DISCRETE-ORDINATES COST OPTIMIZATION OF WEIGHT-DEPENDENT VARIANCE REDUCTION TECHNIQUES FOR MONTE CARLO NEUTRAL PARTICLE TRANSPORT

by

CLELL J. SOLOMON, JR.

B.S. Kansas State University, 2005
M.S., Kansas State University, 2007

AN ABSTRACT OF A DISSERTATION

submitted in partial fulfillment of the
requirements for the degree

DOCTOR OF PHILOSOPHY

Department of Mechanical and Nuclear Engineering
College of Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas
2010

Abstract

A method for deterministically calculating the population variances of Monte Carlo particle transport calculations involving weight-dependent variance reduction has been developed. This method solves a set of equations developed by Booth and Cashwell [1979], but extends them to consider the weight-window variance reduction technique. Furthermore, equations that calculate the duration of a single history in an MCNP5 (RSICC version 1.51) calculation have been developed as well. The calculation cost, defined as the inverse figure of merit, of a Monte Carlo calculation can be deterministically minimized from calculations of the expected variance and expected calculation time per history.

The method has been applied to one- and two-dimensional multi-group and mixed material problems for optimization of weight-window lower bounds. With the adjoint (importance) function as a basis for optimization, an optimization mesh is superimposed on the geometry. Regions of weight-window lower bounds contained within the same optimization mesh element are optimized together with a scaling parameter. Using this additional optimization mesh restricts the size of the optimization problem, thereby eliminating the need to optimize each individual weight-window lower bound.

Application of the optimization method to a one-dimensional problem, designed to replicate the variance reduction iron-window effect, obtains a gain in efficiency by a factor of 2 over standard deterministically generated weight windows. The gain in two dimensional problems varies. For a 2-D block problem and a 2-D two-legged duct problem, the efficiency gain is a factor of about 1.2. The top-hat problem sees an efficiency gain of 1.3, while a 2-D 3-legged duct problem sees an efficiency gain of only 1.05.

This work represents the first attempt at deterministic optimization of Monte Carlo calculations with weight-dependent variance reduction. However, the current work is limited in the size of problems that can be run by the amount of computer memory available in computational systems. This limitation results primarily from the added discretization of the Monte Carlo particle weight required to perform the weight-dependent analyses. Alternate discretization methods for the Monte Carlo weight should be a topic of future investigation. Furthermore, the accuracy with which the MCNP5 calculation times can be calculated deterministically merits further study.

DISCRETE-ORDINATES COST OPTIMIZATION OF WEIGHT-DEPENDENT VARIANCE REDUCTION TECHNIQUES FOR MONTE CARLO NEUTRAL PARTICLE TRANSPORT

by

CLELL J. SOLOMON, JR.

B.S. Kansas State University, 2005
M.S., Kansas State University, 2007

A DISSERTATION

submitted in partial fulfillment of the
requirements for the degree

DOCTOR OF PHILOSOPHY

Department of Mechanical and Nuclear Engineering
College of Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas
2010

Approved by:

Major Professor
Dr. J. Kenneth Shultis

Abstract

A method for deterministically calculating the population variances of Monte Carlo particle transport calculations involving weight-dependent variance reduction has been developed. This method solves a set of equations developed by Booth and Cashwell [1979], but extends them to consider the weight-window variance reduction technique. Furthermore, equations that calculate the duration of a single history in an MCNP5 (RSICC version 1.51) calculation have been developed as well. The calculation cost, defined as the inverse figure of merit, of a Monte Carlo calculation can be deterministically minimized from calculations of the expected variance and expected calculation time per history.

The method has been applied to one- and two-dimensional multi-group and mixed material problems for optimization of weight-window lower bounds. With the adjoint (importance) function as a basis for optimization, an optimization mesh is superimposed on the geometry. Regions of weight-window lower bounds contained within the same optimization mesh element are optimized together with a scaling parameter. Using this additional optimization mesh restricts the size of the optimization problem, thereby eliminating the need to optimize each individual weight-window lower bound.

Application of the optimization method to a one-dimensional problem, designed to replicate the variance reduction iron-window effect, obtains a gain in efficiency by a factor of 2 over standard deterministically generated weight windows. The gain in two dimensional problems varies. For a 2-D block problem and a 2-D two-legged duct problem, the efficiency gain is a factor of about 1.2. The top-hat problem sees an efficiency gain of 1.3, while a 2-D 3-legged duct problem sees an efficiency gain of only 1.05.

This work represents the first attempt at deterministic optimization of Monte Carlo calculations with weight-dependent variance reduction. However, the current work is limited in the size of problems that can be run by the amount of computer memory available in computational systems. This limitation results primarily from the added discretization of the Monte Carlo particle weight required to perform the weight-dependent analyses. Alternate discretization methods for the Monte Carlo weight should be a topic of future investigation. Furthermore, the accuracy with which the MCNP5 calculation times can be calculated deterministically merits further study.

Table of Contents

List of Figures

List of Tables

Acknowledgments

Many people have been invaluable in completion of this work by providing guidance and insight as well as technical and moral support. The following section contains acknowledgments to these people.

First I would like to thank Dr. Ken Shultis, my major professor for not just this Ph.D. work but also my master's work, for being a guide and source of invaluable advice while I was pursuing this research. I would also like to thank Dr. Shultis for being the one who initially corrupted me into the world of computational neutronics and simulation.

Next I would like to thank Dr. Avneet Sood for being a committee member and mentor to me during my summers at LANL and over the past year as I've completed this work. Furthermore, I would like to thank Dr. Sood for the funding to perform this work and the other work we have pursued over the past five years.

I would also like to acknowledge my other committee members Dr. Bill Dunn, Dr. Larry Weaver, Dr. John Maginnis. Thanks to Dr. Dunn and Dr. Weaver for the classes, the guidance, and serving on my doctoral committee, and thanks to Dr. Maginnis for serving as the outside chairmember on my committee.

I would like to offer much thanks to Dr. Tom Booth, Dr. Art Forster, Dr. Tom Hill, and Dr. Kent Parsons. Dr. Booth has provided me with invaluable insight into the world of variance reduction and has been a substantial guide in approaching this research. I would like to thank Dr. Forster for his review of this work and insight into its possible extensions. I thank Dr. Tom Hill and Dr. Kent Parsons for their guidance in learning the S_n method and providing help when needed.

Also, I would like to thank my fellow graduate students at Kansas State University and my fellow student interns at LANL for their moral support and friendships as I have pursued this research and degree. Additionally, I would like to thank the staff of the Mechanical and Nuclear Engineering Department at Kansas State University for their friendships and help over the past years.

A special thanks to Tadeusz Raven, Dave Sayre, and Brian Hnath for helping assemble and setup computers that could actually run the calculations presented in this work. Also, thanks to the members of the X-Division XCS team for their aid with software issues.

Most, I would like to thank my parents, Clell and Melody Solomon, and my brothers and sister, Philip, Peter, and Tiffany, for all their support, encouragement, and guidance throughout college and my life. Thanks for always being there and for all of your love.

Lastly, I want to thank my wife, Emilia. Thanks for all your support and love as I have completed this degree.

Chapter 1

Introduction and Background

The Monte Carlo method is extensively applied to a wide variety of radiation transport calculations. Because of its ability to utilize continuous-energy cross sections and to transport particles in all, rather than discrete, directions, the Monte Carlo method has some distinct advantages over traditional deterministic S_n and P_n methods. One disadvantage of Monte Carlo is that it lacks the general ability to obtain statistically-sound results at distances far away from the simulated radiation source, and, for this reason, many variance reduction techniques have been been developed to obtain better results for this class of problems.

Study of the theory of Monte Carlo uncertaintiess began in the late 1970s with work by Amster and Djomehri [1976]. In Amster and Djomehri's work, a set of equations is developed for the moments of the score distribution resulting from a Monte Carlo tally. Equally important to theory of Monte Carlo uncertainties is Amster and Djomehri's proof that the equations developed are adjoint to the linearized Boltzmann transport equation commonly used to describe radiation transport processes. The first moment equation is equivalent to the solution of the adjoint transport equation and provides the expected score of a particle. The second moment, along with the first, provides the population variance of the underlying score distribution the Monte Carlo seeks to estimate.

The work of Amster and Djomehri was later independently generalized by Booth and Cashwell [1979] and Sarkar and Prasad [1979] to include the variance reduction schemes current to that day. The result of these two works was a set of equations that not only offered a means of calculating the population variance of a Monte Carlo score distribution but one that does so with the inclusion of multiple variance reduction effects. The work of Sarkar and Prasad took the additional step of considering the optimization of the Monte Carlo calculation specifically for the exponential transform game. In Sarkar and Prasad's work, the number of collisions was used as an approximation of the required calculation time. With an estimate of the calculation time and variances from P_1 calculations, predictions about the minimum Monte Carlo calculation cost were made in one-group 1-D slab problems. Shortly after the works by Booth and Cashwell and Sarkar and Prasad, Lux [1980] made analytical predictions about a novel expected leakage estimator and quasi-optimum rouletting parameters using the history-score moment equations and compared the predictions to actual Monte Carlo calculations.

Similar calculations to those of Sarkar and Prasad were performed by Juzaitis [1982]. In these calculations the optimum locations of one and two splitting surfaces were determined for one-group, 1-D slabs. Again, using the expected number of collisions as a replacement for the calculation time, Juzaitis was able to determine the location of splitting surfaces that provides a minimum

Monte-Carlo cost using the S_n method. Neither Sarkar and Prasad nor Juzaitis [1982] considered the case of weight-dependent variance reduction schemes, e.g. weight windows, in their work.

Current state-of-the-art variance reduction techniques generally employ the importance (adjoint) function as basis for determining a set of variance reduction techniques. This importance function may be calculated using either deterministic [Wagner and Haghighat, 1998] or Monte Carlo [Booth, 1982; Booth and Hendricks, 1984] methods. Using the importance function for Monte Carlo biasing has proved very applicable to a wide range of problems. However, the importance function provides information only about the expected score a Monte Carlo particle will make and contains no information about the variance introduced by that particle nor the additional calculation time introduced by the biasing process. Thus, simple application of the importance function for biasing offers no insight into the efficiency of the calculation until after the calculation has been performed.

The works of Sarkar and Prasad and Juzaitis [1982] provided an initial investigation into the deterministic optimization of Monte Carlo transport problems. However, little record of any additional attempts to predict the cost of Monte Carlo calculations with deterministic calculations was found. Perhaps one reason for the lack of additional work with the deterministic methods was the development of the Monte Carlo weight-window generator [Booth, 1982] that made the calculation of importance estimations possible directly in the Monte Carlo. Methods to predict variances and minimize calculation costs directly in the Monte Carlo calculation were considered by Dubi [1985a; 1985b] for splitting games at surfaces. Dubi's initial work was later generalized to the Direct Statistical Approach (DSA) [Dubi et al., 1985, 1986], and even later extended by Burn [1992, 1995, 1997] to include a cell importance model, particle bifurcation between surfaces, and a weight-dependent approach.

Burn's modification of the DSA method to include weight dependence was the first consideration of weight-dependent variance reduction with respect to the score distribution moments and optimization. Similarly, this work seeks to optimize Monte Carlo cost with weight-dependent variance-reduction parameters using strictly deterministic calculations. Specifically, this work is interested in the optimization of weight-window lower bounds for the weight-window game in MCNP5 [X-5 Monte Carlo Team, 2003]. Though optimization of weight-window lower bounds is the ultimate goal of this work, the method developed to solve the moment equations and predict the Monte Carlo cost is shown to be applicable to other variance reduction techniques as well.

As this work was being completed, a similar study was presented that considers weight-dependent variance reduction as well [van Wijk and Hoogenboom, 2010]. Despite some similarities in the work, the work by van Wijk and Hoogenboom considers extremely simplified versions of the transport equation, namely bi-directional transport in a slab and inifinte medium solutions. Furthermore, the definition of weight window employed by van Wijk and Hoogenboom is not that typically used in production codes such as MCNP. Furthermore, the practical implementation into the PARTISN [Alcouffe et al., 2008] code presented by van Wijk and Hoogenboom is limited to a single weight window uniform across energy and space, which is nearly never the case. van Wijk and Hoogenboom's multigroup considerations are limited to cases for which the weight window in one group is an exact multiple of that in another, which is also generally not the case, and an infinite medium. Finally, van Wijk and Hoogenboom's work is limited to collisional estimators. van Wijk and Hoogenboom's work does provide useful analytical results for the cases of bidirectional transport in a slab and a two group infinite medium problem.

This work seeks a general numerical solution method for the history-score moment equations.

The moment equations defined by Booth and Cashwell [1979] are extended to consider the weight window as implimented by general transport codes such as MCNP. Moreover, the one-dimensional transport problems considered here are not limited to simple bidirectional transport, and this work considers even two-dimensional transport problems. The method developed herein has no restriction on the weight windows in multiple energy groups and does not require that the weight window be uniform in space and energy for a practical implementation. Furthermore, this work attempts to directly calculate the expected time to process a single history rather than using the number of collisions directly as a stand-in for the calculation effort.

The remainder of this work is structured as follows. Chapter 2 derives the history-score moment equations being solved following the work of Booth and Cashwell [1979]. Furthermore, the integral equations developed by Booth and Cashwell are converted to an integro-differential form for solution by an S_n code specifically written for this work. Chapter 2 also extends the history-score moment equations to consider the weight-window game as defined by the MCNP transport code [X-5 Monte Carlo Team, 2003].

Chapter 3 shows how the moment equations are discretized using discrete ordinates. Chapter 3 introduces the discretization applied to the Monte Carlo particle weight and illustrates how individual variance reduction games affect the particle weight domain. Finally, Chapter 3 derives an expected track-length estimator and describes its application and limitations to discrete ordinates solutions of the history-score moment equations.

Chapter 4 presents the development of the equations used to estimate the time required to perform an MCNP5 (RSICC version 1.51) calculation. The timing studies of MCNP5 required to obtain estimates of the calculation times and the effects different variance reduction games have on the calculations times are also presented in Chapter 4.

Optimization results are given in Chapter 5. The optimization proceedure utilized is described along with how the optimization parameter domain is reduced. Results of 1-D one-group and two-group optimizations and 2-D one-group optimization are given. Chapter 6 sumarizes this work and suggests future work to be pursued.

Appendix A contains an extensive amount of verification data to show that the discrete ordinates code written for this work is capable of variance calculations for a substantial number of variance-reduction methods and combinations of variance-reduction methods. Appendix B demonstrates the inverse relationship between the integro-differential streaming operator and the integral transition operator necessary for some of the derivations in Chapter 2.

Chapter 2

Derivation and Extension of the History-Score Moment Equations

This work is concerned with the deterministic calculation of variances in Monte Carlo particle transport calculations and the optimization of the efficiencies of these calculations. In this chapter, Monte Carlo fundamentals and their application to particle transport are outlined. Moreover, variance reduction techniques commonly applied to particle-transport Monte Carlo are discussed and the deterministice equations governing the variances, originally developed by Booth and Cashwell [1979], are derived and extended to the weight-window variance reduction technique.

2.1 Monte Carlo Fundamentals

This section highlights the basics of the Monte Carlo method. Statistical considerations pertinent to the Monte Carlo method are discussed with particular focus on the moments of probability distribution functions. Also, the general application of the Monte Carlo method to particle transport is summarized along with some variance reduction techniques.

