightarrow E, \hat{\Omega}' \cdot \hat{\Omega})$ | Boltzmann transport equation macroscopic scattering cross-section from energy $E'$ to $E$ with a deflection cosine of $\hat{\Omega}' \cdot \hat{\Omega}$ |
| $\vec{J}$ | Analytical current vector |
| $\vec{j}$ | Analytical current vector density |
| $\xi$ | Direction cosine corresponding to the $k$-direction ($z$) |
| $J_+$ | Partial current; $+r$ or $+z$ in $r$-$z$, $+r$ or $+\theta$ in $r$-$\theta$, $+x$ or $+y$ in $x$-$y$ |
| $J_-$ | Partial current; $-r$ or $-z$ in $r$-$z$, $-r$ or $-\theta$ in $r$-$\theta$, $-x$ or $-y$ in $x$-$y$ |
| $J_{\text{net}}$ | Analytical net current |
| $J_n$ | Partial current density in 3D cases; $n = 1$ corresponds to leakage through the $-i$ face, $n = 2$ through $+i$, $n = 3$ through $-j$, $n = 4$ through $+j$, $n = 5$ through $-k$ and $n = 6$ through the $+k$ face |
| $q_{\text{ex}}\left(\vec{r}, \hat{\Omega}, E\right)$ | Boltzmann transport equation external source term |
| $w$ | Quadrature set weight corresponding to a particular direction |
| 2D | two-dimensional |
| 3D | three-dimensional |
| ACT2 | Westinghouse code for processing 2D and 3D scalar flux data as well as activation and/or depletion cross-sections and decay constants to create discrete ordinates activation sources |
| CDF | Cumulative Distribution Function |
| DISCO | DetermInistic-Stochastic Coupling Operation code created to satisfy the objective of this work |

| | |
|---|---|
| DOMINO | General-purpose utility for coupling $r$-$z$ discrete ordinates calculations and Monte Carlo radiation transport calculations |
| FIXSOUR | Westinghouse code for generating fixed, distributed, sources from a 2D combination of fuel nozzle and fuel region activity information |
| MCNPBQ | Westinghouse code for sampling DOMINO CDFs to generate SPFs for MCNP |
| PDF | Probability Density Function |
| SORCERY | Westinghouse code for generating fixed, distributed, sources for 2D and 3D discrete ordinates calculations using fuel assembly burnup and enrichment information |
| SPF | MCNPBQ-produced Source Particle File |
| V&V | Verification and validation |

# Chapter 1

# Introduction

## 1.1 History

A wide range of problems in the field of nuclear engineering require the solution of the time-independent linear Boltzmann transport equation for uncharged particles. Many conventions have been used in expressing this equation so the convention considered herein is the one expressed by Lewis and Miller (1993) as

$$\left[\hat{\Omega}\cdot\vec{\nabla}+\sigma\left(\vec{r},E\right)\right]\psi\left(\vec{r},\hat{\Omega},E\right) = q_{\mathrm{ex}}\left(\vec{r},\hat{\Omega},E\right)+\int dE'\int d\Omega'\sigma_{\mathrm{s}}\left(\vec{r},E'\rightarrow E,\hat{\Omega}'\cdot\hat{\Omega}\right)\psi\left(\vec{r},\hat{\Omega}',E'\right), \qquad (1.1)$$

where $\hat{\Omega}\cdot\vec{\nabla}$ is the Boltzmann transport equation streaming operator, $\sigma(\vec{r},E)$ is loss due to particle interaction (e.g., absorption), $\psi(\vec{r},\hat{\Omega},E)$ is spatial-, angle-, and energy-dependent particle flux (i.e., angular flux), $q_{\mathrm{ex}}\left(\vec{r},\hat{\Omega},E\right)$ is the sum of any external sources (e.g., fission, delayed neutron, or fixed sources) and $\sigma_{\mathrm{s}}(\vec{r},E'\rightarrow E,\hat{\Omega}'\cdot\hat{\Omega})$ is the macroscopic scattering cross-section from energy $E'$ to $E$ with a deflection cosine of $\hat{\Omega}'\cdot\hat{\Omega}$. It should be noted that the quantity that is typically of interest is the scalar flux, or $\phi(\vec{r},E)=\int_{4\pi}\psi(\vec{r},\hat{\Omega},E)d\Omega$. Common computational methods for solving this integro-differential equation are of a stochastic or deterministic nature. Generally

speaking, stochastic transport calculates the average behavior of particles interacting in the system on a pseudo-continuous basis whereas deterministic transport discretizes the spatial, angular, and/or energy domains and finds the bulk particle behavior in each subdomain.

In the early to middle 1970's, the DOMINO (**D**iscrete **O**rdinates **M**onte Carlo **In**terface **O**peration) code was created as a general-purpose utility for the coupling of $r$-$z$ discrete ordinates and Monte Carlo radiation transport calculations (Emmett et al., 1973). In order to do this, DOMINO processes boundary angular flux data produced by a discrete ordinates code, such as DOT or DORT, to produce a series of cumulative distribution functions (CDFs) representing particle spatial, angular, and energy distributions. These CDFs would subsequently be sampled in order to generate initial particle positions, directions, and energies for use in a Monte Carlo code such as MORSE-CGA (Emmett, 1985) or MCNP (Girard et al., 2008). With these CDFs in hand, it was left up to the user to appropriately sample and post-process the information for use in a Monte Carlo code.

Several recent applications of the DOMINO code include the work by Kulesza (2009), Hagler and Fero (2005), and Fero et al. (1996). In all three instances, sampling and post-processing was performed with the Westinghouse code MCNPBQ (**M**onte **C**arlo **N**-**P**article Code **B**oundary **Q**-Source Generator) to prepare source particle information for MCNP. As such, the deterministic transport provided a convenient method to transport particles from the source to the edge of a region of interest without any need for variance reduction. The subsequent stochastic transport allowed fine localized modeling and perturbation study (Kulesza) and freedom from worry about ray effects (Hagler and Fero et al.). Furthermore, added benefit was derived from auxiliary codes that generate deterministic source terms (such as Westinghouse's SORCERY and FIXSOUR codes) rather than using the built-in Monte Carlo source definition routines. Finally, Hagler could have also applied an activation/depletion/decay code (such as Westinghouse's ACT2) to the

discrete ordinates results before processing by DOMINO to give a "best estimate" set of results rather than the more conservative approach that was taken.

## 1.2 Motivation

The unique applications of the DOMINO code cited previously demonstrate its utility in solving certain problems efficiently. However, there are several disadvantages to the approach taken by DOMINO. As the methodology is currently implemented, angular flux data can only be processed from $r$-$z$ discrete ordinates calculations. It should be noted that Kurasawa (2005) developed an application for use with TORT (Rhoades and Simpson, 1997), but only for $x$-$y$-$z$ geometries. While it is possible to use any two-dimensional angular flux in the `bndrys` format (Emmett et al., 1992) with DOMINO, geometry coefficients are hard-coded into DOMINO (discussed in Section 2.1.1) that would require significant processing of the angular flux data to make it appropriate for the preferred geometry. Furthermore, there is a substantial approximation made when converting from the two-dimensional $r$-$z$ geometry to a three-dimensional geometry in MCNP; namely that, there is no azimuthal dependence (in space) in the discrete ordinates calculation. This is solved by a uniform azimuthal sampling when used in the Monte Carlo code which may not be appropriate for certain problems such as leakage from a reactor pressure vessel where there is significant azimuthal variation in the flux profile due to the stair step geometry of the fuel and corresponding varying water region thickness between the baffles, barrel, and pressure vessel. Finally, with the rising prevalence of three-dimensional (3D) calculations, particularly in $r$-$\theta$-$z$ geometries for power reactor calculations, additional geometric flexibility is desirable.

Also, Westinghouse's current work practices (shown schematically in Figure 1.1) requires an intermediate processing step of the CDFs produced by DOMINO before they are used with the Monte Carlo code. That intermediate step entails using MCNPBQ to sample the DOMINO CDF file to produce a file bound for MCNP's

`source.F90` routine containing discrete source particles (often referred to as a source particle file, or SPF). Since MCNP serially reads from a finite set of discrete

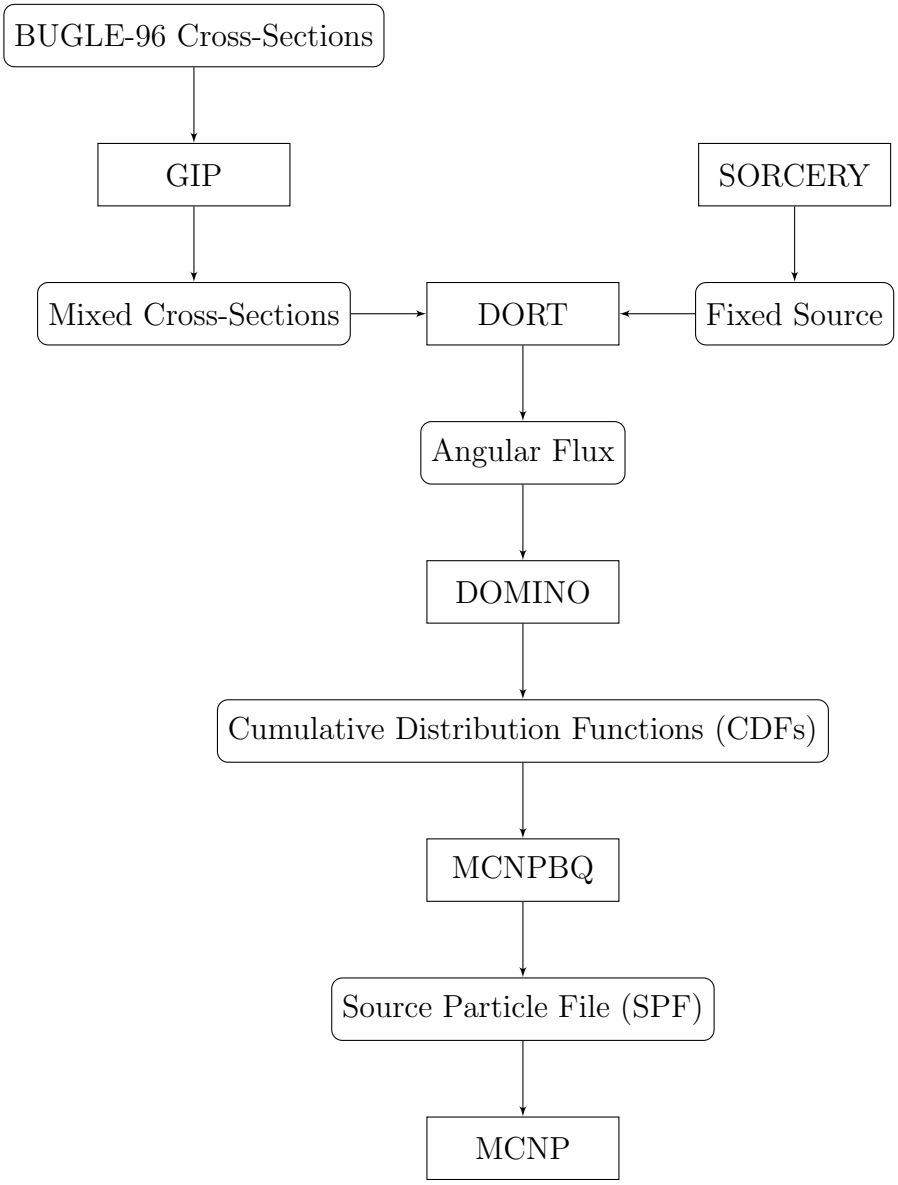



**Figure 1.1:** Current DORT-DOMINO-MCNPBQ-MCNP Process

source particles, there is potential to exceed the amount of particles produced with MCNPBQ. Currently, that limitation is handled by rewinding the SPF. However, rewinding may introduce a bias in the source particle initial positions, directions, and energies. For example, if 1000 particles are generated in the SPF and 3000 particles are

called for in MCNP, each particle will be used three times (i.e., 1-1000, rewind, 1001-2000, rewind, 2001-3000) with each instance as an independent particle. Therefore, the analyst is cautioned and advised to generate a SPF with enough particles to adequately sample the source region. Parallel execution of MCNP has a similar effect that is exacerbated by increasing the amount of processors used in the calculation. For example, if 1000 particles are generated in the SPF and 1000 particles are called for in MCNP with 10 processors, each processor will use the first 100 particles in the SPF. Due to these two sources of bias, it is desirable to bring CDF sampling into MCNP's `source.F90` routine. An ancillary benefit to this is that, for most 2D calculations, the data file containing the CDFs will be much smaller than the SPF file. One downside to this approach is that by using a `source.F90` that serially reads from a list of source particles, an analyst is able to precisely define a source outside of the DOMINO-MCNPBQ-MCNP process. Since this capability is desirable, particularly for diagnostic purposes, flexibility should be maintained in the `source.F90` to perform sampling *or* serially read from a source particle listing.

## 1.3 Objective

Based on the history presented previously describing the coupling of deterministic and stochastic calculations as well as the motivations to perform those coupling operations, the objective of this body of work is to produce a code capable of processing boundary angular flux data from $r$-$z$, $r$-$\theta$, $x$-$y$, $x$-$y$-$z$, and $r$-$\theta$-$z$ geometries into cumulative probability density functions that can be used with the MCNP `source.F90` routine to define neutron and photon initial positions, directions, and energies. This method should be applicable in serial and parallel execution of MCNP. In addition, the capability to use the `source.F90` routine to serially read predefined (i.e., non-sampled) particle initial positions, directions, and energies shall be maintained.

Indeed, this objective was accomplished and the DISCO (**D**eterm**I**nistic-**S**tochastic **C**oupling **O**peration) code and accompanying MCNP `source.F90` routine was created. As such, the remainder of this work will discuss the methodology, formulations, considerations, numerical implementation, and test cases for this code and subroutine. For the sake of discussion, this work will consider all deterministic calculations as having been performed with the DOORS package (i.e., DORT for 2D and TORT for 3D calculations). Furthermore, while this method should be applicable to most Monte Carlo transport codes, the focus of this work will be on MCNP.

Chapter 2 of this document will focus on the methodology and analytical considerations that form the basis of this work. Chapter 3 will describe the numerical implementation of DISCO and the modification of the MCNP `source.F90` routine at the programmatic level. Chapter 4 will present the test cases used to verify and validate the functionality of DISCO. Finally, Chapter 5 will present a summary of this work and recommendations for future work. Appendix A gives the DISCO input file specification as well as a few representative input files in their entirety. Appendix B provides graphics that show MCNP initial particle positions and directions in 2D and 3D that are the end result of applying DISCO.

# Chapter 2

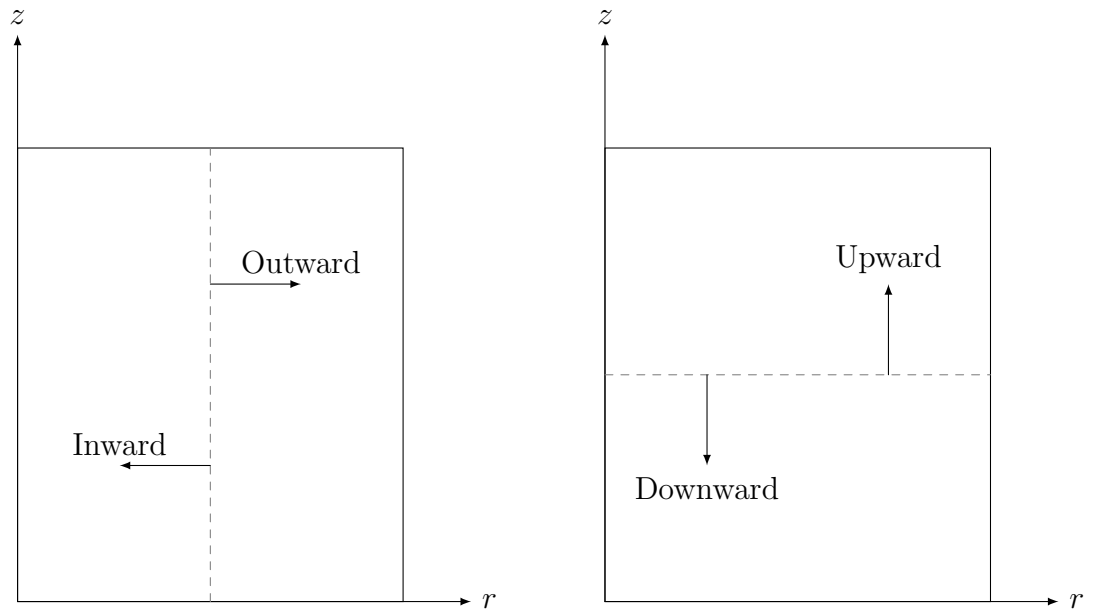
# Methodology

Any discussion of the methodology implemented in DISCO would be incomplete without a survey of the DOMINO methodology. Therefore, that will now be discussed. It should be noted that the solution (analytically or numerically) of the Boltzmann equation is not the focus of this work, but merely the resulting angular flux and its processing; therefore, it will not be discussed herein. A number of excellent references are available that cover this aspect of neutral particle transport such as Lewis and Miller (1993) and Ronen (1986).

## 2.1 Previous Work in DOMINO

### 2.1.1 Generating CDFs with DOMINO

As described in Section 1.1, DOMINO produces normalized cumulative distribution functions (CDFs) that fully describe source particle behavior in terms of energy, spatial position, and polar and azimuthal direction. DOMINO does this by using the angular flux, $\psi$, along a boundary from a discrete ordinates calculation in $r$-$z$ geometry that is processed into partial angular currents, shown schematically in Figure 2.1a and Figure 2.1b, respectively.




**(a)** Fixed Radius — Inward/Outward **(b)** Fixed Elevation — Upward/Downward

**Figure 2.1:** DOMINO Partial Current Options

Some discussion of the analytical development of net and partial currents may be useful, so that will be described in brief. To begin, one may define a current vector density as

$$\vec{j}\left(\vec{r},\hat{\Omega},E\right) \equiv \hat{\Omega}\psi\left(\vec{r},\hat{\Omega},E\right). \tag{2.1}$$

Using the current vector density, one can find the current vector as

$$\vec{J}\left(\vec{r},E\right) = \int_{4\pi} d\Omega\hat{\Omega}\psi\left(\vec{r},\hat{\Omega},E\right). \tag{2.2}$$

One may again integrate the total current to find the net number of particles crossing a surface with area $A$ defined by surface normal vector $\hat{n}$, or net current, to be

$$J_{\text{net}}\left(E\right) = \int_{A} dA \int_{4\pi} d\Omega\hat{n}\cdot\hat{\Omega}\psi\left(\vec{r},\hat{\Omega},E\right) \tag{2.3}$$

that can be decomposed into partial currents such that

$$J_{\text{net}}(E) = J_{+}(E) - J_{-}(E) \tag{2.4}$$

where

$$J_{+}(E) = \int_{A} dA \int_{\hat{\Omega}\cdot\hat{n}>0} d\Omega \hat{n}\cdot\hat{\Omega}\psi\left(\vec{r},\hat{\Omega},E\right) \tag{2.5}$$

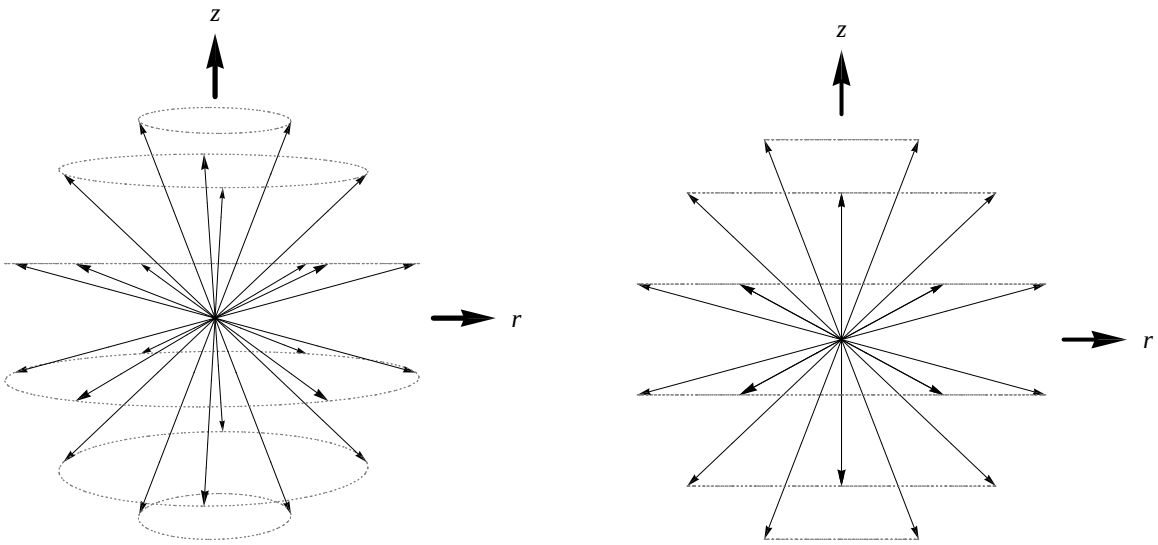
and

$$J_{-}(E) = \int_{A} dA \int_{\hat{\Omega}\cdot\hat{n}<0} d\Omega \left|\hat{n}\cdot\hat{\Omega}\right| \psi\left(\vec{r},\hat{\Omega},E\right). \tag{2.6}$$

For the purpose of this discussion in $r$-$z$ geometries, the convention will be used that for a partial current at a fixed radius, the $J_{+}$ partial current is composed of particles leaking radially outward ($+r$) and the $J_{-}$ partial current is composed of particles leaking radially inward ($-r$). Correspondingly, for a partial current at a fixed elevation, the $J_{+}$ partial current is composed of particles leaking upward ($+z$) and the $J_{-}$ partial current is composed of particles leaking downward ($-z$). As noted previously, the angular flux is expressed analytically as $\psi(\vec{r},\hat{\Omega},E)$, that is, as a function of position, direction, and energy. However, DORT discretizes the position $\vec{r}$ into its radial ($r$) and axial ($z$) components along a fixed spatial mesh, the direction $\hat{\Omega}$ into a quadrature set with direction cosine $\mu$ corresponding to the $r$ direction and $\xi$ corresponding to the $z$ direction (with corresponding weights $w$), and the energy into various energy groups. As an aside, it should be noted that for the purpose of this work, the direction cosine $\mu$ will always correspond to the $i$-direction (either $x$ or $r$), $\eta$ will always correspond to the $j$-direction (either $y$ or $\theta$), and $\xi$ will always correspond to the $k$-direction ($z$) with $w$ representing the weight along the particular direction in both the 2D and 3D case. The energy groups are ordered such that groups of lower value represent higher particle energies and the cross-section sets applied contain both neutron and photon information where the photons exist in a block at the top of the group range. For example, neutrons occupy groups 1–47 and photons occupy groups

48–67 as in the BUGLE-96 cross-section set (White et al., 1999). These details will become particularly relevant later when other geometries are discussed.

With the DORT angular flux $\psi(r, z, \mu, \xi, E)$ in hand, it can be reduced to one spatial dimension by acknowledging that it exists only along a fixed radius or elevation corresponding to $\psi(z, \mu, \xi, E)$ and $\psi(r, \mu, \xi, E)$, respectively. For convenience and consistency with DOMINO's documentation, the discrete index $g$ will represent the energy group, $i$ will represent the spatial position along the fixed dimension (e.g., a particular radius at a fixed elevation), $l$ will represent the polar level defined by a fixed $\xi$, and $m$ will represent each azimuthal segment defined by $\mu$. The polar levels and azimuthal segments are best visualized as existing on a unit sphere formed by the two direction cosines (with a portion, generally the upward half, commonly referred to elsewhere as an "igloo") shown schematically in Figure 2.2. Therefore, the current




**(a)** Perspective **(b)** Orthogonal

**Figure 2.2:** Two-dimensional $S_6$ Quadrature Set Sphere

density inward or outward is calculated as

$$J_{gilm} = \psi_{gilm} \left| \mu_{lm} \right| \tag{2.7}$$

where $i$ is used to denote $z$-levels on the cylindrical surface and the current density upward or downward is calculated as

$$J_{gilm} = \psi_{gilm} \left| \xi_{lm} \right| \tag{2.8}$$

where $i$ is used to denote $r$-levels on the disc surface. Using these current densities, DOMINO then calculates four unnormalized arrays that correspond to the angular and spatial integrations expressed in Equation 2.2 and Equation 2.3

$$E^*_{gilm} = J_{gilm}, \tag{2.9}$$

$$D^*_{gil} = \sum_m E^*_{gilm} w_{lm}, \tag{2.10}$$

$$C^*_{gi} = \sum_l D^*_{gil}, \quad \text{and} \tag{2.11}$$

$$B^*_g = \sum_i C^*_{gi} \Delta S_i, \tag{2.12}$$

or, expanded into their full form,

$$E^*_{gilm} = J_{gilm}, \tag{2.13}$$

$$D^*_{gil} = \sum_m J_{gilm} w_{lm}, \tag{2.14}$$

$$C^*_{gi} = \sum_l \left( \sum_m J_{gilm} w_{lm} \right), \quad \text{and} \tag{2.15}$$

$$B^*_g = \sum_i \left( \sum_l \left( \sum_m J_{gilm} w_{lm} \right) \right) \Delta S_i \tag{2.16}$$

where $\Delta S_i$ represents the area of the spatial mesh that leakage is occurring through, that is, $\Delta S_i = 2\pi R \left( z_{i+1} - z_i \right)$ for fixed radius $R$ and $\Delta S_i = \pi \left( r^2_{i+1} - r^2_i \right)$ for fixed elevation $Z$. Finally, an overall leakage value is calculated as

$$A = \sum_g B^*_g = \sum_g \left( \sum_i \left( \sum_l \left( \sum_m J_{gilm} w_{lm} \right) \right) \Delta S_i \right) \tag{2.17}$$

which may act as a normalization constant for the Monte Carlo calculation if the tallies are taken on a per source particle basis. Next, DOMINO normalizes Equation 2.13, Equation 2.14, Equation 2.15, and Equation 2.16 into the four PDFs

$$E_{gilm} = \frac{J_{gilm} w_{lm}}{\sum_m J_{gilm} w_{lm}}, \tag{2.18}$$

$$D_{gil} = \frac{\sum_m J_{gilm} w_{lm}}{\sum_l \left(\sum_m J_{gilm} w_{lm}\right)}, \tag{2.19}$$

$$C_{gi} = \frac{\sum_l \left(\sum_m J_{gilm} w_{lm}\right)}{\sum_i \left(\sum_l \left(\sum_m J_{gilm} w_{lm}\right)\right) \Delta S_i}, \quad \text{and} \tag{2.20}$$

$$B_g = \frac{\sum_i \left(\sum_l \left(\sum_m J_{gilm} w_{lm}\right)\right) \Delta S_i}{\sum_g \left(\sum_i \left(\sum_l \left(\sum_m J_{gilm} w_{lm}\right)\right) \Delta S_i\right)}. \tag{2.21}$$

Equation 2.18, Equation 2.19, Equation 2.20, and Equation 2.21 are then made cumulative through element-by-element accumulation and written to disk. As such, one now has all the information needed to sample for particle initial energy, position, and direction.

### 2.1.2 Generating Starting Particles in MCNP

As described in Section 1.1, DOMINO left CDF sampling and source particle generation up to the user. Therefore, each user was allowed to use a potentially different scheme to sample from the DOMINO CDFs and generate source particle conditions. The approach that the author is most familiar with is the one taken by Westinghouse whereby an intermediate code, MCNPBQ, is used to sample the CDFs using the sampling methodology adopted from MORSE to generate a Source Particle File (SPF) that contains discrete particle initial positions, directions, and energies. However, the methodology applied in MCNPBQ is outside the scope of this document and will not be discussed more extensively.

## 2.2 CDF Implementation in DISCO

### 2.2.1 Indexing of Discrete Directions

As described in Section 2.1.1, DOMINO considered the two direction cosines that define the polar and azimuthal direction of a particle separately. This treatment allowed for a more efficient use of computer storage than the approach taken in DISCO. Indeed, while the concern for storage was paramount in the 1970's, that concern has largely vanished except for extreme cases. In these extreme cases the user must be cautioned and he or she must acknowledge the practical limits of the code and execution platform.

The approach taken in DISCO is to refer to each direction by a unique index rather than by each of its component direction cosines. The main benefit of this approach is that only one CDF is required to ascertain the initial direction of a particle regardless of whether the geometry is two-dimensional or three-dimensional. Issues of independence do not arise because the angular flux along each direction is independent of the other directions at a particular spatial position and energy. For convenience, the $l$ index is eliminated and instead the $m$ index is used to represent discrete direction with corresponding direction cosines $\mu_m$, $\eta_m$, and $\xi_m$ and corresponding weight $w_m$.

### 2.2.2 CDF Generation in DISCO for $r$-$z$ Cases

In $r$-$z$ geometries, the CDF generation process in DISCO is similar to DOMINO except that one less CDF is needed because of how the directions are now referred to. As such, Equation 2.13, Equation 2.14, and Equation 2.15 become

$$E^*_{gim} = J_{gim} w_m, \tag{2.22}$$

$$D^*_{gi} = \sum_m J_{gim} w_m, \quad \text{and} \tag{2.23}$$

$$C^*_g = \sum_i \left( \sum_m J_{gim} w_m \right) \Delta S_i, \tag{2.24}$$

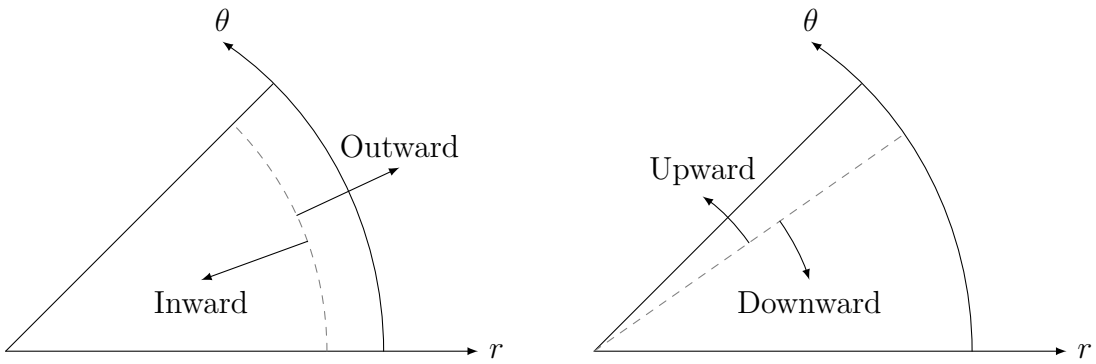
where $J_{gim} = \psi_{gim} |\mu_m|$ for inward/outward cases and $J_{gim} = \psi_{gim} |\xi_m|$ for upward/downward cases. The spatial mesh area, $\Delta S_i$, is identical to the term by the same name in Section 2.1.1. The normalization factor is then calculated as

$$A = \sum_g C_g^* = \sum_g \left( \sum_i \left( \sum_m J_{gim} w_m \right) \Delta S_i \right). \tag{2.25}$$

As before, Equation 2.22, Equation 2.23, and Equation 2.24 are normalized and made cumulative through element-by-element accumulation and written to disk.

### 2.2.3 CDF Generation in DISCO for $r$-$\theta$ Cases

DISCO is capable of generating CDFs in $r$-$\theta$ geometries by using the angular flux, $\psi$, along a boundary from a discrete ordinates calculation in $r$-$\theta$ geometry that is processed into partial angular currents, shown schematically in Figure 2.3. For




**(a)** Fixed Radius — Inward/Outward **(b)** Fixed Angle — Upward/Downward

**Figure 2.3:** DISCO $r$-$\theta$ Partial Current Options

the purpose of this discussion in $r$-$\theta$ geometries, the convention will be used that for a partial current at a fixed radius, the $J_+$ partial current is composed of particles leaking radially outward ($+r$) and the $J_-$ partial current is composed of particles leaking radially inward ($-r$). Correspondingly, for a partial current at a fixed angle, the $J_+$ partial current is composed of particles leaking upward ($+\theta$) and the $J_-$ partial current is composed of particles leaking downward ($-\theta$).

Similar to the $r$-$z$ case in DISCO, only three CDFs are necessary to determine the source particle energy, position, and direction. These are

$$E^*_{gim} = J_{gim} w_m, \tag{2.26}$$

$$D^*_{gi} = \sum_m J_{gim} w_m, \quad \text{and} \tag{2.27}$$

$$C^*_g = \sum_i \left( \sum_m J_{gim} w_m \right) \Delta S_i, \tag{2.28}$$

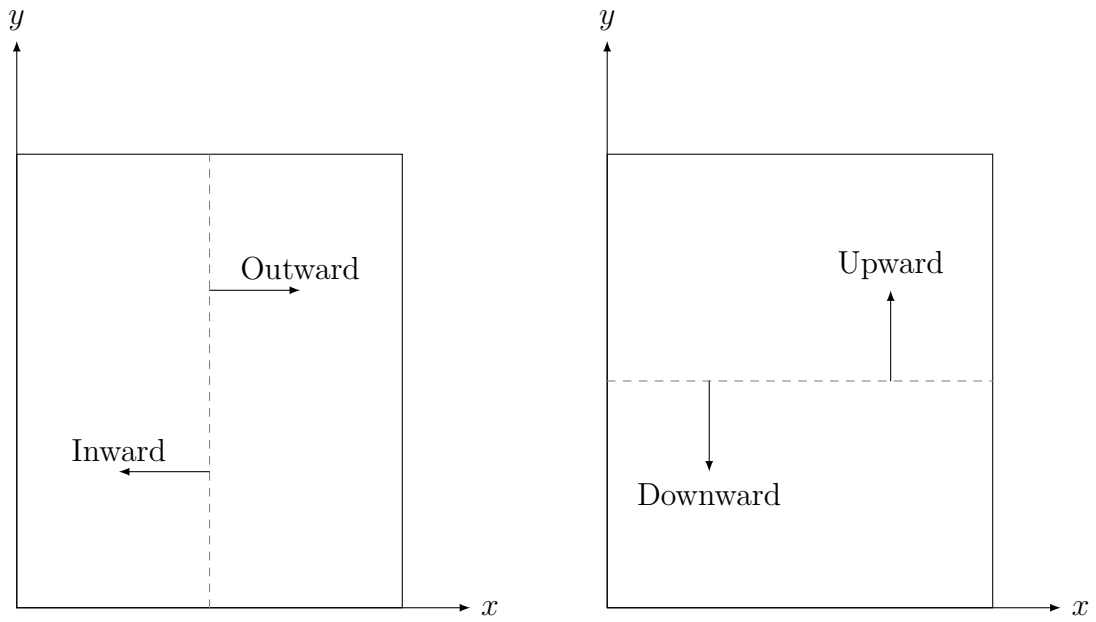
where $J_{gim} = \psi_{gim} |\mu_m|$ and $\Delta S_i = 2\pi R (\theta_{i+1} - \theta_i)$ (at fixed radius $R$ with $\theta$ in revolutions) for inward/outward cases and $J_{gim} = \psi_{gim} |\eta_m|$ and $\Delta S_i = r_{i+1} - r_i$ for upward/downward cases. The normalization factor is again calculated as

$$A = \sum_g C^*_g = \sum_g \left( \sum_i \left( \sum_m J_{gim} w_m \right) \Delta S_i \right). \tag{2.29}$$

Finally, Equation 2.26, Equation 2.27, and Equation 2.28 are normalized and made cumulative through element-by-element accumulation and written to disk.

### 2.2.4 CDF Generation in DISCO for $x$-$y$ Cases

DISCO is capable of generating CDFs in $x$-$y$ geometries by using the angular flux, $\psi$, along a boundary from a discrete ordinates calculation in $x$-$y$ geometry that is processed into partial angular currents, shown schematically in Figure 2.4. For the purpose of this discussion in $x$-$y$ geometries, the convention will be used that for a partial current at a fixed $x$, the $J_+$ partial current is composed of particles leaking outward ($+x$) and the $J_-$ partial current is composed of particles leaking inward ($-x$). Correspondingly, for a partial current at a fixed $y$, the $J_+$ partial current is composed of particles leaking upward ($+y$) and the $J_-$ partial current is composed of particles leaking downward ($-y$).