2.1.1 Statistics Considerations

The Monte Carlo method fundamentally seeks to determine a generally unknown probability distribution function (PDF). The average, or mean, of the PDF is typically the quantity of interest, and, unlike other numerical methods, the Monte Carlo technique provides a quantification of the uncertainty in the mean. From samples of the unknown PDF, the properties of the PDF may be determined. In this section, a few fundamental properties of PDFs are discussed along with the mathematical theorems that make the Monte Carlo method possible.

Probability Distribution Functions

Consider the function $f(\mathbf{x})$ where $\mathbf{x} \in V$. This function is a PDF if

1. $f(\mathbf{x}) \geq 0 \; \forall \; \mathbf{x}$, and
2. $\int_V d\mathbf{x} \; f(\mathbf{x}) = 1$.

The first requirement indicates the relative likelihood of an event in a differential region about **x** cannot be less than zero. The second requirement indicates that, when all possibilities **x** times the likelihood of that possibility, $f(\mathbf{x})$, are summed, represented here by an integral, then the result must be unity. In other words, the probability that one of the possible events **x** will occur is unity.

Moments of Probability Distribution Functions

Most important to this work are moments of probability distribution functions and their relationship to the mean and variance of the PDF. If one considers a PDF of a single variable $f(x)$, then the rth moment M_r of the PDF with respect to some function $z(x)$ is defined as

$$M_r[z(x)] = \int_{-\infty}^{\infty} dx\, z^r(x) f(x). \tag{2.1}$$

Of special interest for the Monte Carlo method are the first and second moments of the probability distribution function. The population, or true, mean μ is defined to the be the first moment of the PDF, i.e.,

$$\mu[z(x)] = M_1[z(x)] = \int_{-\infty}^{\infty} dx\, z(x) f(x). \tag{2.2}$$

The population variance $\sigma^2[z(x)]$ is defined as

$$\sigma^2[z(x)] = \int_{-\infty}^{\infty} dx\, (x-\mu)^2 f(x), \tag{2.3}$$

and is equivalent to

$$\sigma^2[z(x)] = M_2[z(x)] - M_1^2[z(x)]. \tag{2.4}$$

By calculating the first two moments of a known probability distribution function, both the population mean and population variance can be determined.

Estimation of Moments by Monte Carlo

In order to determine the moments of a PDF, as outlined above, one must know the PDF being randomly sampled. Often the PDF is not known and, even if it is, many PDFs do not lend themselves to simple analytic integration. It is in these cases that the Monte Carlo method for estimating the moments of a PDF is quite useful. Rather than calculate the exact population mean and population variance of a PDF, the Monte Carlo technique estimates the population mean and population variance with the sample mean $\bar{z}$ and sample variance $v^2(z)$[1], respectively. The sample mean and sample variance are obtained by repeated random sampling, obtaining individual data points, of the PDF. In such a manner the moments of a non-integrable or unknown PDF can be estimated as

$$\widehat{M}_r[z(x)] = \frac{1}{N} \sum_{i=1}^{N} z^r(x_i), \tag{2.5}$$

[1]Traditionally, the notation s^2 is used to represent sample variance; however, to avoid confusion with s, which is used here to represent the score to a tally, this alternative notation is utilized.

where N is the number of random samples, x_i is the ith random sample from the PDF, and $\widehat{M}_r$ indicates the r*th* sample moment. The sample mean and first sample moment are equal and given by

$$\bar{z} = \widehat{M}_1[z(x)] = \frac{1}{N}\sum_{i=1}^{N} z(x_i), \tag{2.6}$$

and the sample variance is given by

$$v^2(z) = \widehat{M}_2[z(x)] - \bar{z}^2, \tag{2.7}$$

where $\widehat{M}_2[z(x)]$ is the second sample moment given by

$$\widehat{M}_2[z(x)] = \frac{1}{N}\sum_{i=1}^{N} z^2(x_i). \tag{2.8}$$

Although the population variance is a fundamental property of a distribution, independent of the number of samples and the sample variance is an estimate of the population variance, the variance of the sample mean $v^2(\bar{z})$ decreases with the number of histories as [Dunn and Shultis, 2008]

$$v^2(\bar{z}) = v^2(z)/N. \tag{2.9}$$

To simplify the quantification of uncertainty of the mean, the sample standard deviation of the mean $v(\bar{z}) = \sqrt{v^2(\bar{z})} = \sqrt{v^2(z)/N}$ or relative error $R = v(\bar{z})/\bar{z}$ is typically employed.

Success of the Monte Carlo method is rooted in two fundamental mathematical theorems, stated here without proof. The first is the law of large number which states that in the limit as the number of random samples approaches infinity, the sample moments of a PDF approach the population moments, i.e., [Dunn and Shultis, 2008]

$$\lim_{N\to\infty} \frac{1}{N}\sum_{i=1}^{N} z^r(x_i) = \int_{-\infty}^{\infty} dx\, z^r(x) f(x). \tag{2.10}$$

Therefore, for a sufficiently large number of samples, the sample mean and sample variance will be a "good" approximation of the population mean and population variance. How "good" that approximation is quantified by the second important theorem, the central limit theorem.

In one form, the central limit theorem states that, in the limit that the number of random samples approaches infinity, the probability that the sample mean differs from the population mean by less than or equal to an amount λ is distributed as a Gaussian distribution, namely [Dunn and Shultis, 2008]

$$\lim_{N\to\infty} \text{Prob}\left\{\frac{|\widehat{M}_r[z(x)] - M_r[z(x)]|}{\sqrt{\sigma^2[z(x)]/N}} \le \lambda\right\} = \frac{1}{\sqrt{2\pi}}\int_{-\lambda}^{\lambda} du\, e^{-u^2/2}. \tag{2.11}$$

With the central limit theorem, it is possible to quantify the probability that the sample mean is within some tolerance of the population mean. The denominator in the probability contains the population variance, which is unknown but also can be estimated by Monte Carlo as discussed above. However, given an estimate of the sample variance alone, it is unknown how accurately the sample variance represents the population variance. To know how well the sample variance approximates the population variance, the *variance of the variance* must also be determined and

the central limit theorem applied to the sample variance. Only with a low variance of the variance is it known that the sample variance accurately approximates the population variance and, in turn, the sample mean accurately approximates the population mean.

2.1.2 Monte Carlo Neutral Particle Transport

The Monte Carlo method has been used for almost 70 years to simulate neutral particle radiation transport [Carter and Cashwell, 1975]. The stochastic behavior of radiation particles traversing a medium lends itself well to a solution by stochastic means. Based on the probabilities involved in all the processes a radiation particle can undergo, e.g., sourcing the particle, computing distances to collisions, computing collision events, etc., a solution to the linearized Boltzmann transport equation can be estimated.

Typically, the Monte Carlo procedure of generating a random sample for radiation transport is as follows. A particle's starting position, energy, and direction are sampled from a given source PDF. The distance to the next collision event is sampled from another PDF, and the properties of the particle(s) emerging from the collision event, if any, are sampled from yet another PDF. This process of sampling distances to collision events and then sampling the properties of particles emerging from the event continues until either the particle leaves the problem domain or is terminated by some natural, e.g., absorption, or artificial, e.g., variance reduction, event. Then, the entire process, called a *histroy*, is repeated for another initial particle.

Quantities of interest, such as flux talies, current tallies, reaction rate tallies, etc., are estimated during the process described above. As the particle progresses through the randomly sampled transport process, it may enter regions or cross surfaces over which quantities of interest are desired and contribute a score to the tally or tallies. Even though a particle may contribute to a tally multiple times during the transport process for a single history, it is the sum of these multiple contributions that is the score for that history, not each separate event. In this way, each history has associated with it a single score. If a particle never reaches a tally, the score is zero. Therefore, each history has a probability of contributing a score s described by an unknown PDF $p(s)$ known as the *history-score distribution function*. The probability that a history contributes a score in ds about s is then given by $p(s)ds$, and, from the properties of PDFs, the probability that some score results from a history is

$$\int_{-\infty}^{\infty} ds\ p(s) = 1. \tag{2.12}$$

Typically, tallies are chosen such that they produce purely positive scores and the bounds of the above integral are instead $[0, \infty)$.

The history-score distribution function is unknown a priori, hence the transport simulation process described above is used to obtain samples of the distribution. As samples are obtained, the history-score distribution function may be inferred, but, more importantly, moments of the history-score distribution function can be calculated as

$$M_r[s] = \frac{1}{N} \sum_{i=1}^{N} s^r. \tag{2.13}$$

From these moments the sample mean and sample variance of a tally can be determined. With a sufficiently high number of samples, application of the law of large numbers, and the central limit

theorem, the accuracy with which the sample mean and variance represent the population mean and variance can be determined.

2.1.3 Efficiency of Monte Carlo Techniques

Ideally, a description of the Monte Carlo simulation is desired such that every history contributes a score and the set of scores is sampled proportionately from the history-score distribution function. Describing the problem in such a manner is often difficult. Consider an isotropic source and a tally 100 mean free paths away. It is highly unlikely that a particle emitted from the source will traverse all 100 mean free paths and contribute a non-zero score to the tally. A direct or analog simulation of the such a problem is inefficient.

A metric commonly used to evaluate the efficiency of a Monte Carlo simulation is the figure of merit (FOM), defined as

$$\text{FOM} = \frac{1}{R^2 T}, \tag{2.14}$$

where R is the relative error obtained from sampling the history-score distribution for a given tally and T is the amount of time (typically computational) required to obtain that relative error. Because R^2 decreases as $1/N$ and T is proportional to the number of histories N, one can see that the FOM should be roughly constant

$$\text{FOM} = \frac{\bar{z}^2}{\frac{v^2(z)}{N} N\tau} = \frac{\bar{z}^2}{v^2(z)\tau}, \tag{2.15}$$

where τ is the expected time required to process a single history. Alternatively, the *cost* C of a Monte Carlo calculation is the inverse of the FOM, namely

$$C = \frac{v^2(z)\tau}{\bar{z}^2}. \tag{2.16}$$

Provided the sample mean and sample variance are adequate representations of the population mean and population variance, respectively, the cost may be written

$$C \simeq \frac{\sigma^2 \tau}{\mu^2}, \tag{2.17}$$

or in terms of the fundamental moments of the history-score distribution

$$C \simeq \frac{(M_2[s] - M_1^2[s])\tau}{M_1^2[s]}. \tag{2.18}$$

2.1.4 Variance Reduction Techniques of Particle Transport

Variance reduction is the process by which the underlying PDF being sampled in a Monte Carlo process is *fairly* altered such that the first moment or mean of the distribution remains unchanged and the second moment of the distribution is reduced, thereby reducing the variance. Many different variance reduction techniques, or "non-analog games," have been developed for application to radiation-particle transport Monte Carlo. Typically, the usefulness of any one technique is

problem dependent, although the weight-windows technique [X-5 Monte Carlo Team, 2003] has demonstrated its applicability to a wide variety of problems.

In particle transport Monte Carlo, variance reduction games are typically implemented by altering the probabilities of certain events. To maintain the fairness of the game, i.e., preserve the first moment, a non-physical quantity called particle weight w is introduced. Typically, particles from an unbiased source are born with unit weight. As the particle progresses through the transport process discussed above, aspects of the process may be biased to promote specific outcomes, and the particle weight is reduced in proportion to the magnitude of the bias so that the means of the tallies are preserved.

Generally, the penalty for using variance reduction is an increased computational time per history and, correspondingly, an increased computational cost, as indicated in Eq. (2.18). There is a balance between the variance reduction methods employed and the time required to perform the calculation. In some situations, a variance reduction technique does not reduce the variance commensurately with the additional time introduced by the technique. One common example is excessive splitting where most of the particles follow nearly the same random walk from the point of the split onward.

Variance reduction methods can be categorized into two types: weight-independent methods and weight-dependent methods. Weight-independent techniques, such as implicit capture, exponential transform, and importance splitting/rouletting, act on the particle regardless of the particle's weight. Weight-dependent games, such as weight cutoff and weight windows, only act on a particle if its weight meets some criterion. The following subsections discuss some particle transport variance reduction methods with which this work is concerned. This work compares exclusively to the Monte Carlo N-Particle (MCNP) transport code [X-5 Monte Carlo Team, 2003]. The implementations of the variance reduction techniques discussed is that of MCNP and other transport codes may have different implementations.

Implicit Capture

The implicit capture game biases the Monte Carlo transport process by assuming that the particle is never captured resulting in a termination. The probability that a particle is captured when colliding is Σ_c/Σ, where Σ_c and Σ are the macroscopic capture and total cross sections, respectively. When a particle collides and the implicit capture game is being played the particle always survives the collision, and, to compensate for the bias, the new weight of the surviving particle w' is modified as

$$w' = w\,(1 - \Sigma_c/\Sigma)\,. \tag{2.19}$$

If the medium the particle undergoes a collision in is highly absorbing, then the ratio Σ_c/Σ is nearly unity. In such a case, the new weight of the particle is reduced. On the other hand, if the medium is very diffuse, then $\Sigma_c \approx 0$ and the new weight is approximately that of the initial weight. When this latter situation is the case, the additional time introduced by the implicit capture game does not produce a benefit.

Weight Cutoff

The weight-cutoff game is actually a time reduction technique not a variance reduction technique, and, in general, increases the variance of a Monte Carlo calculation. The weight cutoff method

utilizes two user-specified parameters: the cutoff weight ω_c and the survival weight ω_s. If during the transport process a particle's weight w drops below the cutoff weight, then with a probability w/ω_s the particle's weight is promoted to ω_s, otherwise the particle is fairly terminated. Such a technique prevents the further processing of low weight particles that will simply consume computational effort and contribute little to either the means or variances of tallies.

The fact that the weight cutoff technique is variance increasing can be seen by considering a batch of particles, some with weights just above and some with weights just below the cutoff weight. Those particles with weights below the cutoff weight will undergo the weight cutoff game and a fraction of them will have their weights promoted to the survival weight. This introduces a nearly bimodal distribution of particle weights. One mode is those particles with weights slightly above the weight cutoff that did not experience the weight cutoff game, and the second mode is those particles now at the survival weight. The introduction of this weight discrepancy introduces greater variances in scores contributing to tallies.

Exponential Transform

The exponential transform game biases the distance particles travel between collisions, increasing that distance in some preferred direction and decreasing it in the opposite direction. The probability that a particle undergoes a collision a distance dh about h away from its current location is $\Sigma \exp(-\Sigma h) dh$. The exponential transform method introduces a parameter p such that $-1 \leq p \leq 1$ and creates a new fictitious total cross section Σ' based on the original as $\Sigma' = \Sigma(1 - p\mu)$. The effect is a stretching of the collision distance. Here, μ is the cosine of the angle between the particle's direction of travel $\mathbf{\Omega}$ and some desired direction $\mathbf{\Omega}_d$, i.e., $\mu = \mathbf{\Omega} \cdot \mathbf{\Omega}_d$. The weight correction w_c must be such that

$$\Sigma e^{-\Sigma h} dh = w_c \Sigma' e^{-\Sigma' h} dh, \tag{2.20}$$

to preserve the expected collided weight. Solving for w_c one obtains $\exp(-p\mu\Sigma h)/(1 - p\mu)$.

Because this method depends on the particle's direction of travel, the exponential transform game can introduce extreme variances in the weights of particles in regions where this technique is employed. These extreme differences in turn can lead to substantial variances in the score contributions to tallies. For this reason, good Monte Carlo practitioners always recommend that weight control methods be combined with the exponential transform technique.