**(a)** Fixed $x$ — Inward/Outward **(b)** Fixed $y$ — Upward/Downward

**Figure 2.4:** DOMINO $x$-$y$ Partial Current Options

Similar to the previous cases described for DISCO, only three CDFs are necessary to determine the source particle energy, position, and direction. These are

$$E^*_{gim} = J_{gim} w_m, \tag{2.30}$$

$$D^*_{gi} = \sum_m J_{gim} w_m, \quad \text{and} \tag{2.31}$$

$$C^*_g = \sum_i \left( \sum_m J_{gim} w_m \right) \Delta S_i, \tag{2.32}$$

where $J_{gim} = \psi_{gim} |\mu_m|$ and $\Delta S_i = y_{i+1} - y_i$ for inward/outward cases and $J_{gim} = \psi_{gim} |\eta_m|$ and $\Delta S_i = x_{i+1} - x_i$ for upward/downward cases. The normalization factor is again calculated as

$$A = \sum_g C^*_g = \sum_g \left( \sum_i \left( \sum_m J_{gim} w_m \right) \Delta S_i \right). \tag{2.33}$$

Finally, Equation 2.30, Equation 2.31, and Equation 2.32 are normalized and made cumulative through element-by-element accumulation and written to disk. It should be noted that the $x$-$y$ formulations described previously can be confirmed with the work performed by Lloyd (1997).

### 2.2.5 Considerations in the Transition from 2D to 3D

When extending DISCO from operating in two dimensions to three, there are several differences to be taken account of and handled. First, one must consider how to handle the presence of the third direction cosine. Due to the way that the direction cosines were handled in Section 2.2.2, Section 2.2.3, and Section 2.2.4, there is a natural extension into 3D. By using indexed directions, one may continue on with that approach and merely apply the proper direction cosine along the axis of interest, as appropriate.

Next, particle spatial position representation becomes more complicated. In the 2D cases, one dimension is fixed and the particle initial position is only allowed to vary along the one remaining dimension. As such, only one dimension needs to be represented in the CDF. In 3D, one may have the option of using a volume to generate source points from or only the boundary surface (in part or in whole) of a volume. The approach taken in DISCO is to use only the boundary surface of a volume. Furthermore, the boundary surface is split into its six faces (lower and upper limits in three dimensions leading to six 2D surfaces). As such, an additional CDF is required to represent the leakage from each face. The six partial current densities of interest are $J_n$ with $n = 1, 2, ..., 6$ where $n$ denotes which face of the 3D volume that leakage is occurring through. Table 2.1 describes the relationship of $n$ to each face of a general 3D source in both the $x$-$y$-$z$ and $r$-$\theta$-$z$ coordinate systems.

Furthermore, since six separate 2D surfaces are being used, the spatial location within each face must be handled. An approach is taken similar to that of the direction cosines. Since each face is on a regular grid (whether Cartesian or cylindrical), each

**Table 2.1:** Relationship of Boundary Face Indicies $n$ to 3D Coordinate Systems

| $n$ | **General Geometry** | $x$-$y$-$z$ | $r$-$\theta$-$z$ |
|---|---|---|---|
| 1 | $-i$ Face | $y$-$z$ Face at $x = x_{\min}$ | $\theta$-$z$ Face at $r = r_{\min}$ |
| 2 | $+i$ Face | $y$-$z$ Face at $x = x_{\max}$ | $\theta$-$z$ Face at $r = r_{\max}$ |
| 3 | $-j$ Face | $x$-$z$ Face at $y = y_{\min}$ | $r$-$z$ Face at $\theta = \theta_{\min}$ |
| 4 | $+j$ Face | $x$-$z$ Face at $y = y_{\max}$ | $r$-$z$ Face at $\theta = \theta_{\max}$ |
| 5 | $-k$ Face | $x$-$y$ Face at $z = z_{\min}$ | $r$-$\theta$ Face at $z = z_{\min}$ |
| 6 | $+k$ Face | $x$-$y$ Face at $z = z_{\max}$ | $r$-$\theta$ Face at $z = z_{\max}$ |

mesh on the face is assigned a unique index that refers to its two component position identifiers. By way of example, one might use Figure 2.5 which shows a $x$-$y$ slice at some $z$ elevation with 10×10 mesh leading to 100 unique mesh. These mesh are assigned a unique index (shown as the upper value) that corresponds back to their unique point in space (shown as the lower value). Therefore, if one knows which of the six faces he or she is on as well as the mesh index, the absolute position in space can be determined.

In the following discussion of CDF generation in both 3D coordinate systems, familiar CDF indicies will be used with one addition. As before, $g$ will represent discrete energy group, $i$ will represent the unique mesh index on a particular face, $m$ will represent a particular direction defined by the three direction cosines ($\mu$, $\eta$, and $\xi$), and $n$ will be used to represent which of the six faces leakage is occurring through.

### 2.2.6 CDF Generation in DISCO for $x$-$y$-$z$ Cases

Having taken account of the caveats described in Section 2.2.5, the approach to creating the necessary CDFs in Cartesian geometry is similar to that described in Section 2.2.2, Section 2.2.3, and Section 2.2.4 with the addition of a fourth CDF to represent the face that leakage is occurring through. However, the leakage partial current densities must first be denoted. As mentioned in Section 2.2.5, the six partial

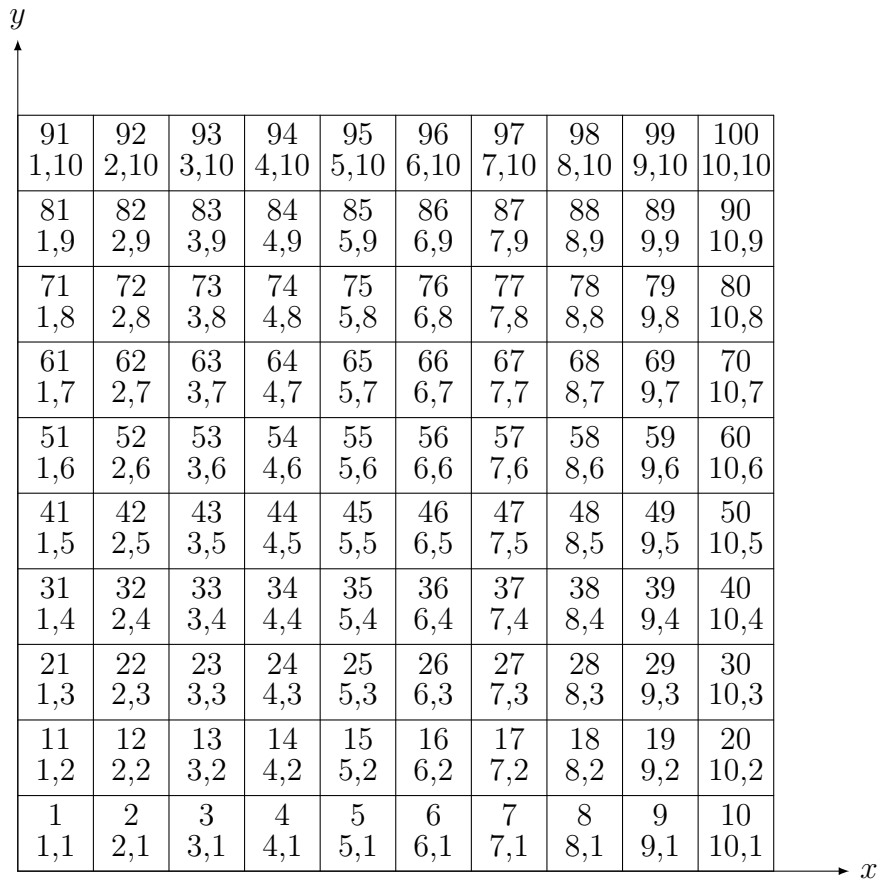



**Figure 2.5:** DISCO Cartesian Face Example with Indexed (Upper) & Absolute (Lower) Positions

current densities are formed on the faces of the source volume as

$$J_1 = \psi_{ngim} \, |\mu_m| \qquad (\mu_m < 0), \tag{2.34}$$

$$J_2 = \psi_{ngim} \, |\mu_m| \qquad (\mu_m > 0), \tag{2.35}$$

$$J_3 = \psi_{ngim} \, |\eta_m| \qquad (\eta_m < 0), \tag{2.36}$$

$$J_4 = \psi_{ngim} \, |\eta_m| \qquad (\eta_m > 0), \tag{2.37}$$

$$J_5 = \psi_{ngim} \, |\xi_m| \qquad (\xi_m < 0), \quad \text{and} \tag{2.38}$$

$$J_6 = \psi_{ngim} \, |\xi_m| \qquad (\xi_m > 0), \tag{2.39}$$

where $\psi_{ngim}$ is the angular flux for energy group $g$ at mesh index $i$ on face $n$ in direction $m$. For the purposes of this discussion, one can consider a general current density $J_{ngim}$. As such, the four PDFs created are

$$E^*_{ngim} = J_{ngim} w_m, \tag{2.40}$$

$$D^*_{ngi} = \sum_m J_{ngim} w_m, \tag{2.41}$$

$$C^*_{ng} = \sum_i \left( \sum_m J_{ngim} w_m \right) \Delta S_{ni}, \quad \text{and} \tag{2.42}$$

$$B^*_n = \sum_g \left( \sum_i \left( \sum_m J_{ngim} w_m \right) \Delta S_{ni} \right), \tag{2.43}$$

where

$$\Delta S_{1i} = (y_{i+1} - y_i)(z_{i+1} - z_i), \tag{2.44}$$

$$\Delta S_{2i} = (y_{i+1} - y_i)(z_{i+1} - z_i), \tag{2.45}$$

$$\Delta S_{3i} = (x_{i+1} - x_i)(z_{i+1} - z_i), \tag{2.46}$$

$$\Delta S_{4i} = (x_{i+1} - x_i)(z_{i+1} - z_i), \tag{2.47}$$

$$\Delta S_{5i} = (x_{i+1} - x_i)(y_{i+1} - y_i), \quad \text{and} \tag{2.48}$$

$$\Delta S_{6i} = (x_{i+1} - x_i)(y_{i+1} - y_i). \tag{2.49}$$

The overall leakage from the 3D volume, to be used as a normalization factor, is then

$$A = \sum_n B_n^* = \sum_n \left( \sum_g \left( \sum_i \left( \sum_m J_{ngim} w_m \right) \Delta S_{ni} \right) \right). \tag{2.50}$$

Next, Equation 2.40, Equation 2.41, Equation 2.42, and Equation 2.43 are normalized to become

$$E_{ngim} = \frac{J_{ngim} w_m}{\sum_m J_{ngim} w_m}, \tag{2.51}$$

$$D_{ngi} = \frac{\sum_m J_{ngim} w_m}{\sum_i \left( \sum_m J_{ngim} w_m \right) \Delta S_{ni}}, \tag{2.52}$$

$$C_{ng} = \frac{\sum_i \left( \sum_m J_{ngim} w_m \right) \Delta S_{ni}}{\sum_g \left( \sum_i \left( \sum_m J_{ngim} w_m \right) \Delta S_{ni} \right)}, \quad \text{and} \tag{2.53}$$

$$B_n = \frac{\sum_g \left( \sum_i \left( \sum_m J_{ngim} w_m \right) \Delta S_{ni} \right)}{\sum_n \left( \sum_g \left( \sum_i \left( \sum_m J_{ngim} w_m \right) \Delta S_{ni} \right) \right)}. \tag{2.54}$$

Finally, Equation 2.51, Equation 2.52, Equation 2.53, and Equation 2.54 are made cumulative through element-by-element accumulation and written to disk.

### 2.2.7 CDF Generation in DISCO for $r$-$\theta$-$z$ Cases

The generation of CDFs for cylindrical geometries is very similar to the Cartesian case described in Section 2.2.6. Indeed, Equation 2.34 through Equation 2.54 are identical. However, the mesh wise leakage areas, $\Delta S_{ni}$, are different; they become

$$\Delta S_{1i} = 2\pi r_{\min}((\theta_{i+1} - \theta_i)(z_{i+1} - z_i)), \tag{2.55}$$

$$\Delta S_{2i} = 2\pi r_{\max}((\theta_{i+1} - \theta_i)(z_{i+1} - z_i)), \tag{2.56}$$

$$\Delta S_{3i} = (r_{i+1} - r_i)(z_{i+1} - z_i), \tag{2.57}$$

$$\Delta S_{4i} = (r_{i+1} - r_i)(z_{i+1} - z_i), \tag{2.58}$$

$$\Delta S_{5i} = \pi(r_{i+1}^2 - r_i^2)(\theta_{i+1} - \theta_i), \quad \text{and} \tag{2.59}$$

$$\Delta S_{6i} = \pi(r_{i+1}^2 - r_i^2)(\theta_{i+1} - \theta_i), \tag{2.60}$$

where $r_{\min}$ and $r_{\max}$ are the minimum and maximum radii of the 3D volume that is the source region and not the problem, respectively.

### 2.2.8 Summary of CDF Associations

So far many CDFs have been described with some alphabetic identifiers ($E$, $D$, $C$, and $B$) in keeping with DOMINO's practice. However, many have different meanings from those used in DOMINO. As such, the meanings of $E$, $D$, $C$, and $B$ for both DOMINO and DISCO for all geometries of interest are summarized in Table 2.2. This summary will become useful later in Chapter 3 because the CDFs described previously map directly to multidimensional arrays in the programmatic implementation.

**Table 2.2:** Summary of DOMINO and DISCO CDF Associations

| **Code** | **Geometry** | $E$ **CDF** | $D$ **CDF** | $C$ **CDF** | $B$ **CDF** |
|---|---|---|---|---|---|
| DOMINO | $r$-$z$ | Azimuthal | Polar | Space | Energy |
| DISCO | $r$-$z$ | Direction | Space | Energy | Unused |
| DISCO | $r$-$\theta$ | Direction | Space | Energy | Unused |
| DISCO | $x$-$y$ | Direction | Space | Energy | Unused |
| DISCO | $x$-$y$-$z$ | Direction | Space | Energy | Face |
| DISCO | $r$-$\theta$-$z$ | Direction | Space | Energy | Face |

## 2.3 Process for Randomly Sampling CDFs

As noted in Section 1.1, sampling of the DOMINO CDFs is left up to the user. However, the approach taken with implementing DISCO was to generate CDFs and use them directly with the Monte Carlo code. Since the process of sampling from a CDF is germane to DISCO inasmuch as MCNP's `source.F90` routine must be made compatible with it, the CDF sampling process applied will now be described. As seen previously, a series of three or four CDFs are formed along with a normalization factor. This normalization factor is not relevant to sampling the CDFs and will be

ignored for the time being. However, the order for sampling the CDFs is paramount. One must start with the lowest-dimensioned CDF and work progressively to higher dimensions. As an example in 2D, one first samples from the $C$ CDF to determine the starting energy of a particle as well as the group index to use in the $D$ CDF when sampling to find the spatial position. The spatial index found when sampling from $D$ is then used to sample from $E$ to find the direction index that defines the two appropriate direction cosines. Similarly in 3D, one first samples from the $B$ CDF to determine the face that the particle is starting on and then from the $C$ array to determine the group index to use in the $D$ CDF when sampling to find the spatial position. The spatial index found when sampling from $D$ is then used to sample from $E$ to find the direction index that defines the three appropriate direction cosines. This is very similar to the approach taken historically in MCNPBQ where one first samples from the $B$ CDF to determine the starting energy of a particle as well as the group index to use in the $C$ CDF when sampling to find the spatial position. The spatial index found when sampling from $C$ is then used to sample from $D$ to find the polar level and that polar level is then used to sample the $E$ CDF to find the azimuthal segment.

There are many methods available for sampling from probability density functions such as "mapping," "direct," "rejection," "multidimensional," "weighting," and "Metropolis" (Pevey, 2008). However, for the purpose of this work the focus will be on mapping since the discrete CDF is readily available. As such, one may consider a general CDF, $\Pi(x)$, which is formed as

$$\Pi(x) \equiv \int_{-\infty}^{x} \pi(x')dx' \tag{2.61}$$

where $\pi(x')dx'$ is any proper probability density function where the probability of event $x_i$ lies in $(x', x' + dx')$ and where $\Pi(x)$ is the probability that $x_i < x$. As one considers this CDF, the following three properties become evident:

1. $\Pi(-\infty) = 0$,

2. $\Pi(\infty) = 1$, and

3. $\Pi(x)$ is non-decreasing.

Furthermore, when considered over a finite range $[a, b]$, the first two properties reduce to $\Pi(a) = 0$ and $\Pi(b) = 1$, respectively. Due to the integral relationship of the CDF to the PDF, it may be noted that the difference in range between the CDF values at two dissimilar points is the probability of an event occurring between those points. When one considers a discrete case, the CDF is merely the accumulation of probabilities for each event preceding a particular event. Using this fact, it becomes evident that the magnitude of the step change in a CDF for a particular event is the probability of the particular event occurring.

Considering the facts presented in the previous discussion, one can sample from a discrete CDF given a uniform random variable $X \in [0, 1]$. Once $X$ is found, its value is "mapped" from its intersection with the CDF to the corresponding bin. Using Figure 2.6, one can see that the random variable $X$ maps to the fourth event bin, which coincidentally has the highest probability of occurrence since it has the largest step change in value of any bin. The process of obtaining a uniform (pseudo-)random variable and mapping it with a CDF to a particular event will become important in Section 3.3 when the modifications to the MCNP `source.F90` subroutine are discussed.

## 2.4 Starting Monte Carlo Particles

### 2.4.1 Particle Initial Energy

When sampling the CDF for particle energy, there are as many bins as there are neutron and photon energy groups (for a coupled cross-section set). As such, when one determines which bin the particle will arise from he or she knows both the particle type and energy group. At this point, a decision must be made about how to handle

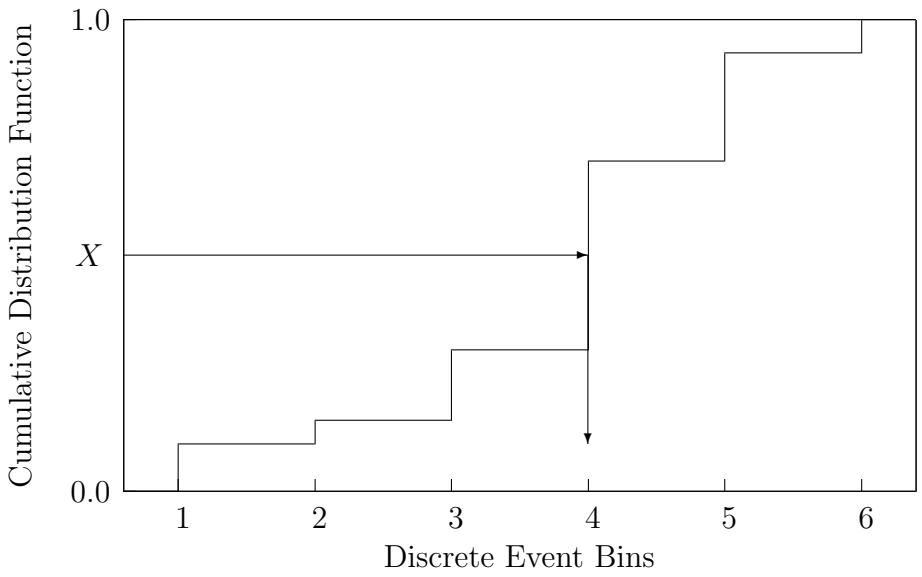



**Figure 2.6:** Example of Mapping a Uniform Random Variable to an Event with a CDF

the energy group versus the particle's initial energy. One has two options: each particle in a group can be assigned the group average energy or some energy from within the group where the most likely scenario is to randomly select an energy from the group in a uniform fashion.

For the purposes of this work, all particles arising from a particular energy group will begin with the average energy of that group.

### 2.4.2 Particle Initial Position

Since most Monte Carlo codes operate with a Cartesian coordinate system, whatever position is selected from the relevant CDFs must be transformed into Cartesian coordinates. For the $x$-$y$-$z$ case, this transformation is unnecessary because all coordinates are defined. In the $x$-$y$ case, there is no information about axial position. For this work, a base coordinate and distance to extrude along the corresponding axis is provided such that the unknown dimension is randomly chosen in the interval between the base ($b$) and the extruded length ($l$) in a uniform fashion. As an

example, the Monte Carlo position $(\bar{x}, \bar{y}, \bar{z})$ determined from an $x$-$y$ discrete ordinates calculation would be $(x, y, b + Xl)$ where one again uses uniform random variable $X \in [0, 1]$.

For the $r$-$\theta$-$z$ case, a coordinate transformation between cylindrical and Cartesian coordinates is all that is necessary. Therefore, the Monte Carlo particle's position is defined with

$$\bar{x} = r\cos(\theta), \tag{2.62}$$

$$\bar{y} = r\sin(\theta), \quad \text{and} \tag{2.63}$$

$$\bar{z} = z. \tag{2.64}$$

For the $r$-$\theta$ case, $\bar{x}$ and $\bar{y}$ are identical to Equation 2.62 and Equation 2.63, respectively. Meanwhile, $\bar{z}$ is defined as in the $x$-$y$ case. For the $r$-$z$ case, $\bar{z}$ is identical to Equation 2.64. However, now $\theta$ is selected using the base/extrusion technique leading to $\theta = b + Xl$. Once $\theta$ is selected, $\bar{x}$ and $\bar{y}$ are identical to Equation 2.62 and Equation 2.63.

### 2.4.3 Particle Initial Direction

The initial direction of the Monte Carlo particle is complicated by two factors. First, two-dimensional discrete ordinates calculations generally specify only two of the three direction cosines necessary to determine particle direction in a 3D coordinate system. Second, for cylindrical cases the particle direction is dependent on the position of the particle because the $i$ and $j$ direction cosines are relative to the radial direction of the particle.

For the 2D cases, the approach taken in this work is to use the information from the two direction cosines available and to not calculate the third direction cosine. As such, the unspecified and uncalculated direction cosine will be invariant. This leads to the situation that for an $r$-$z$ case all particles are directed outward along the radial

direction and for an $r$-$\theta$ or $x$-$y$ case all particles are directed with no axial trajectory (i.e., parallel to the $x$-$y$ plane).

As such, for the $x$-$y$ case the Monte Carlo initial particle direction vector with components $(\bar{u}, \bar{v}, \bar{w})$ is defined with

$$\bar{u} = \mu/\sqrt{\mu^2 + \eta^2}, \tag{2.65}$$

$$\bar{v} = \eta/\sqrt{\mu^2 + \eta^2}, \quad \text{and} \tag{2.66}$$

$$\bar{w} = 0. \tag{2.67}$$

However, the $r$-$z$ and $r$-$\theta$ cases are complicated by the fact mentioned previously that $\mu$ and $\eta$ are relative to the radial direction of the particle. This is illustrated in Figure 2.7 where $\bar{x}$ and $\bar{y}$ define the particle starting position at point $p$ and the direction cosines define the particle starting direction in a local frame of reference. As

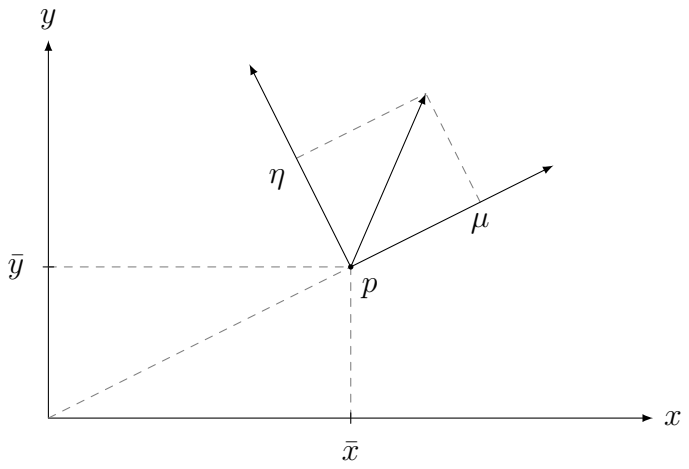



**Figure 2.7:** Global (Monte Carlo Cartesian) and Local (Discrete Ordinates Cylindrical) Frames of Reference with Particle Direction Vector

such, for these direction cosines to become useful in the global Monte Carlo coordinate system, the direction of the particle must be transformed into the global coordinate system from the local one. This is done by considering a local coordinate system $x'$-$y'$ with its origin at $p$ and its axes parallel to the global coordinate system $x$-$y$ and a unit vector in the local system from the origin along the radial direction of the

particle of interest terminating at point $(\bar{x}', \bar{y}')$. Another local coordinate system $\mu'$-$\eta'$ is defined with the $\mu'$ axis is coincident with the radial direction of the particle of interest. The direction cosines are used to find angle $\theta' = \arctan(\eta/\mu)$ that is used to

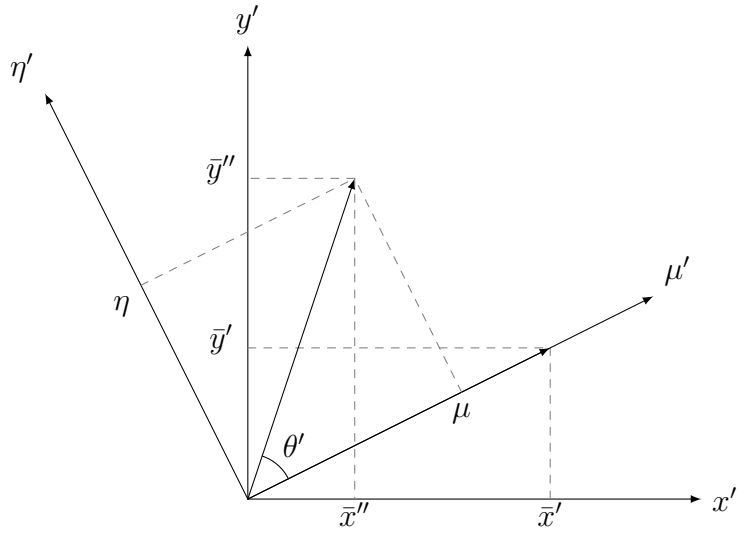



**Figure 2.8:** Local Coordinate Systems used to Transform Direction Cosine-based Particle Direction to Global Direction

rotate the local $x'$-$y'$ unit vector into the $\mu'$-$\eta'$ system to give a vector terminating at point $(\bar{x}'', \bar{y}'')$. Once rotated, that vector defines the $u$ and $v$ components of the Monte Carlo particle. In order to rotate the vectors as described previously, two equations as expressed by Korn and Korn (2000) are used where

$$\bar{x}'' = \bar{x}' \cos(\theta') - \bar{y}' \sin(\theta'), \quad \text{and} \tag{2.68}$$

$$\bar{y}'' = \bar{x}' \sin(\theta') + \bar{y}' \cos(\theta'). \tag{2.69}$$

Since $(\bar{x}', \bar{y}')$ were unitized, $\bar{u} = \bar{x}''$, $\bar{v} = \bar{y}''$, and $\bar{w} = 0$.

In 3D for the $x$-$y$-$z$ case, the direction cosines $\mu$, $\eta$, and $\xi$ are along local orthogonal axes that directly correspond to the global axes. As such, one can simply define a

vector with components

$$\bar{u} = \mu, \tag{2.70}$$

$$\bar{v} = \eta, \quad \text{and} \tag{2.71}$$

$$\bar{w} = \xi \tag{2.72}$$

to obtain the Monte Carlo initial particle direction vector. However, in the $r$-$\theta$-$z$ case one must perform operations similar to the $r$-$z$ and $r$-$\theta$ cases with the addition of an axial term that *is* parallel to the global $z$ axis. Therefore, once the rotation by $\theta'$ is performed, the unit vector is scaled down by the projection of the direction cosine vector onto the $x$-$y$ plane. That is, the Monte Carlo direction vector becomes

$$\bar{u} = \frac{\xi}{\tan(\arcsin(\xi))} \left(\bar{x}' \cos(\theta') - \bar{y}' \sin(\theta')\right), \tag{2.73}$$

$$\bar{v} = \frac{\xi}{\tan(\arcsin(\xi))} \left(\bar{x}' \sin(\theta') + \bar{y}' \cos(\theta')\right), \quad \text{and} \tag{2.74}$$

$$\bar{w} = \xi. \tag{2.75}$$

# Chapter 3

# Numerical Implementation

In order to implement the methodology given in Section 2.2, a new code was written dubbed DISCO (**D**eterm**I**nistic-**S**tochastic **C**oupling **O**peration). Due to the need to interface with DORT/TORT binary files as well as MCNP, DISCO was written primarily in FORTRAN 90 with some C extensions on a GNU/Linux 2.6 system and compiled with the Intel® compilers for Linux, `ifort` and `icc`, as appropriate. To implement the methodology given in Section 2.3 and subsequently in Section 2.4, the standard MCNP `source.F90` routine was modified and compiled with the other MCNP source subroutines into a non-standard executable with the tools listed previously. It should be noted that the `source.F90` subroutine is a standard part of MCNP meant to allow a user to define their own source functionality, so no other modifications to MCNP were necessary except for the inclusion of appropriate variable definitions in the `mcnp_global.F90` subroutine.

## 3.1 DISCO Code Structure

As described previously, DISCO was written primarily in FORTRAN 90. Furthermore, it was written in a modular fashion whereby each geometric case was given its own subroutine. Since the processing is very similar between different geometries,

this lead to significant amounts of duplicate code. While convenient in the short-term, this code should be condensed into a more maintainable format in the future. A program flow diagram is given in Figure 3.1 that describes the high-level logic and subroutine calls performed as DISCO is executed. The specific functionality of the subroutines cited in Figure 3.1 and those supplementary subroutines not cited is given in Section 3.1.1 through Section 3.1.12.

### 3.1.1 DISCO Subroutine: `main.f90`

This subroutine governs overall program execution and acts as the logical entry point. It first opens file handles for the input and output files and then prints configuration control and copyright information. It then calls `read_input.f90`. Once returned, it calls reads other subroutines to read the appropriate angular flux binary file and another to construct the CDFs for the geometry specified by the user.

### 3.1.2 DISCO Subroutine: `read_input.f90`

This subroutine only reads user input from a file assumed to be named or linked in as "`input`." As input is processed, it is echoed back out into an output file named "`output`." The user input file is roughly separated into two sections: namelist and FIDO array.

The namelist section governs non-array information. There are four namelists used in DISCO. The `INPUT` namelist contains general problem information that is common between all DISCO executions. The `DOMINO_INPUTS` namelist is only expected if DOMINO mode is enabled in the `INPUT` namelist and contains input fields consistent with those in DOMINO. Finally, the `TWODEEINPUT` and `THREEDEEINPUT` namelists contain geometry specific fields.

The FIDO section contains arrays that define the problem parameters such as quadrature set parameters, spatial mesh, and energy group boundaries.

The input file specification with accompanying options is available in Appendix A.

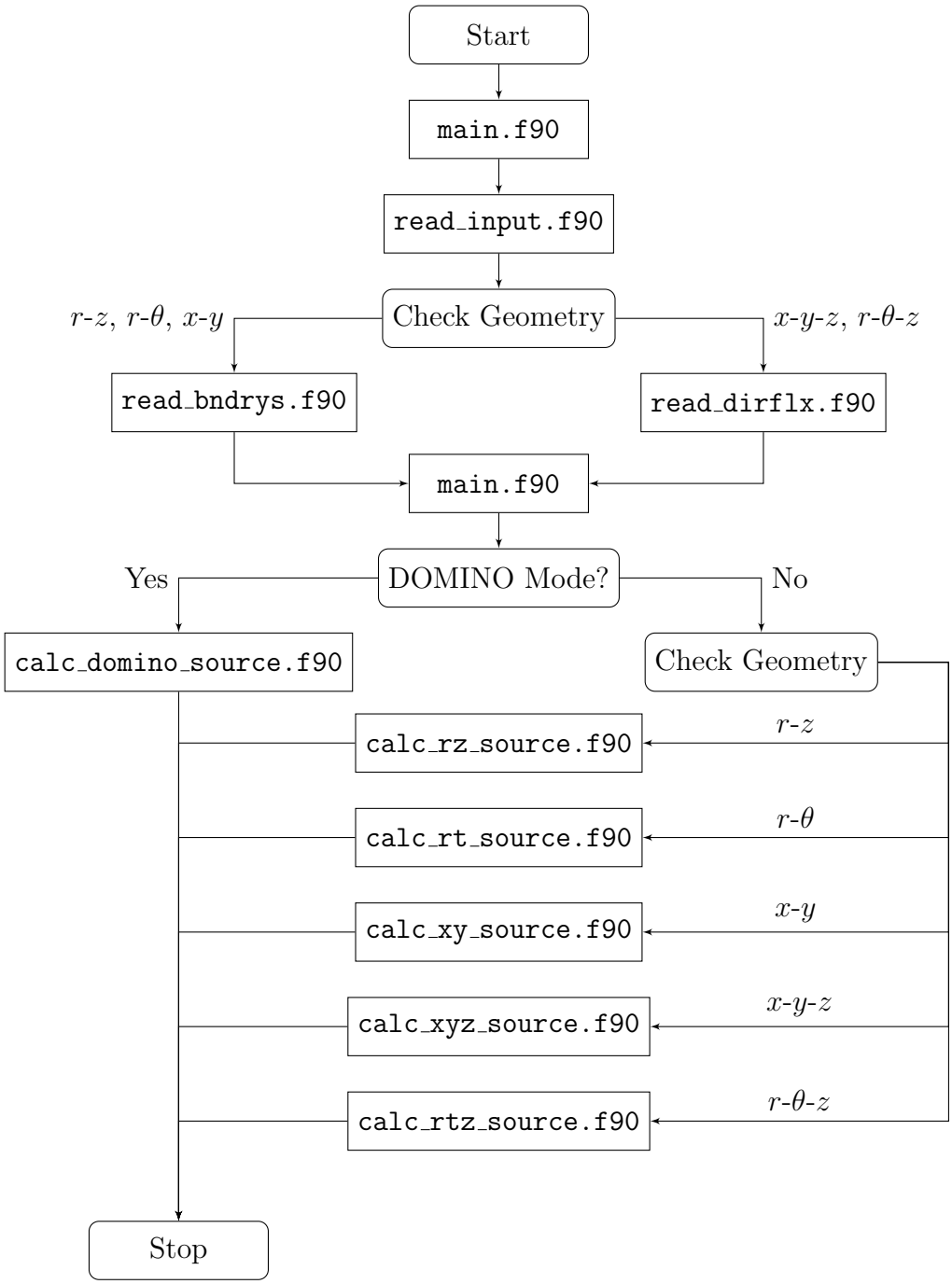



**Figure 3.1:** DISCO Program Flow

### 3.1.3 DISCO Subroutine: `read_bndrys.f90`

This subroutine is only used in cases where 2D geometries are being considered. The `bndrys` format is specified by Emmett et al. (1992) on pages 4-50–4-52. Based on the geometry specified by the user, directional flux data is read (from a file assumed to be named or linked in as "fort.12") and processed into a general `bndrys` array within DISCO. Negative angular flux values are set to zero.

### 3.1.4 DISCO Subroutine: `read_dirflx.f90`

This subroutine is only used in cases where 3D geometries are being considered. The `dirflx` format is specified by Rhoades and Simpson (1997) on pages B-2–B-4. Based on the geometry specified by the user, directional flux data is read (from a file assumed to be named or linked in as "fort.12") and processed into a general `dirflx` array within DISCO. Negative angular flux values are set to zero.

One consideration for future work is whether to include the ability to read TORT-produced "reduced-size" angular flux files. Using the TORT `ntdir` input field, a user specifies the unit number to record angular flux information to. If the user chooses a negative integer, a "reduced-size" file is written. This reduced size file is a record of the directional flux only on the boundary of the volume specified by the user. As such, it is exactly what DISCO needs. However, no file format specification was available for the "reduced-size" file during code development. Since the ability to use the volumetric source could potentially be useful to a user, the "full-size" file was used instead. For the test problems considered in this work, either approach is suitable. However, in an industrial application, the "reduced-size" file will almost certainly be necessary. Using a 96-direction, 67-group problem, a source for $100\times100\times100$ mesh is approximately 25GB. On a typical modern computer, this will overwhelm the available physical memory. Some additional discussion of future work in this area is available in Section 5.2.

### 3.1.5 DISCO Subroutine: `calc_domino_source.f90`

This subroutine uses the user input and 2D, $r$-$z$, angular flux information processed previously to calculate CDFs identical in form and content to those formed by DOMINO. Each CDF described in Section 2.1.1 is an array in this subroutine. Furthermore, this subroutine writes the CDF arrays and associated information to the "fort.9" unformatted binary file. The "fort.9" file is readable by MCNPBQ. The reproduction of DOMINO functionality is important to be able to reproduce historical calculations and perform thorough verification and validation (V&V) testing on DISCO.

### 3.1.6 DISCO Subroutine: `calc_rz_source.f90`

This subroutine uses the user input and 2D, $r$-$z$, angular flux information processed previously to calculate CDFs according to the equations found in Section 2.2.2. Each CDF is an array in this subroutine. Furthermore, this subroutine writes the CDF arrays and associated information to the "fort.9" unformatted binary file. The "fort.9" file is readable by MCNP using the modified `source.F90` subroutine described in Section 3.3.

### 3.1.7 DISCO Subroutine: `calc_rt_source.f90`

This subroutine uses the user input and 2D, $r$-$\theta$, angular flux information processed previously to calculate CDFs according to the equations found in Section 2.2.3. Each CDF is an array in this subroutine. Furthermore, this subroutine writes the CDF arrays and associated information to the "fort.9" unformatted binary file. The "fort.9" file is readable by MCNP using the modified `source.F90` subroutine described in Section 3.3.

### 3.1.8 DISCO Subroutine: `calc_xy_source.f90`

This subroutine uses the user input and 2D, $x$-$y$, angular flux information processed previously to calculate CDFs according to the equations found in Section 2.2.4. Each CDF is an array in this subroutine. Furthermore, this subroutine writes the CDF arrays and associated information to the "fort.9" unformatted binary file. The "fort.9" file is readable by MCNP using the modified `source.F90` subroutine described in Section 3.3.

### 3.1.9 DISCO Subroutine: `calc_xyz_source.f90`

This subroutine uses the user input and 3D, $x$-$y$-$z$, angular flux information processed previously to calculate CDFs according to the equations found in Section 2.2.6. Each CDF is an array in this subroutine. Furthermore, this subroutine writes the CDF arrays and associated information to the "fort.9" unformatted binary file. The "fort.9" file is readable by MCNP using the modified `source.F90` subroutine described in Section 3.3.

### 3.1.10 DISCO Subroutine: `calc_rtz_source.f90`

This subroutine uses the user input and 3D, $r$-$\theta$-$z$, angular flux information processed previously to calculate CDFs according to the equations found in Section 2.2.7. Each CDF is an array in this subroutine. Furthermore, this subroutine writes the CDF arrays and associated information to the "fort.9" unformatted binary file. The "fort.9" file is readable by MCNP using the modified `source.F90` subroutine described in Section 3.3.

### 3.1.11 DISCO FIDO Subroutines

FIDO was selected for use in DISCO because of user familiarity, commonality with the discrete ordinates codes that DISCO works in conjunction with, and the

availability of processing routines for FIDO information. As such, several FIDO subroutines were extracted from the SCALE6 package as documented in Petrie (2009). These subroutines are `ffpack.f90`, `fido_M.f90`, `Vast_kind_param_M.f90`, and `Y0trns_M.f90`. These four subroutines work in conjunction to read and return pointer-referenced arrays of user input data in the `read_input.f90` subroutine.

### 3.1.12 DISCO LAPACK Sorting Subroutine: `slasrt.f90`

In order to reproduce the functionality of DOMINO, ordering of the quadrature set arrays needed to be performed by polar level and azimuthal segment depending on the source type (disc or cylinder) and source direction (inward, outward, upward, or downward). As such, the LAPACK sorting routine `slasrt.f` (Anderson et al., 1999) was retrieved and modified to fit the current need and to become FORTRAN 90 compliant. In order to meet the current need, the subroutine was modified to not only sort an array but to also rearrange a corresponding index array so that the sorted array can be traced back to its original order. This is crucial to ensure that directional fluxes are multiplied by the correct direction cosine parameters $\mu$, $\eta$, and $w$ without the need to also rearrange the `bndrys` array.

## 3.2 DISCO Error Checking

There are several subroutines in DISCO that contain significant error checking. Most of this is error checking of strictly the user input but since the `bndrys` and `dirflx` files contain some internal parameters, there are some consistency checks between the user input and those binary files. In the case of all error checks, if a check is failed execution is halted with a `stop` and an error message is printed.

### 3.2.1 Error Checks in `calc_domino_source.f90`

Since the user is free to enter position values in terms of centimeters rather than mesh interval number, there is potential for the user to enter off-mesh values. This is handled in four separate checks on the `alr1`, `alr2`, `aud1`, and `aud2` values. If any of these checks fail the user must adjust the offending parameter to include a value in the $r$-mesh or $z$-mesh array, as appropriate.

### 3.2.2 Error Checks in `read_bndrys.f90`

DORT is capable of writing out multiple boundary sources along both the $i$ and $j$ directions to the same `bndrys` file. However, DISCO expects only one boundary source and expects it to be consistent with the user input. As such, `bndrys` values `nintsr` and `njntsr` must sum to one.

If the problem geometry is $r$-$z$, the only valid source types are "disc" or "cylinder." For each type, there are three error checks. One checks whether the source is of the appropriate type, the other two check if there are more than one of that type, or none of that type. A check is also made to ensure that the number of energy groups in the `bndrys` match the amount specified in user input. The mesh spacing in the `bndrys` file are also checked to ensure that `ima` is equal to the user input for `im` and the `bndrys jm` value is equal to the user input for `km`. Finally, the `bndrys mma` value is checked against the user's `ndirs` value for consistency.

If the problem geometry is $r$-$\theta$, the only valid source types are "cylinder" or "plane." For each type, there are three error checks. One checks whether the source is of the appropriate type, the other two check if there are more than one of that type, or none of that type. A check is also made to ensure that the number of energy groups in the `bndrys` match the amount specified in user input. The mesh spacing in the `bndrys` file are also checked to ensure that `ima` is equal to the user input for `im` and the `bndrys jm` value is equal to the user input for `jm`. Finally, the `bndrys mma` value is checked against the user's `ndirs` value for consistency.

If the problem geometry is $x$-$y$, the only valid source types are "xplane" or "yplane." For each type, there are three error checks. One checks whether the source is of the appropriate type, the other two check if there are more than one of that type, or none of that type. A check is also made to ensure that the number of energy groups in the `bndrys` match the amount specified in user input. The mesh spacing in the `bndrys` file are also checked to ensure that `ima` is equal to the user input for `im` and the `bndrys jm` value is equal to the user input for `jm`. Finally, the `bndrys mma` value is checked against the user's `ndirs` value for consistency.

### 3.2.3 Error Checks in `read_dirflx.f90`

TORT is capable of writing out significantly more helpful information to the `dirflx` file than DORT is capable of writing to the corresponding `bndrys` file. This presents a conflict to the developer. While useful information can be gleaned from the `dirflx` file reducing the need for some user inputs, input files become inconsistent in structure. This inconsistency was deemed undesirable so the user must specify all parameters in the user input and not rely on the `dirflx` file for anything other than angular flux data.

The first error check is whether the quadrature set in the `dirflx` file varies by energy group. Currently, DISCO cannot handle this capability. Next, the amount of energy groups and quadrature set directions are confirmed to match those input by the user. Finally, the `dirflx` mesh size variables (`im`, `jm`, and `km`) are checked against the user input `im`, `jm`, and `km` values for consistency.

Since TORT is capable of outputting a spatial volume that is less than that of the whole problem, the volume boundaries are checked against the user input. As such, the following six conditions are confirmed:

1. `dirflx ipch1` must be $\leq$ `imin`,

2. `dirflx ipch2` must be $\geq$ `imax`,

3. `dirflx jpch1` must be $\leq$ `jmin`,

4. `dirflx jpch2` must be $\geq$ `jmax`,

5. `dirflx kpch1` must be $\leq$ `kmin`, and

6. `dirflx kpch2` must be $\geq$ `kmax`.

As long as the six preceding conditions are satisfied, the volume source generated by the user in TORT is guaranteed to encompass the region required by DISCO.

### 3.2.4 Error Checks in `read_input.f90`

The majority of the error checking in DISCO is present in the `read_input.f90` to ensure cross-consistency in user inputs. The file is structured with a series of nested conditionals to enter into and out of to ensure that the appropriate error checks are performed as well as to echo and operate on input values, as appropriate.

If the user is using DISCO in DOMINO-compatibility mode, many error checks are made to ensure that the `DOMINO_INPUTS` readlist is consistent with the `INPUT` readlist. First, the geometry of the execution must be $r$-$z$ since that is all DOMINO is capable of operating on. Next, the `isides` and `ibot` values must be assigned to -1, 0, or 1. Following that, a check is performed to ensure that the sum of `isides` and `ibot` equals -1 or 1. Four checks are performed (two source types in two directions) to ensure consistency between user input and DOMINO input for the `isides` and `ibot` values. If a "disc" source is specified, `aud1` must equal `aud2`. Correspondingly, if a "cylinder" source is specified, `alr1` must equal `alr2`. The energy group to halt at is checked for consistency. Finally, the `imin`, `imax`, `jmin`, `kmax`, `kmin`, and `kmax` values are all confirmed to equal zero. Additional information regarding DOMINO input is available in Emmett et al. (1973).

If the user is using DISCO in its native form, other checks are made. First, the `geometry` is confirmed to be "rz," "rt," "xy," "rtz," or "xyz." A warning is issued if the energy group to begin sampling lies below the lowest-energy neutron group and the

group to end sampling occurs above it (i.e., a CDF is being created for both neutrons and photons). While this is valid, it may lead to poor sampling of both particle types if one particle dominates such as in non-power reactor cases. Furthermore, particular care must be taken in the application of the $A$, $A_{\mathrm{n}}$, and $A_{\mathrm{p}}$ normalization factors. The `verbosity` trigger is checked whether it is equal to an integer value 0, 1, 2 or 3. Checks are made to ensure that geometries are consistent with the source type desired with valid relationships listed in Table 3.1.

**Table 3.1:** DISCO Source Types and Directions that Correspond to Each Geometry

| Geometry | Valid Source Types | Valid Directions |
|---|---|---|
| $r$-$z$ | "disc" | "upward" or "downward" |
| $r$-$z$ | "cylinder" | "inward" or "outward" |
| $r$-$\theta$ | "plane" | "upward" or "downward" |
| $r$-$\theta$ | "cylinder" | "inward" or "outward" |
| $x$-$y$ | "xplane" | "inward" or "outward" |
| $x$-$y$ | "yplane" | "upward" or "downward" |
| $x$-$y$-$z$ | "volume" | "outward" |
| $r$-$\theta$-$z$ | "volume" | "outward" |

For $x$-$y$-$z$ or $r$-$\theta$-$z$ cases, additional checks are made on the `THREEDEEINPUT namelist` entries. Values `isrc_in`, `isrc_out`, `jsrc_in`, `jsrc_out`, `ksrc_in`, and `ksrc_out` are checked whether they are each equal either "yes" or "no" with at least one "yes."

Finally, as the `FIDO` routines are executed, the error code is returned and checked.

## 3.3 MCNP Subroutine: `source.F90`

The MCNP `source.F90` is called by MCNP when no other source definition (`sdef`, `ssr`, or `kcode`) is found in the input deck. As such, the `source.F90` subroutine was modified to read the CDF files (fort.9) produced by DISCO. Furthermore, the MCNP `mcnp_global.F90` subroutine was modified to include the necessary global

arrays so that the CDF array information need not be read in for every starting particle. Furthermore, when either subroutine was modified to include a DISCO-specific variable or array, the variable or array was prefixed with `cdf_` to create a pseudo-namespace to avoid conflict with preexisting MCNP variables.

MCNP expects to receive certain information from `source.F90`. This information includes the source particle initial position (`xxx`, `yyy`, `zzz` in centimeters), initial direction (`uuu`, `vvv`, `www` which define a unit vector), initial energy (`erg` in units of MeV), and starting cell (assumed to be "999"). Furthermore, variables which are not of interest to this discussion are set to default values (`vel` = 0., `jsu` = 0., `wgt` = 1., and `tme` = 0.). It is for the first three sets of variables that sampling is performed based on the DISCO information. Sampling is performed according to Section 2.3 where the MCNP `rang()` function is used to generate a pseudorandom number between 0 and 1.

The general process followed when not using MCNPBQ information is:

1. Determine DISCO geometry ($r$-$z$, $r$-$\theta$, $x$-$y$, $x$-$y$-$z$, or $r$-$\theta$-$z$).
2. Read auxiliary information such as energy group structure, spatial mesh structure, and quadrature set information.
3. Read appropriate CDF array information for $E$, $D$, and $C$ (as well as $B$, if 3D).
4. Sample from lowest dimensioned array ($C$ for 2D, $B$ for 3D).
5. Operate on data, as applicable (e.g., determine mean group energy if sampling energy or determine mean mesh spatial position if sampling position).
6. Sample progressively higher arrays based on previously found indicies from lower-ordered arrays.
7. Return source particle information to calling function and continue.

# Chapter 4

# Test Cases

## 4.1 Description of Test Cases

In order to test DISCO, 24 test cases were created. A summary of the test cases is given in Table 4.1 where the source and direction values correspond to those described in Section 2.2. All 2D test cases were executed with DORT and all 3D cases were executed with TORT. These cases were defined throughout the DISCO development process to provide diagnostic functional checks to the DISCO methodology while remaining simple and small to allow for rapid execution. One benefit of using DORT was the availability of zone-wise balance tables at the end of execution. These were instrumental in verifying DISCO at a bulk level. For all 2D test cases, the boundary source was generated at the zone 3/4 interface along the entire length of the problem. As such, the DORT balance table entries for zone outflow could be directly compared against the DISCO calculated total leakage value, $A$. Unfortunately, a similar check was unavailable for TORT which does not contain zone-wise balance tables. In order to further validate DISCO in 3D, zone-wise balance tables should be incorporated into TORT or another suitable 3D discrete ordinates code should be used to test against.

**Table 4.1:** DISCO Test Case Summary

| Case | Geometry | Source | Direction | Notes |
|---|---|---|---|---|
| 1 | $r$-$z$ | Disc | Upward | DOMINO Mode |
| 2 | $r$-$z$ | Disc | Downward | DOMINO Mode |
| 3 | $r$-$z$ | Cylinder | Outward | DOMINO Mode |
| 4 | $r$-$z$ | Cylinder | Inward | DOMINO Mode |
| 5 | $r$-$z$ | Disc | Upward | |
| 6 | $r$-$z$ | Disc | Downward | |
| 7 | $r$-$z$ | Cylinder | Outward | |
| 8 | $r$-$z$ | Cylinder | Inward | |
| 9 | $r$-$\theta$ | Cylinder | Outward | |
| 10 | $r$-$\theta$ | Cylinder | Inward | |
| 11 | $r$-$\theta$ | Plane | Upward | |
| 12 | $r$-$\theta$ | Plane | Downward | |
| 13 | $x$-$y$ | XPlane | Outward | |
| 14 | $x$-$y$ | XPlane | Inward | |
| 15 | $x$-$y$ | YPlane | Upward | |
| 16 | $x$-$y$ | YPlane | Downward | |
| 17 | $x$-$y$-$z$ | Volume | Outward | |
| 18 | $x$-$y$-$z$ | Volume | Outward | Shifted Source ($-x$) |
| 19 | $x$-$y$-$z$ | Volume | Outward | Shifted Source ($+x$) |
| 20 | $x$-$y$-$z$ | Volume | Outward | Shifted Source ($-y$) |
| 21 | $x$-$y$-$z$ | Volume | Outward | Shifted Source ($+y$) |
| 22 | $x$-$y$-$z$ | Volume | Outward | Shifted Source ($-z$) |
| 23 | $x$-$y$-$z$ | Volume | Outward | Shifted Source ($+z$) |
| 24 | $r$-$\theta$-$z$ | Volume | Outward | |

Throughout all of the test cases, only five materials were used which are all based on the work of White et al. (1999). These materials are "source" (0.0% enriched U-238), "water" (downcomer-weighted), "steel" (core-weighted stainless steel 304), "air", and "concrete" (concrete-weighted). These materials were arbitrarily chosen to provide a pseudo-realistic environment to transport particles through.

All test cases used a source defined as "1" for each energy group within each mesh of the source/fuel zone and "0" elsewhere. No additional normalization was performed. As such, one should not be surprised to see leakage flux values less than one particle/cm$^2$-s.

For test cases 5 through 24, a simple MCNP model was set up to validate particle starting behavior. Since the `source.F90` carries the assumption that particles would be starting in cell 999, a sphere with radius of 100 cm was centered at the origin. Naturally, in a realistic analysis it is incumbent upon the user to ensure that cell 999 adequately covers the true source volume.

### 4.1.1 DORT Geometry for $r$-$z$ Test Cases

The $r$-$z$ test case geometries were used for all comparisons between DISCO and DOMINO (since DOMINO only handles $r$-$z$ geometry) and to test the direct DISCO-to-MCNP scenario. Test cases 1, 2, 5, and 6 used the geometry shown in Figure 4.1 to generate disc sources. Test cases 3, 4, 7, and 8 used the geometry shown in Figure 4.2 to generate cylindrical sources. Both Figure 4.1 and Figure 4.2 show the problem mesh in light gray and zone boundaries in black with zone labels and axis labels, as appropriate. As noted previously, all boundary sources were generated on the zone 3/4 boundary (i.e., between the steel and air zones). Test cases 5, 6, 7, and 8 (used to validate the direct DISCO-to-MCNP scenario) generated a MCNP source with a base at 0.0 revolutions and an azimuthal extrusion of 0.125 revolutions.

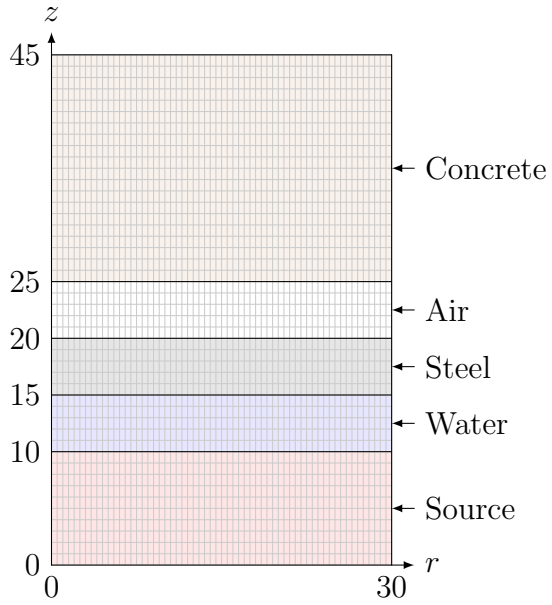



**Figure 4.1:** Test Geometry for $r$-$z$ Upward/Downward Source

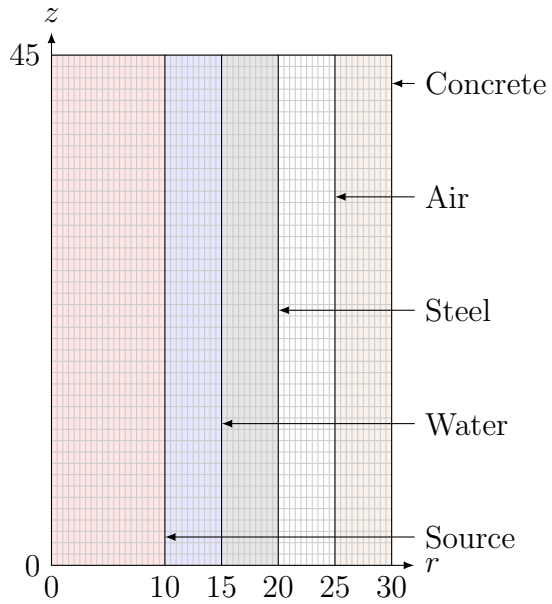



**Figure 4.2:** Test Geometry for $r$-$z$ Inward/Outward Source

### 4.1.2 DORT Geometry for $r$-$\theta$ Test Cases

The $r$-$\theta$ test case geometries were used only to test the direct DISCO-to-MCNP scenario. Test cases 9 and 10 used the geometry shown in Figure 4.3 to generate cylindrical sources. Test cases 11 and 12 used the geometry shown in Figure 4.4 to generate planar sources. Both Figure 4.3 and Figure 4.4 show the problem mesh in light gray and zone boundaries in black with zone labels and axis labels, as appropriate. As noted previously, all boundary sources were generated on the zone 3/4 boundary (i.e., between the steel and air zones). Test cases 9, 10, 11, and 12 (used to validate the direct DISCO-to-MCNP scenario) generated a MCNP source with a base at 5.0 cm and an axial extrusion of 10.0 cm.

### 4.1.3 DORT Geometry for $x$-$y$ Test Cases

The $x$-$y$ test case geometries were used only to test the direct DISCO-to-MCNP scenario. Test cases 13 and 14 used the geometry shown in Figure 4.5 to generate planar sources at a fixed $x$ position. Test cases 15 and 16 used the geometry shown in Figure 4.6 to generate planar sources at a fixed $y$ position. Both Figure 4.5 and Figure 4.6 show the problem mesh in light gray and zone boundaries in black with zone labels and axis labels, as appropriate. As noted previously, all boundary sources were generated on the zone 3/4 boundary (i.e., between the steel and air zones). Test cases 13, 14, 15, and 16 (used to validate the direct DISCO-to-MCNP scenario) generated a MCNP source with a base at 5.0 cm and an axial extrusion of 10.0 cm.

### 4.1.4 TORT Geometry for $x$-$y$-$z$ Test Cases

The $x$-$y$-$z$ test case geometry was used to verify correct reading of the angular flux (fort.12) file and to test the direct DISCO-to-MCNP scenario. Test case 17 used a series of three concentric cubes with a single mesh source within a $5 \times 5 \times 5$ mesh steel cube within a $25 \times 25 \times 25$ water cube as shown in Figure 4.7. This was expected to give equal leakage on all outside faces of the steel cube where the boundary source

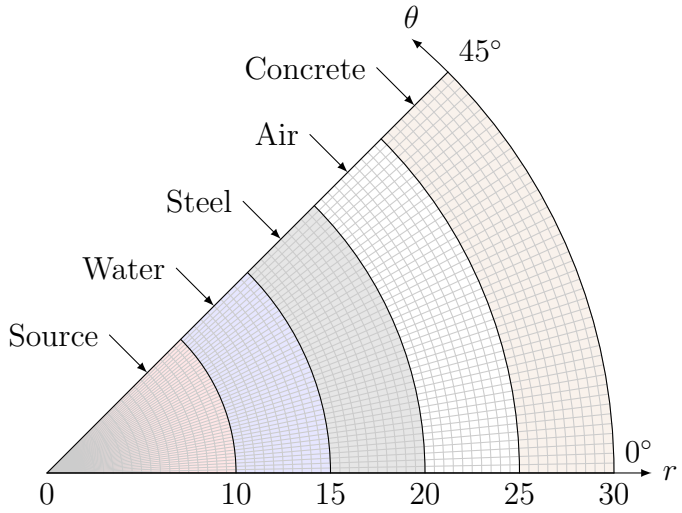



**Figure 4.3:** Test Geometry for $r$-$\theta$ Inward/Outward Source

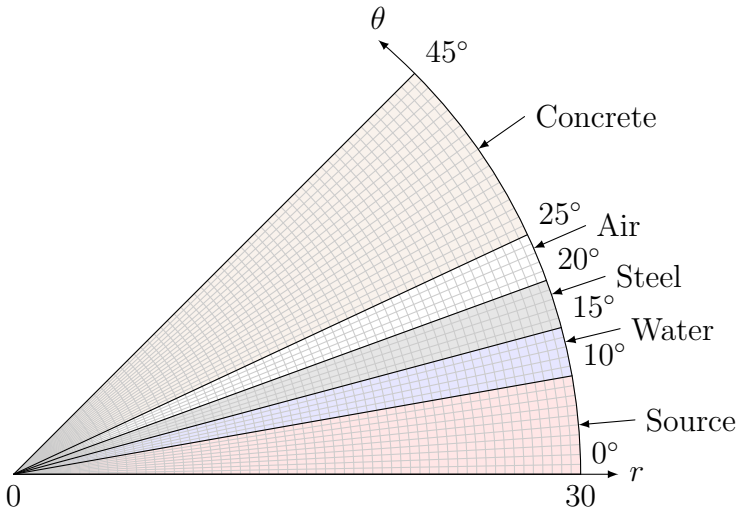



**Figure 4.4:** Test Geometry for $r$-$\theta$ Upward/Downward Source

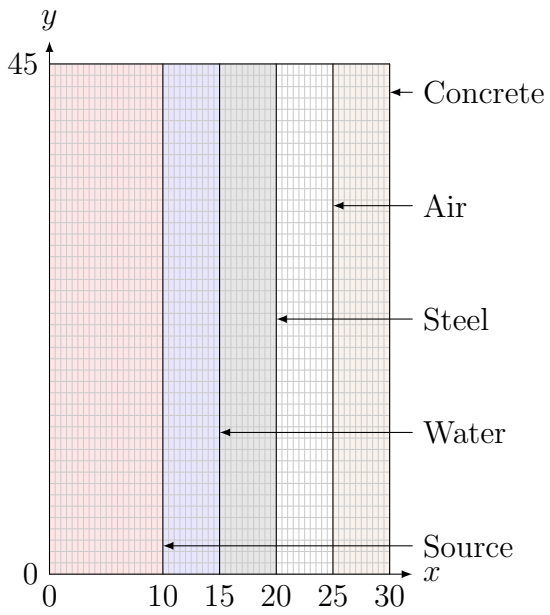



**Figure 4.5:** Test Geometry for $x$-$y$ Inward/Outward Source

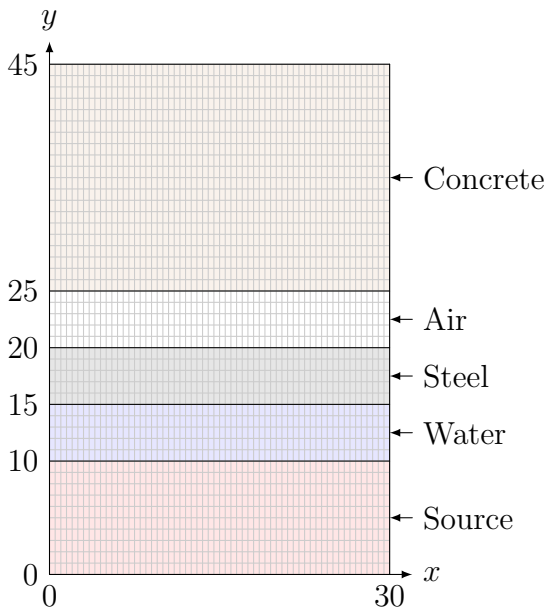



**Figure 4.6:** Test Geometry for $x$-$y$ Upward/Downward Source

was retrieved. Both Figure 4.7a and Figure 4.7b show a 2D projection of the problem mesh in light gray and zone boundaries in black with zone labels and axis labels, as appropriate. Test cases 18 through 23 then shifted the source cell within the steel cube to ensure that as the source shifted the leakage through the closest face increased accordingly. The shifts are all for one mesh in the direction denoted in Table 4.1. Since the discrete ordinates calculation fully defines the particle positions in space, no additional spatial information needs to be provided to DISCO.

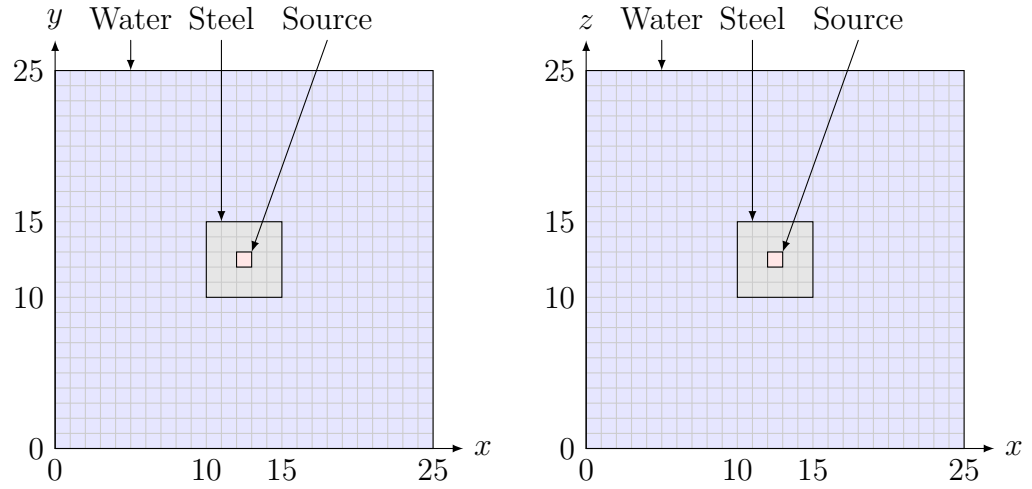
### 4.1.5 TORT Geometry for $r$-$\theta$-$z$ Test Cases

The $r$-$\theta$-$z$ test case geometry was used to test the direct DISCO-to-MCNP scenario. Test case 24 used a series of three 45° segments of concentric cylinders with a source cylinder within a steel cylinder within a water cylinder as shown in Figure 4.8. The boundary source was retrieved on the surface of the outside of the steel cylinder. Both Figure 4.8a and Figure 4.8b show a 2D projection of the problem mesh in light gray and zone boundaries in black with zone labels and axis labels, as appropriate. Since the discrete ordinates calculation fully defines the particle positions in space, no additional spatial information needs to be provided to DISCO.

## 4.2 DISCO Test Case Results Summary

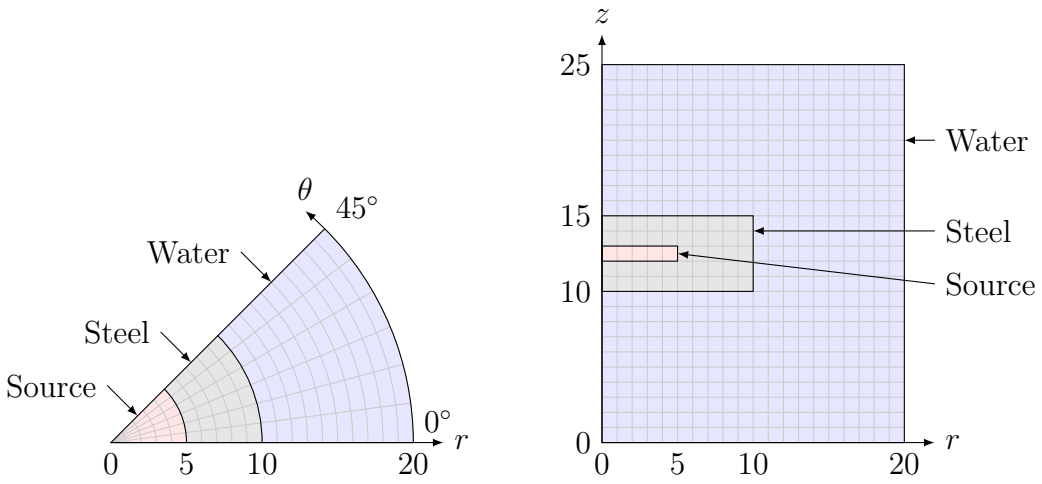
### 4.2.1 Results for $r$-$z$ Test Cases

For the comparisons between DISCO and DOMINO, both codes were executed for equivalent boundary source positions and extents. The final $A$ values for both calculations were compared and are shown in Table 4.2. The values agree with each other which suggests that the total leakage calculation is correct. Furthermore, the CDFs for energy are shown in Figure 4.9, Figure 4.10, Figure 4.11, and Figure 4.12 for the case 1–5, case 2–6, case 3–7, and case 4–8 comparisons, respectively. The energy CDFs match exactly in all cases. Additional extensive comparisons were made with




**(a)** $x$-$y$ Slice at $z = 12.5$cm **(b)** $x$-$z$ Slice at $y = 12.5$cm

**Figure 4.7:** Test Geometry for $x$-$y$-$z$




**(a)** $r$-$\theta$ Slice at $z = 12.5$cm **(b)** $r$-$z$ Slice at $\theta = 0°$

**Figure 4.8:** Test Geometry for $r$-$\theta$-$z$

the other CDFs which confirm agreement throughout the calculation process for both source types and directions. Therefore, one may conclude that the functionality of DOMINO has been sufficiently replicated in DISCO.

**Table 4.2:** DISCO/DOMINO $r$-$z$ Test Case Results

| Case | DISCO $A$ Value | DOMINO $A$ Value | DISCO $A$ / DOMINO $A$ |
|---|---|---|---|
| 1 | $1.47195 \times 10^{-01}$ | $1.472 \times 10^{-01}$ | 1.000 |
| 2 | $8.39026 \times 10^{-02}$ | $8.390 \times 10^{-02}$ | 1.000 |
| 3 | $2.19770 \times 10^{-01}$ | $2.198 \times 10^{-01}$ | 1.000 |
| 4 | $4.12925 \times 10^{-02}$ | $4.129 \times 10^{-02}$ | 1.000 |

For the comparisons between DISCO and DORT zone-wise balance tables in $r$-$z$ geometries, the DORT value was extracted from the output and compared with the direct DISCO-to-MCNP executions (test cases 5, 6, 7, and 8). The final $A$ value from DISCO was then compared with the DORT value which are both shown in Table 4.3.

**Table 4.3:** DISCO/DORT $r$-$z$ Test Case Results

| Case | DISCO $A$ Value | DORT Value | DISCO $A$ / DORT |
|---|---|---|---|
| 5 | $1.47195 \times 10^{-01}$ | $1.47195 \times 10^{-01}$ | 1.000 |
| 6 | $8.39026 \times 10^{-02}$ | $8.39026 \times 10^{-02}$ | 1.000 |
| 7 | $2.19770 \times 10^{-01}$ | $2.19770 \times 10^{-01}$ | 1.000 |
| 8 | $4.12925 \times 10^{-02}$ | $4.12925 \times 10^{-02}$ | 1.000 |

### 4.2.2 Results for $r$-$\theta$ Test Cases

For the comparisons between DISCO and DORT zone-wise balance tables in $r$-$\theta$ geometries, the DORT value was extracted from the output and compared with the direct DISCO-to-MCNP executions (test cases 9, 10, 11, and 12). The final $A$ value from DISCO was then compared with the DORT value which are both shown in Table 4.4.