Importance Splitting/Rouletting

The importance splitting and rouletting game biases the transport process by splitting individual particles into more particles of lesser weight when a particle traverses from a user-defined low-importance region to a high importance region. Contrarily, when a particle transitions from a user-defined high-importance region to a low importance region the particle's weight is increased in proportion to the importances or terminated.

Consider two regions, one with importance a and the other with importance b such that $b > a$. When a particle transitions from importance a to importance b it is split. With a probability a/b the split produces $\lfloor b/a \rfloor$[2] particles and with a probability $(1 - a/b)$ the split produces $\lceil b/a \rceil$ particles. In

[2]The notation $\lfloor c \rfloor$ indicates the floor of c or the maximum integer less than c. Similarly, the notation $\lceil c \rceil$ indicates the ceiling of c or the minimum integer greater than c. If c is itself an integer, then $\lfloor c \rfloor = \lceil c \rceil$.

either case the weight w' of the new particles or the surviving particle is $w' = wa/b$, where w is the weight of the original particle. Assigning the new particles' weights in such a manner decreases the dispersion of particle weights and helps to minimize variances in contributions to scores.

When a particle transitions from importance b to importance a, then with a probability of $(1 - a/b)$ the particle is rouletted (terminated). If the particle is not rouletted, then it survives with a probability a/b and its new weight w' becomes $w' = wb/a$.

Weight Windows

The weight-window technique essentially combines the weight cutoff technique with a weight dependent splitting game. The weight window game requires four user specified parameters: the weight-window lower bound ω_l, the weight-window upper bound ω_u, a weight-window survival weight ω_s such that $\omega_l \leq \omega_s \leq \omega_u$, and a maximum splitting/rouletting parameter K. Particles entering into a weight window region and/or experiencing a collision in the weight window region will do one of five things depending on their weight.

If the particle's weight is such that $\omega_l \leq w \leq \omega_u$ then the particle continues with weight unchanged. If the particles weight is such that $w < \omega_l/K$, then with a probability $1/K$ the particle survives and its new weight becomes $w' = Kw$. If the particle's weight is such that $\omega_l/K < w < \omega_l$, then with a probability w/ω_s the particle survives and its new wight becomes ω_s. If the particle's weight is such that $\omega_u < w < K\omega_u$ then the particle is split into $\lceil w/\omega_u \rceil$ particles and the new weight of each particle is $w' = w/\lceil w/\omega_u \rceil$. Finally, if the particle's weight is such that $w > K\omega_u$ then the particle is split into K particles and the new weight of each particle is $w' = w/K$.

The weight window method seeks to control the weight distribution of particles in a region. Ideally, the window maintains each particle's weight between ω_l and ω_u. However, particles with weights less than ω_l/K will not be rouletted into the window, nor will particles with weights in excess of $K\omega_u$ be split into the window. The cap on rouletting prevents an excessive introduction of variances in the particle weights and the cap on splitting prevents an excessive number of particles being created that need to be tracked and, thereby, prevents excessive additions to the computational time.

2.2 Derivation of the History-Score Moment Equations

Although the underlying PDF the Monte Carlo process seeks to sample may not be known a priori, the distribution and its moments typically exist. Booth and Cashwell [1979][3] have derived a set of integral equations for the moments of the history-score distribution function which account for particle bifurcation by importance splitting/rouletting, implicit capture, and weight cutoff methods. The section that follows reproduces and adds helpful details to their derivation as it is essential to the understanding of this work.

2.2.1 Description of the Scoring Functions and Transport Kernels

This section derives an integral equation for the history-score moment equations (or simply moment equations) by first developing an equation for the history-score distribution and then deriving

[3] A similar derivation, though somewhat less complete, was concurrently published by Sarkar and Prasad [1979]

moments of that distribution. Before the integral form of the moment equation can be derived, the scoring functions and integral kernels describing the transition of score probability must be described.

Phase Space

The phase space $\mathbf{P}$ is defined as position $\mathbf{r}$, direction $\mathbf{\Omega}$, energy E, and particle weight w, namely

$$\mathbf{P} = (\mathbf{r}, \mathbf{\Omega}, E, w). \tag{2.21}$$

Time t is considered separately. The subset of phase space $\mathbf{R} \in \mathbf{P}$ is defined as position $\mathbf{r}$, direction $\mathbf{\Omega}$, and energy E, namely

$$\mathbf{R} = (\mathbf{r}, \mathbf{\Omega}, E), \tag{2.22}$$

such that

$$\mathbf{P} = (\mathbf{R}, w). \tag{2.23}$$

For this derivation, $\mathbf{P}^+$ represents phase space after a transition but before collision, $\mathbf{P}'$ represents phase space after collision but before the effects of the collision, $\mathbf{P}''$ represents phase space after collision and its effects, and $\mathbf{P}^c$ represents phase space after collision and variance reduction applied at collision.

Scoring Functions

Each of the following scoring functions describes the probability that a particle contributes a score s in ds in its next event:

$$p_d(\mathbf{P}, \mathbf{P}', s)ds = \boxed{\text{probability of a score in } ds \text{ about } s \text{ for a collisionless free flight from } \mathbf{P} \text{ to } \mathbf{P}'}, \tag{2.24}$$

$$p_0(\mathbf{P}, \mathbf{P}', s)ds = \boxed{\text{probability of a score in } ds \text{ about } s \text{ for a departure from } \mathbf{P} \text{ followed directly by absorption at } \mathbf{P}'}, \tag{2.25}$$

$$p_1(\mathbf{P}, \mathbf{P}', s)ds = \boxed{\text{probability of a score in } ds \text{ about } s \text{ for a departure from } \mathbf{P} \text{ followed directly by a scattering at } \mathbf{P}'}, \tag{2.26}$$

$$p_k(\mathbf{P}, \mathbf{P}', s)ds = \boxed{\text{probability of a score in } ds \text{ about } s \text{ for a departure from } \mathbf{P} \text{ followed directly by a collision at } \mathbf{P}' \text{ that produces } k \text{ particles}}, \tag{2.27}$$

$$p_{sk}(\mathbf{P}, \mathbf{P}', s)ds = \boxed{\text{probability of a score in } ds \text{ about } s \text{ for a departure from } \mathbf{P} \text{ followed by a } k\text{-for-one split at } \mathbf{P}'}, \tag{2.28}$$

$$p_r(\mathbf{P}, \mathbf{P}', s)ds = \boxed{\text{probability of a score in } ds \text{ about } s \text{ for a departure from } \mathbf{P} \text{ followed by a rouletting game at } \mathbf{P}'}. \tag{2.29}$$

Weight Multiplication Functions

For nonanalog games, the quantity $\omega(\mathbf{R}, \mathbf{R}')$ describes the factor by which a particle's weight is multiplied for a specific transition event. Note that these factors are independent of the particle's initial weight.

$$\omega_t(\mathbf{R}, \mathbf{R}') = \boxed{\text{factor by which the particle's weight is multiplied for an eventless transition from } \mathbf{R} \text{ to } \mathbf{R}'}, \quad (2.30)$$

$$\omega_\Sigma(\mathbf{R}') = \boxed{\text{factor by which the particle's weight is multiplied if the particle collides at } \mathbf{R}'}, \quad (2.31)$$

$$\omega_{\epsilon_1}(\mathbf{R}', \mathbf{R}'') = \boxed{\text{factor by which the particle's weight is multiplied if the particle collides at } \mathbf{R}' \text{ and exits at } \mathbf{R}''}, \quad (2.32)$$

$$\omega_{\epsilon_k}(\mathbf{R}', \mathbf{R}_1, \ldots, \mathbf{R}_k) = \boxed{\text{factor by which the } k \text{ secondary particles' weights resulting from the collision of a single particle at } \mathbf{R}' \text{ are multiplied}}, \quad (2.33)$$

$$\omega_c = \boxed{\text{the weight cutoff—no particle may exit a collision with a weight less than } \omega_c}, \quad (2.34)$$

$$\omega_s = \boxed{\text{survival weight assigned to a particle if it survives a weight cutoff game}}. \quad (2.35)$$

Transport Kernels

The following transport kernels describe the probability of transitioning from one phase space region to another. Not all phase-space variables change in each kernel.

Transmission Kernel:

$$\begin{aligned} T(\mathbf{P}, \mathbf{P}^+) d\mathbf{\Omega}^+ dE^+ dw^+ &= T(\mathbf{R}, \mathbf{R}^+) \delta(w^+ - \omega_t w) d\mathbf{R}^+ dw^+ \\ &= T(\mathbf{r}, \mathbf{r}^+, E, \mathbf{\Omega}) \delta(E^+ - E)\, \delta(\mathbf{\Omega}^+ - \mathbf{\Omega}) \delta(w^+ - \omega_t w) dE^+ d\mathbf{\Omega}^+ dw^+ \\ &= \boxed{\begin{array}{l}\text{the probability that a particle at } \mathbf{r} \text{ will transition to } \mathbf{r}^+ \\ \text{while maintaining a constant energy } E \text{ and direction} \\ \mathbf{\Omega}. \text{ Upon arrival at } \mathbf{r}^+ \text{ the weight of the particle is} \\ \text{multiplied by } \omega_t \end{array}}. \end{aligned} \quad (2.36)$$

Collision Kernel:

$$\Sigma(\mathbf{P}^+, \mathbf{P}') d\mathbf{r}^+ d\mathbf{P}' = \Sigma(\mathbf{r}^+) \delta(\mathbf{r}' - \mathbf{r}^+) \delta(E' - E^+) \delta(\mathbf{\Omega}' - \mathbf{\Omega}^+) \delta(w' - \omega_\Sigma w^+)$$

$$= \boxed{\begin{array}{l}\text{the probability that a particle at } \mathbf{r}^+ \text{ will collide in } d\mathbf{r}^+ \text{ about} \\ \mathbf{r}^+ \text{ and emerge in } d\mathbf{P}' \text{ about } \mathbf{P}'. \text{ The particle's position, en-} \\ \text{ergy, and direction remain unchanged and only the particle's} \\ \text{weight may change. } \Sigma(\mathbf{r}^+) \text{ is the total macroscopic cross sec-} \\ \text{tion}\end{array}} . \quad (2.37)$$

Splitting Kernel:

$$S_k(\mathbf{P}^+)d\mathbf{r}^+ = \boxed{\begin{array}{l}\text{the probability that a particle at } \mathbf{r}^+ \text{ traveling in direction } \mathbf{\Omega}^+ \\ \text{with energy } E^+ \text{ and weight } w^+ \text{ will undergo a } k\text{-for-one split} \\ \text{in } d\mathbf{r}^+ \text{ about } \mathbf{r}^+\end{array}} . \quad (2.38)$$

Russian Roulette Probability:

$$R_o(\mathbf{r}^+)d\mathbf{r}^+ = \boxed{\begin{array}{l}\text{the probability that a particle at } \mathbf{r}^+ \text{ traveling in direction } \mathbf{\Omega}^+ \\ \text{with energy } E^+ \text{ and weight } w^+ \text{ will play Russian roulette in} \\ d\mathbf{r}^+ \text{ about } \mathbf{r}^+\end{array}} . \quad (2.39)$$

Collisionless Free-Flight (Drift) Probability:

$$\begin{aligned} D(\mathbf{P}, \mathbf{P}^+, t)d\mathbf{r}^+ &= T(P, P^+)\delta(|\mathbf{r}^+ - \mathbf{r}| - v(E)t)d\mathbf{r}^+ \quad (2.40) \\ &= T(\mathbf{r}, \mathbf{r}^+, \mathbf{\Omega}, E)\delta(\mathbf{\Omega}^+ - \mathbf{\Omega})\delta(E^+ - E)\delta(|\mathbf{r}^+ - \mathbf{r}| - v(E)t) \\ &\quad \times\delta(w^+ - \omega_t w)d\mathbf{r}^+ d\mathbf{\Omega}^+ dE^+ dw^+ \\ &= \boxed{\begin{array}{l}\text{the probability that a particle departing } \mathbf{P} \text{ will undergo a col-} \\ \text{lisionless flight (drift) for time } t \text{ with energy } E \text{ along direc-} \\ \text{tion } \mathbf{\Omega} \text{ and with weight } w \text{ and end up in } d\mathbf{r}^+ \text{ about } \mathbf{r}^+ \text{ with} \\ \text{the same energy and direction, but a modified weight}\end{array}} . \quad (2.41) \end{aligned}$$

Absorption Probability:

$$A(\mathbf{P}') = \boxed{\begin{array}{l}\text{the probability that a particle is absorbed when colliding at} \\ \mathbf{P}'\end{array}} . \quad (2.42)$$

Multiplication Probability:

$$\begin{aligned} \epsilon_k(\mathbf{P}', \mathbf{P}_1, \ldots, \mathbf{P}_k)d\mathbf{P}_1 \ldots d\mathbf{P}_k &= \epsilon_k(E', \mathbf{\Omega}', E_1, \mathbf{\Omega}_1, \ldots, E_k, \mathbf{\Omega}_k) \\ &\quad \times\delta(\mathbf{r}_1 - \mathbf{r}') \ldots \delta(\mathbf{r}_k - \mathbf{r}')\delta(w_1 - \omega_{\epsilon_k} w') \ldots \delta(w_k - \omega_{\epsilon_k} w') \\ &\quad \times d\mathbf{r}_1 d\mathbf{\Omega}_1 dE_1 dw_1 \ldots d\mathbf{r}_k d\mathbf{\Omega}_k dE_k dw_k \\ &= \boxed{\begin{array}{l}\text{the probability that a particle colliding at } \mathbf{P}' \\ \text{produces } k \text{ secondaries, the } j\text{th of which ends} \\ \text{up in } d\mathbf{P}_j \text{ about } \mathbf{P}_j\end{array}} . \quad (2.43) \end{aligned}$$

Scattering Probability:

$$\begin{aligned} E(\mathbf{P}', \mathbf{P}'')d\mathbf{P}'' &= \epsilon_1(\mathbf{P}', \mathbf{P}_1)d\mathbf{P}_1 \\ &= E(E', \mathbf{\Omega}', E'', \mathbf{\Omega}'')\delta(\mathbf{r}'' - \mathbf{r}')\delta(w'' - \omega_{\epsilon_1} w')d\mathbf{r}''d\mathbf{\Omega}''dE''dw'' \\ &= \boxed{\text{the probability that a particle colliding at } \mathbf{P}' \text{ will scatter into } d\mathbf{P}'' \text{ about } \mathbf{P}''} \end{aligned} . \quad (2.44)$$

Splitting Weight Change Probability:

$$\begin{aligned} B_{sk}(\mathbf{P}^+, \mathbf{P}'')d\mathbf{P}'' &= \delta(\mathbf{r}'' - \mathbf{r}^+)\delta(\mathbf{\Omega}'' - \mathbf{\Omega}^+)\delta(E'' - E^+) \\ &\quad \times\delta(w'' - w^+/k)d\mathbf{r}''d\mathbf{\Omega}''dE''dw'' \\ &= \boxed{\begin{array}{l}\text{the probability that a particle experiencing a } k\text{-for-one split} \\ \text{ends up in } d\mathbf{P}'' \text{ about } \mathbf{P}'', \text{ where only the weight has been} \\ \text{divided by } k\end{array}} \end{aligned} . \quad (2.45)$$