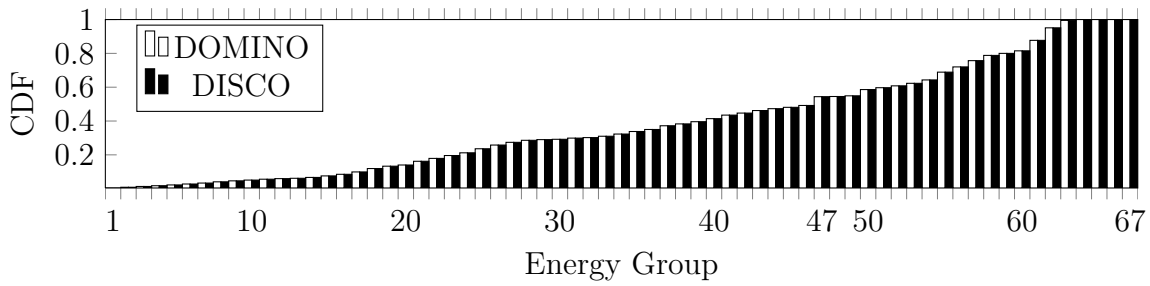



**Figure 4.9:** DOMINO Case 1 / DISCO Case 5 Energy CDF Comparison

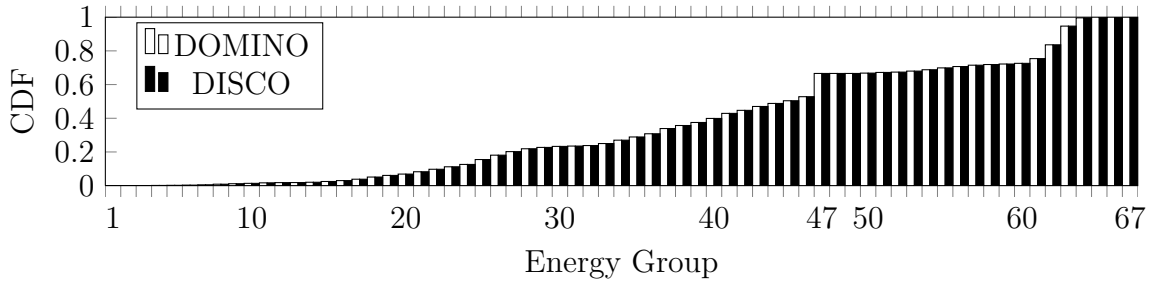



**Figure 4.10:** DOMINO Case 2 / DISCO Case 6 Energy CDF Comparison

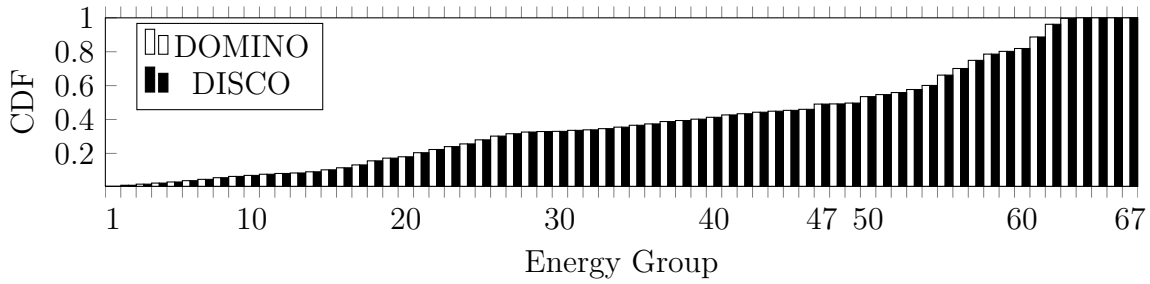



**Figure 4.11:** DOMINO Case 3 / DISCO Case 7 Energy CDF Comparison

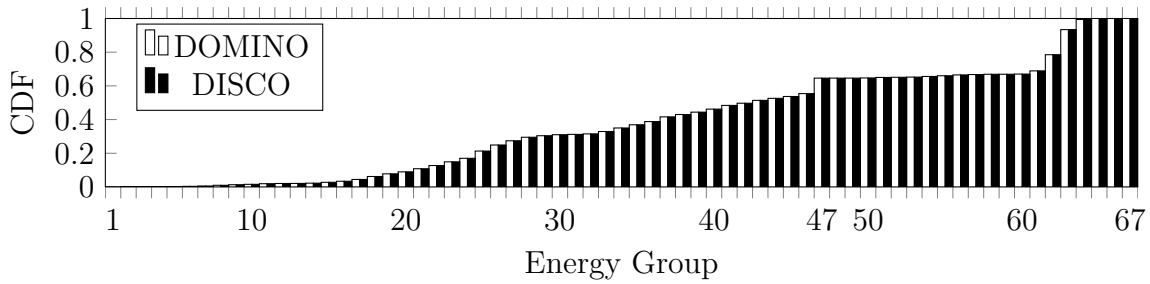



**Figure 4.12:** DOMINO Case 4 / DISCO Case 8 Energy CDF Comparison

**Table 4.4:** DISCO/DORT $r$-$\theta$ Test Case Results

| Case | DISCO $A$ Value | DORT Value | DISCO $A$ / DORT |
|---|---|---|---|
| 9 | $3.15380 \times 10^{-01}$ | $3.15380 \times 10^{-01}$ | 1.000 |
| 10 | $8.24948 \times 10^{-02}$ | $8.24944 \times 10^{-02}$ | 1.000 |
| 11 | $1.08554 \times 10^{00}$ | $1.08554 \times 10^{00}$ | 1.000 |
| 12 | $8.56119 \times 10^{-01}$ | $8.56119 \times 10^{-01}$ | 1.000 |

### 4.2.3 Results for $x$-$y$ Test Cases

For the comparisons between DISCO and DORT zone-wise balance tables in $x$-$y$ geometries, the DORT value was extracted from the output and compared with the direct DISCO-to-MCNP executions (test cases 13, 14, 15, and 16). The final $A$ value from DISCO was then compared with the DORT value which are both shown in Table 4.5.

**Table 4.5:** DISCO/DORT $x$-$y$ Test Case Results

| Case | DISCO $A$ Value | DORT Value | DISCO $A$ / DORT |
|---|---|---|---|
| 13 | $1.19546 \times 10^{-01}$ | $1.19546 \times 10^{-01}$ | 1.000 |
| 14 | $4.36713 \times 10^{-02}$ | $4.36713 \times 10^{-02}$ | 1.000 |
| 15 | $1.47324 \times 10^{-01}$ | $1.47324 \times 10^{-01}$ | 1.000 |
| 16 | $8.41416 \times 10^{-02}$ | $8.41416 \times 10^{-02}$ | 1.000 |

### 4.2.4 Results for $x$-$y$-$z$ Test Cases

As noted in Section 4.1.4, seven test cases were created to investigate DISCO for $x$-$y$-$z$ geometries. The first uses a centered source which should theoretically lead to equal leakage through each of the six faces of the steel zone surrounding the source. The subsequent six test cases shifted the source to investigate the impact of facial leakage. The CDFs for leakage through each face for each test case are shown in Figure 4.13 and the corresponding PDFs are shown in Figure 4.14. In each figure, the (C)PDF

represents leakage through the six boundary faces corresponding to the $-x$, $+x$, $-y$, $+y$, $-z$, and $+z$ faces (moving left to right for each case), respectively. Using the PDF for each case, one can confirm that the facial leakage probability increases as the source is moved toward the face with a corresponding drop in the opposite face.

### 4.2.5 Results for $r$-$\theta$-$z$ Test Cases

Since no self-consistency test cases were created in the $r$-$\theta$-$z$ geometry, there is a dearth of comparative information to allow one to confirm that DISCO is behaving properly. However, given the geometry described in Section 4.1.5, one might immediately conclude that there should be zero leakage through the $-r$ face since it is the inner boundary. Indeed, when one consults the facial leakage CDF shown in Figure 4.15, this behavior is confirmed. Furthermore, using Figure B.39 and Figure B.40, one can see that the leakage decreases as radius increases (i.e., as one moves away from the source). These figures also give a qualitative sense of source symmetry which further reinforces the argument that DISCO is operating properly.

## 4.3 MCNP Test Case Results Summary

As noted in Section 4.1, a simple MCNP test case was created for DISCO test cases 5–24 in order to validate the passing of data from DISCO as well as the modified `source.F90` subroutine. For all test cases a $x$-$y$ 2D projection, $x$-$z$ 2D projection, and 3D perspective projection of the first 500 particles' initial positions and directions is available in Appendix B. These particles were extracted from a modified `print table 110` edit. Furthermore, these particles were all taken from a serial execution of MCNP. Separate parallel runs of MCNP were made for each test case and the particle behavior was confirmed to match.

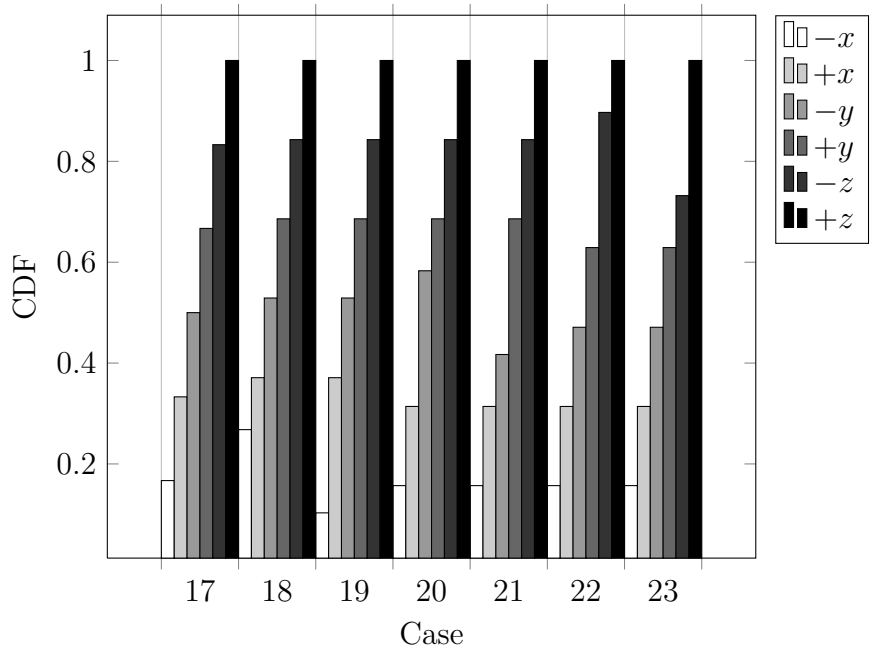



**Figure 4.13:** Test Case 17–23 Surface Leakage CDFs

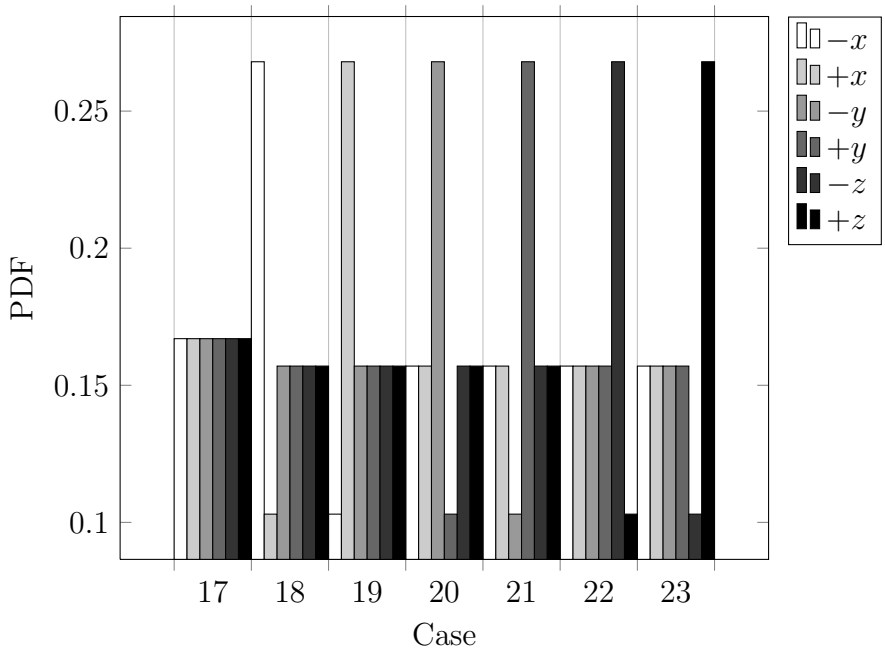



**Figure 4.14:** Test Case 17–23 Surface Leakage PDFs

Since a single sphere was used to contain the source with void inside and outside, two surface tallies were defined on the sphere to track the amount of neutrons and photons leaking out of the sphere. Since the exterior volume is a voided graveyard, there is no chance of particle re-entry. As such, the tallies can be compared with the DISCO $A_{\mathrm{n}}/A$ and $A_{\mathrm{p}}/A$ values to confirm that the ratio of the fraction of neutrons and photons is consistent between DISCO and those transported in MCNP. Indeed, Table 4.6 and Table 4.7 shows good agreement ($\pm 5\%$ for neutrons and $\pm 13\%$ for photons) while noting that the MCNP executions used 1000 source particles (total). As such, better agreement is expected with a greater amount of source particles.

**Table 4.6:** DISCO/MCNP Source Neutron Fraction Comparison

| **Case** | **DISCO $A_{\mathrm{n}}/A$ Value** | **MCNP Value** | **DISCO / MCNP** |
|---|---|---|---|
| 5 | 0.544 | 0.542 | 1.003 |
| 6 | 0.666 | 0.673 | 0.990 |
| 7 | 0.491 | 0.497 | 0.988 |
| 8 | 0.646 | 0.650 | 0.994 |
| 9 | 0.497 | 0.501 | 0.992 |
| 10 | 0.655 | 0.661 | 0.991 |
| 11 | 0.687 | 0.698 | 0.984 |
| 12 | 0.707 | 0.716 | 0.988 |
| 13 | 0.528 | 0.532 | 0.993 |
| 14 | 0.653 | 0.659 | 0.991 |
| 15 | 0.544 | 0.542 | 1.003 |
| 16 | 0.666 | 0.673 | 0.989 |
| 17 | 0.821 | 0.802 | 1.024 |
| 18 | 0.816 | 0.794 | 1.028 |
| 19 | 0.816 | 0.793 | 1.029 |
| 20 | 0.816 | 0.798 | 1.022 |
| 21 | 0.816 | 0.793 | 1.029 |
| 22 | 0.816 | 0.801 | 1.019 |
| 23 | 0.816 | 0.801 | 1.019 |
| 24 | 0.839 | 0.815 | 1.029 |

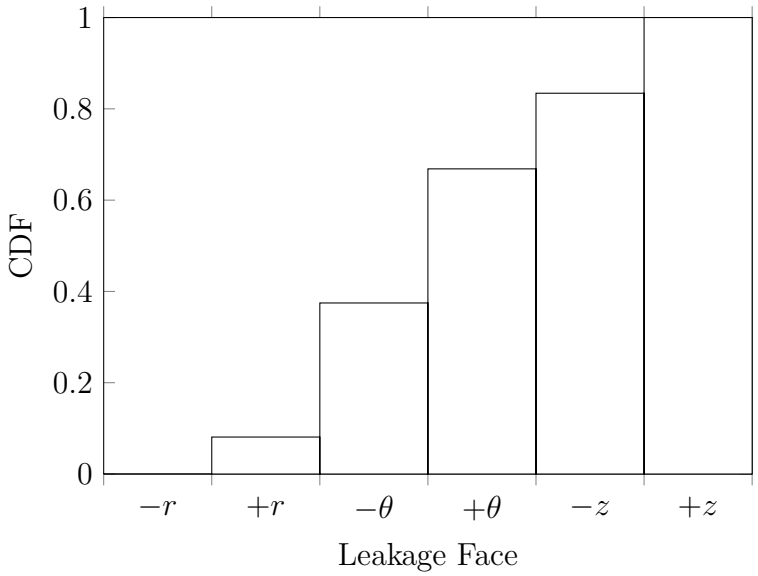



**Figure 4.15:** DISCO Case 24 Surface Leakage CDF

**Table 4.7:** DISCO/MCNP Source Photon Fraction Comparison

| Case | DISCO $A_p/A$ Value | MCNP Value | DISCO / MCNP |
|---|---|---|---|
| 5 | 0.456 | 0.458 | 0.996 |
| 6 | 0.334 | 0.327 | 1.021 |
| 7 | 0.509 | 0.503 | 1.012 |
| 8 | 0.354 | 0.350 | 1.011 |
| 9 | 0.503 | 0.499 | 1.008 |
| 10 | 0.345 | 0.339 | 1.018 |
| 11 | 0.313 | 0.302 | 1.036 |
| 12 | 0.293 | 0.284 | 1.030 |
| 13 | 0.472 | 0.468 | 1.008 |
| 14 | 0.347 | 0.341 | 1.018 |
| 15 | 0.456 | 0.458 | 0.997 |
| 16 | 0.334 | 0.327 | 1.022 |
| 17 | 0.179 | 0.198 | 0.904 |
| 18 | 0.184 | 0.206 | 0.894 |
| 19 | 0.184 | 0.207 | 0.889 |
| 20 | 0.184 | 0.202 | 0.911 |
| 21 | 0.184 | 0.207 | 0.889 |
| 22 | 0.184 | 0.199 | 0.925 |
| 23 | 0.184 | 0.199 | 0.925 |
| 24 | 0.161 | 0.185 | 0.872 |

# Chapter 5

# Summary & Future Work

## 5.1 Summary

The DISCO code created herein is capable of processing boundary angular flux data from $r$-$z$, $r$-$\theta$, $x$-$y$, $x$-$y$-$z$, and $r$-$\theta$-$z$ geometries into cumulative probability density functions that can be used with the MCNP `source.F90` routine to define neutron and photon initial positions, directions, and energies. Furthermore, MCNP is able to operate on this data in either serial or parallel execution modes. Finally, the `source.F90` routine created for MCNP is sufficiently modular such that it is capable of serially reading predefined particle initial positions, directions, and energies.

The quantitative test case results given in Chapter 4 and the qualitative initial particle positions and directions shown in Appendix B provide a high level of confidence that the methodology described in Chapter 2 was implemented correctly for all geometries. All comparisons between DOMINO and DISCO results match exactly. Furthermore, leakage values computed by DISCO match DORT balance table leakage values exactly. Particle behavior in 3D cases react accordingly to source perturbations.

While there are several areas for future investigation, the DISCO code and `source.F90` subroutine created and documented herein comprise a complete software

package ready for use. Furthermore, they incorporate several novel features such as compatibility with all 2D and 3D Cartesian and cylindrical geometries, an angular and spatial indexing scheme to reduce random sampling operations, a streamlining of process execution, and the ability for the resulting Monte Carlo code to operate in either serial and parallel mode. Therefore, one may conclude that the objective of this body of work was fully accomplished.

## 5.2 Future Work

There are several areas for future investigation and improvement to further validate DISCO and make it more robust, flexible, and suitable for use in industry.

First, all functional tests performed herein are for simplistic and small-scale geometries. No benchmarking against previous analyses and/or plant measurements was performed (such as the calculation performed by Fero et al. (1996)). As such, these sorts of calculations should be performed to further validate DISCO's functionality. Furthermore, only a loose comparison was available in the 3D cases because TORT lacks balance table output that yields zone-wise surface leakage values. Therefore, TORT should be modified or a different 3D discrete ordinates code should be identified in order to perform direct test case comparisons in 3D similar to those performed between DORT and DISCO for the 2D test cases.

Next, there are several opportunities available to incorporate additional flexibility in DISCO. As mentioned in Section 3.1.4, the reduced-size angular flux file from TORT is most appropriate for use in DISCO. However, it was not incorporated due to the lack of documentation describing the binary file format. Nevertheless, it would be beneficial to incorporate the ability to read this type of file to extend DISCO beyond being a proof-of-concept code to a code that is applicable to industrial-scale problems. For larger problems where even a true boundary angular flux in 3D exceeds the computer memory available, it may be suitable to read in the scalar flux data and use it to create a source term with isotropic particle initial directions. Rather than

using only the boundary values from the TORT full-size angular flux file, there may be some value in generating a 3D volumetric source. The usefulness and implications of all of these situations should be investigated and considered in future versions of DISCO.

Finally, there are some long term actions that could also take DISCO beyond being a proof-of-concept code to one that is more suitable for industrial use. DISCO (and the corresponding `source.F90` subroutine) assumes that the deterministic and stochastic coordinate systems exist with a coincident origin. However, future versions of DISCO should incorporate the ability to shift the Monte Carlo particle initial positions in $x$, $y$, and/or $z$ to allow for offset coordinate systems. The ability to rotate and/or scale the Monte Carlo coordinate system based on the discrete ordinates coordinate system is less important but may also be incorporated to provide total flexibility to the user. Also, due to the manner in which each geometry type was implemented, there is an immense amount of duplicate code as described in Section 3.1. In order to make DISCO more maintainable, this code should be condensed and the duplicate code should be removed. Furthermore, the two most common coordinate systems in commercial power reactor analyses (Cartesian and cylindrical) were incorporated in DISCO. In the future spherical and unstructured mesh geometries could be incorporated.

# Bibliography

# Bibliography

Anderson, E., Bai, Z., Bischof, C., Blackford, S., Demmel, J., Dongarra, J., Croz, J. D., Greenbaum, A., Hammarling, S., McKenney, A., and Sorensen, D. (1999). *LAPACK User's Guide Third Edition*. Philadelphia, Pennsylvania.

Emmett, M. B. (1985). *MORSE-CGA, A Monte Carlo Radiation Transport Code with Array Geometry Capability*. Oak Ridge, Tennessee. ORNL-6174.

Emmett, M. B., Burgart, C. E., and Hoffman, T. J. (1973). *DOMINO, A General Purpose Code for Coupling Discrete Ordinates and Monte Carlo Radiation Transport Calculations*. Oak Ridge, Tennessee. ORNL-4853.

Emmett, M. B., Rhoades, W. A., Childs, R. L., and Johnson, J. O. (1992). *the DORT section of: A User's Manual for MASH 1.0 — A Monte Carlo Adjoint Shielding Code System*. Oak Ridge, Tennessee. ORNL/TM-13221.

Fero, A. H., Sterritt, R. C., and Tarzia, J. P. (1996). Analysis and Measurements of Neutron Streaming Shields at Seabrook. In *ANS Radiation Protection & Shielding Topical Meeting: Advancements and Applications in Radiation Protection Shielding*, North Falmouth, Massachusetts. American Nuclear Society.

Girard, S. M., Booth, T. E., Brown, F. B., Bull, J. S., Forster, R. A., Goorley, J. T., Hughes, H. G., Martz, R., Prael, R. E., Sood, A., Sweezy, J. E., Zukaitis, A., Little, R. C., White, M. C., Lee, M. B., and Trellue, H. (2008). *MCNP — A General Monte Carlo N-Particle Transport Code, Version 5*. Los Alamos, New Mexico. LA-UR-03-1987.

Hagler, R. J. and Fero, A. H. (2005). ISFSI Site Boundary Radiation Dose Rate Analyses. *Radiation Protection Dosimetry*, 116(1-4):411–416.

Korn, G. A. and Korn, T. M. (2000). *Mathematical Handbook for Scientists and Engineers*. Dover, Mineola, NY.

Kulesza, J. A. (2009). Sensitivity Study to Optimize the Thickness of a Moderator for use in the AP1000 Source, Intermediate, and Power Range Excore Detectors. In *International Conference on Mathematics, Computational Methods & Reactor Physics (M&C 2009)*, Saratoga Springs, NY. American Nuclear Society.

Kurasawa, M. (2005). TORT/MCNP Coupling Method for the Calculation of Neutron Flux around a Core of BWR. *Radiation Protection Dosimetry*, 116(1-4):513–517.

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

Lloyd, T. M. (1997). Investigation of the Radiation Environment of PWR Baffle-Former Bolts. Master's thesis, The Pennsylvania State University, State College, PA.

Petrie, L. M. (2009). *FIDO Input System*. Oak Ridge, Tennessee. ORNL/TM-2005/39, Version 6, Vol. III, Sec. M10.

Pevey, R. E. (2008). *Lecture Notes from Nuclear Engineering Course NE582: Monte Carlo Analysis*. University of Tennessee: Knoxville, Knoxville, Tennessee.

Rhoades, W. A. and Simpson, D. B. (1997). *The TORT Three-Dimensional Discrete Ordinates Neutron/Photon Transport Code*. Oak Ridge, Tennessee. ORNL/TM-13221.

Ronen, Y. (1986). *CRC Handbook of Nuclear Reactors Calculations*, volume 1. CRC Press, Inc., Boca Raton, FL.

White, J. E., Ingersoll, D. T., Wright, R. Q., Hunter, H. T., Slater, C. O., Greene, N. M., MacFarlane, R. E., and Roussin, R. W. (1999). *Production and Testing of the Revised VITAMIN-B6 Fine-Group and the BUGLE-96 Broad-Group Neutron/Photon Cross-Section Libraries Derived from ENDF/B-VI.3 Nuclear Data*. Oak Ridge, Tennessee. DLC-185.

# Appendix

# Appendix A

# DISCO Input File Specification

As noted in Section 3.1.2, the DISCO input file is effectively split into into two sections: namelist and FIDO array. However, the first line is expected to be a title card no greater than 80 characters long. Following that is the `&INPUT` namelist. Both of these components are expected in every input file and an example is given in Listing A.1.

**Listing A.1:** DISCO Input File Header

```
Test Problem Title
&INPUT
   geometry       =
   source_type    =
   direction      =
   domino_mode    =
   verbosity      =
   ndirs          =
   igm            =
   nneut          =
   ngstart        =
   ngstop         =
   im             =
   jm             =
   km             =
   imin           =
   imax           =
   jmin           =
   jmax           =
   kmin           =
```

```
21    kmax              =
22 /
```

Valid entries for "geometry" are "rz", "rt", "xy", "xyz", and "rtz". Valid entries for "source_type" and "direction" are given in Table A.1. Valid entries

**Table A.1:** DISCO Geometry, Source, and Direction Input Options

| Geometry | Source Type | Direction |
|---|---|---|
| rz | Disc | Upward |
| rz | Disc | Downward |
| rz | Cylinder | Outward |
| rz | Cylinder | Inward |
| rt | Cylinder | Outward |
| rt | Cylinder | Inward |
| rt | Plane | Upward |
| rt | Plane | Downward |
| xy | XPlane | Outward |
| xy | XPlane | Inward |
| xy | YPlane | Upward |
| xy | YPlane | Downward |
| xyz | Volume | Outward |
| rtz | Volume | Outward |

for "domino_mode" are "yes" or "no". If "yes" is selected, another namelist (`&DOMINO_INPUTS`) is required following the `&INPUT` namelist.

The "verbosity" flag takes integer values 0, 1, 2, or 3 and controls the level of output that DISCO produces. By default (verbosity = 0) in 2D, DISCO will print the energy ($C$) CDF array as well as $A$, $A_{\mathrm{n}}$, and $A_{\mathrm{p}}$, and in 3D DISCO will print the facial leakage ($B$) CDF array as well as $A$, $A_{\mathrm{n}}$, and $A_{\mathrm{p}}$. If "verbosity" is set to 1 then all unnormalized arrays ($E^*$, $D^*$, $C^*$, and $B^*$ if 3D or $E^*$, $D^*$, $C^*$ if 2D) are printed. If "verbosity" is set to 2 then all unnormalized arrays listed previously and normalized PDF arrays ($E$, $D$, $C$, and $B$ if 3D) are printed. If "verbosity" is set to 3 then all unnormalized arrays listed previously, normalized PDF arrays, and CDF arrays ($E$, $D$, $C$, and $B$ if 3D) are printed.

The "ndirs" value is the total amount of directions (zero and non-zero weighted) in the quadrature set. The "igm" value is the total number of energy groups in the discrete ordinates calculation. The "nneut" value is the last neutron group in the discrete ordinates calculation. The "ngstart" and "ngstop" values govern the starting (lowest numbered) and ending (highest numbered) energy groups for DISCO to process. The "im", "jm", and "km" values define the amount of spatial mesh in each direction. In 2D, the unrepresented dimension takes a value of 1. The "imin", "imax", "jmin", "jmin", "kmin", and "kmax" values are the mesh cells over which DISCO will operate. In 2D, the unrepresented dimension takes a value of 0.

If "domino_mode" is set to yes, the `&DOMINO_INPUTS` namelist is required as shown in Listing A.2. All values other than "tomcnpbq" function identically to standard DOMINO and will not be addressed herein since they are specified by Emmett et al. (1973). The "tomcnpbq" value accepts "yes" or "no" where "yes" will produce a fort.9 file suitable for processing by MCNPBQ.

**Listing A.2:** DISCO `&DOMINO_INPUTS` Namelist

```
23 &DOMINO_INPUTS
24    iggg            =
25    isides          =
26    ibot            =
27    alr1            =
28    alr2            =
29    aud1            =
30    aud2            =
31    tomcnpbq        =
32 /
```

If "domino_mode" is set to no, and "geometry" is either "rz", "rt", or "xy" then the `&TWODEEINPUT` namelist is required as shown in Listing A.3. The "base" value is the base dimension described in Section 2.4.2 and "extrusion" is distance to extrude the source also described in Section 2.4.2. For cylindrical sources in $r$-$z$ or $r$-$\theta$, "base" and "extrusion" are in units of revolutions. For all other 2D sources, both of these values are in centimeters.

**Listing A.3:** DISCO `&TWODEEINPUT` Namelist

```
23 &TWODEEINPUT
24    base            =
25    extrusion       =
26 /
```

If "domino_mode" is set to no, and "geometry" is either "xyz" or "rtz" then the `&THREEDEEINPUT` namelist is required as shown in Listing A.4. Each value in the `&THREEDEEINPUT` namelist can take a value of "yes" or "no" which governs whether leakage from a particular face will be considered.

**Listing A.4:** DISCO `&THREEDEEINPUT` Namelist

```
23 &THREEDEEINPUT
24    isrc_in         =
25    isrc_out        =
26    jsrc_in         =
27    jsrc_out        =
28    ksrc_in         =
29    ksrc_out        =
30 /
```

The "in" faces correspond to the lower extent for each dimension (e.g., $i-$, $j-$, $k-$) and the "out" faces correspond to the upper extent for each dimension (e.g., $i+$, $j+$, $k+$).

This concludes the namelist portion. Following the appropriate namelists, there are several FIDO arrays used to inform DISCO of the discrete ordinates calculation's parameters. Details regarding the FIDO specification are given by Petrie (2009) and will not be discussed here. However, depending on the problem geometry, some arrays may or may not be used. For a given geometry, the required arrays are listed in Table A.2.

The various FIDO arrays used in DISCO are as follows:

- 81** : Quadrature Set Directional Weights [ndirs entries],
- 82** : Quadrature Set $\mu$ Direction Cosines [ndirs entries],
- 83** : Quadrature Set $\eta$ Direction Cosines [ndirs entries],
- 84** : Quadrature Set $\xi$ Direction Cosines [ndirs entries],

**Table A.2:** DISCO Quadrature Set and Spatial Mesh FIDO Array Associations

| **Geometry** | **Quadrature Set Arrays** | **Spatial Mesh Arrays** |
|---|---|---|
| rz | 81**, 82**, 84** | 2**, 4** |
| rt | 81**, 82**, 83** | 2**, 3** |
| xy | 81**, 82**, 83** | 2**, 3** |
| xyz | 81**, 82**, 83**, 84** | 2**, 3**, 4** |
| rtz | 81**, 82**, 83**, 84** | 2**, 3**, 4** |

- 2** : $i^{\text{th}}$ Dimension Spatial Mesh Boundaries [im+1 entries],
- 3** : $j^{\text{th}}$ Dimension Spatial Mesh Boundaries [jm+1 entries],
- 4** : $k^{\text{th}}$ Dimension Spatial Mesh Boundaries [km+1 entries],
- 11** : Neutron Energy Group Boundaries, from High to Low Energy, in eV [nneut+1 entries], and
- 12** : Photon Energy Group Boundaries, from High to Low Energy, in eV [igm-nneut+1 entries].

## A.1 Example $r$-$z$ DISCO, DOMINO-Compatible, Input File

```
Test Problem Title
&INPUT
   geometry       = rz
   source_type    = disc
   direction      = upward
   domino_mode    = yes
   verbosity      = 3
   ndirs          = 160
   igm            = 67
   nneut          = 47
   ngstart        = 1
   ngstop         = 67
   im             = 60
   jm             = 1
   km             = 45
   imin           = 0
   imax           = 0
   jmin           = 0
   jmax           = 0
   kmin           = 0
   kmax           = 0
/
&DOMINO_INPUTS
   iggg           = 67
   isides         = 0
   ibot           = 1
   alr1           =   0.00
   alr2           =  30.00
   aud1           =  20.00
   aud2           =  20.00
   tomcnpbq       = yes
/
/ Directional Weights
/
81**
   +00000+0 2r+13586-6   +00000+0   +97681-7 2r+97681-7   +97681-7
   +00000+0   +64738-7   +50390-7 2r+64738-7   +50390-7   +64738-7
   +00000+0   +64634-7   +71124-7   +71124-7 2r+64634-7   +71124-7
   +71124-7   +64634-7   +00000+0   +64634-7   +14381-7   +36342-7
   +14381-7 2r+64634-7   +14381-7   +36342-7   +14381-7   +64634-7
   +00000+0   +64738-7   +71124-7   +36342-7   +36342-7   +71124-7
 2r+64738-7   +71124-7   +36342-7   +36342-7   +71124-7   +64738-7
   +00000+0   +97681-7   +50390-7   +71124-7   +14381-7   +71124-7
   +50390-7 2r+97681-7   +50390-7   +71124-7   +14381-7   +71124-7
   +50390-7   +97681-7   +00000+0   +13586-6   +97681-7   +64738-7
   +64634-7   +64634-7   +64738-7   +97681-7 2r+13586-6   +97681-7
   +64738-7   +64634-7   +64634-7   +64738-7   +97681-7   +13586-6
   +00000+0 2r+13586-6   +00000+0   +97681-7 2r+97681-7   +97681-7
   +00000+0   +64738-7   +50390-7 2r+64738-7   +50390-7   +64738-7
   +00000+0   +64634-7   +71124-7   +71124-7 2r+64634-7   +71124-7
   +71124-7   +64634-7   +00000+0   +64634-7   +14381-7   +36342-7
   +14381-7 2r+64634-7   +14381-7   +36342-7   +14381-7   +64634-7
   +00000+0   +64738-7   +71124-7   +36342-7   +36342-7   +71124-7
 2r+64738-7   +71124-7   +36342-7   +36342-7   +71124-7   +64738-7
   +00000+0   +97681-7   +50390-7   +71124-7   +14381-7   +71124-7
   +50390-7 2r+97681-7   +50390-7   +71124-7   +14381-7   +71124-7
   +50390-7   +97681-7   +00000+0   +13586-6   +97681-7   +64738-7
   +64634-7   +64634-7   +64738-7   +97681-7 2r+13586-6   +97681-7
   +64738-7   +64634-7   +64634-7   +64738-7   +97681-7   +13586-6
/
/ mu (Cosine of Angle Relative to R Direction)
/
82**
 -21082-5 -14907-5 +14907-5 -42164-5 -39441-5 -14907-5
 +14907-5 +39441-5 -55777-5 -53748-5 -39441-5 -14907-5
 +14907-5 +39441-5 +53748-5 -66667-5 -64979-5 -53748-5
 -39441-5 -14907-5 +14907-5 +39441-5 +53748-5 +64979-5
 -76012-5 -74536-5 -64979-5 -53748-5 -39441-5 -14907-5
 +14907-5 +39441-5 +53748-5 +64979-5 +74536-5 -84327-5
 -82999-5 -74536-5 -64979-5 -53748-5 -39441-5 -14907-5
 +14907-5 +39441-5 +53748-5 +64979-5 +74536-5 +82999-5
 -91894-5 -90676-5 -82999-5 -74536-5 -64979-5 -53748-5
 -39441-5 -14907-5 +14907-5 +39441-5 +53748-5 +64979-5
 +74536-5 +82999-5 +90676-5 -98883-5 -97753-5 -90676-5
 -82999-5 -74536-5 -64979-5 -53748-5 -39441-5 -14907-5
 +14907-5 +39441-5 +53748-5 +64979-5 +74536-5 +82999-5
 +90676-5 +97753-5 -21082-5 -14907-5 +14907-5 -42164-5
 -39441-5 -14907-5 +14907-5 +39441-5 -55777-5 -53748-5
 -39441-5 -14907-5 +14907-5 +39441-5 +53748-5 -66667-5
 -64979-5 -53748-5 -39441-5 -14907-5 +14907-5 +39441-5
```

```
 +53748-5 +64979-5 -76012-5 -74536-5 -64979-5 -53748-5
 -39441-5 -14907-5 +14907-5 +39441-5 +53748-5 +64979-5
 +74536-5 -84327-5 -82999-5 -74536-5 -64979-5 -53748-5
 -39441-5 -14907-5 +14907-5 +39441-5 +53748-5 +64979-5
 +74536-5 +82999-5 -91894-5 -90676-5 -82999-5 -74536-5
 -64979-5 -53748-5 -39441-5 -14907-5 +14907-5 +39441-5
 +53748-5 +64979-5 +74536-5 +82999-5 +90676-5 -98883-5
 -97753-5 -90676-5 -82999-5 -74536-5 -64979-5 -53748-5
 -39441-5 -14907-5 +14907-5 +39441-5 +53748-5 +64979-5
 +74536-5 +82999-5 +90676-5 +97753-5
/
/ csi (Cosine of Angle Relative to Z Direction)
/
83**
 3r-97753-5  5r-90676-5  7r-82999-5  9r-74536-5 11r-64979-5 13r-53748-5
15r-39441-5 17r-14907-5  3r+97753-5  5r+90676-5  7r+82999-5  9r+74536-5
11r+64979-5 13r+53748-5 15r+39441-5 17r+14907-5
/
/ I-mesh (X or R) Interval Boundaries
/
2**
  0.000000e+00  5.000000e-01  1.000000e+00  1.500000e+00  2.000000e+00
  2.500000e+00  3.000000e+00  3.500000e+00  4.000000e+00  4.500000e+00
  5.000000e+00  5.500000e+00  6.000000e+00  6.500000e+00  7.000000e+00
  7.500000e+00  8.000000e+00  8.500000e+00  9.000000e+00  9.500000e+00
  1.000000e+01  1.050000e+01  1.100000e+01  1.150000e+01  1.200000e+01
  1.250000e+01  1.300000e+01  1.350000e+01  1.400000e+01  1.450000e+01
  1.500000e+01  1.550000e+01  1.600000e+01  1.650000e+01  1.700000e+01
  1.750000e+01  1.800000e+01  1.850000e+01  1.900000e+01  1.950000e+01
  2.000000e+01  2.050000e+01  2.100000e+01  2.150000e+01  2.200000e+01
  2.250000e+01  2.300000e+01  2.350000e+01  2.400000e+01  2.450000e+01
  2.500000e+01  2.550000e+01  2.600000e+01  2.650000e+01  2.700000e+01
  2.750000e+01  2.800000e+01  2.850000e+01  2.900000e+01  2.950000e+01
  3.000000e+01
/
/ K-mesh (Z) Interval Boundaries
/
4**
  0.000000e+00  1.000000e+00  2.000000e+00  3.000000e+00  4.000000e+00
  5.000000e+00  6.000000e+00  7.000000e+00  8.000000e+00  9.000000e+00
  1.000000e+01  1.100000e+01  1.200000e+01  1.300000e+01  1.400000e+01
  1.500000e+01  1.600000e+01  1.700000e+01  1.800000e+01  1.900000e+01
  2.000000e+01  2.100000e+01  2.200000e+01  2.300000e+01  2.400000e+01
  2.500000e+01  2.600000e+01  2.700000e+01  2.800000e+01  2.900000e+01
  3.000000e+01  3.100000e+01  3.200000e+01  3.300000e+01  3.400000e+01
  3.500000e+01  3.600000e+01  3.700000e+01  3.800000e+01  3.900000e+01
  4.000000e+01  4.100000e+01  4.200000e+01  4.300000e+01  4.400000e+01
  4.500000e+01
/
/ Neutron Energy Group Boundaries
/
11**
1.73300E+07 1.41900E+07 1.22100E+07 1.00000E+07 8.60700E+06 7.40800E+06
6.06500E+06 4.96600E+06 3.67900E+06 3.01200E+06 2.72500E+06 2.46600E+06
2.36500E+06 2.34600E+06 2.23100E+06 1.92000E+06 1.65300E+06 1.35300E+06
1.00300E+06 8.20800E+05 7.42700E+05 6.08100E+05 4.97900E+05 3.68800E+05
2.97200E+05 1.83200E+05 1.11100E+05 6.73800E+04 4.08700E+04 3.18300E+04
2.60600E+04 2.41800E+04 2.18800E+04 1.50300E+04 7.10200E+03 3.35500E+03
1.58500E+03 4.54000E+02 2.14400E+02 1.01300E+02 3.72700E+01 1.06800E+01
5.04300E+00 1.85500E+00 8.76400E-01 4.14000E-01 1.00000E-01 1.00000E-05
/
/ Photon Energy Group Boundaries
/
12**
1.40000E+07 1.00000E+07 8.00000E+06 7.00000E+06 6.00000E+06 5.00000E+06
4.00000E+06 3.00000E+06 2.00000E+06 1.50000E+06 1.00000E+06 8.00000E+05
7.00000E+05 6.00000E+05 4.00000E+05 2.00000E+05 1.00000E+05 6.00000E+04
3.00000E+04 2.00000E+04 1.00000E+04
t
```

## A.2 Example $r$-$\theta$ DISCO Input File

```
Test Problem Title
&INPUT
   geometry       = rt
   source_type    = cylinder
   direction      = outward
   domino_mode    = no
   verbosity      = 0
   ndirs          = 160
   igm            = 67
   nneut          = 47
   ngstart        = 1
   ngstop         = 67
   im             = 60
   jm             = 45
   km             = 1
   imin           = 41
   imax           = 41
   jmin           = 1
   jmax           = 45
   kmin           = 0
   kmax           = 0
/
&TWODEEINPUT
   base           =  5.0
   extrusion      = 10.0
/
/ Directional Weights
/
81**
0 2r13586-6 0 4r97681-7 0 64738-7 5039-6 2r64738-7 5039-6 64738-7 0
64634-7 2r71124-7 2r64634-7 2r71124-7 64634-7 0 64634-7 14381-7 36342-7
14381-7 2r64634-7 14381-7 36342-7 14381-7 64634-7 0 64738-7 71124-7
2r36342-7 71124-7 2r64738-7 71124-7 2r36342-7 71124-7 64738-7 0 97681-7
5039-6 71124-7 14381-7 71124-7 5039-6 2r97681-7 5039-6 71124-7 14381-7
71124-7 5039-6 97681-7 0 13586-6 97681-7 64738-7 2r64634-7 64738-7
97681-7 2r13586-6 97681-7 64738-7 2r64634-7 64738-7 97681-7 13586-6 0
2r13586-6 0 4r97681-7 0 64738-7 5039-6 2r64738-7 5039-6 64738-7 0
64634-7 2r71124-7 2r64634-7 2r71124-7 64634-7 0 64634-7 14381-7 36342-7
14381-7 2r64634-7 14381-7 36342-7 14381-7 64634-7 0 64738-7 71124-7
2r36342-7 71124-7 2r64738-7 71124-7 2r36342-7 71124-7 64738-7 0 97681-7
5039-6 71124-7 14381-7 71124-7 5039-6 2r97681-7 5039-6 71124-7 14381-7
71124-7 5039-6 97681-7 0 13586-6 97681-7 64738-7 2r64634-7 64738-7
97681-7 2r13586-6 97681-7 64738-7 2r64634-7 64738-7 97681-7 13586-6
/
/ mu (Cosine of Angle Relative to R Direction)
/
82**
-21082-5 -14907-5 14907-5 -42164-5 -39441-5 -14907-5 14907-5 39441-5
-55777-5 -53748-5 -39441-5 -14907-5 14907-5 39441-5 53748-5 -66667-5
-64979-5 -53748-5 -39441-5 -14907-5 14907-5 39441-5 53748-5 64979-5
-76012-5 -74536-5 -64979-5 -53748-5 -39441-5 -14907-5 14907-5 39441-5
53748-5 64979-5 74536-5 -84327-5 -82999-5 -74536-5 -64979-5 -53748-5
-39441-5 -14907-5 14907-5 39441-5 53748-5 64979-5 74536-5 82999-5
-91894-5 -90676-5 -82999-5 -74536-5 -64979-5 -53748-5 -39441-5 -14907-5
14907-5 39441-5 53748-5 64979-5 74536-5 82999-5 90676-5 -98883-5
-97753-5 -90676-5 -82999-5 -74536-5 -64979-5 -53748-5 -39441-5 -14907-5
14907-5 39441-5 53748-5 64979-5 74536-5 82999-5 90676-5 97753-5 -21082-5
-14907-5 14907-5 -42164-5 -39441-5 -14907-5 14907-5 39441-5 -55777-5
-53748-5 -39441-5 -14907-5 14907-5 39441-5 53748-5 -66667-5 -64979-5
-53748-5 -39441-5 -14907-5 14907-5 39441-5 53748-5 64979-5 -76012-5
-74536-5 -64979-5 -53748-5 -39441-5 -14907-5 14907-5 39441-5 53748-5
64979-5 74536-5 -84327-5 -82999-5 -74536-5 -64979-5 -53748-5 -39441-5
-14907-5 14907-5 39441-5 53748-5 64979-5 74536-5 82999-5 -91894-5
-90676-5 -82999-5 -74536-5 -64979-5 -53748-5 -39441-5 -14907-5 14907-5
39441-5 53748-5 64979-5 74536-5 82999-5 90676-5 -98883-5 -97753-5
-90676-5 -82999-5 -74536-5 -64979-5 -53748-5 -39441-5 -14907-5 14907-5
39441-5 53748-5 64979-5 74536-5 82999-5 90676-5 97753-5
/
/ eta (Cosine of Angle Relative to T Direction)
/
83**
-1-5 2r-14904-5 -1-5 -14909-5 2r-39442-5 -14909-5 -1-5 -1491-4 -39441-5
2r-53749-5 -39441-5 -1491-4 -1-5 -14904-5 -3944-4 -53747-5 2r-64978-5
-53747-5 -3944-4 -14904-5 -1-5 -14904-5 -39439-5 -53748-5 -64978-5
2r-74535-5 -64978-5 -53748-5 -39439-5 -14904-5 -1-5 -1491-4 -3944-4
-53748-5 -64979-5 -74536-5 2r-83-2 -74536-5 -64979-5 -53748-5 -3944-4
-1491-4 -1-5 -14909-5 -39441-5 -53747-5 -64978-5 -74536-5 -82999-5
2r-90676-5 -82999-5 -74536-5 -64978-5 -53747-5 -39441-5 -14909-5 -1-5
-14904-5 -39442-5 -53749-5 -64978-5 -74535-5 -83-2 -90676-5 2r-97753-5
-90676-5 -83-2 -74535-5 -64978-5 -53749-5 -39442-5 -14904-5 1-5
2r14904-5 1-5 14909-5 2r39442-5 14909-5 1-5 1491-4 39441-5 2r53749-5
39441-5 1491-4 1-5 14904-5 3944-4 53747-5 2r64978-5 53747-5 3944-4
14904-5 1-5 14904-5 39439-5 53748-5 64978-5 2r74535-5 64978-5 53748-5
39439-5 14904-5 1-5 1491-4 3944-4 53748-5 64979-5 74536-5 2r83-2 74536-5
```

```
64979-5 53748-5 3944-4 1491-4 1-5 14909-5 39441-5 53747-5 64978-5
74536-5 82999-5 2r90676-5 82999-5 74536-5 64978-5 53747-5 39441-5
14909-5 1-5 14904-5 39442-5 53749-5 64978-5 74535-5 83-2 90676-5
2r97753-5 90676-5 83-2 74535-5 64978-5 53749-5 39442-5 14904-5
/
/ I-mesh (X or R) Interval Boundaries
/
2**
  0.000000e+00  5.000000e-01  1.000000e+00  1.500000e+00  2.000000e+00
  2.500000e+00  3.000000e+00  3.500000e+00  4.000000e+00  4.500000e+00
  5.000000e+00  5.500000e+00  6.000000e+00  6.500000e+00  7.000000e+00
  7.500000e+00  8.000000e+00  8.500000e+00  9.000000e+00  9.500000e+00
  1.000000e+01  1.050000e+01  1.100000e+01  1.150000e+01  1.200000e+01
  1.250000e+01  1.300000e+01  1.350000e+01  1.400000e+01  1.450000e+01
  1.500000e+01  1.550000e+01  1.600000e+01  1.650000e+01  1.700000e+01
  1.750000e+01  1.800000e+01  1.850000e+01  1.900000e+01  1.950000e+01
  2.000000e+01  2.050000e+01  2.100000e+01  2.150000e+01  2.200000e+01
  2.250000e+01  2.300000e+01  2.350000e+01  2.400000e+01  2.450000e+01
  2.500000e+01  2.550000e+01  2.600000e+01  2.650000e+01  2.700000e+01
  2.750000e+01  2.800000e+01  2.850000e+01  2.900000e+01  2.950000e+01
  3.000000e+01
/
/ J-mesh (T) Interval Boundaries
/
3**
  0.000000e+00  2.777778e-03  5.555556e-03  8.333333e-03  1.111111e-02
  1.388889e-02  1.666667e-02  1.944444e-02  2.222222e-02  2.500000e-02
  2.777778e-02  3.055556e-02  3.333333e-02  3.611111e-02  3.888889e-02
  4.166667e-02  4.444444e-02  4.722222e-02  5.000000e-02  5.277778e-02
  5.555556e-02  5.833333e-02  6.111111e-02  6.388889e-02  6.666667e-02
  6.944444e-02  7.222222e-02  7.500000e-02  7.777778e-02  8.055556e-02
  8.333333e-02  8.611111e-02  8.888889e-02  9.166667e-02  9.444444e-02
  9.722222e-02  1.000000e-01  1.027778e-01  1.055556e-01  1.083333e-01
  1.111111e-01  1.138889e-01  1.166667e-01  1.194444e-01  1.222222e-01
  1.250000e-01
/
/ Neutron Energy Group Boundaries
/
11**
1.73300E+07 1.41900E+07 1.22100E+07 1.00000E+07 8.60700E+06 7.40800E+06
6.06500E+06 4.96600E+06 3.67900E+06 3.01200E+06 2.72500E+06 2.46600E+06
2.36500E+06 2.34600E+06 2.23100E+06 1.92000E+06 1.65300E+06 1.35300E+06
1.00300E+06 8.20800E+05 7.42700E+05 6.08100E+05 4.97900E+05 3.68800E+05
2.97200E+05 1.83200E+05 1.11100E+05 6.73800E+04 4.08700E+04 3.18300E+04
2.60600E+04 2.41800E+04 2.18800E+04 1.50300E+04 7.10200E+03 3.35500E+03
1.58500E+03 4.54000E+02 2.14400E+02 1.01300E+02 3.72700E+01 1.06800E+01
5.04300E+00 1.85500E+00 8.76400E-01 4.14000E-01 1.00000E-01 1.00000E-05
/
/ Photon Energy Group Boundaries
/
12**
1.40000E+07 1.00000E+07 8.00000E+06 7.00000E+06 6.00000E+06 5.00000E+06
4.00000E+06 3.00000E+06 2.00000E+06 1.50000E+06 1.00000E+06 8.00000E+05
7.00000E+05 6.00000E+05 4.00000E+05 2.00000E+05 1.00000E+05 6.00000E+04
3.00000E+04 2.00000E+04 1.00000E+04
t
```

## A.3 Example $x$-$y$-$z$ DISCO Input File

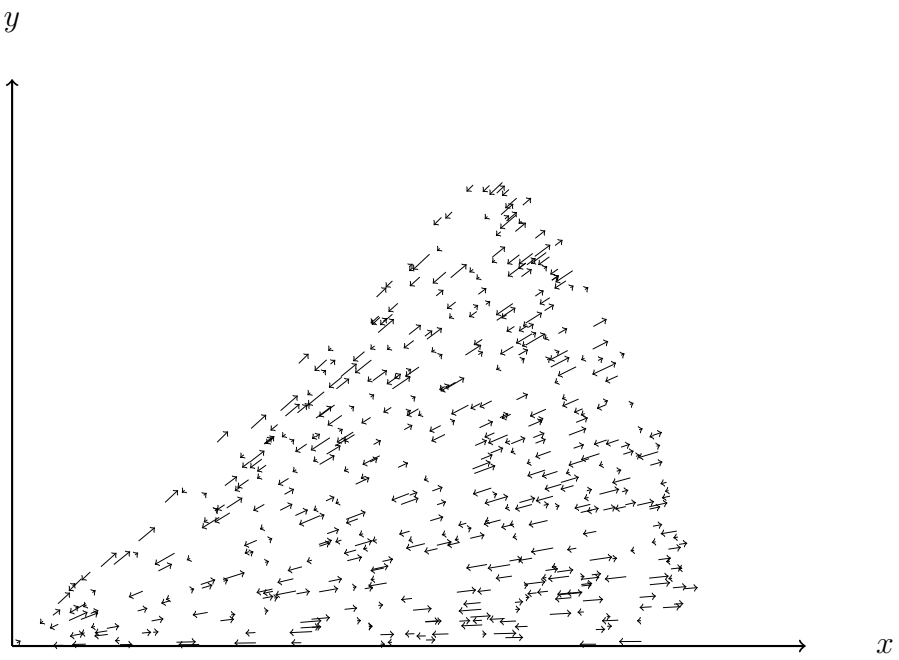
```
Test Problem Title
&INPUT
   geometry       = xyz            ! Geometry: rz, rt, xy, xyz, rtz
   source_type    = volume         ! Source type: disc, cylinder, plane, xplane, yplane, volume
   direction      = outward        ! Particle direction: upward, downward, inward, outward
   domino_mode    = no             ! Calculate and output like DOMINO
   verbosity      = 0              ! Detail level of output, 0, 1, 2, or 3
   ndirs          = 96             ! Number of directions
   igm            = 67             ! Number of energy groups
   nneut          = 47             ! Number of neutron groups
   ngstart        = 1              ! Group to begin sampling
   ngstop         = 67             ! Group to finish sampling
   im             = 25             ! Number of mesh in I direction
   jm             = 25             ! Number of mesh in J direction
   km             = 25             ! Number of mesh in K direction
   imin           = 11             ! Minimum I mesh to sample
   imax           = 15             ! Maximum I mesh to sample
   jmin           = 11             ! Minimum J mesh to sample
   jmax           = 15             ! Maximum J mesh to sample
   kmin           = 11             ! Minimum K mesh to sample
   kmax           = 15             ! Maximum K mesh to sample
/
&THREEDEEINPUT
   isrc_in        = yes            ! Generate Source on I- Surface at imin
   isrc_out       = yes            ! Generate Source on I+ Surface at imax
   jsrc_in        = yes            ! Generate Source on J- Surface at jmin
   jsrc_out       = yes            ! Generate Source on J+ Surface at jmax
   ksrc_in        = yes            ! Generate Source on K- Surface at kmin
   ksrc_out       = yes            ! Generate Source on K+ Surface at kmax
/
/ Directional Weights
/
81**
0 2r151235-7 0 4r113425-7 0 113425-7 11574-6 2r113425-7 11574-6 113425-7
0 151235-7 2r113425-7 2r151235-7 2r113425-7 151235-7
  3q  24
/
/ mu (Cosine of Angle Relative to X Direction)
/
82**
-30861-5 -21822-5 21822-5 -61721-5 -57735-5 -21822-5 21822-5 57735-5
-8165-4 -7868-4 -57735-5 -21822-5 21822-5 57735-5 7868-4 -9759-4
-95119-5 -7868-4 -57735-5 -21822-5 21822-5 57735-5 7868-4 95119-5
  3q  24
/
/ eta (Cosine of Angle Relative to Y Direction)
/
83**
  0.00000e+00 -2.18215e-01 -2.18215e-01  0.00000e+00 -2.18204e-01
 -5.77344e-01 -5.77344e-01 -2.18204e-01  0.00000e+00 -2.18204e-01
 -5.77351e-01 -7.86795e-01 -7.86795e-01 -5.77351e-01 -2.18204e-01
  0.00000e+00 -2.18215e-01 -5.77344e-01 -7.86795e-01 -9.51189e-01
 -9.51189e-01 -7.86795e-01 -5.77344e-01 -2.18215e-01  0.00000e+00
  2.18215e-01  2.18215e-01  0.00000e+00  2.18204e-01  5.77344e-01
  5.77344e-01  2.18204e-01  0.00000e+00  2.18204e-01  5.77351e-01
  7.86795e-01  7.86795e-01  5.77351e-01  2.18204e-01  0.00000e+00
  2.18215e-01  5.77344e-01  7.86795e-01  9.51189e-01  9.51189e-01
  7.86795e-01  5.77344e-01  2.18215e-01  0.00000e+00 -2.18215e-01
 -2.18215e-01  0.00000e+00 -2.18204e-01 -5.77344e-01 -5.77344e-01
 -2.18204e-01  0.00000e+00 -2.18204e-01 -5.77351e-01 -7.86795e-01
 -7.86795e-01 -5.77351e-01 -2.18204e-01  0.00000e+00 -2.18215e-01
 -5.77344e-01 -7.86795e-01 -9.51189e-01 -9.51189e-01 -7.86795e-01
 -5.77344e-01 -2.18215e-01  0.00000e+00  2.18215e-01  2.18215e-01
  0.00000e+00  2.18204e-01  5.77344e-01  5.77344e-01  2.18204e-01
  0.00000e+00  2.18204e-01  5.77351e-01  7.86795e-01  7.86795e-01
  5.77351e-01  2.18204e-01  0.00000e+00  2.18215e-01  5.77344e-01
  7.86795e-01  9.51189e-01  9.51189e-01  7.86795e-01  5.77344e-01
  2.18215e-01
/
/ csi (Cosine of Angle Relative to Z Direction)
/
84**
3r-95119-5 5r-7868-4 7r-57735-5 9r-21822-5
   q  24 g  48
/
/ I-mesh (X or R) Interval Boundaries
/
2**
  0.000000e+00  1.000000e+00  2.000000e+00  3.000000e+00  4.000000e+00
  5.000000e+00  6.000000e+00  7.000000e+00  8.000000e+00  9.000000e+00
  1.000000e+01  1.100000e+01  1.200000e+01  1.300000e+01  1.400000e+01
  1.500000e+01  1.600000e+01  1.700000e+01  1.800000e+01  1.900000e+01
  2.000000e+01  2.100000e+01  2.200000e+01  2.300000e+01  2.400000e+01
  2.500000e+01
```

```
/
/ J-mesh (Y or T) Interval Boundaries
/
3**
  0.000000e+00  1.000000e+00  2.000000e+00  3.000000e+00  4.000000e+00
  5.000000e+00  6.000000e+00  7.000000e+00  8.000000e+00  9.000000e+00
  1.000000e+01  1.100000e+01  1.200000e+01  1.300000e+01  1.400000e+01
  1.500000e+01  1.600000e+01  1.700000e+01  1.800000e+01  1.900000e+01
  2.000000e+01  2.100000e+01  2.200000e+01  2.300000e+01  2.400000e+01
  2.500000e+01
/
/ K-mesh (Z) Interval Boundaries
/
4**
  0.000000e+00  1.000000e+00  2.000000e+00  3.000000e+00  4.000000e+00
  5.000000e+00  6.000000e+00  7.000000e+00  8.000000e+00  9.000000e+00
  1.000000e+01  1.100000e+01  1.200000e+01  1.300000e+01  1.400000e+01
  1.500000e+01  1.600000e+01  1.700000e+01  1.800000e+01  1.900000e+01
  2.000000e+01  2.100000e+01  2.200000e+01  2.300000e+01  2.400000e+01
  2.500000e+01
/
/ Neutron Energy Group Boundaries
/
11**
1.73300E+07 1.41900E+07 1.22100E+07 1.00000E+07 8.60700E+06 7.40800E+06
6.06500E+06 4.96600E+06 3.67900E+06 3.01200E+06 2.72500E+06 2.46600E+06
2.36500E+06 2.34600E+06 2.23100E+06 1.92000E+06 1.65300E+06 1.35300E+06
1.00300E+06 8.20800E+05 7.42700E+05 6.08100E+05 4.97900E+05 3.68800E+05
2.97200E+05 1.83200E+05 1.11100E+05 6.73800E+04 4.08700E+04 3.18300E+04
2.60600E+04 2.41800E+04 2.18800E+04 1.50300E+04 7.10200E+03 3.35500E+03
1.58500E+03 4.54000E+02 2.14400E+02 1.01300E+02 3.72700E+01 1.06800E+01
5.04300E+00 1.85500E+00 8.76400E-01 4.14000E-01 1.00000E-01 1.00000E-05
/
/ Photon Energy Group Boundaries
/
12**
1.40000E+07 1.00000E+07 8.00000E+06 7.00000E+06 6.00000E+06 5.00000E+06
4.00000E+06 3.00000E+06 2.00000E+06 1.50000E+06 1.00000E+06 8.00000E+05
7.00000E+05 6.00000E+05 4.00000E+05 2.00000E+05 1.00000E+05 6.00000E+04
3.00000E+04 2.00000E+04 1.00000E+04
t
```

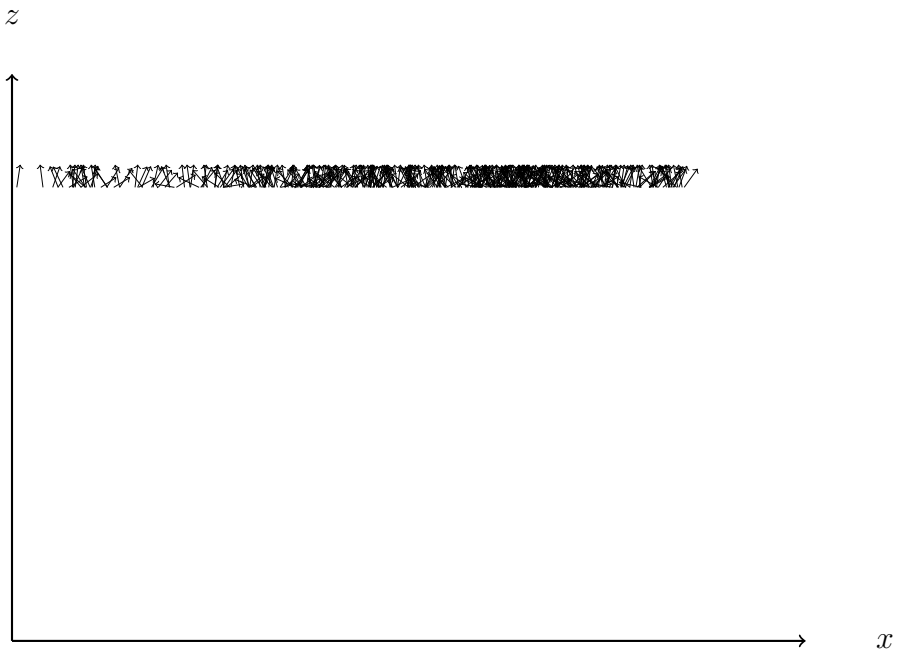
# Appendix B

# MCNP Test Case Initial Particle Behavior

## B.1 Test Case 5




**(a)** $x$-$y$ Projection




**(b)** $x$-$z$ Projection

**Figure B.1:** MCNP Test Case 5 2D Projections

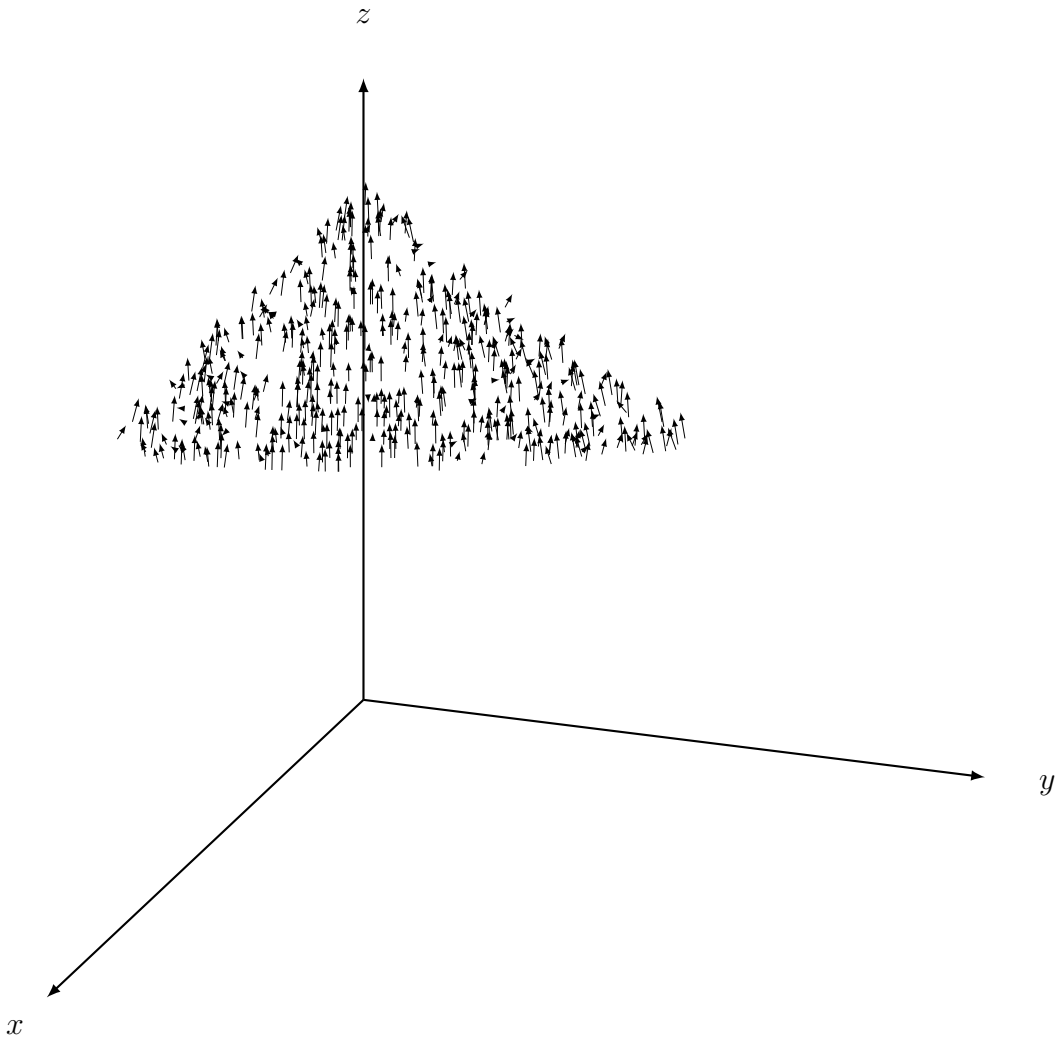
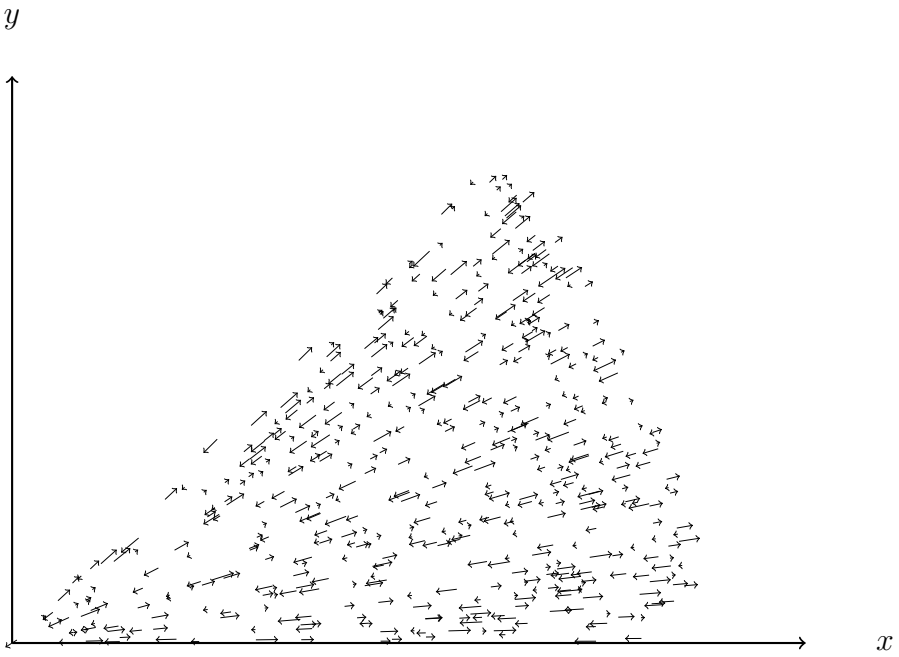
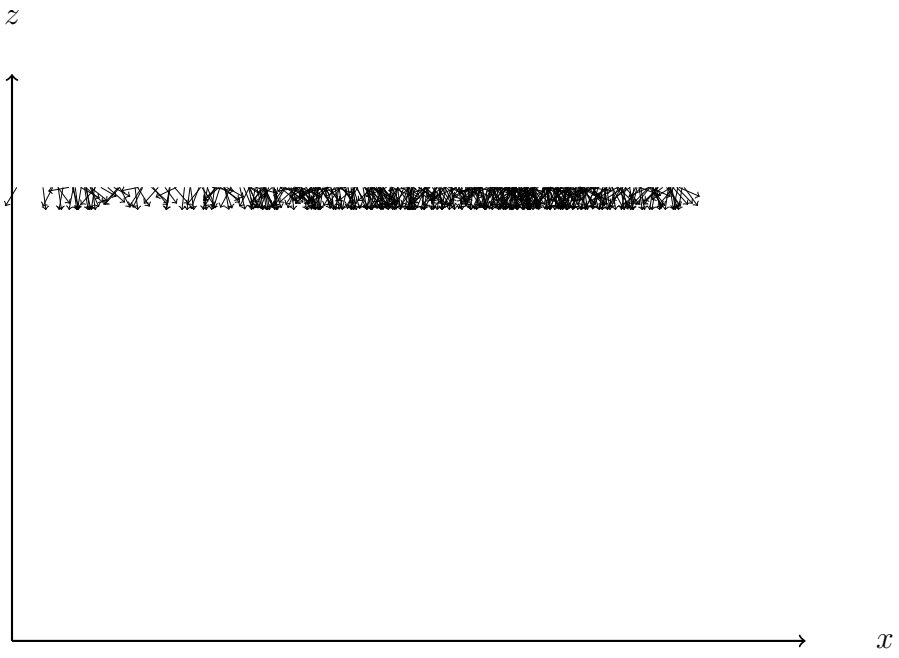



**Figure B.2:** MCNP Test Case 5 3D Projection

## B.2 Test Case 6




**(a)** $x$-$y$ Projection




**(b)** $x$-$z$ Projection

**Figure B.3:** MCNP Test Case 6 2D Projections

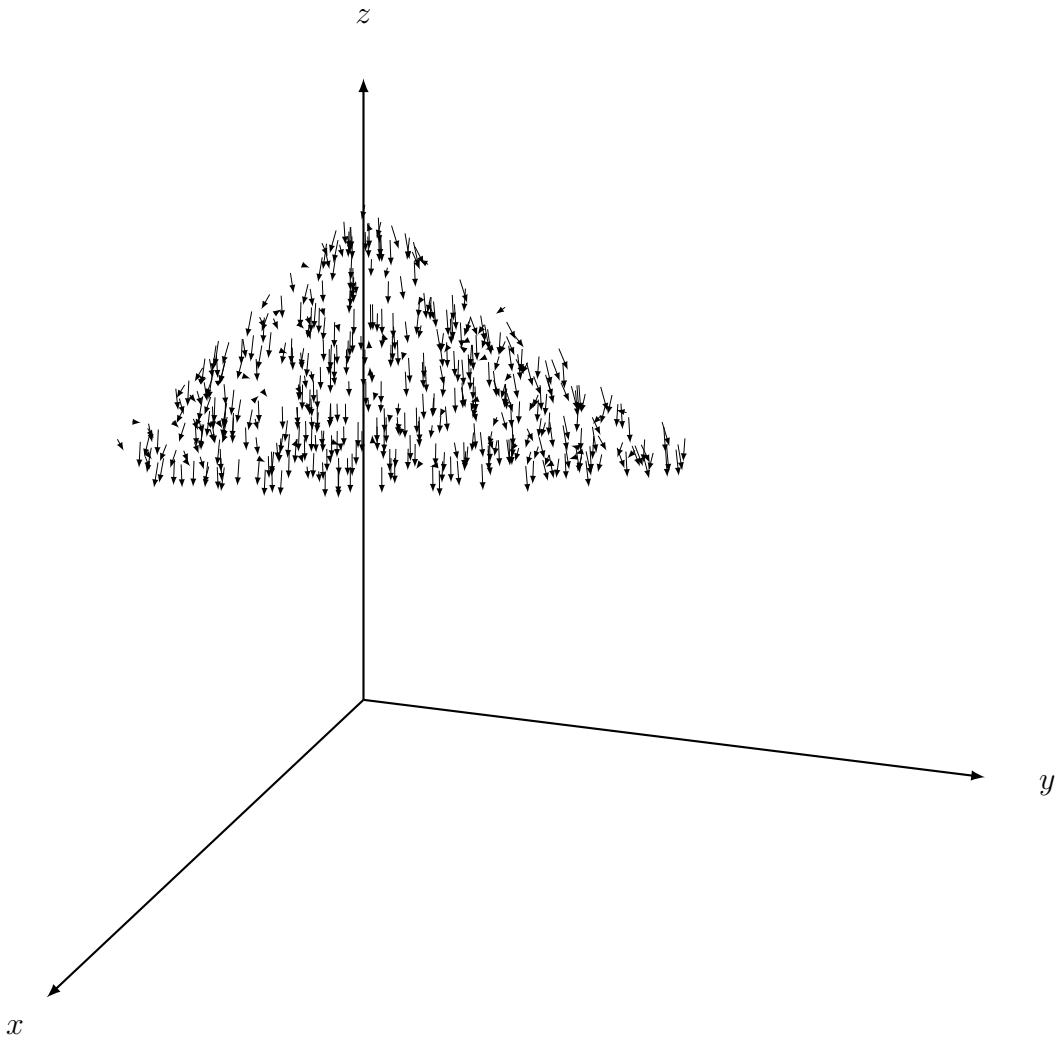
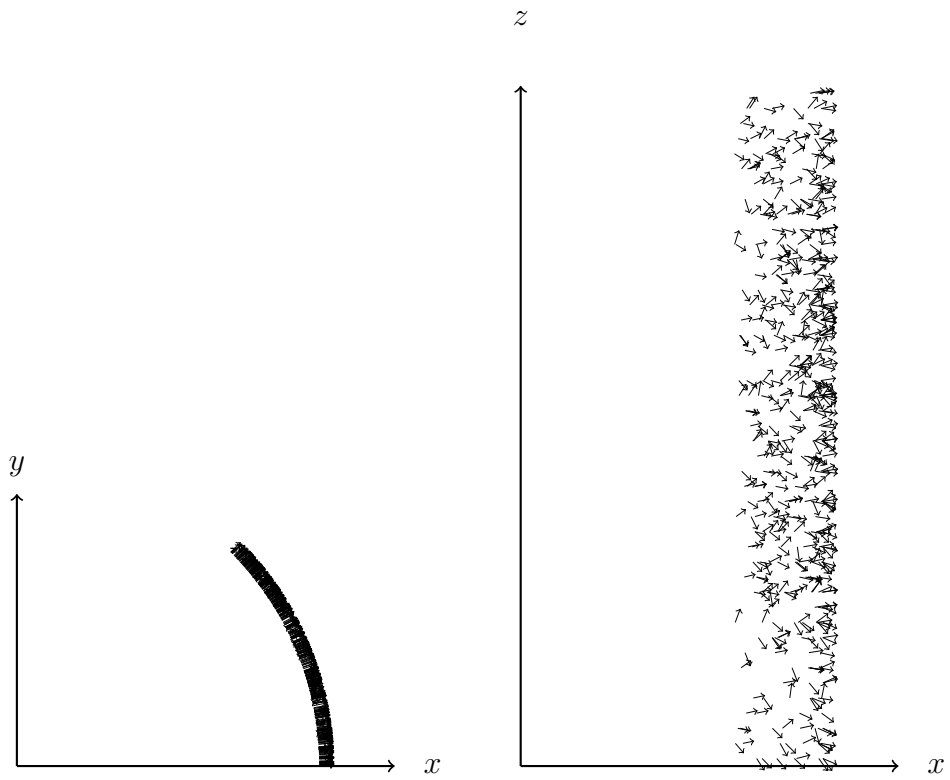