Russian Roulette Weight Change Probability:

$$\begin{aligned} B_o(\alpha, \mathbf{P}^+, \mathbf{P}'')d\mathbf{P}''d\alpha &= \delta(\mathbf{r}'' - \mathbf{r}^+)\delta(\mathbf{\Omega}'' - \mathbf{\Omega}^+)\delta(E'' - E^+) \\ &\quad \times \left[\alpha\delta(w'' - w^+/\alpha) + (1-\alpha)\delta(w'')\right] d\mathbf{r}''d\mathbf{\Omega}''dE''dw''d\alpha \\ &= \boxed{\begin{array}{l}\text{the probability that a particle undergoing Russian} \\ \text{roulette with a survival probability of } \alpha \text{ exits in } d\mathbf{P}'' \\ \text{about } \mathbf{P}'', \text{ where the particle is either killed } (w'' = 0) \\ \text{with a probability of } (1-\alpha) \text{ or the weight is increased} \\ \text{by a factor of } 1/\alpha \text{ with a probability of } \alpha\end{array}} \end{aligned} . \quad (2.46)$$

Weight Cutoff Weight Change Probability:

$$\begin{aligned} B_c(\mathbf{P}'', \mathbf{P}^c)d\mathbf{P}^c &= \delta(\mathbf{r}^c - \mathbf{r}'')\delta(\mathbf{\Omega}^c - \delta'')\delta(E^c - E'') \\ &\quad \times\{\Theta(w'' - \omega_c)\delta(w^c - w'') + [1 - \Theta(w'' - \omega_c)] \\ &\quad \times [(w''/\omega_e)\delta(w^c - \omega_c) + (1 - w''/\omega_e)\delta(w^c)]\} d\mathbf{r}^c d\mathbf{\Omega}^c dE^c dw^c \\ &= \boxed{\begin{array}{l}\text{the probability that a particle exiting a collision at } \mathbf{r}'' \text{ will} \\ \text{undergo weight cutoff and have resulting phase space coor-} \\ \text{dinates in } d\mathbf{P}^c \text{ about } \mathbf{P}^c\end{array}} \end{aligned} . \quad (2.47)$$

2.2.2 Derivation of the History-Score Distribution Function

The *history-score distribution function* $\psi(\mathbf{P}, s)$, the probability that a particle at $\mathbf{P}$ and all its progeny will yield a score in ds about s, is derived by considering a probability balance of all the possible ways a particle in $\mathbf{P}$ at time t_o can end a free flight by time $t > t_o$. There are six possible ways to end the free flight as illustrated in Fig. 2.1:

1. stream for the entire time $t - t_o$ and arrive at $\mathbf{P}^+$ at time t,

2. be absorbed at time $t' \in (t_o, t)$ upon arriving at $\mathbf{P}'$ at time t',

3. undergo Russian roulette at time $t' \in (t_o, t)$ upon arriving at $\mathbf{P}^+$ at time t' and surviving with probability α,

4. split k-to-one at time $t' \in (t_o, t)$ upon arriving at $\mathbf{P}^+$ at t' and all progeny begin at $\mathbf{P}^+$ and time t' with weight w/k,

5. scatter at time $t' \in (t_o, t)$ upon arriving at $\mathbf{P}'$ at time t'.

6. multiply at time $t' \in (t_o, t)$ upon arriving at $\mathbf{P}'$ at time t' and all progeny begin at $\mathbf{P}'$ and t' with weight w.

Each of the above possibilities is a probability of the form

$$\boxed{\text{probability of event occurring}} \times \boxed{\begin{array}{c}\text{probability that event}\\ \text{contributes a score in}\\ ds' \text{ about } s'\end{array}} \times \boxed{\begin{array}{c}\text{probability that all subsequent}\\ \text{events experienced by the}\\ \text{particle and progeny lead to a}\\ \text{score in } ds \text{ about } s - s' \text{ in the}\\ \text{remaining time } t - t'\end{array}}, \tag{2.48}$$

and each will now be considered in terms of the transport kernels presented above.

Case 1: Stream for Entire Time $t - t_o$

The probability of the event occurring is $D(\mathbf{P}, \mathbf{P}^+, t)d\mathbf{P}^+$. The probability of a score in ds' about s' is $p_d(\mathbf{P}, \mathbf{P}^+, s')ds'$. The probability that the particle and progeny produce a score in ds about $s - s'$ in the remaining time is $\delta(s - s')ds$, because no time remains. Thus we have the expected score distribution for a free flight from $\mathbf{P}$ is

$$\left[D(\mathbf{P}, \mathbf{P}^+, t)d\mathbf{P}^+\right] \times \left[p_d(\mathbf{P}, \mathbf{P}^+, s')ds'\right] \times \left[\delta(s - s')ds\right]. \tag{2.49}$$

Case 2: Absorption at Time t' after Transition from P to P′

The probability of the event occurring is the probability of transition from $\mathbf{P}$ to $\mathbf{P}^+$, $T(\mathbf{P}, \mathbf{P}^+)d\mathbf{P}^+$, times the probability of collision at $\mathbf{P}^+$, $\Sigma(\mathbf{P}^+, \mathbf{P}')d\mathbf{P}'$, times the probability that the collision is an absorption event, $A(\mathbf{P}')$. The probability of generating a score in ds' about ds is $p_0(\mathbf{P}, \mathbf{P}', s')ds'$, and, because absorption ends the history, the probability that the particle and progeny produce a score in ds about $(s - s')$ is $\delta(s - s')ds$. Thus, the expected score distribution from absorption is

$$\left[T(\mathbf{P}, \mathbf{P}^+)\Sigma(\mathbf{P}^+, \mathbf{P}')A(\mathbf{P}')d\mathbf{P}^+d\mathbf{P}'\right] \times \left[p_0(\mathbf{P}, \mathbf{P}', s')ds'\right] \times \left[\delta(s - s')ds\right]. \tag{2.50}$$

Case 3: Russian Roulette at Time t' after Transition from P to P+

The probability of the event occurring is the probability of transition from $\mathbf{P}$ to $\mathbf{P}^+$, $T(\mathbf{P}, \mathbf{P}^+)d\mathbf{P}^+$, times the probability of Russian roulette at $\mathbf{P}^+$, $R_o(\mathbf{P}^+)$, times the probability of surviving Russian roulette and exiting at $d\mathbf{P}''$, $B_o(\alpha, \mathbf{P}^+, \mathbf{P}'')d\alpha d\mathbf{P}''$. The probability of generating a score in ds' about s' *before* the rouletting game is $p_r(\mathbf{P}, \mathbf{P}^+, s')ds'$. The probability that a total score in ds about $(s - s')$

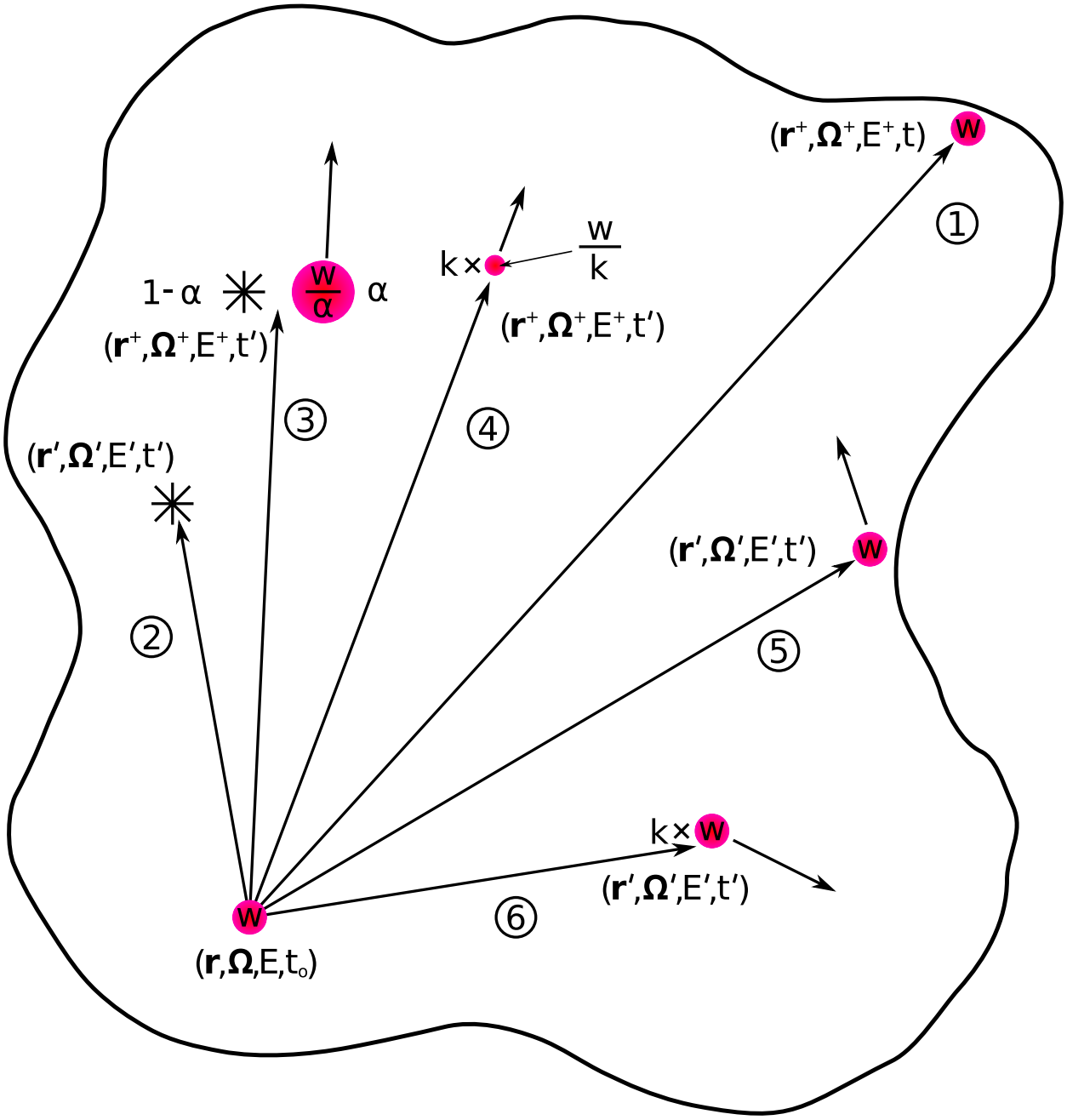


Figure 2.1. the possible methods a particle can end its free flight (not all possible sources of weight change, such as changing during transition, are depicted). Numbers in circles correspond to the six cases of Section 2.2.2.

by all particles surviving roulette is then $\psi(\mathbf{P}'', t-t', s-s')ds$. Thus, the expected score distribution from Russian roulette is

$$\begin{aligned} & [T(\mathbf{P},\mathbf{P}^+)R_o(\mathbf{P}^+)B_o(\alpha,\mathbf{P}^+,\mathbf{P}'')d\mathbf{P}^+ d\alpha d\mathbf{P}''] \\ \times \quad & [p_r(\mathbf{P},\mathbf{P}^+,s')ds'] \times [\psi(\mathbf{P}'',t-t',s-s')ds]\,. \end{aligned} \tag{2.51}$$

Case 4: Splitting at Time t' after Transition from P to P$^+$

The probability of the event occurring is the probability of transition from $\mathbf{P}$ to $\mathbf{P}^+$, $T(\mathbf{P},\mathbf{P}^+)d\mathbf{P}^+$, times the probability of splitting k-to-one at $\mathbf{P}^+$, $S_k(\mathbf{P}^+)$, times the weight change from splitting $B_{sk}(\mathbf{P}^+,\mathbf{P}'')d\mathbf{P}''$. The score generated in ds' about s' for the k-to-one split *before* the split is $p_{sk}(\mathbf{P},\mathbf{P}^+,s')ds'$. The probability that the total score generated by the progeny is in ds about $(s-s')$ is the probability that the first $k-1$ progeny produce scores in ds_j about s_j and the kth progeny produces a score in ds about $(s-(s'+s_1+\ldots+s_{k-1}))$. Because each progeny produced is a separate random sample, the result is the product of probabilities for each particle, namely

$$\prod_j^{k-1} \psi(\mathbf{P}'',t-t',s_j)ds_j \times \psi(\mathbf{P}'',t-t',s-(s'+s_1+\ldots+s_{k-1}))ds.$$

The total expected score distribution resulting from a split is therefore

$$\begin{aligned} & [T(\mathbf{P},\mathbf{P}^+)S_k(\mathbf{P}^+)B_{sk}(\mathbf{P}^+,\mathbf{P}'')d\mathbf{P}^+ d\mathbf{P}''] \times [p_{sk}(\mathbf{P},\mathbf{P}^+,s')ds'] \times \\ & \left[\prod_j^{k-1} \psi(\mathbf{P}'',t-t',s_j)ds_j \times \psi(\mathbf{P}'',t-t',s-(s'+s_1+\ldots+s_{k-1}))ds\right]. \end{aligned} \tag{2.52}$$

Case 5: Scattering at Time t' after Transition from P to P$'$ Followed by Weight-Cutoff Russian Roulette

The probability of the event occurring is the probability of transition from $\mathbf{P}$ to $\mathbf{P}^+$, $T(\mathbf{P},\mathbf{P}^+)d\mathbf{P}^+$, times the probability of colliding at $\mathbf{P}^+$, $\Sigma(\mathbf{P}^+,\mathbf{P}')d\mathbf{P}')$, times the probability that the collision yields a single particle $E(\mathbf{P}',\mathbf{P}'')d\mathbf{P}''$, times the probability of surviving weight-cutoff Russian roulette, $B_c(\mathbf{P}'',\mathbf{P}^c)d\mathbf{P}^c$. The probability that a score in ds' about s' is generated *before* the weight-cutoff game is $p_1(\mathbf{P},\mathbf{P}',s')ds'$. The probability that the recoiling particle generates a score in ds about $(s-s')$ in the remaining amount of time $t-t'$ is $\psi(\mathbf{P}^c,t-t',s-s')ds$. The total score distribution resulting from a scatter is then

$$\begin{aligned} & [T(\mathbf{P},\mathbf{P}^+)\Sigma(\mathbf{P}^+,\mathbf{P}')E(\mathbf{P}',\mathbf{P}'')B_c(\mathbf{P}'',\mathbf{P}^c)d\mathbf{P}^+ d\mathbf{P}' d\mathbf{P}'' d\mathbf{P}^c] \\ & \quad \times [p_1(\mathbf{P},\mathbf{P}',s')ds'] \times [\psi(\mathbf{P}^c,t-t',s-s')ds]\,. \end{aligned} \tag{2.53}$$

Case 6: Multiplication at Time t' after Transition from P to P$'$ Followed by Weight-Cutoff Russian Roulette of all the Progeny

The probability of the event occurring is the probability of transition from $\mathbf{P}$ to $\mathbf{P}^+$, $T(\mathbf{P},\mathbf{P}^+)d\mathbf{P}^+$, times the probability of colliding at $\mathbf{P}^+$, $\Sigma(\mathbf{P}^+,\mathbf{P}')d\mathbf{P}')$, times the probability the collision produces k progeny, the j of which is in $d\mathbf{P}_j$, $\epsilon_k(\mathbf{P}',\mathbf{P}_1,\ldots,\mathbf{P}_k)\prod_{j=1}^k d\mathbf{P}_j$, times the probability that each

of the j particles survives the weight-cutoff game, $\prod_{j=1}^{k} B_c(\mathbf{P}_j, \mathbf{P}_j^c) d\mathbf{P}_j^c$. The score in ds' about s' generated by the collision is $p_k(\mathbf{P}, \mathbf{P}', s')ds'$. The score distributions generated by the progeny are, for the same reasons as Case 4 above,

$$\prod_{j}^{k-1} \psi(\mathbf{P}_j^c, t - t', s_j) ds_j \times \psi(\mathbf{P}_k^c, t - t', s - (s' + s_1 + \ldots + s_{k-1})) ds.$$

Thus, the score distribution generated for a multiplying event is

$$\left[T(\mathbf{P}, \mathbf{P}^+) \Sigma(\mathbf{P}^+, \mathbf{P}') \epsilon_k(\mathbf{P}', \mathbf{P}_1, \ldots, \mathbf{P}_k) \left(\prod_{j=1}^{k} B_c(\mathbf{P}_j, \mathbf{P}_j^c) \right) d\mathbf{P}^+ d\mathbf{P}' \left(\prod_{j=1}^{k} d\mathbf{P}_j \right) \left(\prod_{j=1}^{k} d\mathbf{P}_j^c \right) \right]$$
$$\times \left[p_k(\mathbf{P}, \mathbf{P}', s') ds' \right] \times \left[\prod_{j}^{k-1} \psi(\mathbf{P}_j^c, t - t', s_j) ds_j \right.$$
$$\left. \times \psi(\mathbf{P}_k^c, t - t', s - (s' + s_1 + \ldots + s_{k-1})) ds \right]. \tag{2.54}$$

The History-Score Distribution Function $\psi(\mathbf{P}, t, s)$

The total score distribution function is obtained by summing the individual cases from the previous section and integrating over all phase space variables and over time from t_o to t, namely

$$\begin{aligned}
\psi(\mathbf{P}, t, s) ds &= \int_{t_o}^{t} dt' \int d\mathbf{P}^+ D(\mathbf{P}, \mathbf{P}^+, t) \int_{-\infty}^{\infty} ds' p_d(\mathbf{P}, \mathbf{P}^+, s') \delta(s - s') ds \\
&+ \int_{t_o}^{t} dt' \int d\mathbf{P}^+ T(\mathbf{P}, \mathbf{P}^+) \int d\mathbf{P}' \Sigma(\mathbf{P}^+, \mathbf{P}') A(\mathbf{P}') \int_{-\infty}^{\infty} ds' p_0(\mathbf{P}, \mathbf{P}', s') \delta(s - s') ds \\
&+ \int_{t_o}^{t} dt' \int d\mathbf{P}^+ T(\mathbf{P}, \mathbf{P}^+) R_o(\mathbf{P}^+) \int_{-\infty}^{\infty} ds' p_r(\mathbf{P}, \mathbf{P}^+, s') \\
&\quad \times \int d\mathbf{P}'' \int_0^1 d\alpha B_o(\alpha, \mathbf{P}^+, \mathbf{P}'') \psi(\mathbf{P}'', t - t', s - s') ds \\
&+ \sum_{k=2}^{\infty} \int_{t_o}^{t} dt' \int d\mathbf{P}^+ T(\mathbf{P}, \mathbf{P}^+) S_k(\mathbf{P}^+) \int_{-\infty}^{\infty} ds' p_{sk}(\mathbf{P}, \mathbf{P}^+, s') \\
&\quad \times \int d\mathbf{P}'' B_{sk}(\mathbf{P}^+, \mathbf{P}'') \int_{-\infty}^{\infty} ds_1 \ldots \int_{-\infty}^{\infty} ds_{k-1} \psi(\mathbf{P}'', t - t', s_1) \ldots \\
&\quad \times \psi(\mathbf{P}'', t - t', s_{k-1}) \psi(\mathbf{P}'', t - t', s - (s' + s_1 + \ldots + s_{k-1})) ds \\
&+ \int_{t_o}^{t} dt' \int d\mathbf{P}^+ T(\mathbf{P}, \mathbf{P}^+) \int d\mathbf{P}' \Sigma(\mathbf{P}^+, \mathbf{P}') \int_{-\infty}^{\infty} ds' p_1(\mathbf{P}, \mathbf{P}', s') \\
&\quad \times \int d\mathbf{P}'' E(\mathbf{P}', \mathbf{P}'') \int d\mathbf{P}^c B_c(\mathbf{P}'', \mathbf{P}^c) \psi(\mathbf{P}^c, t - t', s - s') ds \\
&+ \sum_{k=2}^{\infty} \int_{t_o}^{t} dt' \int d\mathbf{P}^+ T(\mathbf{P}, \mathbf{P}^+) \int d\mathbf{P}' \Sigma(\mathbf{P}^+, \mathbf{P}') \int_{-\infty}^{\infty} ds' p_k(\mathbf{P}, \mathbf{P}', s')
\end{aligned}$$

$$
\begin{aligned}
&\times \int d\mathbf{P}_1 \ldots \int d\mathbf{P}_k \epsilon_k(\mathbf{P}', \mathbf{P}_1, \ldots, \mathbf{P}_k) \\
&\times \int d\mathbf{P}_1^c B_c(\mathbf{P}_1, \mathbf{P}_1^c) \ldots \int d\mathbf{P}_k^c B_c(\mathbf{P}_k, \mathbf{P}_k^c) \\
&\times \int_{-\infty}^{\infty} ds_1 \ldots \int_{-\infty}^{\infty} ds_j \psi(\mathbf{P}_1^c, t - t', s_1) \ldots \\
&\times \psi(\mathbf{P}_{k-1}^c, t - t', s_{k-1}) \psi(\mathbf{P}_k^c, t - t', s - (s' + s_1 + \ldots + s_{k-1})) ds. \qquad (2.55)
\end{aligned}
$$

2.2.3 Derivation of the History-Score Moment Equations

The rth moment of the score distribution function can be obtained by multiplying Eq. (2.55) by s^r and integrating over all score s. Each of the seven terms (one on the left hand side of the equal sign and 6 on the right hand side of the equal sign) of Eq. (2.55) will be handled separately.

Moment of the Score Distribution

The moment of the total score distribution is defined as follows:

$$
\int_{-\infty}^{\infty} ds \; s^r \psi(\mathbf{P}, t, s) = M_r(\mathbf{P}, t). \qquad (2.56)
$$

Moment of the Streaming Term

The moment of the streaming term is obtained as follows

$$
\begin{aligned}
&\int_{t_o}^{t} dt' \int d\mathbf{P}^+ D(\mathbf{P}, \mathbf{P}^+, t) \int_{-\infty}^{\infty} ds' p_d(\mathbf{P}, \mathbf{P}^+, s') \int_{-\infty}^{\infty} ds \; s^r \delta(s - s') \\
= &\int_{t_o}^{t} dt' \int d\mathbf{P}^+ D(\mathbf{P}, \mathbf{P}^+, t) \int_{-\infty}^{\infty} ds' \; s'^r p_d(\mathbf{P}, \mathbf{P}^+, s') \\
= &\int_{t_o}^{t} dt' \int d\mathbf{P}^+ D(\mathbf{P}, \mathbf{P}^+, t) \int_{-\infty}^{\infty} ds \; s^r p_d(\mathbf{P}, \mathbf{P}^+, s). \qquad (2.57)
\end{aligned}
$$

Moment of the Absorption Term

The moment of the absorption term is obtained as follows:

$$
\begin{aligned}
&\int_{t_o}^{t} dt' \int d\mathbf{P}^+ T(\mathbf{P}, \mathbf{P}^+) \int d\mathbf{P}' \Sigma(\mathbf{P}^+, \mathbf{P}') A(\mathbf{P}') \int_{-\infty}^{\infty} ds' p_0(\mathbf{P}, \mathbf{P}', s') \int_{-\infty}^{\infty} ds \; s^r \delta(s - s') \\
= &\int_{t_o}^{t} dt' \int d\mathbf{P}^+ T(\mathbf{P}, \mathbf{P}^+) \int d\mathbf{P}' \Sigma(\mathbf{P}^+, \mathbf{P}') A(\mathbf{P}') \int_{-\infty}^{\infty} ds' s'^r p_0(\mathbf{P}, \mathbf{P}', s') \\
= &\int_{t_o}^{t} dt' \int d\mathbf{P}^+ T(\mathbf{P}, \mathbf{P}^+) \int d\mathbf{P}' \Sigma(\mathbf{P}^+, \mathbf{P}') A(\mathbf{P}') \int_{-\infty}^{\infty} ds s^r p_0(\mathbf{P}, \mathbf{P}', s). \qquad (2.58)
\end{aligned}
$$

Moment of the Russian-Roulette Term

The moment of the Russian roulette term is obtained as follows:

$$\int_{t_o}^{t} dt' \int d\mathbf{P}^+ T(\mathbf{P},\mathbf{P}^+) R_o(\mathbf{P}^+) \int_{-\infty}^{\infty} ds' p_r(\mathbf{P},\mathbf{P}^+,s') \\ \times \int d\mathbf{P}'' \int_0^1 d\alpha B_o(\alpha,\mathbf{P}^+,\mathbf{P}'') \int_{-\infty}^{\infty} ds\ s^r \psi(\mathbf{P}'',t-t',s-s')$$

Now, define $q = s - s'$ such that

$$\begin{aligned} s^r &= (q+s')^r \\ &= \sum_{n=0}^{r} \binom{r}{n} q^n s'^{r-n}. \end{aligned}$$

Insertion of q into the expression above for the moment of the Russian roulette term one obtains

$$\begin{aligned} &\int_{t_o}^{t} dt' \int d\mathbf{P}^+ T(\mathbf{P},\mathbf{P}^+) R_o(\mathbf{P}^+) \int_{-\infty}^{\infty} ds' p_r(\mathbf{P},\mathbf{P}^+,s') \\ &\qquad \times \int d\mathbf{P}'' \int_0^1 d\alpha B_o(\alpha,\mathbf{P}^+,\mathbf{P}'') \int_{-\infty}^{\infty} dq \sum_{n=0}^{r} s'^{r-n} q^n \psi(\mathbf{P}'',t-t',q) \\ = &\int_{t_o}^{t} dt' \int d\mathbf{P}^+ T(\mathbf{P},\mathbf{P}^+) R_o(\mathbf{P}^+) \int_{-\infty}^{\infty} ds' p_r(\mathbf{P},\mathbf{P}^+,s') \\ &\qquad \times \int d\mathbf{P}'' \int_0^1 d\alpha B_o(\alpha,\mathbf{P}^+,\mathbf{P}'') \sum_{n=0}^{r} \binom{r}{n} s'^{r-n} M_n(\mathbf{P}'',t-t') \\ = &\int_{t_o}^{t} dt' \int d\mathbf{P}^+ T(\mathbf{P},\mathbf{P}^+) R_o(\mathbf{P}^+) \int_{-\infty}^{\infty} ds\ p_r(\mathbf{P},\mathbf{P}^+,s) \\ &\qquad \times \int d\mathbf{P}'' \int_0^1 d\alpha B_o(\alpha,\mathbf{P}^+,\mathbf{P}'') \sum_{n=0}^{r} \binom{r}{n} s^{r-n} M_n(\mathbf{P}'',t-t'). \end{aligned} \tag{2.59}$$

Moment of the Splitting Term

The moment of the splitting term is obtained as follows:

$$\sum_{k=2}^{\infty} \int_{t_o}^{t} dt' \int d\mathbf{P}^+ T(\mathbf{P},\mathbf{P}^+) S_k(\mathbf{P}^+) \int_{-\infty}^{\infty} ds' p_{sk}(\mathbf{P},\mathbf{P}^+,s') \\ \times \int d\mathbf{P}'' B_{sk}(\mathbf{P}^+,\mathbf{P}'') \int_{-\infty}^{\infty} ds_1 \ldots \int_{-\infty}^{\infty} ds_{k-1} \psi(\mathbf{P}'',t-t',s_1)\ldots \\ \times \psi(\mathbf{P}'',t-t',s_{k-1}) \int_{-\infty}^{\infty} ds\ s^r \psi(\mathbf{P}'',t-t',s-(s'+s_1+\ldots+s_{k-1})).$$

Define $a = s' + s_1 + \ldots + s_{k-1}$ and $q = s - a$, such that

$$\begin{aligned} s^r &= (q+a)^r \\ &= \sum_{n_1=0}^{r} \binom{r}{n_1} q^{n_1} a^{r-n_1}. \end{aligned}$$

Insertion of this binomial expansion into the equation for the moment of the splitting term above, yields

$$\begin{aligned} &\sum_{k=2}^{\infty} \int_{t_o}^{t} dt' \int d\mathbf{P}^+ T(\mathbf{P}, \mathbf{P}^+) S_k(\mathbf{P}^+) \int_{-\infty}^{\infty} ds' p_{sk}(\mathbf{P}, \mathbf{P}^+, s') \\ &\quad \times \int d\mathbf{P}'' B_{sk}(\mathbf{P}^+, \mathbf{P}'') \int_{-\infty}^{\infty} ds_1 \ldots \int_{-\infty}^{\infty} ds_{k-1} \psi(\mathbf{P}'', t-t', s_1) \ldots \\ &\quad \times \psi(\mathbf{P}'', t-t', s_{k-1}) \int_{-\infty}^{\infty} dq \sum_{n_1=0}^{r} \binom{r}{n_1} a^{r-n_1} q^{n_1} \psi(\mathbf{P}'', t-t', q) \\ = &\sum_{k=2}^{\infty} \int_{t_o}^{t} dt' \int d\mathbf{P}^+ T(\mathbf{P}, \mathbf{P}^+) S_k(\mathbf{P}^+) \int_{-\infty}^{\infty} ds' p_{sk}(\mathbf{P}, \mathbf{P}^+, s') \\ &\quad \times \int d\mathbf{P}'' B_{sk}(\mathbf{P}^+, \mathbf{P}'') \int_{-\infty}^{\infty} ds_1 \ldots \int_{-\infty}^{\infty} ds_{k-1} \psi(\mathbf{P}'', t-t', s_1) \ldots \\ &\quad \times \psi(\mathbf{P}'', t-t', s_{k-1}) \sum_{n_1=0}^{r} \binom{r}{n_1} (s' + s_1 + \ldots + s_{k-1})^{r-n_1} M_{n_1}(\mathbf{P}'', t-t'). \end{aligned}$$

Next, define $b = s' + s_1 + \ldots + s_{k-2}$ so that

$$\begin{aligned} (s' + s_1 + \ldots + s_{k-1})^{r-n_1} &= (s_{k-1} + b)^{r-n_1} \\ &= \sum_{n_2=0}^{r-n_1} \binom{r-n_1}{n_2} s_{k-1}^{n_2} b^{r-n_1-n_2}, \end{aligned}$$

and insertion of this binomial expansion into expression for the moment gives

$$\begin{aligned} &\sum_{k=2}^{\infty} \int_{t_o}^{t} dt' \int d\mathbf{P}^+ T(\mathbf{P}, \mathbf{P}^+) S_k(\mathbf{P}^+) \int_{-\infty}^{\infty} ds' p_{sk}(\mathbf{P}, \mathbf{P}^+, s') \\ &\quad \times \int d\mathbf{P}'' B_{sk}(\mathbf{P}^+, \mathbf{P}'') \int_{-\infty}^{\infty} ds_1 \ldots \int_{-\infty}^{\infty} ds_{k-2} \psi(\mathbf{P}'', t-t', s_1) \ldots \psi(\mathbf{P}'', t-t', s_{k-2}) \\ &\quad \times \sum_{n_1=0}^{r} \binom{r}{n_1} M_{n_1}(\mathbf{P}'', t-t') \sum_{n_2=0}^{r-n_1} \binom{r-n_1}{n_2} b^{r-n_1-n_2} \int_{-\infty}^{\infty} ds_{k-1} s_{k-1}^{n_2} \psi(\mathbf{P}'', t-t', s_{k-1}) \\ = &\sum_{k=2}^{\infty} \int_{t_o}^{t} dt' \int d\mathbf{P}^+ T(\mathbf{P}, \mathbf{P}^+) S_k(\mathbf{P}^+) \int_{-\infty}^{\infty} ds' p_{sk}(\mathbf{P}, \mathbf{P}^+, s') \\ &\quad \times \int d\mathbf{P}'' B_{sk}(\mathbf{P}^+, \mathbf{P}'') \int_{-\infty}^{\infty} ds_1 \ldots \int_{-\infty}^{\infty} ds_{k-2} \psi(\mathbf{P}'', t-t', s_1) \ldots \psi(\mathbf{P}'', t-t', s_{k-2}) \end{aligned}$$

$$\times \sum_{n_1=0}^{r} \binom{r}{n_1} M_{n_1}(\mathbf{P}'', t-t') \sum_{n_2=0}^{r-n_1} \binom{r-n_1}{n_2} (s' + s_1 + \ldots + s_{k-2})^{r-n_1-n_2} M_{n_2}(\mathbf{P}'', t-t').$$