**Figure B.4:** MCNP Test Case 6 3D Projection

## B.3 Test Case 7




**(a)** $x$-$y$ Projection **(b)** $x$-$z$ Projection

**Figure B.5:** MCNP Test Case 7 2D Projections

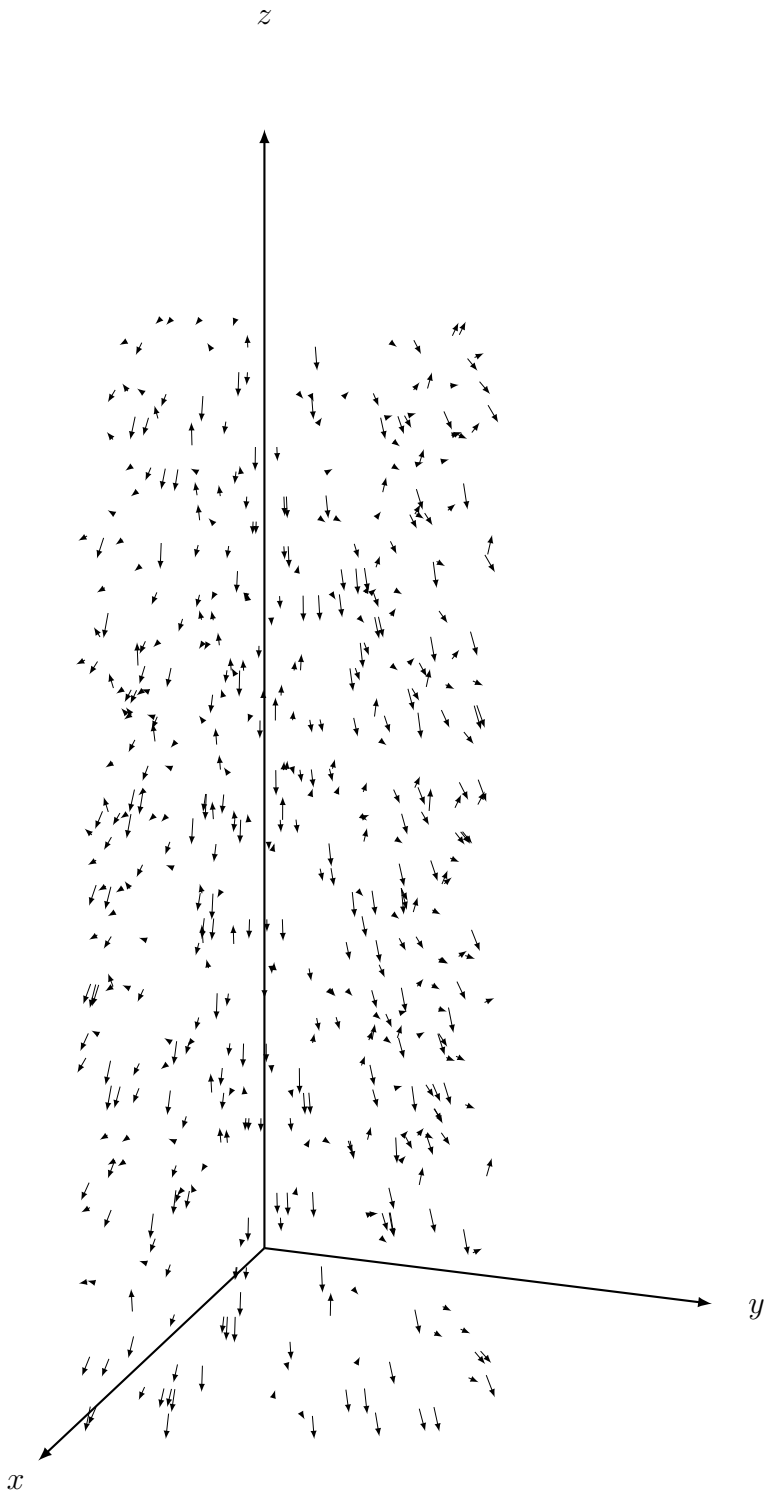
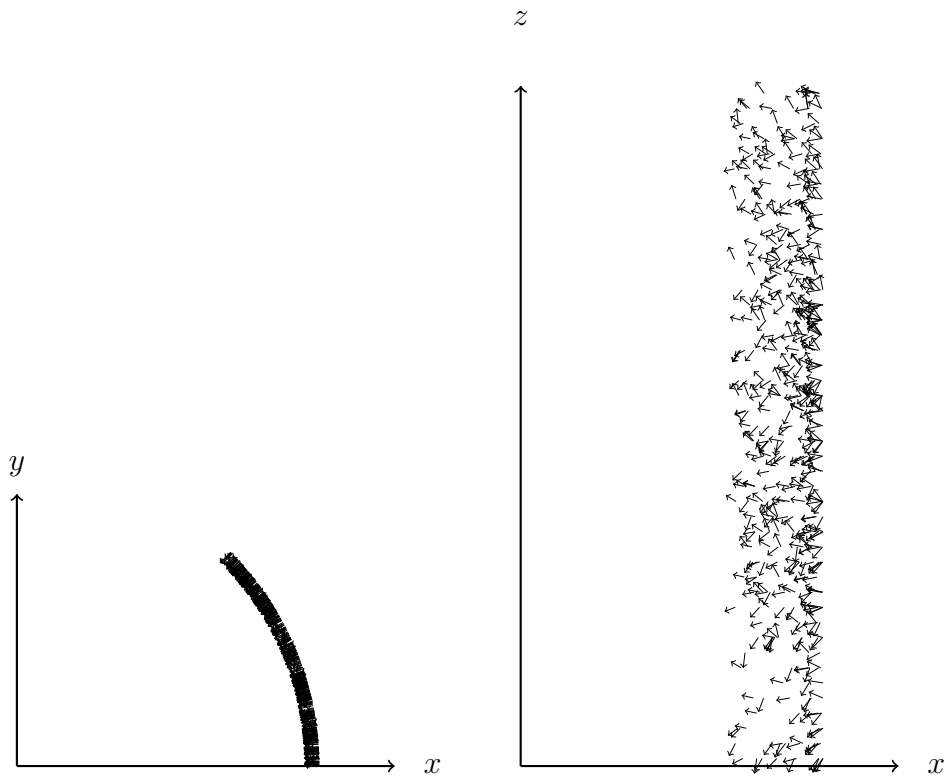



**Figure B.6:** MCNP Test Case 7 3D Projection

## B.4 Test Case 8




**(a)** $x$-$y$ Projection **(b)** $x$-$z$ Projection

**Figure B.7:** MCNP Test Case 8 2D Projections

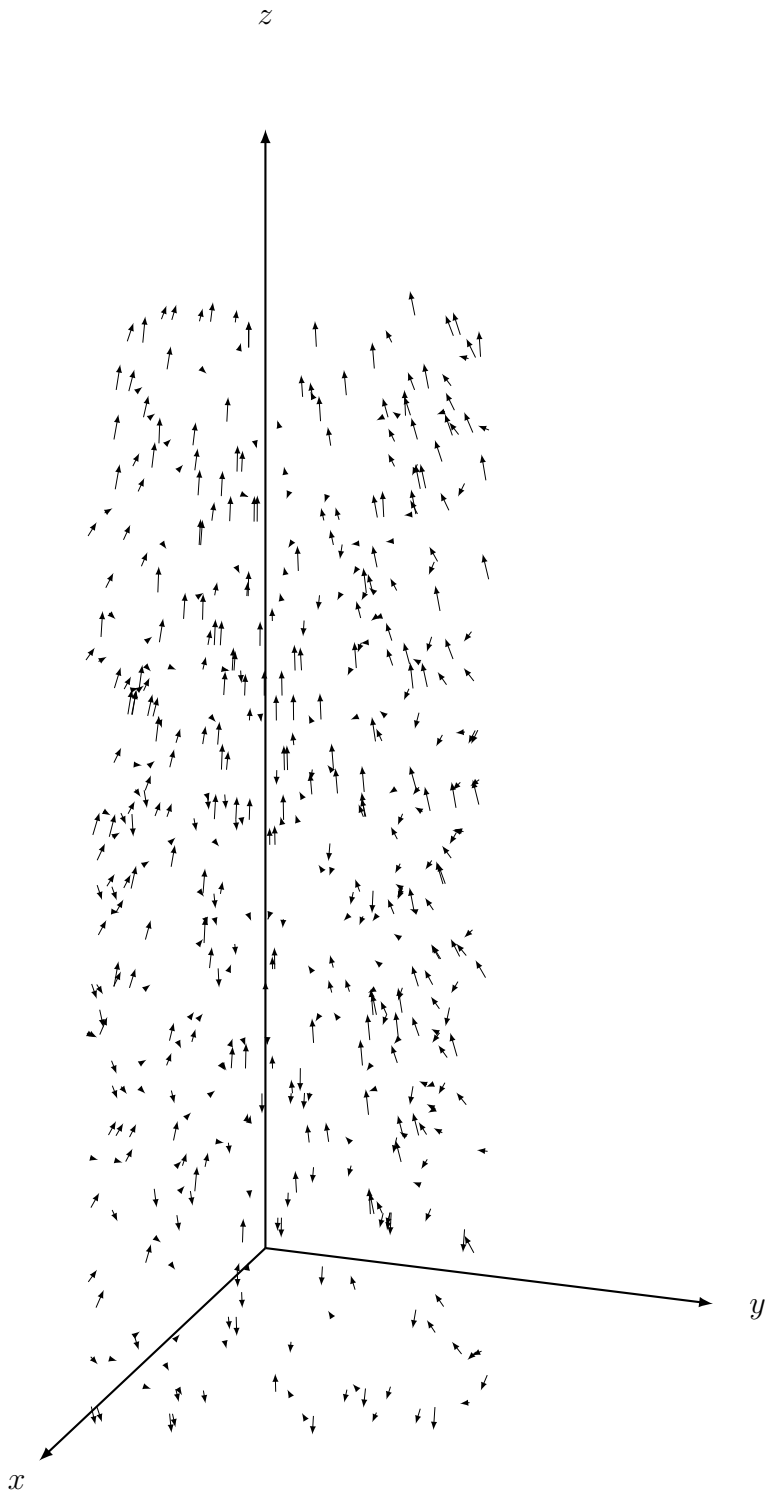
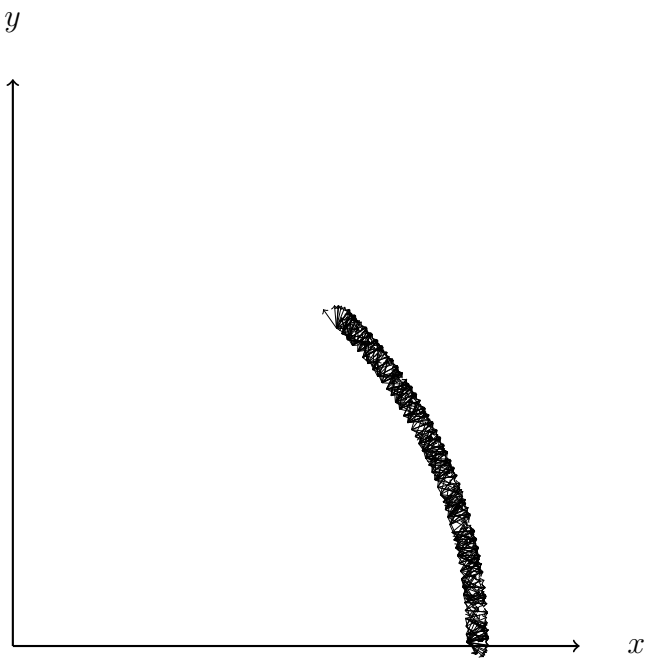
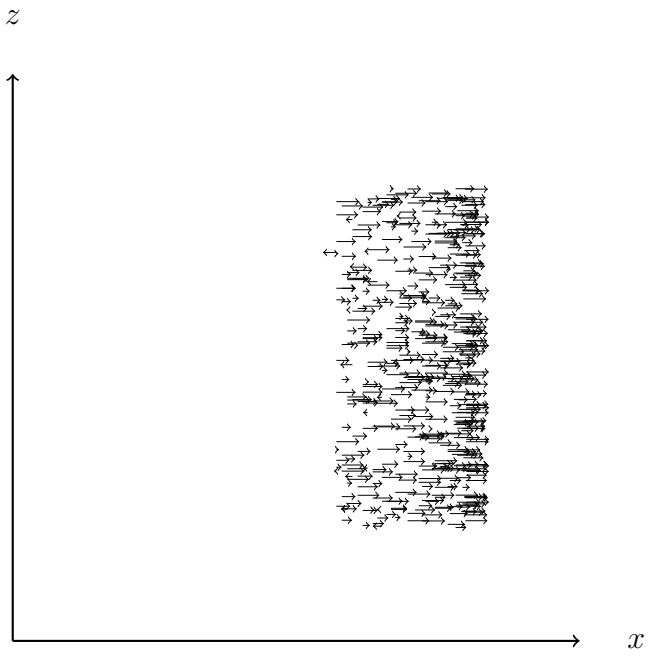



**Figure B.8:** MCNP Test Case 8 3D Projection

## B.5 Test Case 9




**(a)** $x$-$y$ Projection




**(b)** $x$-$z$ Projection

**Figure B.9:** MCNP Test Case 9 2D Projections

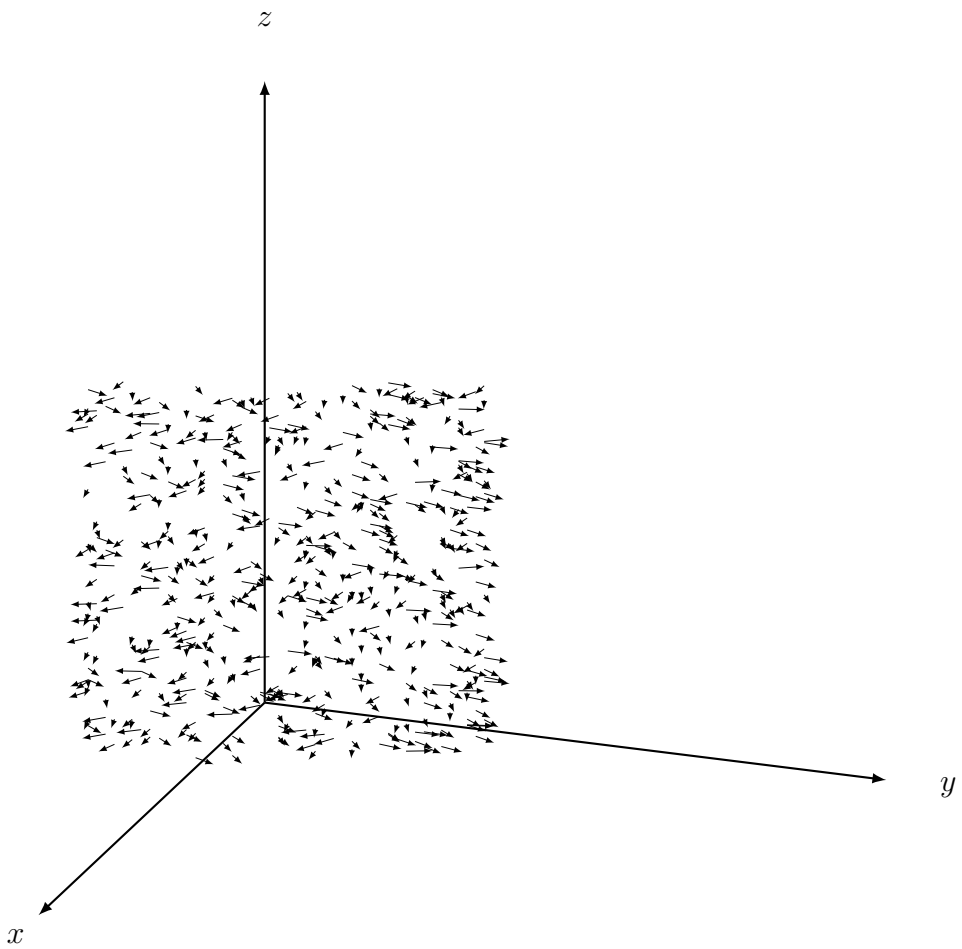
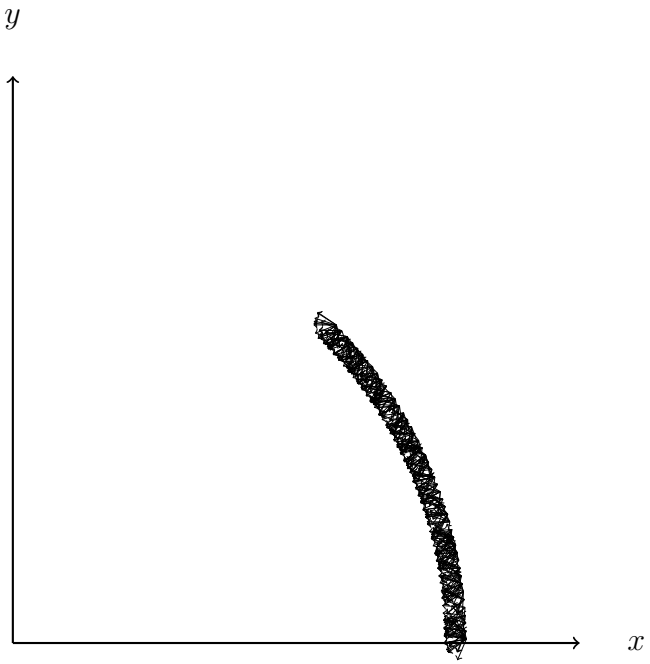
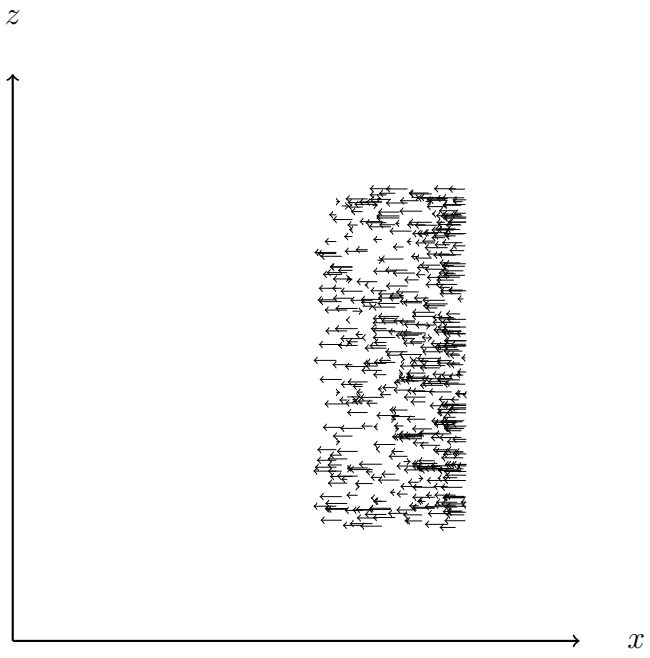



**Figure B.10:** MCNP Test Case 9 3D Projection

## B.6 Test Case 10




**(a)** $x$-$y$ Projection




**(b)** $x$-$z$ Projection

**Figure B.11:** MCNP Test Case 10 2D Projections

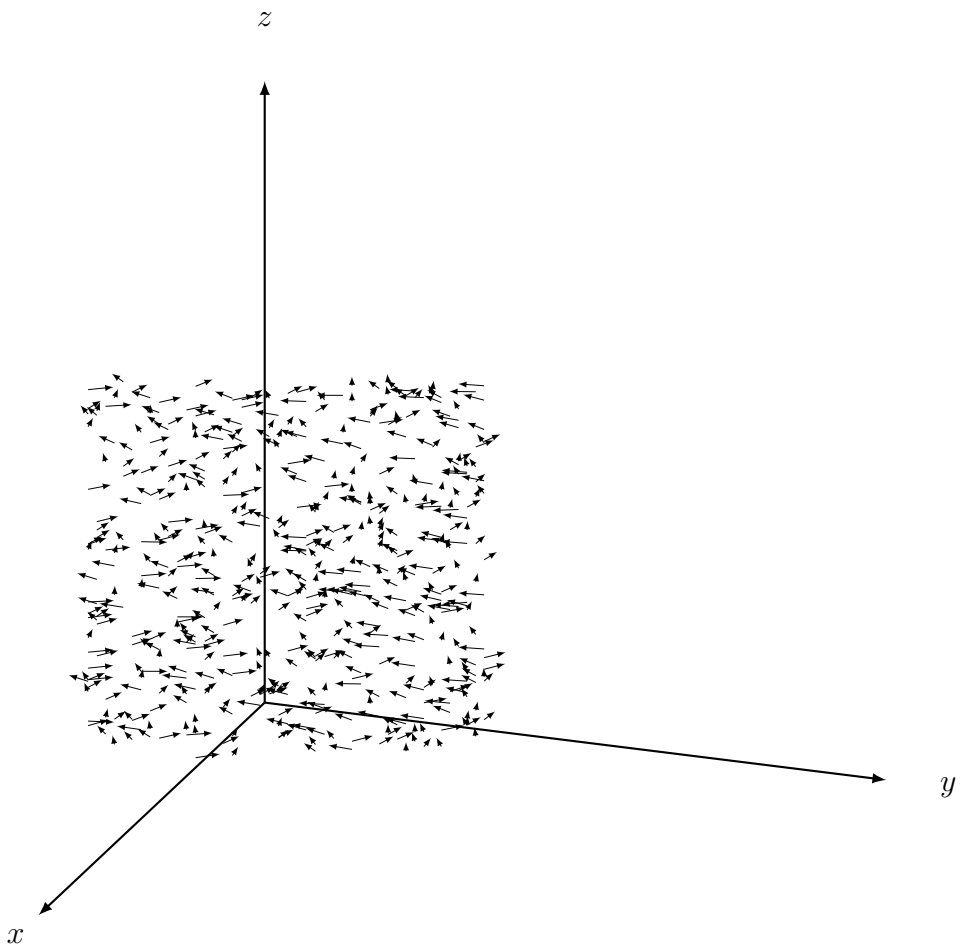
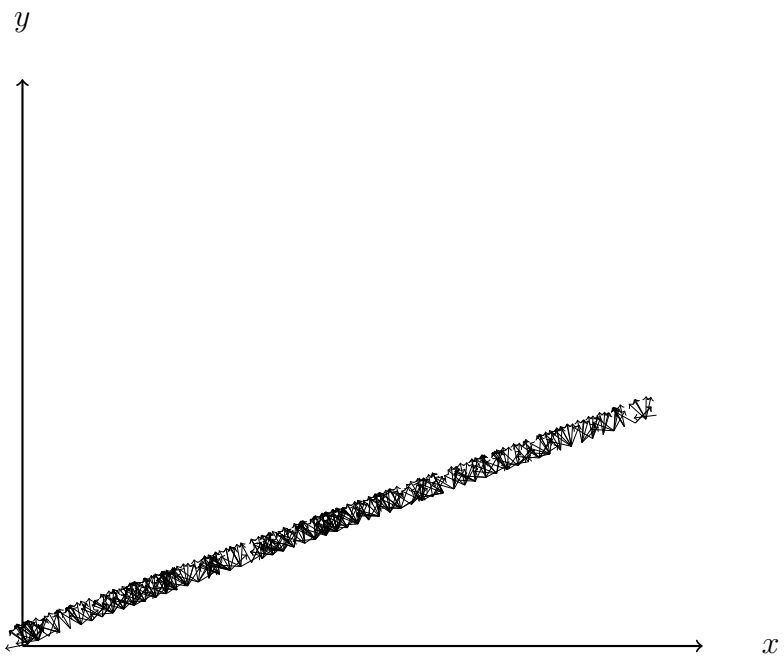
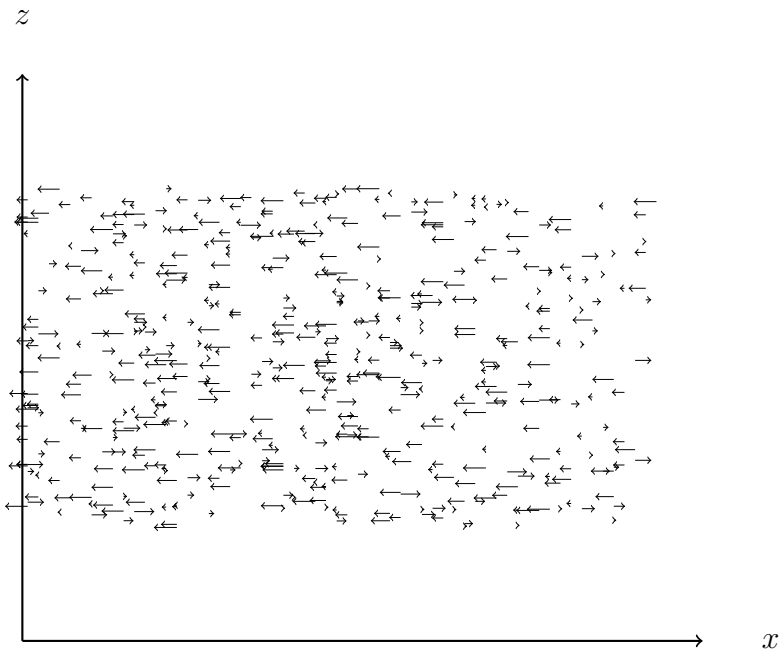



**Figure B.12:** MCNP Test Case 10 3D Projection

## B.7 Test Case 11




**(a)** $x$-$y$ Projection




**(b)** $x$-$z$ Projection

**Figure B.13:** MCNP Test Case 11 2D Projections

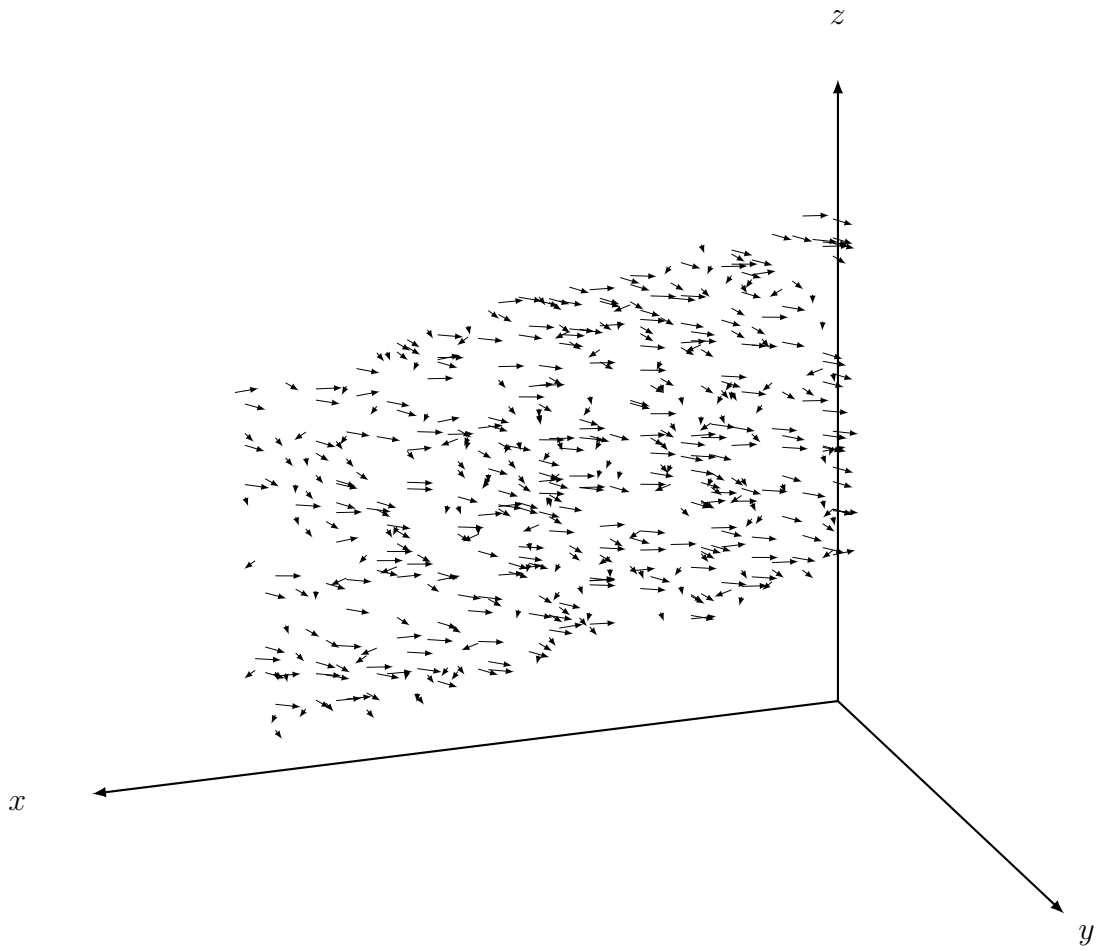
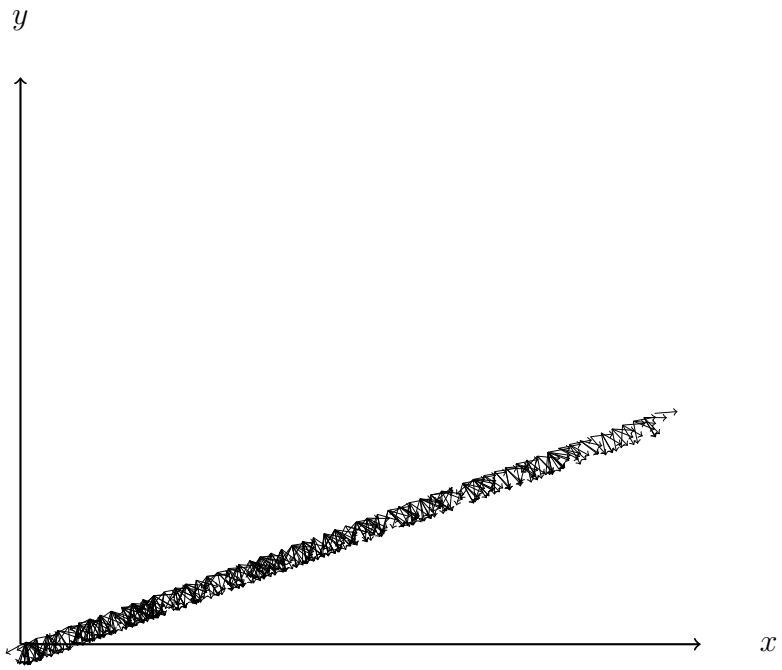
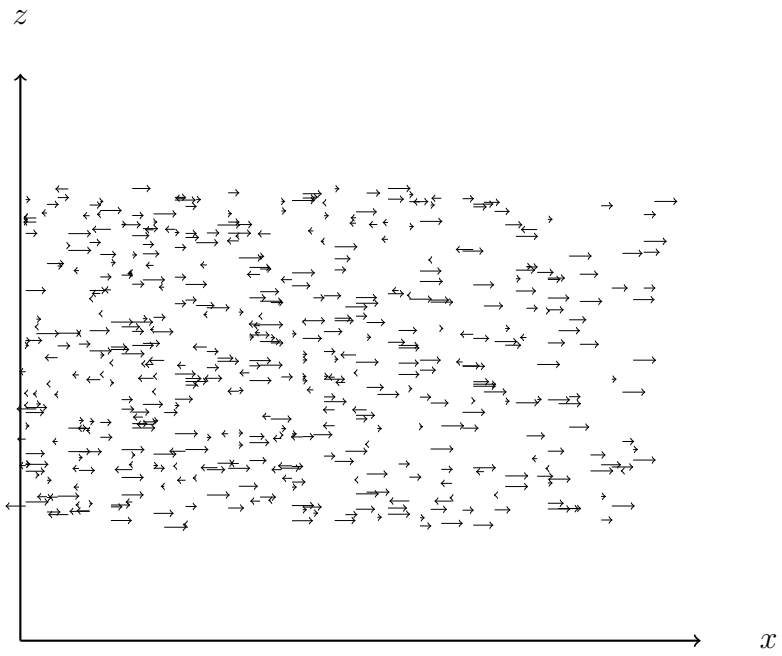



**Figure B.14:** MCNP Test Case 11 3D Projection

## B.8 Test Case 12




**(a)** $x$-$y$ Projection




**(b)** $x$-$z$ Projection

**Figure B.15:** MCNP Test Case 12 2D Projections

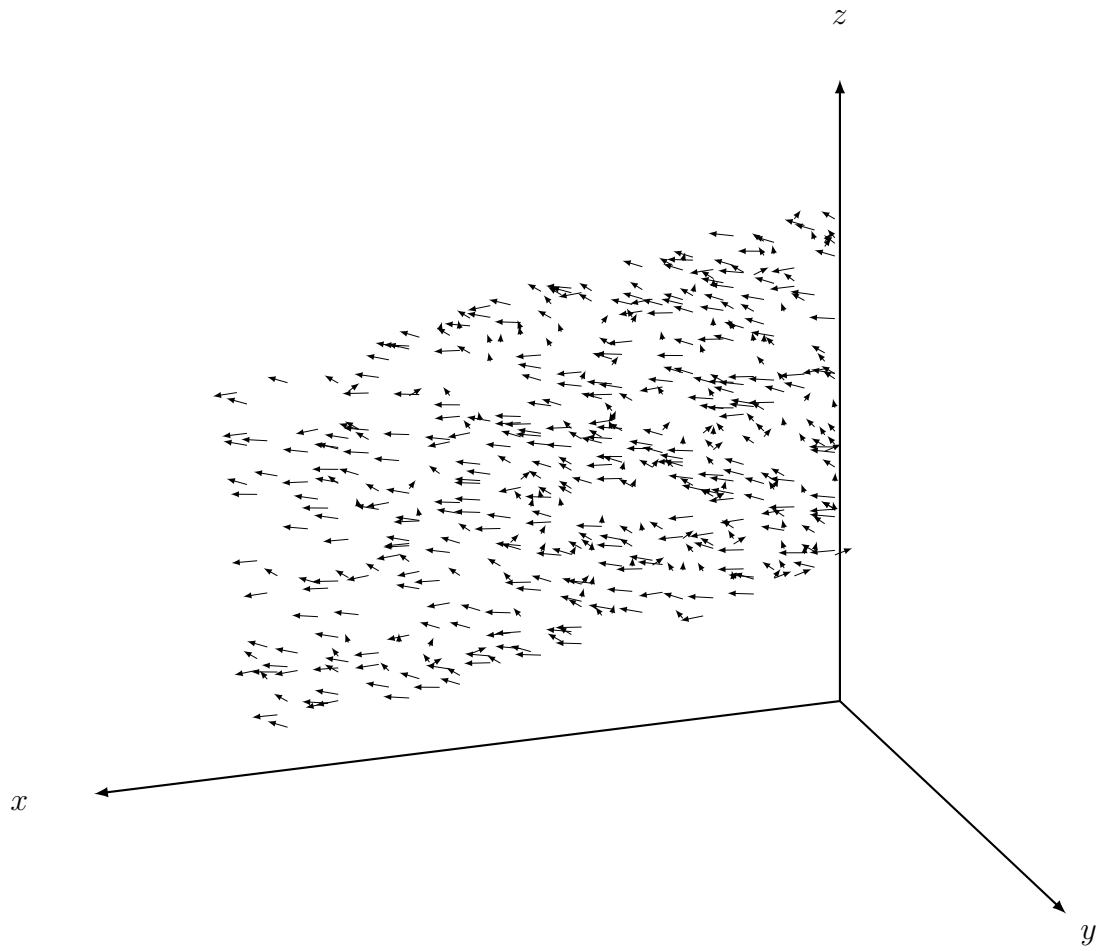
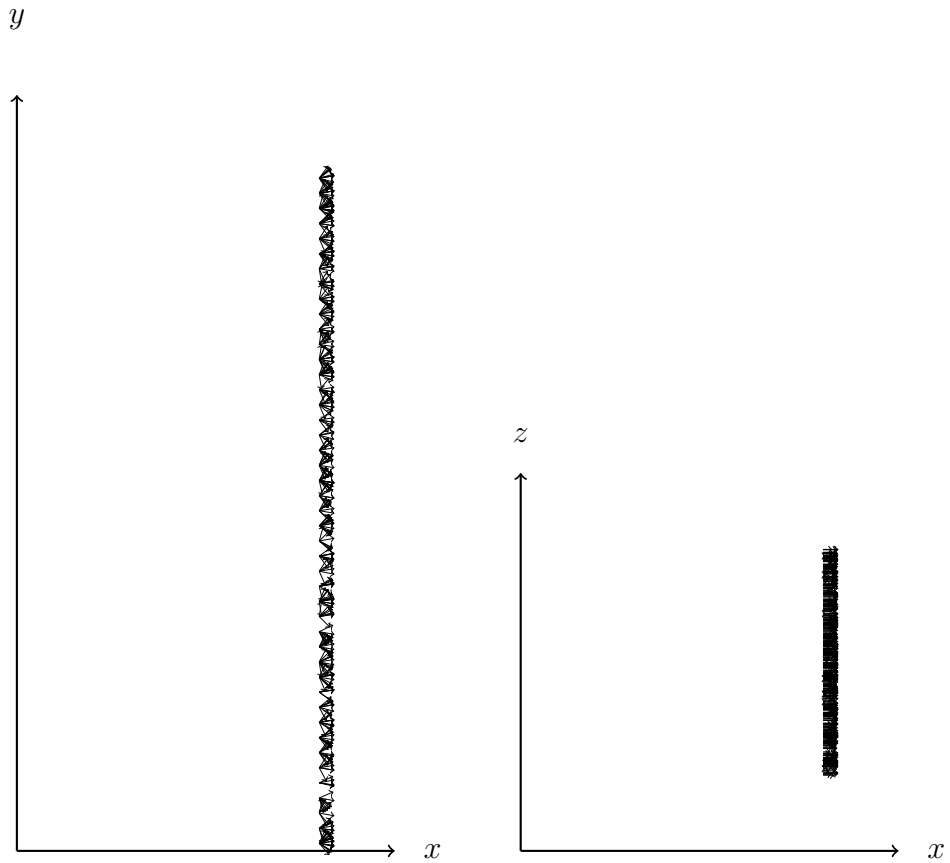



**Figure B.16:** MCNP Test Case 12 3D Projection

## B.9 Test Case 13




**(a)** $x$-$y$ Projection **(b)** $x$-$z$ Projection

**Figure B.17:** MCNP Test Case 13 2D Projections

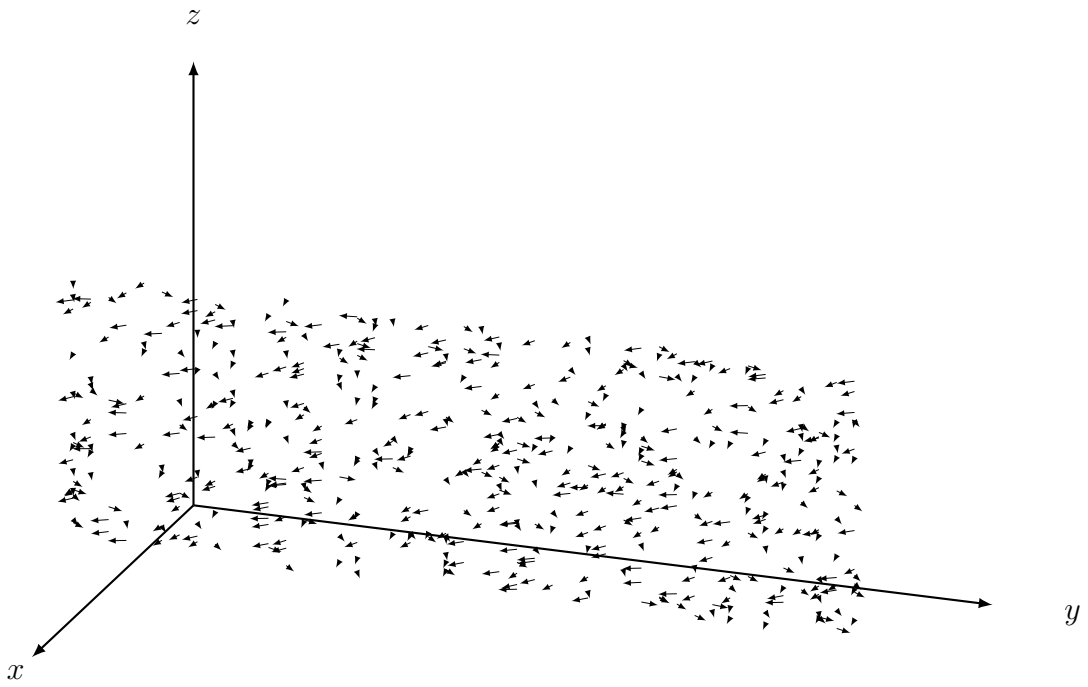
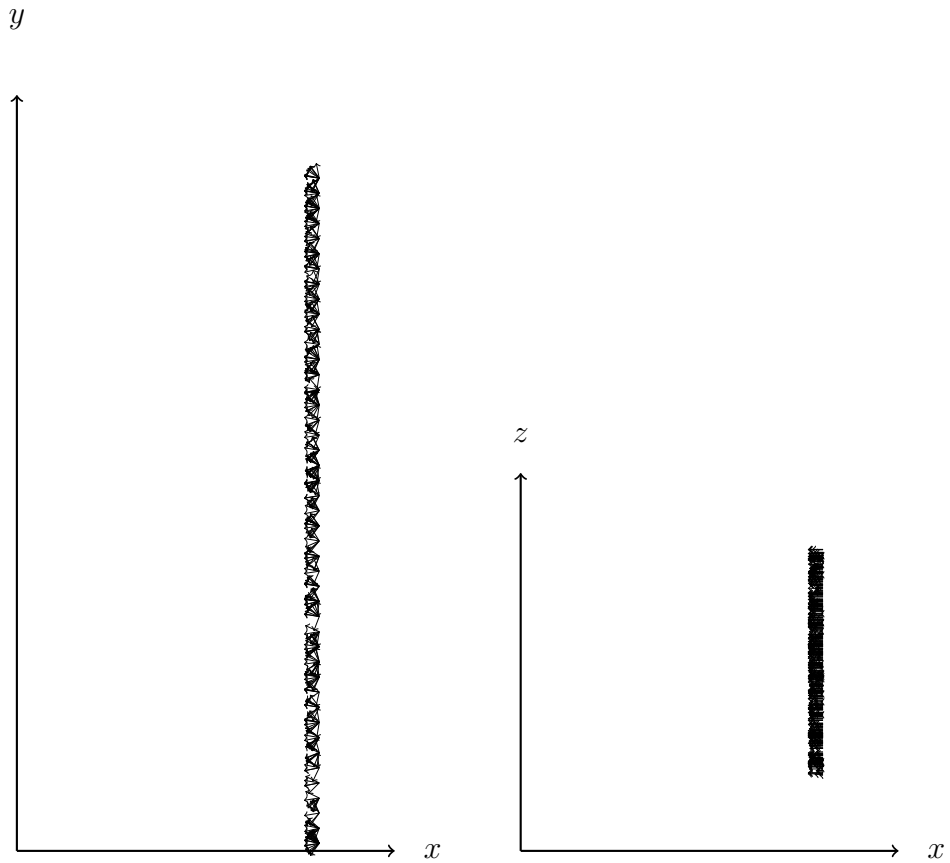



**Figure B.18:** MCNP Test Case 13 3D Projection

## B.10 Test Case 14




**(a)** $x$-$y$ Projection **(b)** $x$-$z$ Projection

**Figure B.19:** MCNP Test Case 14 2D Projections

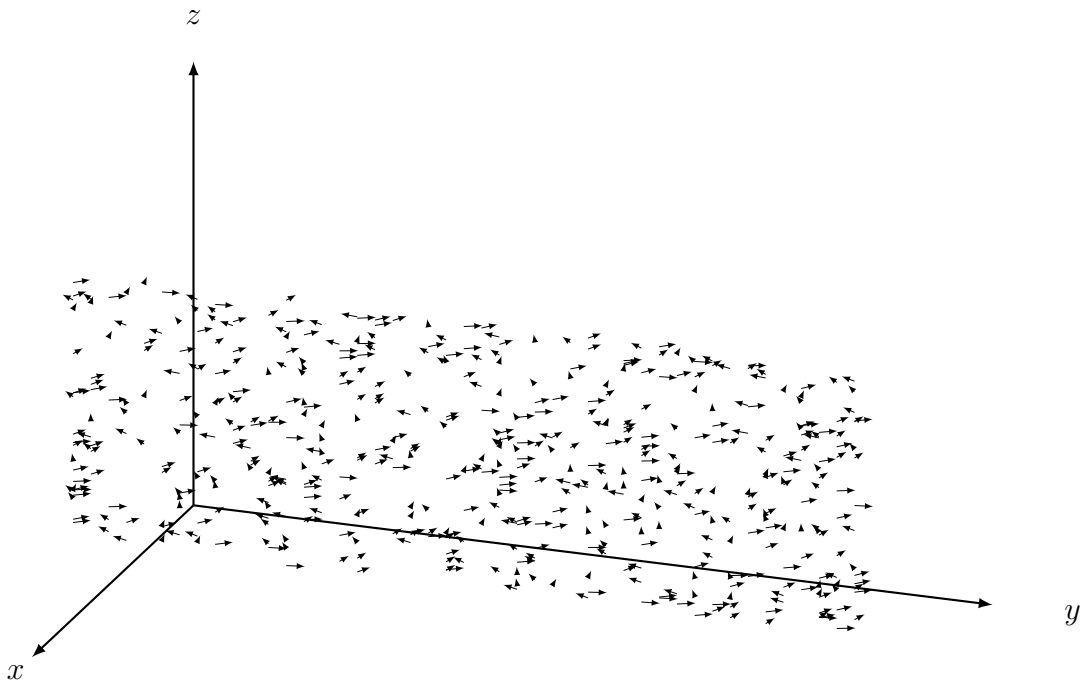
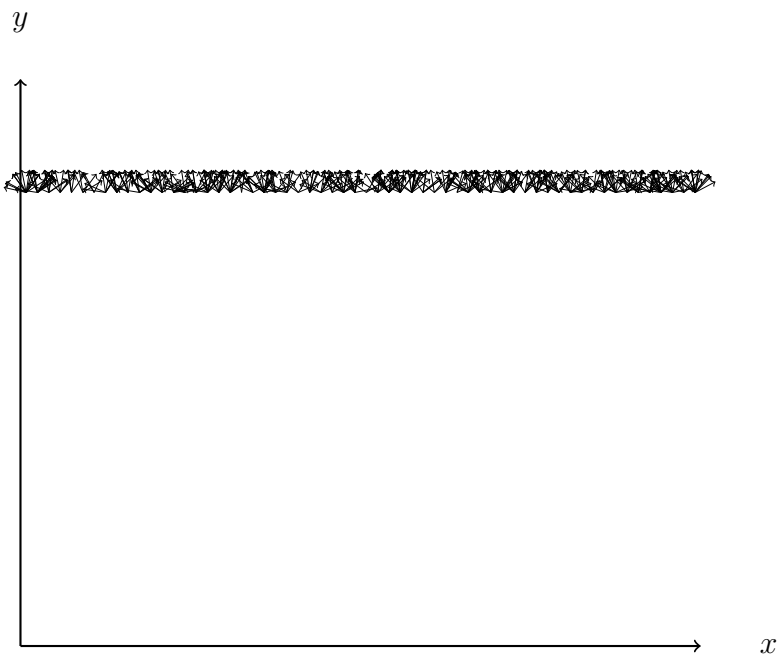
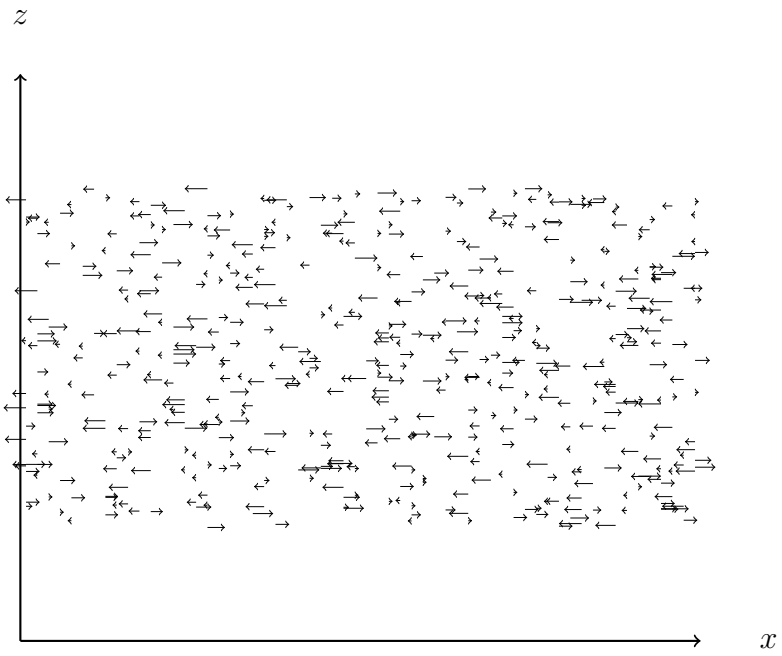



**Figure B.20:** MCNP Test Case 14 3D Projection

## B.11 Test Case 15




**(a)** $x$-$y$ Projection




**(b)** $x$-$z$ Projection

**Figure B.21:** MCNP Test Case 15 2D Projections

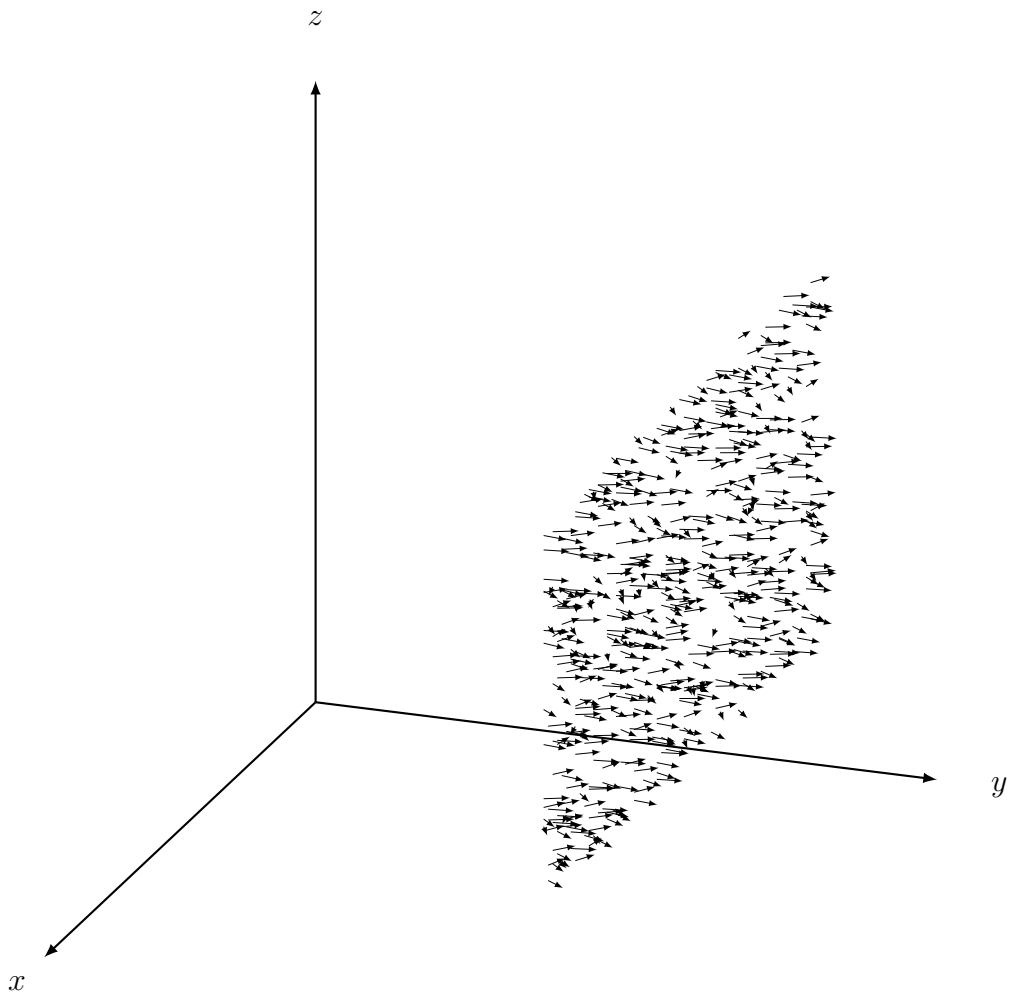
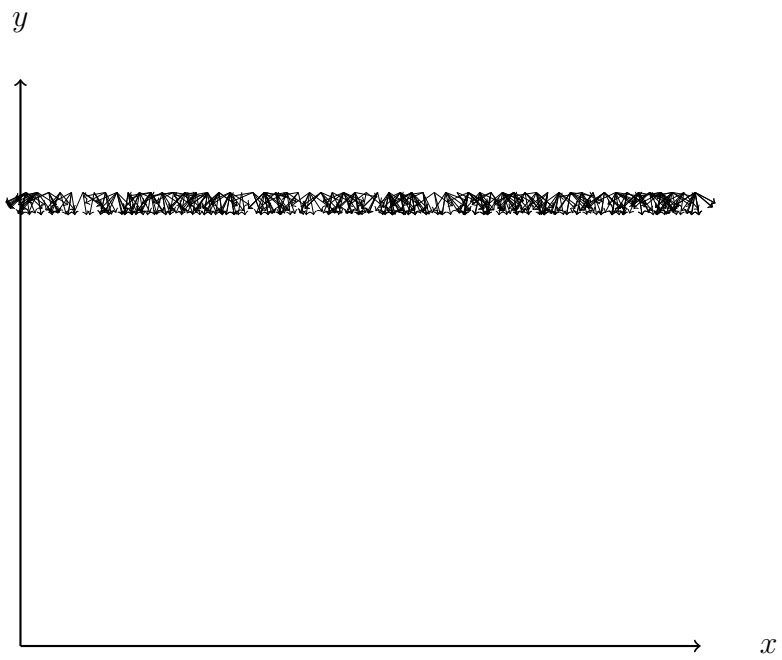
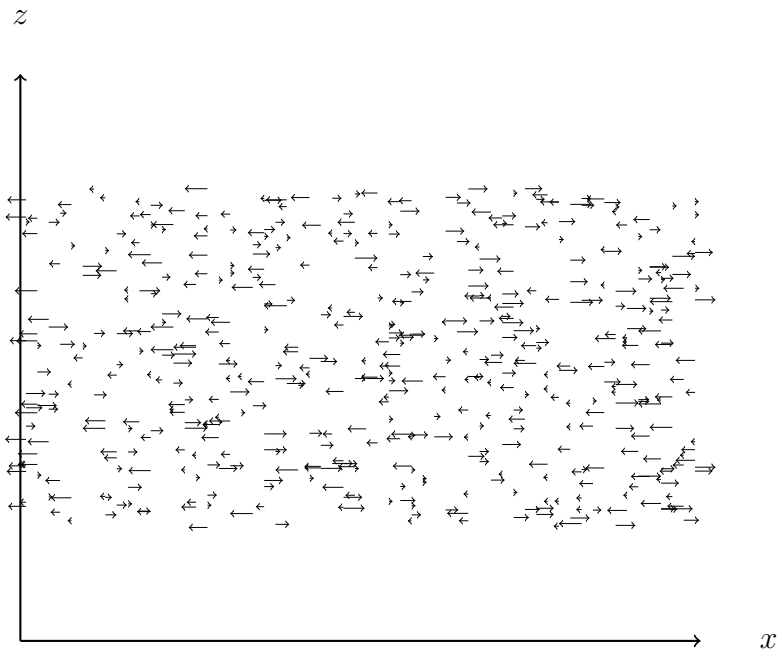



**Figure B.22:** MCNP Test Case 15 3D Projection

## B.12 Test Case 16




**(a)** $x$-$y$ Projection




**(b)** $x$-$z$ Projection

**Figure B.23:** MCNP Test Case 16 2D Projections

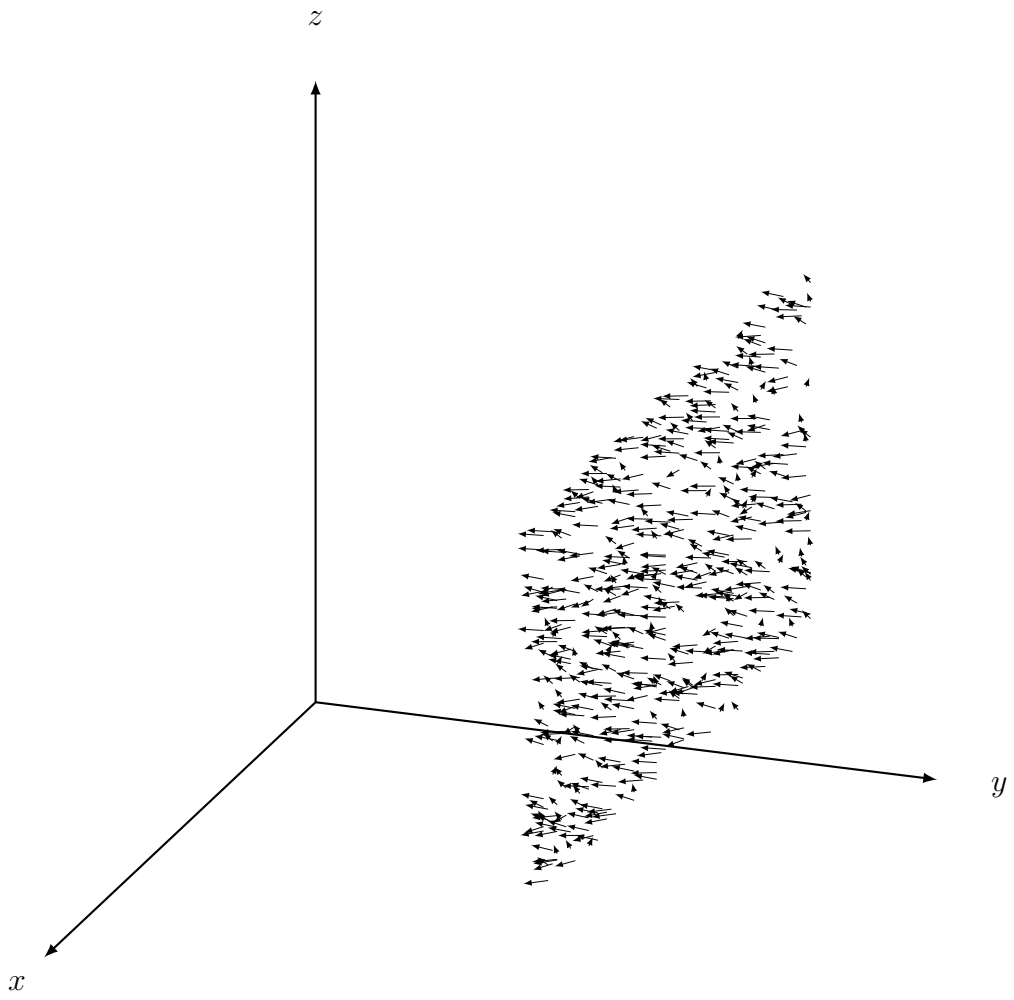
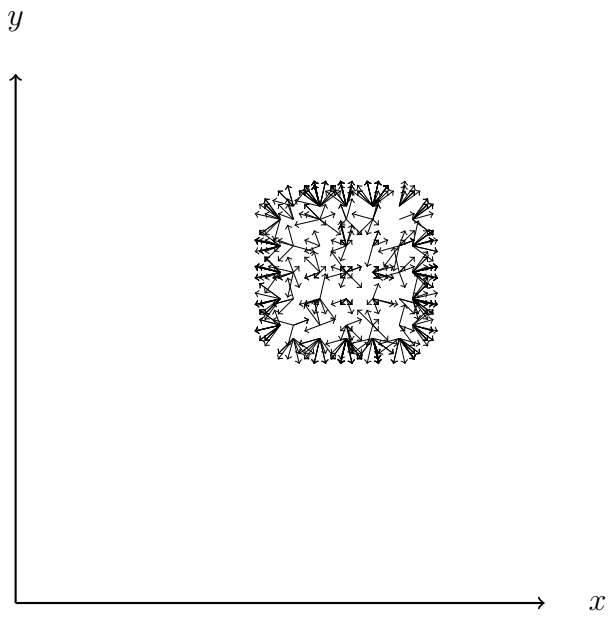
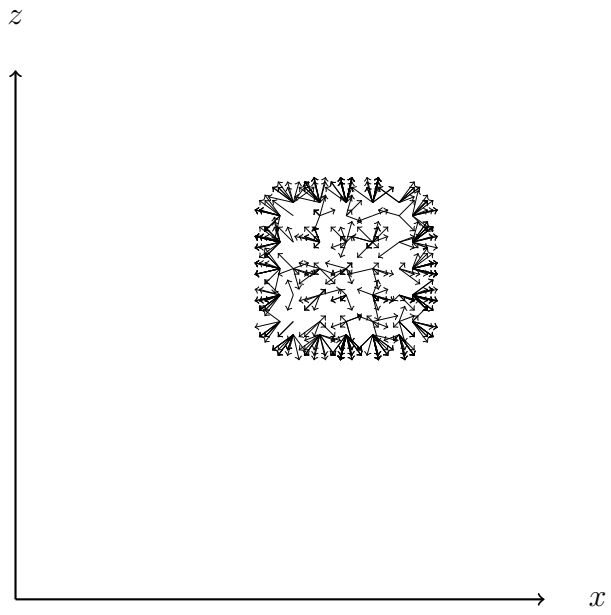



**Figure B.24:** MCNP Test Case 16 3D Projection

## B.13 Test Case 17




**(a)** $x$-$y$ Projection




**(b)** $x$-$z$ Projection

**Figure B.25:** MCNP Test Case 17 2D Projections

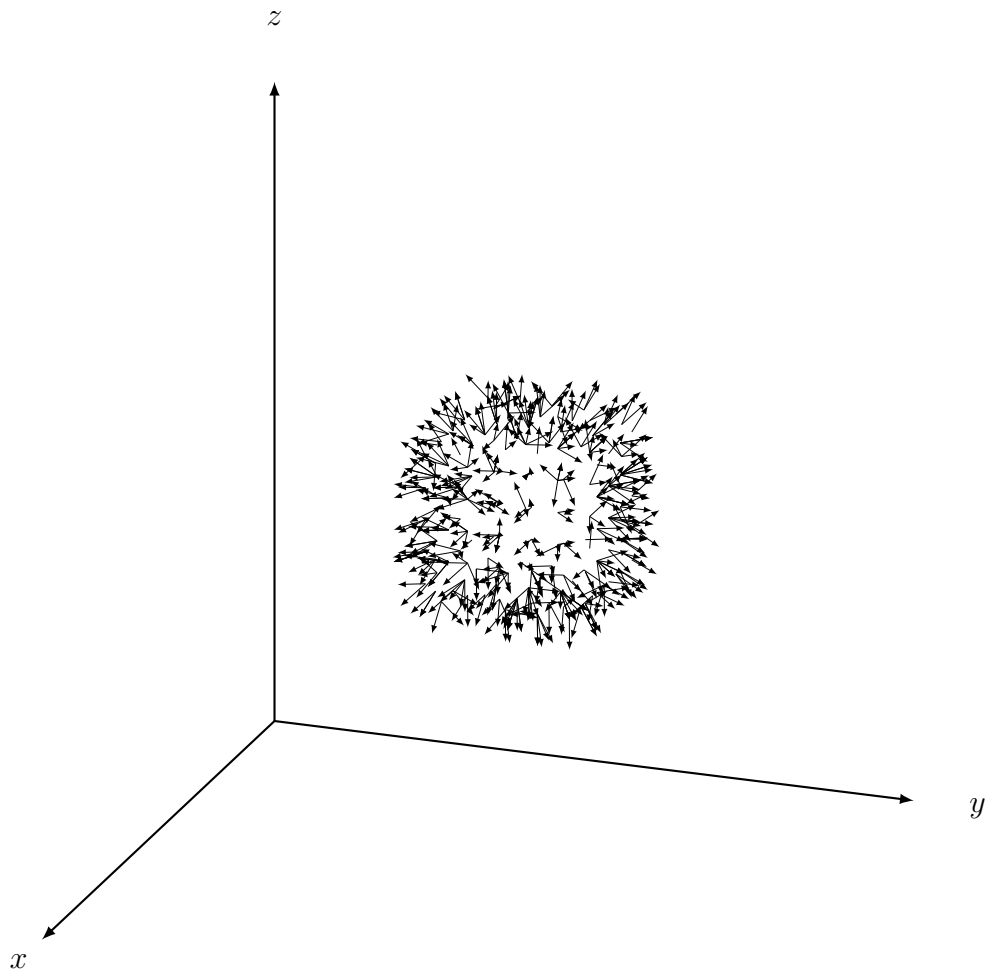
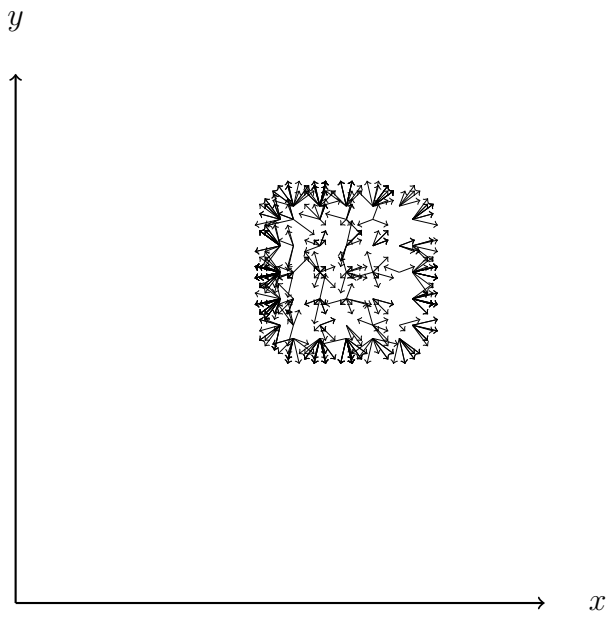
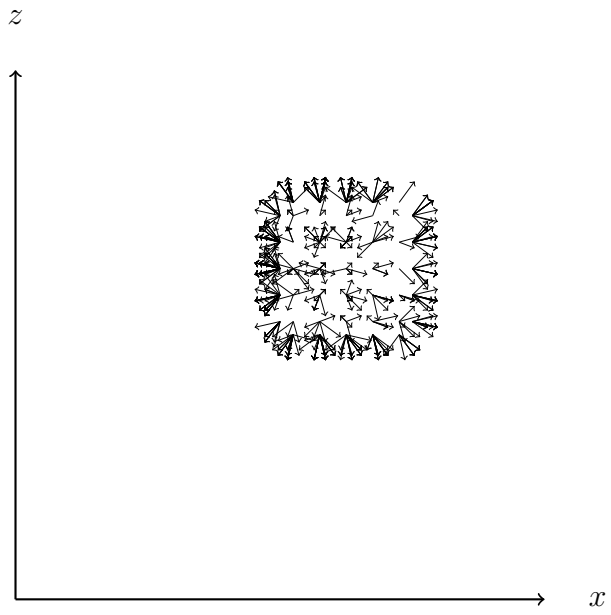