Repeated replacement of all further terms s_1 through s_{k-2} with binomial expansions yields

$$\begin{aligned}
&\sum_{k=2}^{\infty} \int_{t_o}^{t} dt' \int d\mathbf{P}^+ T(\mathbf{P}, \mathbf{P}^+) S_k(\mathbf{P}^+) \int_{-\infty}^{\infty} ds' p_{sk}(\mathbf{P}, \mathbf{P}^+, s') \\
&\qquad \times \int d\mathbf{P}'' B_{sk}(\mathbf{P}^+, \mathbf{P}'') \sum_{n_1=0}^{r} \binom{r}{n_1} M_{n_1}(\mathbf{P}'', t-t') \ldots \\
&\qquad \sum_{n_k=0}^{r-n_1-\ldots-n_{k-1}} \binom{r-n_1-\ldots-n_{k-1}}{n_k} M_{n_k}(\mathbf{P}'', t-t') s'^{r-n_1-\ldots-n_k}. \\
= \quad &\sum_{k=2}^{\infty} \int_{t_o}^{t} dt' \int d\mathbf{P}^+ T(\mathbf{P}, \mathbf{P}^+) S_k(\mathbf{P}^+) \int_{-\infty}^{\infty} ds \; p_{sk}(\mathbf{P}, \mathbf{P}^+, s) \\
&\qquad \times \int d\mathbf{P}'' B_{sk}(\mathbf{P}^+, \mathbf{P}'') \sum_{n_1=0}^{r} \binom{r}{n_1} M_{n_1}(\mathbf{P}'', t-t') \\
&\qquad \ldots \sum_{n_k=0}^{r-n_1-\ldots-n_{k-1}} \binom{r-n_1-\ldots-n_{k-1}}{n_k} M_{n_k}(\mathbf{P}'', t-t') s^{r-n_1-\ldots-n_k}. \qquad (2.60)
\end{aligned}$$

Moment of the Scattering Term

The moment of the scattering term is obtained as follows:

$$\begin{aligned}
&\int_{t_o}^{t} dt' \int d\mathbf{P}^+ T(\mathbf{P}, \mathbf{P}^+) \int d\mathbf{P}' \Sigma(\mathbf{P}^+, \mathbf{P}') \int_{-\infty}^{\infty} ds' p_1(\mathbf{P}, \mathbf{P}', s') \\
&\qquad \times \int d\mathbf{P}'' E(\mathbf{P}', \mathbf{P}'') \int d\mathbf{P}^c B_c(\mathbf{P}'', \mathbf{P}^c) \int_{-\infty}^{\infty} ds \; s^r \psi(\mathbf{P}^c, t-t', s-s').
\end{aligned}$$

Again, define $q = s - s'$ with

$$\begin{aligned}
s^r &= (q + s')^r \\
&= \sum_{n=0}^{r} \binom{r}{n} q^n s'^{r-n},
\end{aligned}$$

and substitute this binomial expression into the moment of the scattering term above to obtain

$$\begin{aligned}
&\int_{t_o}^{t} dt' \int d\mathbf{P}^+ T(\mathbf{P}, \mathbf{P}^+) \int d\mathbf{P}' \Sigma(\mathbf{P}^+, \mathbf{P}') \int_{-\infty}^{\infty} ds' p_1(\mathbf{P}, \mathbf{P}', s') \\
&\qquad \times \int d\mathbf{P}'' E(\mathbf{P}', \mathbf{P}'') \int d\mathbf{P}^c B_c(\mathbf{P}'', \mathbf{P}^c) \int_{-\infty}^{\infty} dq \; \sum_{n=0}^{r} \binom{r}{n} q^n s'^{r-n} \psi(\mathbf{P}^c, t-t', q)
\end{aligned}$$

$$
\begin{aligned}
&= \int_{t_o}^{t} dt' \int d\mathbf{P}^{+} T(\mathbf{P}, \mathbf{P}^{+}) \int d\mathbf{P}' \Sigma(\mathbf{P}^{+}, \mathbf{P}') \int_{-\infty}^{\infty} ds' p_1(\mathbf{P}, \mathbf{P}', s') \\
&\qquad \times \int d\mathbf{P}'' E(\mathbf{P}', \mathbf{P}'') \int d\mathbf{P}^{c} B_c(\mathbf{P}'', \mathbf{P}^{c}) \sum_{n=0}^{r} \binom{r}{n} s''^{r-n} M_n(\mathbf{P}^{c}, t - t') \\
&= \int_{t_o}^{t} dt' \int d\mathbf{P}^{+} T(\mathbf{P}, \mathbf{P}^{+}) \int d\mathbf{P}' \Sigma(\mathbf{P}^{+}, \mathbf{P}') \int_{-\infty}^{\infty} ds \; p_1(\mathbf{P}, \mathbf{P}', s) \\
&\qquad \times \int d\mathbf{P}'' E(\mathbf{P}', \mathbf{P}'') \int d\mathbf{P}^{c} B_c(\mathbf{P}'', \mathbf{P}^{c}) \sum_{n=0}^{r} \binom{r}{n} s^{r-n} M_n(\mathbf{P}^{c}, t - t'). \qquad (2.61)
\end{aligned}
$$

Moment Equation for the Multiplication Term

The moment equation for the multiplication term is obtained as follows:

$$
\begin{aligned}
&\sum_{k=2}^{\infty} \int_{t_o}^{t} dt' \int d\mathbf{P}^{+} T(\mathbf{P}, \mathbf{P}^{+}) \int d\mathbf{P}' \Sigma(\mathbf{P}^{+}, \mathbf{P}') \int_{-\infty}^{\infty} ds' p_k(\mathbf{P}, \mathbf{P}', s') \\
&\qquad \times \int d\mathbf{P}_1 \ldots \int d\mathbf{P}_k \epsilon_k(\mathbf{P}', \mathbf{P}_1, \ldots, \mathbf{P}_k) \\
&\qquad \times \int d\mathbf{P}_1^{c} B_c(\mathbf{P}_1, \mathbf{P}_1^{c}) \ldots \int d\mathbf{P}_k^{c} B_c(\mathbf{P}_k, \mathbf{P}_k^{c}) \\
&\qquad \times \int_{-\infty}^{\infty} ds_1 \ldots \int_{-\infty}^{\infty} ds_j \psi(\mathbf{P}_1^{c}, t - t', s_1) \ldots \\
&\qquad \times \psi(\mathbf{P}_{k-1}^{c}, t - t', s_{k-1}) \int_{-\infty}^{\infty} ds \; \psi(\mathbf{P}_k^{c}, t - t', s - (s' + s_1 + \ldots + s_{k-1})),
\end{aligned}
$$

where repeated application of binomial expansions to s and s_1 through s_{k-1} yields

$$
\begin{aligned}
&\sum_{k=2}^{\infty} \int_{t_o}^{t} dt' \int d\mathbf{P}^{+} T(\mathbf{P}, \mathbf{P}^{+}) \int d\mathbf{P}' \Sigma(\mathbf{P}^{+}, \mathbf{P}') \int_{-\infty}^{\infty} ds' p_k(\mathbf{P}, \mathbf{P}', s') \\
&\qquad \times \int d\mathbf{P}_1 \ldots \int d\mathbf{P}_k \epsilon_k(\mathbf{P}', \mathbf{P}_1, \ldots, \mathbf{P}_k) \\
&\qquad \times \int d\mathbf{P}_1^{c} B_c(\mathbf{P}_1, \mathbf{P}_1^{c}) \ldots \int d\mathbf{P}_k^{c} B_c(\mathbf{P}_k, \mathbf{P}_k^{c}) \\
&\qquad \times \sum_{n_1=0}^{r} M_{n_1}(\mathbf{P}'', t - t') \ldots \sum_{n_k}^{r-n_1-\ldots-n_{k-1}} M_{n_k}(\mathbf{P}'', t - t') s''^{r-n_1-\ldots-n_k} \\
&= \sum_{k=2}^{\infty} \int_{t_o}^{t} dt' \int d\mathbf{P}^{+} T(\mathbf{P}, \mathbf{P}^{+}) \int d\mathbf{P}' \Sigma(\mathbf{P}^{+}, \mathbf{P}') \int_{-\infty}^{\infty} ds \; p_k(\mathbf{P}, \mathbf{P}', s) \\
&\qquad \times \int d\mathbf{P}_1 \ldots \int d\mathbf{P}_k \epsilon_k(\mathbf{P}', \mathbf{P}_1, \ldots, \mathbf{P}_k) \\
&\qquad \times \int d\mathbf{P}_1^{c} B_c(\mathbf{P}_1, \mathbf{P}_1^{c}) \ldots \int d\mathbf{P}_k^{c} B_c(\mathbf{P}_k, \mathbf{P}_k^{c}) \sum_{n_1=0}^{r} \binom{r}{n_1} M_{n_1}(\mathbf{P}'', t - t')
\end{aligned}
$$

$$\cdots \sum_{n_k}^{r-n_1-\ldots-n_{k-1}} \binom{r-n_1-\ldots-n_{k-1}}{n_k} M_{n_k}(\mathbf{P}'', t-t') s^{r-n_1-\ldots-n_k}. \tag{2.62}$$

The Moment Equations

The moment equation is obtained by equating the moment equation for the left hand side of Eq. (2.55) to the sum of the moment equations for the terms on the right hand side, namely

$$\begin{aligned}
M_r(\mathbf{P},t) &= \int_{t_o}^{t} dt' \int d\mathbf{P}^+ D(\mathbf{P},\mathbf{P}^+,t) \int_{-\infty}^{\infty} ds\; p_d(\mathbf{P},\mathbf{P}^+,s) s^r \\
&+ \int_{t_o}^{t} dt' \int d\mathbf{P}^+ T(\mathbf{P},\mathbf{P}^+) \int d\mathbf{P}' \Sigma(\mathbf{P}^+,\mathbf{P}') A(\mathbf{P}') \int_{-\infty}^{\infty} ds p_0(\mathbf{P},\mathbf{P}',s) s^r \\
&+ \int_{t_o}^{t} dt' \int d\mathbf{P}^+ T(\mathbf{P},\mathbf{P}^+) R_o(\mathbf{P}^+) \int_{-\infty}^{\infty} ds\; p_r(\mathbf{P},\mathbf{P}^+,s) \\
&\qquad \times \int d\mathbf{P}'' \int_0^1 d\alpha B_o(\alpha,\mathbf{P}^+,\mathbf{P}'') \sum_{n=0}^{r} \binom{r}{n} M_n(\mathbf{P}'',t-t') s^{r-n} \\
&+ \sum_{k=2}^{\infty} \int_{t_o}^{t} dt' \int d\mathbf{P}^+ T(\mathbf{P},\mathbf{P}^+) S_k(\mathbf{P}^+) \int_{-\infty}^{\infty} ds\; p_{sk}(\mathbf{P},\mathbf{P}^+,s) \\
&\qquad \times \int d\mathbf{P}'' B_{sk}(\mathbf{P}^+,\mathbf{P}'') \sum_{n_1=0}^{r} \binom{r}{n_1} M_{n_1}(\mathbf{P}'',t-t') \\
&\qquad \cdots \sum_{n_k=0}^{r-n_1-\ldots-n_{k-1}} \binom{r-n_1-\ldots-n_{k-1}}{n_k} M_{n_k}(\mathbf{P}'',t-t') s^{r-n_1-\ldots-n_k} \\
&+ \int_{t_o}^{t} dt' \int d\mathbf{P}^+ T(\mathbf{P},\mathbf{P}^+) \int d\mathbf{P}' \Sigma(\mathbf{P}^+,\mathbf{P}') \int_{-\infty}^{\infty} ds\; p_1(\mathbf{P},\mathbf{P}',s) \\
&\qquad \times \int d\mathbf{P}'' E(\mathbf{P}',\mathbf{P}'') \int d\mathbf{P}^c B_c(\mathbf{P}'',\mathbf{P}^c) \sum_{n=0}^{r} \binom{r}{n} M_n(\mathbf{P}^c,t-t') s^{r-n} \\
&+ \sum_{k=2}^{\infty} \int_{t_o}^{t} dt' \int d\mathbf{P}^+ T(\mathbf{P},\mathbf{P}^+) \int d\mathbf{P}' \Sigma(\mathbf{P}^+,\mathbf{P}') \int_{-\infty}^{\infty} ds\; p_k(\mathbf{P},\mathbf{P}',s) \\
&\qquad \times \int d\mathbf{P}_1 \ldots \int d\mathbf{P}_k \epsilon_k(\mathbf{P}',\mathbf{P}_1,\ldots,\mathbf{P}_k) \\
&\qquad \times \int d\mathbf{P}_1^c B_c(\mathbf{P}_1,\mathbf{P}_1^c) \ldots \int d\mathbf{P}_k^c B_c(\mathbf{P}_k,\mathbf{P}_k^c) \sum_{n_1=0}^{r} \binom{r}{n_1} M_{n_1}(\mathbf{P}'',t-t') \\
&\qquad \cdots \sum_{n_k}^{r-n_1-\ldots-n_{k-1}} \binom{r-n_1-\ldots-n_{k-1}}{n_k} M_{n_k}(\mathbf{P}'',t-t') s^{r-n_1-\ldots-n_k}.
\end{aligned} \tag{2.63}$$

This equation can be somewhat simplified by extraction of all terms that do not have an $M_r(\mathbf{P}'', t-t')$ dependence and their combination into a single term $Q_r(\mathbf{P},t)$. This equation is given by Booth and