**Figure B.26:** MCNP Test Case 17 3D Projection

## B.14 Test Case 18




**(a)** $x$-$y$ Projection




**(b)** $x$-$z$ Projection

**Figure B.27:** MCNP Test Case 18 2D Projections

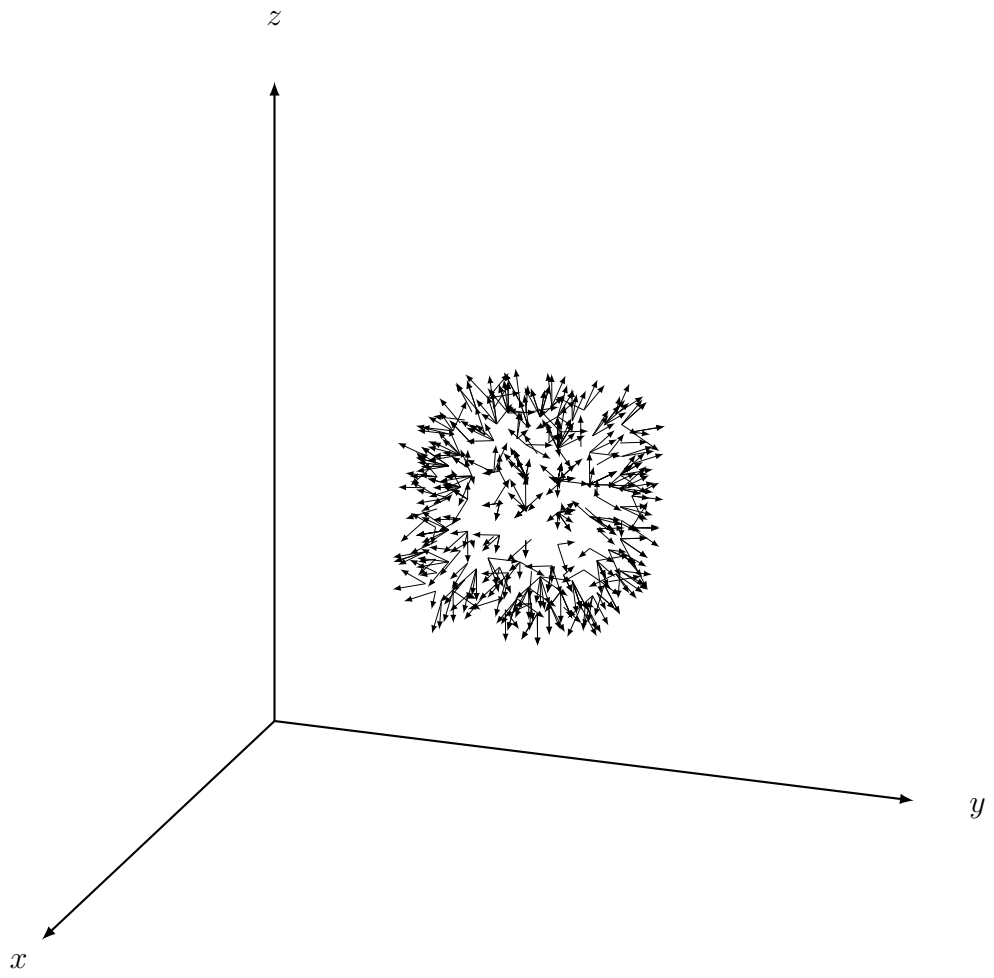
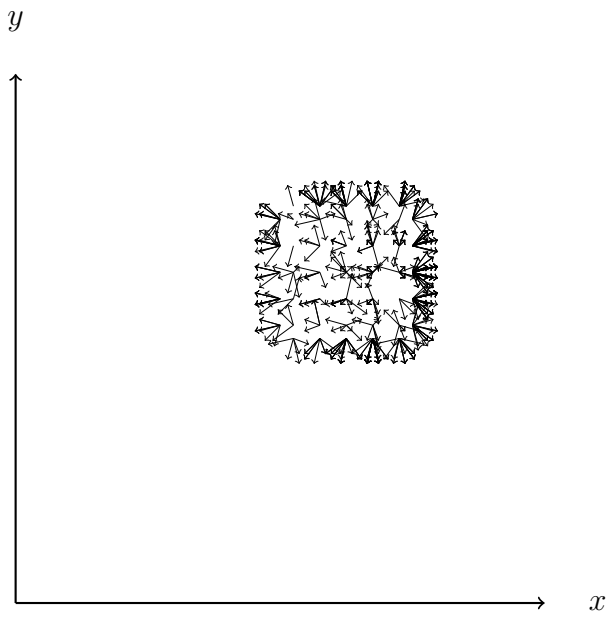
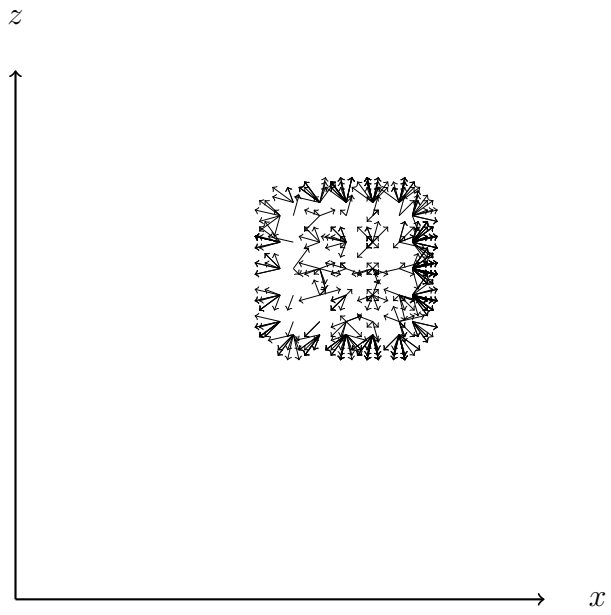



**Figure B.28:** MCNP Test Case 18 3D Projection

## B.15 Test Case 19




**(a)** $x$-$y$ Projection




**(b)** $x$-$z$ Projection

**Figure B.29:** MCNP Test Case 19 2D Projections

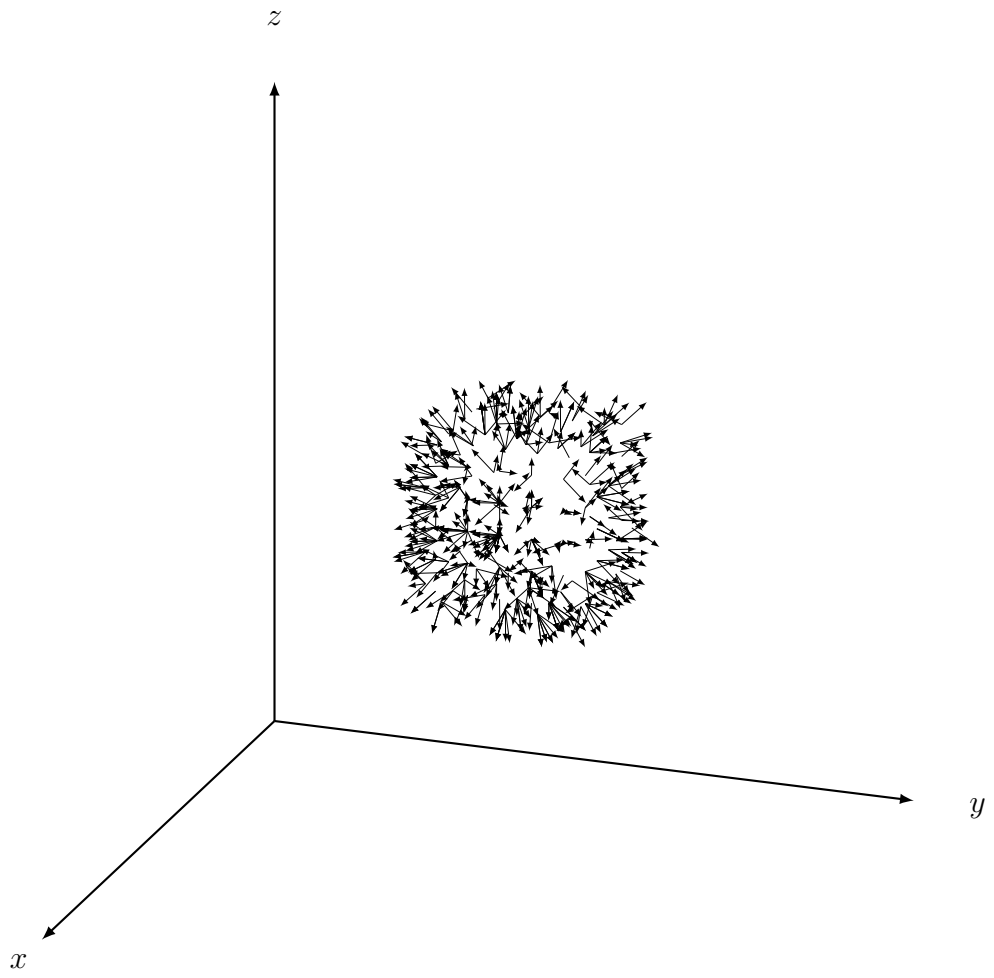
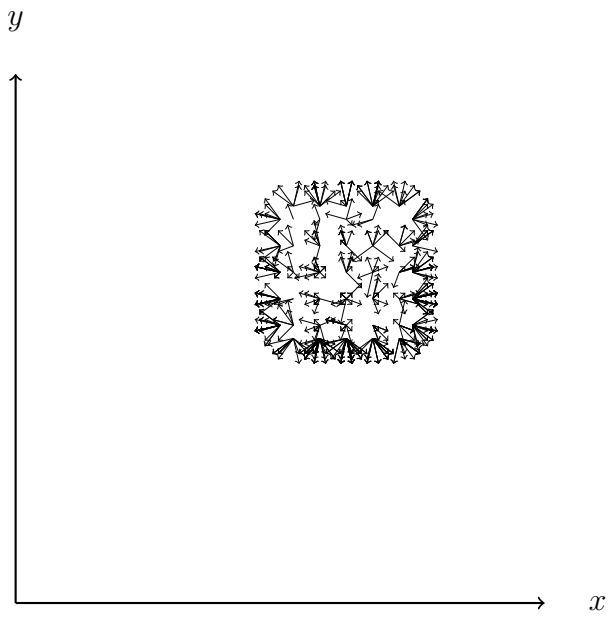
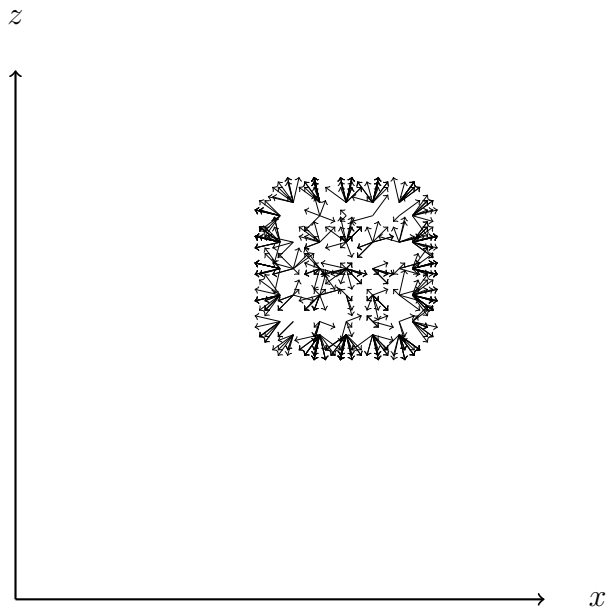



**Figure B.30:** MCNP Test Case 19 3D Projection

## B.16 Test Case 20




**(a)** $x$-$y$ Projection




**(b)** $x$-$z$ Projection

**Figure B.31:** MCNP Test Case 20 2D Projections

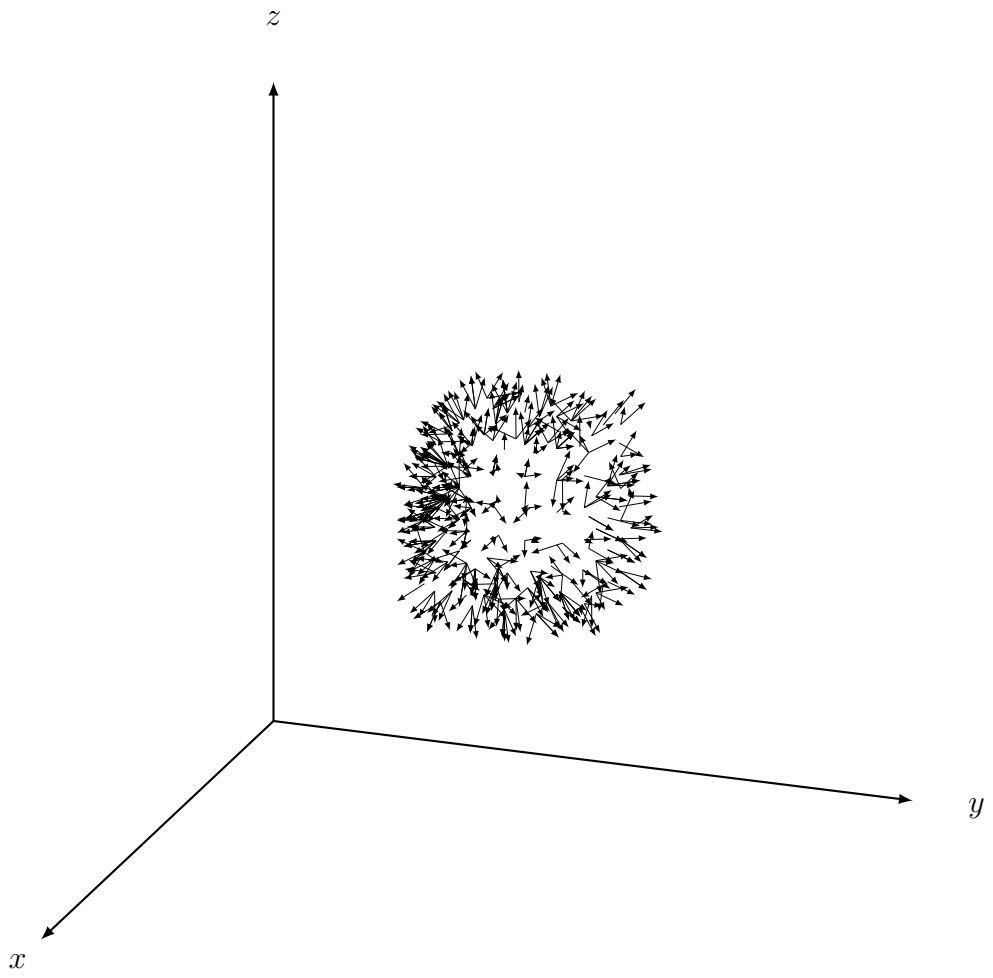
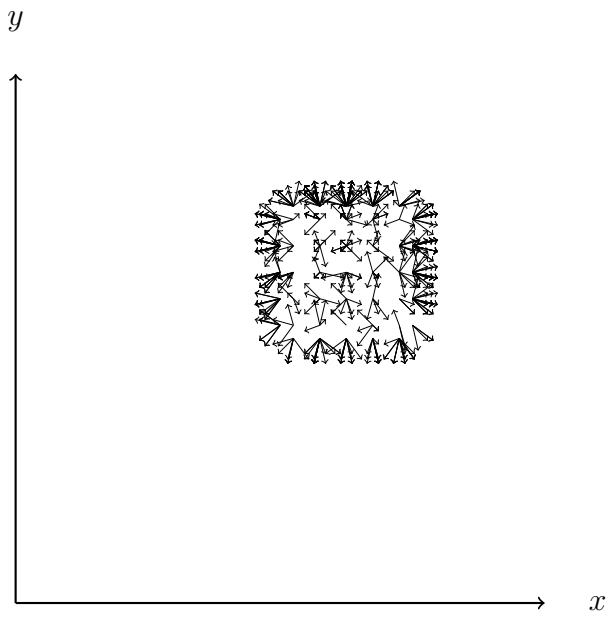
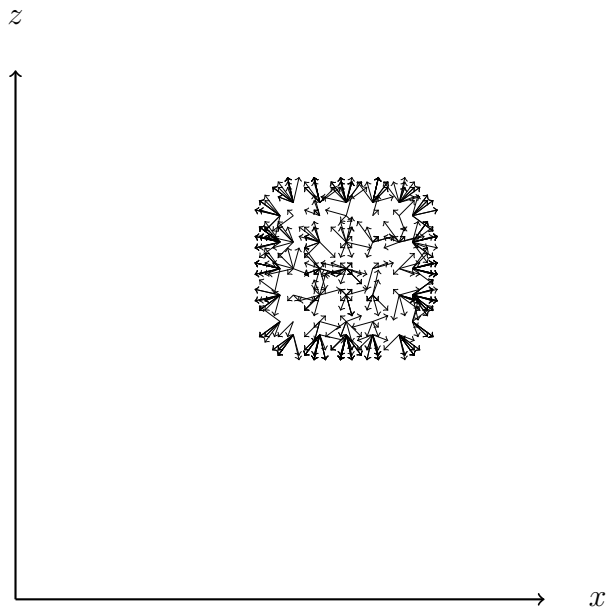



**Figure B.32:** MCNP Test Case 20 3D Projection

## B.17 Test Case 21




**(a)** $x$-$y$ Projection




**(b)** $x$-$z$ Projection

**Figure B.33:** MCNP Test Case 21 2D Projections

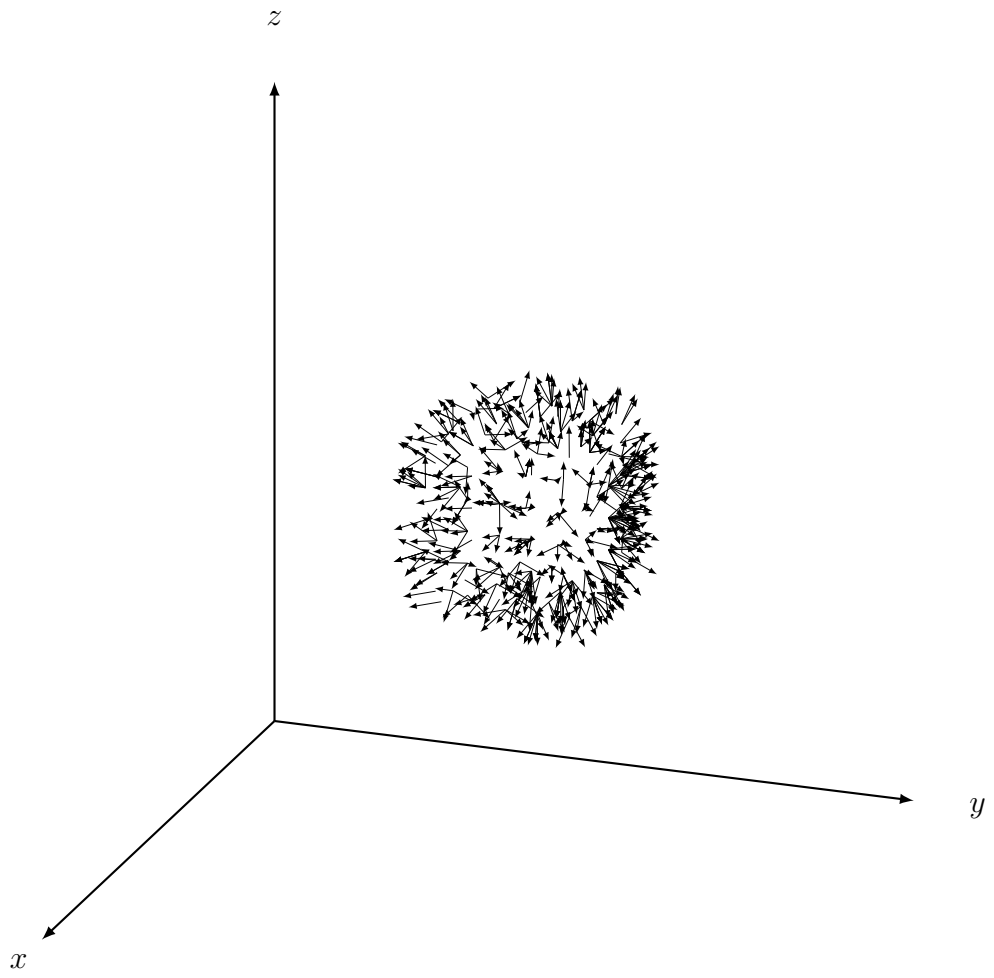
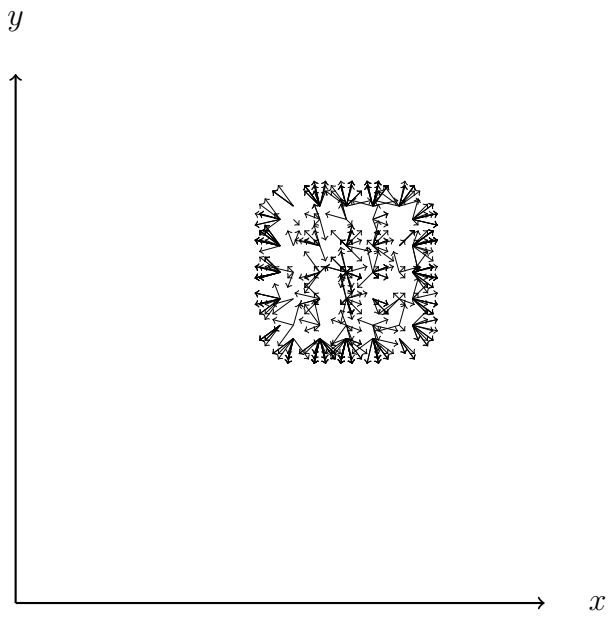
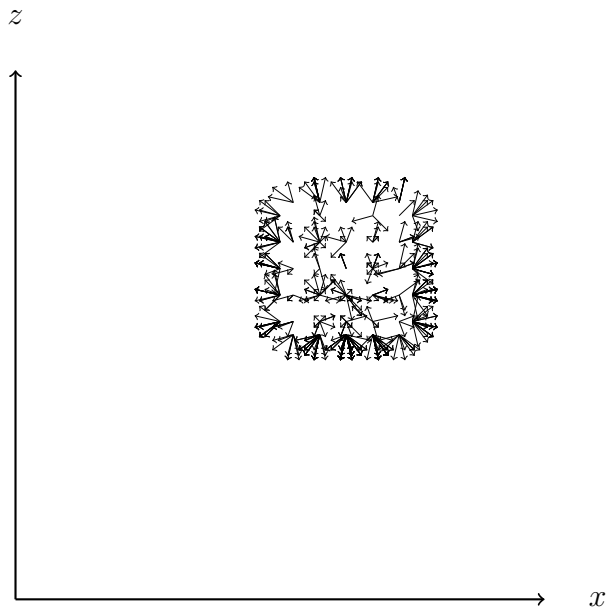



**Figure B.34:** MCNP Test Case 21 3D Projection

## B.18 Test Case 22




**(a)** $x$-$y$ Projection




**(b)** $x$-$z$ Projection

**Figure B.35:** MCNP Test Case 22 2D Projections

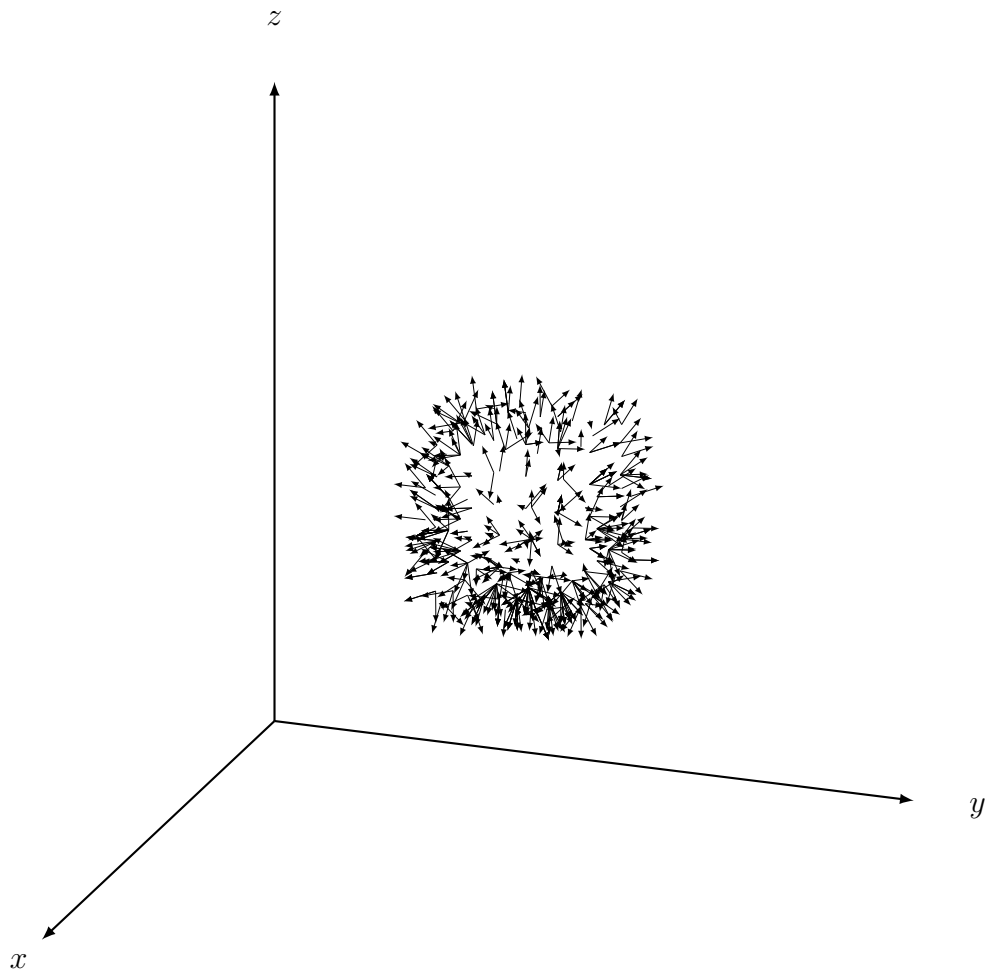
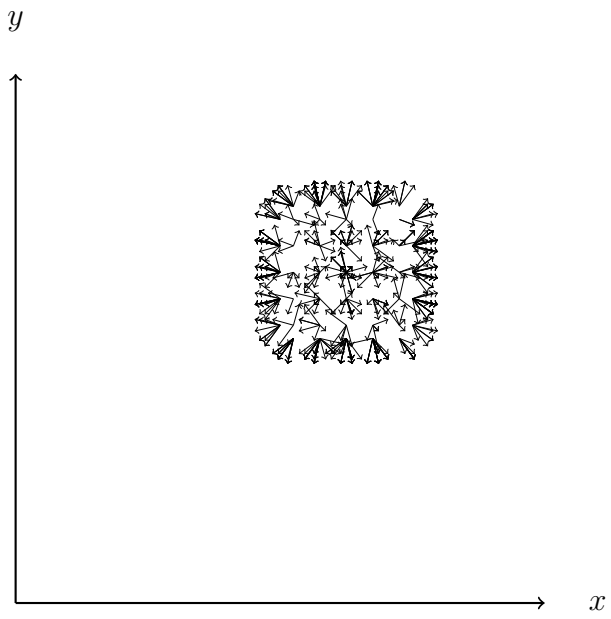
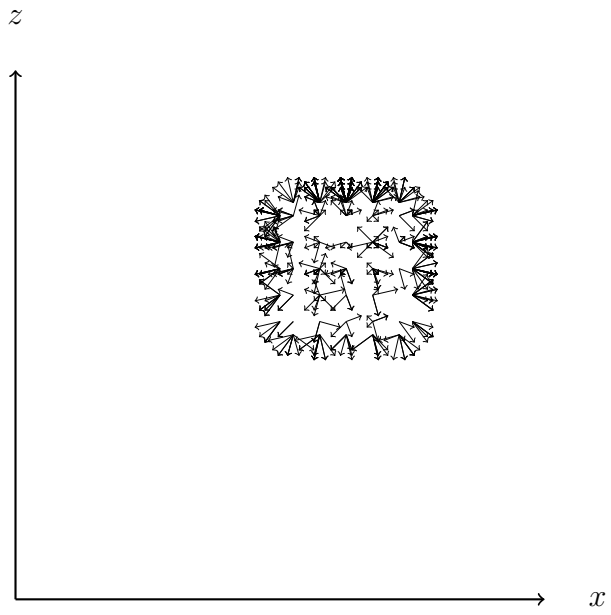



**Figure B.36:** MCNP Test Case 22 3D Projection

## B.19 Test Case 23




**(a)** $x$-$y$ Projection




**(b)** $x$-$z$ Projection

**Figure B.37:** MCNP Test Case 23 2D Projections

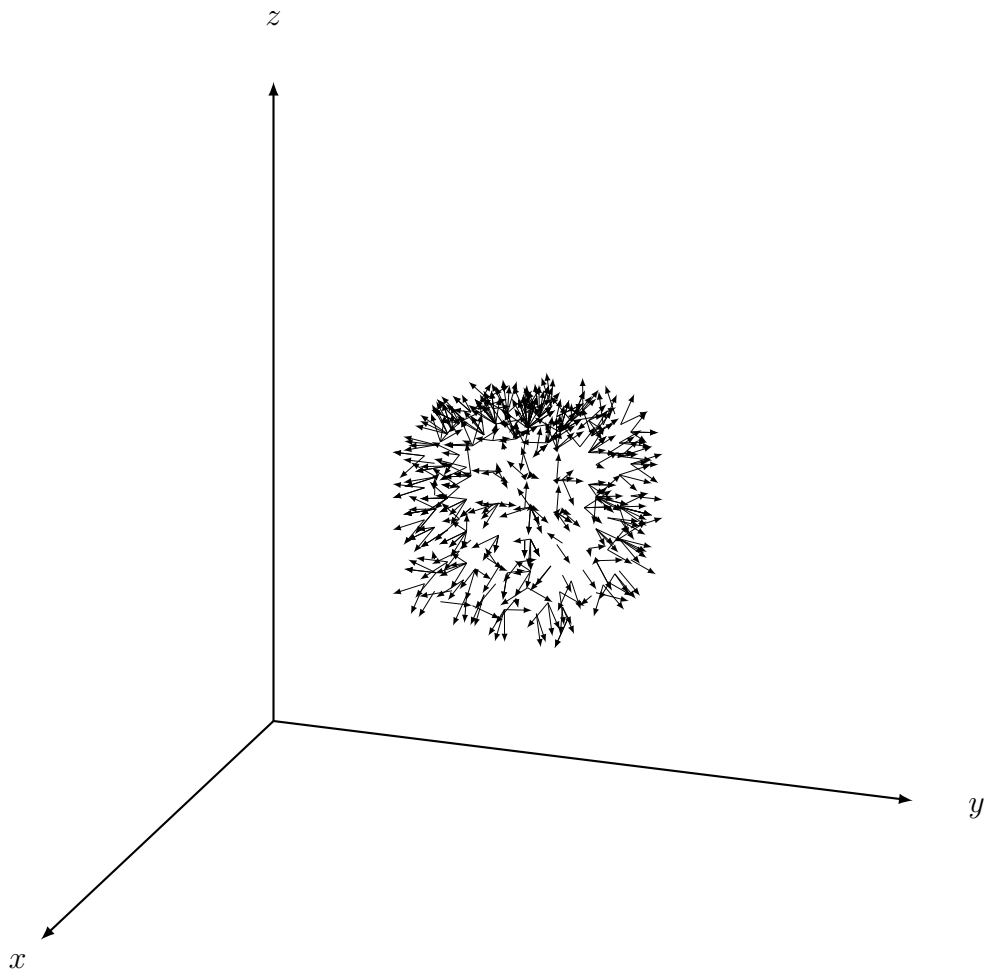
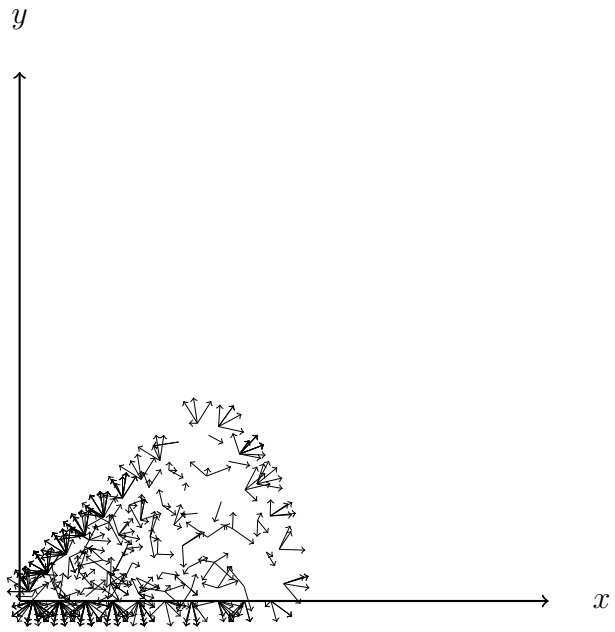
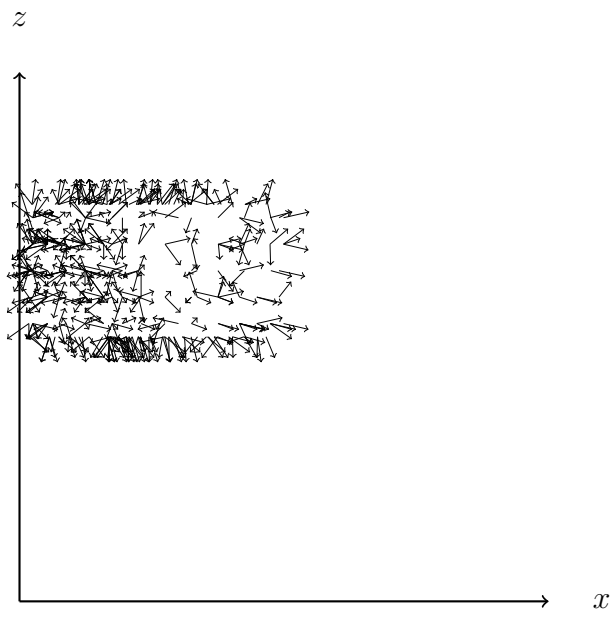



**Figure B.38:** MCNP Test Case 23 3D Projection

## B.20 Test Case 24




**(a)** $x$-$y$ Projection




**(b)** $x$-$z$ Projection

**Figure B.39:** MCNP Test Case 24 2D Projections

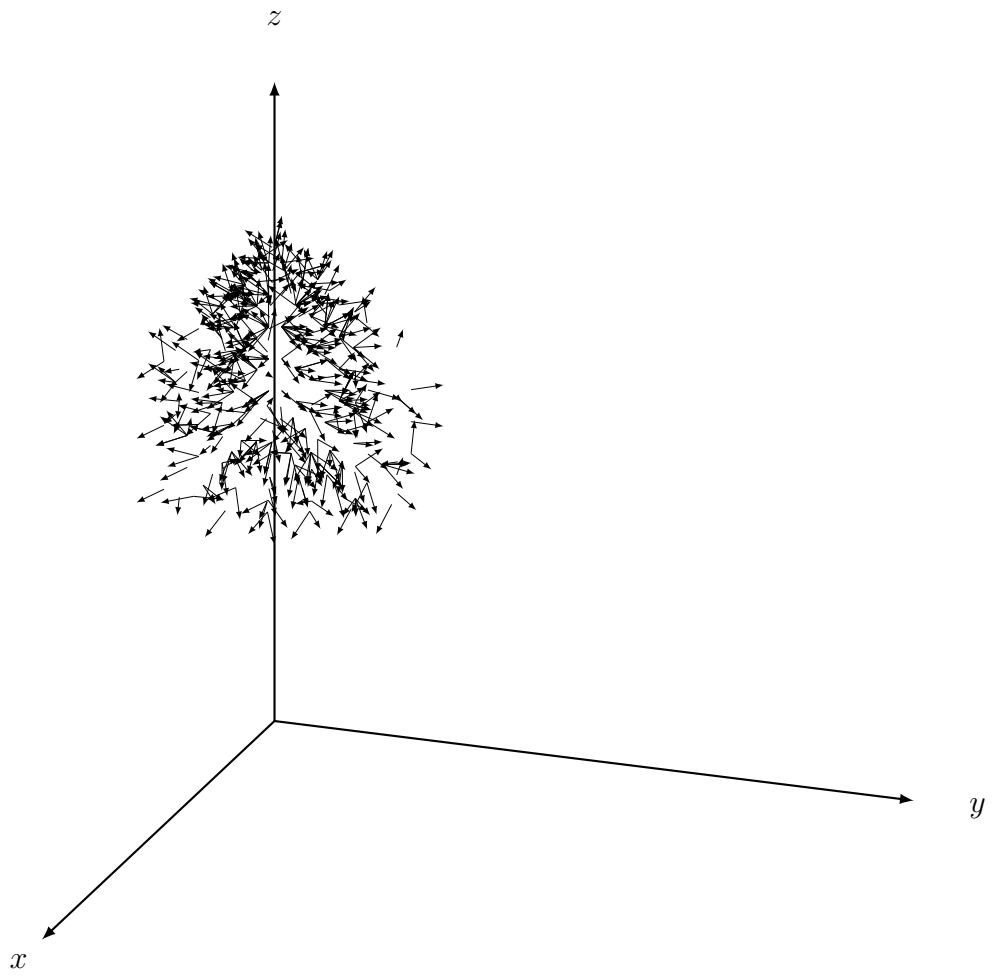



**Figure B.40:** MCNP Test Case 24 3D Projection

# Vita

Joel Kulesza was born in Detroit, Michigan, to the parents of Ronald and Julia Kulesza. He attended Oakley Park Elementary School in Walled Lake, MI, Clifford H. Smart Middle School in Commerce Township, MI, and Walled Lake Central High School in Walled Lake, MI. After graduation, he went to the University of Michigan in Ann Arbor to study engineering. At the outset of his second year, he chose to join the Nuclear Engineering & Radiological Sciences (NERS) department. Shortly after, he was invited to join the Tau Beta Pi (TBΠ) Engineering Honor Society and was formally inducted in April 2004. In May 2006, he graduated Summa Cum Laude with a Bachelor of Science in Engineering degree in Nuclear Engineering & Radiological Sciences with a minor in Mathematics. He then joined Westinghouse Electric Company in Monroeville, Pennsylvania. In September 2007, he began as a distance education student with the University of Tennessee in Knoxville in the Nuclear Engineering department. In August 2011, he received a Master of Science degree in Nuclear Engineering. He continues to be employed with Westinghouse Electric Company and currently resides in Monroeville, Pennsylvania.