Cashwell, is

$$
\begin{aligned}
M_r(\mathbf{P},t) &= \int_{t_o}^{t} dt' \int d\mathbf{P}^{+} T(\mathbf{P},\mathbf{P}^{+}) R_o(\mathbf{P}^{+}) \int d\mathbf{P}'' \int_0^1 d\alpha B_o(\alpha,\mathbf{P}^{+},\mathbf{P}'') M_r(\mathbf{P}'',t-t') \\
&+ \sum_{k=2}^{\infty} \int_{t_o}^{t} dt' \int d\mathbf{P}^{+} T(\mathbf{P},\mathbf{P}^{+}) S_k(\mathbf{P}^{+}) \int d\mathbf{P}'' B_{sk}(\mathbf{P}^{+},\mathbf{P}'') k M_r(\mathbf{P}'',t-t') \\
&+ \int_{t_o}^{t} dt' \int d\mathbf{P}^{+} T(\mathbf{P},\mathbf{P}^{+}) \int d\mathbf{P}' \Sigma(\mathbf{P}^{+},\mathbf{P}') \int d\mathbf{P}'' E(\mathbf{P}',\mathbf{P}'') \\
&\qquad \times \int d\mathbf{P}^{c} B_c(\mathbf{P}'',\mathbf{P}^{c}) M_r(\mathbf{P}^{c},t-t') \\
&+ \sum_{k=2}^{\infty} \int_{t_o}^{t} dt' \int d\mathbf{P}^{+} T(\mathbf{P},\mathbf{P}^{+}) \int d\mathbf{P}' \Sigma(\mathbf{P}^{+},\mathbf{P}') \\
&\qquad \times \int d\mathbf{P}_1 \ldots \int d\mathbf{P}_k \epsilon_k(\mathbf{P}',\mathbf{P}_1,\ldots,\mathbf{P}_k) \int d\mathbf{P}_1^{c} B_c(\mathbf{P}_1,\mathbf{P}_1^{c}) \ldots \int d\mathbf{P}_k^{c} B_c(\mathbf{P}_k,\mathbf{P}_k^{c}) \\
&\qquad \times \sum_{j=1}^{k} M_r(\mathbf{P}'',t-t') \\
&+ Q_r(\mathbf{P},t), \qquad (2.64)
\end{aligned}
$$

where

$$
\begin{aligned}
Q_r(\mathbf{P},t) &= \int_{t_o}^{t} dt' \int d\mathbf{P}^{+} D(\mathbf{P},\mathbf{P}^{+},t) \int_{-\infty}^{\infty} ds\; p_d(\mathbf{P},\mathbf{P}^{+},s) s^r \\
&+ \int_{t_o}^{t} dt' \int d\mathbf{P}^{+} T(\mathbf{P},\mathbf{P}^{+}) \int d\mathbf{P}' \Sigma(\mathbf{P}^{+},\mathbf{P}') A(\mathbf{P}') \int_{-\infty}^{\infty} ds p_0(\mathbf{P},\mathbf{P}',s) s^r \\
&+ \int_{t_o}^{t} dt' \int d\mathbf{P}^{+} T(\mathbf{P},\mathbf{P}^{+}) R_o(\mathbf{P}^{+}) \int_{-\infty}^{\infty} ds\; p_r(\mathbf{P},\mathbf{P}^{+},s) \\
&\qquad \times \int d\mathbf{P}'' \int_0^1 d\alpha B_o(\alpha,\mathbf{P}^{+},\mathbf{P}'') \sum_{n=0}^{r-1} \binom{r}{n} M_n(\mathbf{P}'',t-t') s^{r-n} \\
&+ \sum_{k=2}^{\infty} \int_{t_o}^{t} dt' \int d\mathbf{P}^{+} T(\mathbf{P},\mathbf{P}^{+}) S_k(\mathbf{P}^{+}) \int_{-\infty}^{\infty} ds\; p_{sk}(\mathbf{P},\mathbf{P}^{+},s) \\
&\qquad \times \int d\mathbf{P}'' B_{sk}(\mathbf{P}^{+},\mathbf{P}'') \sum_{n_1=0}^{r} \binom{r}{n_1} M_{n_1}(\mathbf{P}'',t-t') \\
&\qquad \cdots \sum_{n_k=0}^{r-n_1-\ldots-n_{k-1}} \binom{r-n_1-\ldots-n_{k-1}}{n_k} M_{n_k}(\mathbf{P}'',t-t') s^{r-n_1-\ldots-n_k} \prod_{j=1}^{k} (1-\delta_{n_j,r}) \\
&+ \int_{t_o}^{t} dt' \int d\mathbf{P}^{+} T(\mathbf{P},\mathbf{P}^{+}) \int d\mathbf{P}' \Sigma(\mathbf{P}^{+},\mathbf{P}') \int_{-\infty}^{\infty} ds\; p_1(\mathbf{P},\mathbf{P}',s) \\
&\qquad \times \int d\mathbf{P}'' E(\mathbf{P}',\mathbf{P}'') \int d\mathbf{P}^{c} B_c(\mathbf{P}'',\mathbf{P}^{c}) \sum_{n=0}^{r-1} \binom{r}{n} M_n(\mathbf{P}^{c},t-t') s^{r-n}
\end{aligned}
$$

$$
\begin{aligned}
&+ \sum_{k=2}^{\infty} \int_{t_o}^{t} dt' \int d\mathbf{P}^{+} T(\mathbf{P},\mathbf{P}^{+}) \int d\mathbf{P}' \Sigma(\mathbf{P}^{+},\mathbf{P}') \int_{-\infty}^{\infty} ds\; p_k(\mathbf{P},\mathbf{P}',s) \\
&\quad \times \int d\mathbf{P}_1 \ldots \int d\mathbf{P}_k \epsilon_k(\mathbf{P}',\mathbf{P}_1,\ldots,\mathbf{P}_k) \\
&\quad \times \int d\mathbf{P}_1^c B_c(\mathbf{P}_1,\mathbf{P}_1^c) \ldots \int d\mathbf{P}_k^c B_c(\mathbf{P}_k,\mathbf{P}_k^c) \sum_{n_1=0}^{r} \binom{r}{n_1} M_{n_1}(\mathbf{P}'',t-t') \\
&\quad \ldots \sum_{n_k}^{r-n_1-\ldots-n_{k-1}} \binom{r-n_1-\ldots-n_{k-1}}{n_k} M_{n_k}(\mathbf{P}'',t-t') s^{r-n_1-\ldots-n_k} \\
&\quad \times \prod_{j=1}^{k} (1-\delta n_j, r). \qquad (2.65)
\end{aligned}
$$

Generally, one is interested in only the first and second moments. For the special case that $r = 1$, the source term becomes [Booth and Cashwell, 1979]

$$
\begin{aligned}
Q_1(\mathbf{P},t) &= \int_{t_o}^{t} dt' \int d\mathbf{P}^{+} D(\mathbf{P},\mathbf{P}^{+},t) \int_{-\infty}^{\infty} ds\; p_d(\mathbf{P},\mathbf{P}^{+},s) s \\
&+ \int_{t_o}^{t} dt' \int d\mathbf{P}^{+} T(\mathbf{P},\mathbf{P}^{+}) \int d\mathbf{P}' \Sigma(\mathbf{P}^{+},\mathbf{P}') A(\mathbf{P}') \int_{-\infty}^{\infty} ds p_0(\mathbf{P},\mathbf{P}',s) s \\
&+ \int_{t_o}^{t} dt' \int d\mathbf{P}^{+} T(\mathbf{P},\mathbf{P}^{+}) R_o(\mathbf{P}^{+}) \int_{-\infty}^{\infty} ds\; p_r(\mathbf{P},\mathbf{P}^{+},s) s \\
&+ \sum_{k=2}^{\infty} \int_{t_o}^{t} dt' \int d\mathbf{P}^{+} T(\mathbf{P},\mathbf{P}^{+}) S_k(\mathbf{P}^{+}) \int_{-\infty}^{\infty} ds\; p_{sk}(\mathbf{P},\mathbf{P}^{+},s) s \\
&+ \int_{t_o}^{t} dt' \int d\mathbf{P}^{+} T(\mathbf{P},\mathbf{P}^{+}) \int d\mathbf{P}' \Sigma(\mathbf{P}^{+},\mathbf{P}') \int_{-\infty}^{\infty} ds\; p_1(\mathbf{P},\mathbf{P}',s) \\
&\qquad \times \int d\mathbf{P}'' E(\mathbf{P}',\mathbf{P}'') s \\
&+ \sum_{k=2}^{\infty} \int_{t_o}^{t} dt' \int d\mathbf{P}^{+} T(\mathbf{P},\mathbf{P}^{+}) \int d\mathbf{P}' \Sigma(\mathbf{P}^{+},\mathbf{P}') \int_{-\infty}^{\infty} ds\; p_k(\mathbf{P},\mathbf{P}',s) \\
&\qquad \times \int d\mathbf{P}_1 \ldots \int d\mathbf{P}_k \epsilon_k(\mathbf{P}',\mathbf{P}_1,\ldots,\mathbf{P}_k) s, \qquad (2.66)
\end{aligned}
$$

and for the case $r = 2$ the source term becomes [Booth and Cashwell, 1979]

$$
\begin{aligned}
Q_2(\mathbf{P},t) &= \int_{t_o}^{t} dt' \int d\mathbf{P}^{+} D(\mathbf{P},\mathbf{P}^{+},t) \int_{-\infty}^{\infty} ds\; p_d(\mathbf{P},\mathbf{P}^{+},s) s^2 \\
&+ \int_{t_o}^{t} dt' \int d\mathbf{P}^{+} T(\mathbf{P},\mathbf{P}^{+}) \int d\mathbf{P}' \Sigma(\mathbf{P}^{+},\mathbf{P}') A(\mathbf{P}') \int_{-\infty}^{\infty} ds p_0(\mathbf{P},\mathbf{P}',s) s^2 \\
&+ \int_{t_o}^{t} dt' \int d\mathbf{P}^{+} T(\mathbf{P},\mathbf{P}^{+}) R_o(\mathbf{P}^{+}) \int_{-\infty}^{\infty} ds\; p_r(\mathbf{P},\mathbf{P}^{+},s)
\end{aligned}
$$

$$
\begin{aligned}
&\times \int d\mathbf{P}'' \int_0^1 d\alpha B_o(\alpha, \mathbf{P}^+, \mathbf{P}'') \sum_{n=0}^{1} \binom{2}{n} M_n(\mathbf{P}'', t-t') s^{2-n} \\
+ &\sum_{k=2}^{\infty} \int_{t_o}^{t} dt' \int d\mathbf{P}^+ T(\mathbf{P}, \mathbf{P}^+) S_k(\mathbf{P}^+) \int_{-\infty}^{\infty} ds\ p_{sk}(\mathbf{P}, \mathbf{P}^+, s) \\
&\times \int d\mathbf{P}'' B_{sk}(\mathbf{P}^+, \mathbf{P}'') \left[s^2 + 2skM_1(\mathbf{P}'', t-t') + k(k-1) M_1^2(\mathbf{P}'', t-t') \right] \\
+ &\int_{t_o}^{t} dt' \int d\mathbf{P}^+ T(\mathbf{P}, \mathbf{P}^+) \int d\mathbf{P}' \Sigma(\mathbf{P}^+, \mathbf{P}') \int_{-\infty}^{\infty} ds\ p_1(\mathbf{P}, \mathbf{P}', s) \\
&\times \int d\mathbf{P}'' E(\mathbf{P}', \mathbf{P}'') \int d\mathbf{P}^c B_c(\mathbf{P}'', \mathbf{P}^c) \sum_{n=0}^{1} \binom{2}{n} M_n(\mathbf{P}^c, t-t') s^{2-n} \\
+ &\sum_{k=2}^{\infty} \int_{t_o}^{t} dt' \int d\mathbf{P}^+ T(\mathbf{P}, \mathbf{P}^+) \int d\mathbf{P}' \Sigma(\mathbf{P}^+, \mathbf{P}') \int_{-\infty}^{\infty} ds\ p_k(\mathbf{P}, \mathbf{P}', s) \\
&\times \int d\mathbf{P}_1 \ldots \int d\mathbf{P}_k \epsilon_k(\mathbf{P}', \mathbf{P}_1, \ldots, \mathbf{P}_k) \\
&\times \int d\mathbf{P}_1^c B_c(\mathbf{P}_1, \mathbf{P}_1^c) \ldots \int d\mathbf{P}_k^c B_c(\mathbf{P}_k, \mathbf{P}_k^c) \\
&\ldots \left[s^2 + 2s \sum_{j=1}^{k} M_1(\mathbf{P}_j^c, t-t') + 2 \sum_{j=2}^{k} M_1(\mathbf{P}_j^c, t-t') \sum_{l=1}^{j-1} M_1(\mathbf{P}_j^c, t-t'). \right] \qquad (2.67)
\end{aligned}
$$

2.3 The Integro-Differential form of the Moment Equations

Appendix B demonstrates that the differential removal operator $\mathcal{L}$ defined as

$$
\mathcal{L} = \frac{1}{v} \frac{\partial}{\partial t} + \mathbf{\Omega} \cdot \boldsymbol{\nabla} + \Sigma(\mathbf{r}, E),
$$

is inverse to the integral transition operator

$$
\mathcal{T} = \int d\mathbf{P}' T(\mathbf{P}', \mathbf{P}),
$$

such that, for a function $F(\mathbf{P})$,

$$
\begin{aligned}
\left[\frac{1}{v} \frac{\partial}{\partial t} + \mathbf{\Omega} \cdot \boldsymbol{\nabla} + \Sigma(\mathbf{r}, E) \right] \int d\mathbf{P}' T(\mathbf{P}', \mathbf{P}) F(\mathbf{P}') &= \mathcal{L} \left\{ \mathcal{T} \ [F(\mathbf{P})] \right\} \\
&= F(\mathbf{P}).
\end{aligned}
$$

It is known that the adjoint integral transition operator is [Bell and Glasstone, 1970]

$$
\mathcal{T}^\dagger = \int d\mathbf{P}' T(\mathbf{P}, \mathbf{P}'), \qquad (2.68)
$$

where the order of the phase-space variables in the kernel are interchanged. Similarly, the adjoint removal operator is known to be Bell and Glasstone [1970]

$$\mathcal{L}^{\dagger} = -\frac{1}{v}\frac{\partial}{\partial t} - \mathbf{\Omega}\cdot\nabla + \Sigma(\mathbf{r}, E). \tag{2.69}$$

It follows that, for an arbitrary function $F^{\dagger}(\mathbf{P})$,

$$\begin{aligned}\left[-\frac{1}{v}\frac{\partial}{\partial t} - \mathbf{\Omega}\cdot\nabla + \Sigma(\mathbf{r}, E)\right]\int d\mathbf{P}' T(\mathbf{P}, \mathbf{P}')F^{\dagger}(\mathbf{P}') &= \mathcal{L}^{\dagger}\left\{\mathcal{T}^{\dagger}\left[F^{\dagger}(\mathbf{P})\right]\right\} \\ &= F^{\dagger}(\mathbf{P}).\end{aligned} \tag{2.70}$$

The integro-differential moment equation Eq. (2.64) was shown to be adjoint to the linearized Boltzmann transport equation by Amster and Djomehri [1976]. Therefore, the integro-differential form of the history-score moment equations may be obtained by operation on Eq. (2.64) with $\mathcal{L}^{\dagger}$. Keep in mind that $\mathbf{P}$ in Appendix B is defined to be $(\mathbf{r}, \mathbf{\Omega}, E, t)$, although for the moment equations $\mathbf{P}$ is defined to be $(\mathbf{r}, \mathbf{\Omega}, E, w)$. Then operation on the moment equations with the adjoint removal operator gives

$$\begin{aligned}&-\frac{1}{v}\frac{dM_r}{dt} - \mathbf{\Omega}\cdot\nabla M_r(\mathbf{r}, \mathbf{\Omega}, E, t) + \Sigma(\mathbf{r}, E)M_r(\mathbf{r}, \mathbf{\Omega}, E, t) \\
&= \int dw^+\delta(w^+ - \omega_t w)\int d\mathbf{R}^+\delta(\mathbf{R}^+ - \mathbf{R})R_o(\mathbf{P}^+)\int d\mathbf{P}''\int_0^1 d\alpha B_o(\alpha, \mathbf{P}^+, \mathbf{P}'')M_r(\mathbf{P}'', t) \\
&+ \sum_{k=2}^{\infty}\int dw^+\delta(w^+ - \omega_t w)\int d\mathbf{R}^+\delta(\mathbf{R}^+ - \mathbf{R})S_k(\mathbf{P}^+)\int d\mathbf{P}'' B_{sk}(\mathbf{P}^+, \mathbf{P}'')kM_r(\mathbf{P}'', t) \\
&+ \int dw^+\delta(w^+ - \omega_t w)\int d\mathbf{R}^+\delta(\mathbf{R}^+ - \mathbf{R})\int d\mathbf{P}'\Sigma(\mathbf{P}^+, \mathbf{P}')\int d\mathbf{P}'' E(\mathbf{P}', \mathbf{P}'') \\
&\qquad \times \int d\mathbf{P}^c B_c(\mathbf{P}'', \mathbf{P}^c)M_r(\mathbf{P}^c, t) \\
&+ \sum_{k=2}^{\infty}\int dw^+\delta(w^+ - \omega_t w)\int d\mathbf{R}^+\delta(\mathbf{R}^+ - \mathbf{R})\int d\mathbf{P}'\Sigma(\mathbf{P}^+, \mathbf{P}') \\
&\qquad \times \int d\mathbf{P}_1 \ldots \int d\mathbf{P}_k \epsilon_k(\mathbf{P}', \mathbf{P}_1, \ldots, \mathbf{P}_k)\int d\mathbf{P}_1^c B_c(\mathbf{P}_1, \mathbf{P}_1^c)\ldots\int d\mathbf{P}_k^c B_c(\mathbf{P}_k, \mathbf{P}_k^c) \\
&\qquad \times \sum_{j=1}^{k} M_r(\mathbf{P}'', t) \\
&+ \widehat{Q}_r(\mathbf{P}, t),\end{aligned} \tag{2.71}$$

where

$$\begin{aligned}\widehat{Q}_r(\mathbf{P}, t) &= \int dw^+\delta(w^+ - \omega_t w)\int d\mathbf{R}^+\delta(\mathbf{R}^+ - \mathbf{R})\int_{-\infty}^{\infty} ds\; p_d(\mathbf{P}, \mathbf{P}^+, s)s^r \\
&+ \int dw^+\delta(w^+ - \omega_t w)\int d\mathbf{R}^+\delta(\mathbf{R}^+ - \mathbf{R})\int d\mathbf{P}'\Sigma(\mathbf{P}^+, \mathbf{P}')A(\mathbf{P}')\int_{-\infty}^{\infty} ds p_0(\mathbf{P}, \mathbf{P}', s)s^r \\
&+ \int dw^+\delta(w^+ - \omega_t w)R_o(\mathbf{P}^+)\int d\mathbf{R}^+\delta(\mathbf{R}^+ - \mathbf{R})\int_{-\infty}^{\infty} ds\; p_r(\mathbf{P}, \mathbf{P}^+, s)\end{aligned}$$

$$
\begin{aligned}
&\times \int d\mathbf{P}'' \int_0^1 d\alpha B_o(\alpha, \mathbf{P}^+, \mathbf{P}'') \sum_{n=0}^{r-1} \binom{r}{n} M_n(\mathbf{P}'', t) s^{r-n} \\
+ \quad & \sum_{k=2}^{\infty} \int dw^+ \delta(w^+ - \omega_t w) \int d\mathbf{R}^+ \delta(\mathbf{R}^+ - \mathbf{R}) S_k(\mathbf{P}^+) \int_{-\infty}^{\infty} ds\ p_{sk}(\mathbf{P}, \mathbf{P}^+, s) \\
&\times \int d\mathbf{P}'' B_{sk}(\mathbf{P}^+, \mathbf{P}'') \sum_{n_1=0}^{r} \binom{r}{n_1} M_{n_1}(\mathbf{P}'', t) \\
&\cdots \sum_{n_k=0}^{r-n_1-\ldots-n_{k-1}} \binom{r-n_1-\ldots-n_{k-1}}{n_k} M_{n_k}(\mathbf{P}'', t) s^{r-n_1-\ldots-n_k} \prod_{j=1}^{k} (1-\delta_{n_j,r}) \\
+ \quad & \int dw^+ \delta(w^+ - \omega_t w) \int d\mathbf{R}^+ \delta(\mathbf{R}^+ - \mathbf{R}) \int d\mathbf{P}' \Sigma(\mathbf{P}^+, \mathbf{P}') \int_{-\infty}^{\infty} ds\ p_1(\mathbf{P}, \mathbf{P}', s) \\
&\times \int d\mathbf{P}'' E(\mathbf{P}', \mathbf{P}'') \int d\mathbf{P}^c B_c(\mathbf{P}'', \mathbf{P}^c) \sum_{n=0}^{r-1} \binom{r}{n} M_n(\mathbf{P}^c, t) s^{r-n} \\
+ \quad & \sum_{k=2}^{\infty} \int dw^+ \delta(w^+ - \omega_t w) \int d\mathbf{R}^+ \delta(\mathbf{R}^+ - \mathbf{R}) \int d\mathbf{P}' \Sigma(\mathbf{P}^+, \mathbf{P}') \int_{-\infty}^{\infty} ds\ p_k(\mathbf{P}, \mathbf{P}', s) \\
&\times \int d\mathbf{P}_1 \ldots \int d\mathbf{P}_k \epsilon_k(\mathbf{P}', \mathbf{P}_1, \ldots, \mathbf{P}_k) \\
&\times \int d\mathbf{P}_1^c B_c(\mathbf{P}_1, \mathbf{P}_1^c) \ldots \int d\mathbf{P}_k^c B_c(\mathbf{P}_k, \mathbf{P}_k^c) \sum_{n_1=0}^{r} \binom{r}{n_1} M_{n_1}(\mathbf{P}'', t) \\
&\cdots \sum_{n_k}^{r-n_1-\ldots-n_{k-1}} \binom{r-n_1-\ldots-n_{k-1}}{n_k} M_{n_k}(\mathbf{P}'', t) s^{r-n_1-\ldots-n_k} \\
&\times \prod_{j=1}^{k} (1 - \delta n_j, r). \qquad (2.72)
\end{aligned}
$$

The integro-differential first and second moment sources are, respectively,

$$
\begin{aligned}
\widehat{Q}_1(\mathbf{P}, t) \quad = \quad & \int dw^+ \delta(w^+ - \omega_t w) \int d\mathbf{R}^+ \delta(\mathbf{R}^+ - \mathbf{R}) \int_{-\infty}^{\infty} ds\ p_d(\mathbf{P}, \mathbf{P}^+, s) s \\
+ \quad & \int dw^+ \delta(w^+ - \omega_t w) \int d\mathbf{R}^+ \delta(\mathbf{R}^+ - \mathbf{R}) \int d\mathbf{P}' \Sigma(\mathbf{P}^+, \mathbf{P}') A(\mathbf{P}') \int_{-\infty}^{\infty} ds p_0(\mathbf{P}, \mathbf{P}', s) s \\
+ \quad & \int dw^+ \delta(w^+ - \omega_t w) \int d\mathbf{R}^+ \delta(\mathbf{R}^+ - \mathbf{R}) R_o(\mathbf{P}^+) \int_{-\infty}^{\infty} ds\ p_r(\mathbf{P}, \mathbf{P}^+, s) s \\
+ \quad & \sum_{k=2}^{\infty} \int dw^+ \delta(w^+ - \omega_t w) \int d\mathbf{R}^+ \delta(\mathbf{R}^+ - \mathbf{R}) S_k(\mathbf{P}^+) \int_{-\infty}^{\infty} ds\ p_{sk}(\mathbf{P}, \mathbf{P}^+, s) s \\
+ \quad & \int dw^+ \delta(w^+ - \omega_t w) \int d\mathbf{R}^+ \delta(\mathbf{R}^+ - \mathbf{R}) \int d\mathbf{P}' \Sigma(\mathbf{P}^+, \mathbf{P}') \int_{-\infty}^{\infty} ds\ p_1(\mathbf{P}, \mathbf{P}', s) \\
&\times \int d\mathbf{P}'' E(\mathbf{P}', \mathbf{P}'') s
\end{aligned}
$$

$$+ \sum_{k=2}^{\infty} \int dw^+ \delta(w^+ - \omega_t w) \int d\mathbf{R}^+ \delta(\mathbf{R}^+ - \mathbf{R}) \int d\mathbf{P}' \Sigma(\mathbf{P}^+, \mathbf{P}') \int_{-\infty}^{\infty} ds\ p_k(\mathbf{P}, \mathbf{P}', s)$$
$$\times \int d\mathbf{P}_1 \ldots \int d\mathbf{P}_k \epsilon_k(\mathbf{P}', \mathbf{P}_1, \ldots, \mathbf{P}_k) s, \tag{2.73}$$

and

$$\begin{aligned}
\widehat{Q}_2(\mathbf{P}, t) &= \int dw^+ \delta(w^+ - \omega_t w) \int d\mathbf{R}^+ \delta(\mathbf{R}^+ - \mathbf{R}) \int_{-\infty}^{\infty} ds\ p_d(\mathbf{P}, \mathbf{P}^+, s) s^2 \\
&+ \int dw^+ \delta(w^+ - \omega_t w) \int d\mathbf{R}^+ \delta(\mathbf{R}^+ - \mathbf{R}) \int d\mathbf{P}' \Sigma(\mathbf{P}^+, \mathbf{P}') A(\mathbf{P}') \int_{-\infty}^{\infty} ds p_0(\mathbf{P}, \mathbf{P}', s) s^2 \\
&+ \int dw^+ \delta(w^+ - \omega_t w) \int d\mathbf{R}^+ \delta(\mathbf{R}^+ - \mathbf{R}) R_o(\mathbf{P}^+) \int_{-\infty}^{\infty} ds\ p_r(\mathbf{P}, \mathbf{P}^+, s) \\
&\quad \times \int d\mathbf{P}'' \int_0^1 d\alpha B_o(\alpha, \mathbf{P}^+, \mathbf{P}'') \sum_{n=0}^{1} \binom{2}{n} M_n(\mathbf{P}'', t) s^{2-n} \\
&+ \sum_{k=2}^{\infty} \int dw^+ \delta(w^+ - \omega_t w) \int d\mathbf{R}^+ \delta(\mathbf{R}^+ - \mathbf{R}) S_k(\mathbf{P}^+) \int_{-\infty}^{\infty} ds\ p_{sk}(\mathbf{P}, \mathbf{P}^+, s) \\
&\quad \times \int d\mathbf{P}'' B_{sk}(\mathbf{P}^+, \mathbf{P}'') \left[s^2 + 2sk M_1(\mathbf{P}'', t) + k(k-1) M_1^2(\mathbf{P}'', t) \right] \\
&+ \int dw^+ \delta(w^+ - \omega_t w) \int d\mathbf{R}^+ \delta(\mathbf{R}^+ - \mathbf{R}) \int d\mathbf{P}' \Sigma(\mathbf{P}^+, \mathbf{P}') \int_{-\infty}^{\infty} ds\ p_1(\mathbf{P}, \mathbf{P}', s) \\
&\quad \times \int d\mathbf{P}'' E(\mathbf{P}', \mathbf{P}'') \int d\mathbf{P}^c B_c(\mathbf{P}'', \mathbf{P}^c) \sum_{n=0}^{1} \binom{2}{n} M_n(\mathbf{P}^c, t) s^{2-n} \\
&+ \sum_{k=2}^{\infty} \int dw^+ \delta(w^+ - \omega_t w) \int d\mathbf{R}^+ \delta(\mathbf{R}^+ - \mathbf{R}) \int d\mathbf{P}' \Sigma(\mathbf{P}^+, \mathbf{P}') \int_{-\infty}^{\infty} ds\ p_k(\mathbf{P}, \mathbf{P}', s) \\
&\quad \times \int d\mathbf{P}_1 \ldots \int d\mathbf{P}_k \epsilon_k(\mathbf{P}', \mathbf{P}_1, \ldots, \mathbf{P}_k) \\
&\quad \times \int d\mathbf{P}_1^c B_c(\mathbf{P}_1, \mathbf{P}_1^c) \ldots \int d\mathbf{P}_k^c B_c(\mathbf{P}_k, \mathbf{P}_k^c) \\
&\quad \ldots \left[s^2 + 2s \sum_{j=1}^{k} M_1(\mathbf{P}_j^c, t) + 2 \sum_{j=2}^{k} M_1(\mathbf{P}_j^c, t) \sum_{l=1}^{j-1} M_1(\mathbf{P}_j^c, t) \right]
\end{aligned} \tag{2.74}$$

The moment equations have been cast in an integro-differential form so that common solution methods in neutral particle transport can be applied. Specifically this work is interested in the application of the discrete ordinates (S_n) solution method discussed in Chapter 3. Of special interest in the integro-differential form is the integration over weight

$$\int dw^+ \ \delta(w^+ - \omega_t w), \tag{2.75}$$

preceeding each term. The differential removal operator is inverse to the integral transition operator for phase space $(\mathbf{r}, \mathbf{\Omega}, E, t)$, but not w. In practice, the only variance reduction method that

continuously modifies particle weight during transition is the exponential transform. The changes in weight resulting from exponential transform can be accounted for in the deterministic solution of the moment equations. However, such a deterministic solution requires that the spatial differencing (discussed in Chapter 3) become increasingly small as the magnitude of the exponential transform stretching parameter p approaches unity. Because the iteration time increases considerably with an increase in the spatial resolution, the application of the S_n method to optimizing exponential transform parameters is prohibited by the cost of a single calculation.

2.4 Weight-Separable Cases of the *r*th Moment

In certain instances, the weight-dependence of the rth moment is separable Booth and Cashwell [1979]. In particular, the weight-dependence is separable when exclusively weight-independent variance reduction games are employed. When only weight-independent variance reduction is utilized, the rth moment can be expressed as

$$M_r(\mathbf{P}, \mathbf{\Omega}, E, w) = w^r M_r(\mathbf{P}, \mathbf{\Omega}, E, \underline{w = 1}). \tag{2.76}$$

With Eq. (2.76), the first moment can be expressed as

$$M_1(\mathbf{P}, \mathbf{\Omega}, E, w) = w M_1(\mathbf{P}, \mathbf{\Omega}, E, \underline{w = 1}). \tag{2.77}$$

The first moment, but not higher moments, always scales as Eq. (2.77), even for weight-dependent variance reduction. If the first moment did not scale with weight, the expected score a particle would make would not either.

With Eq. (2.76), the second moment can be expressed as

$$M_2(\mathbf{P}, \mathbf{\Omega}, E, w) = w^2 M_2(\mathbf{P}, \mathbf{\Omega}, E, \underline{w = 1}). \tag{2.78}$$

Using exclusively weight-independent variance reduction greatly simplifies the calculation by removing the necessity to treat the weight dependence explicitly. With a solution of both the first and second moments for $w = 1$, the moments for other weights can be calculated by scaling the moment for $w = 1$. It is this scaling relationship that the work by Juzaitis [1982] and Sarkar and Prasad [1979] used to perform optimizations. This work solves the moments using weight-dependent variance reduction, specifically the weight-window variance reduction technique, and cannot assume the scaling relationships. Rather, the weight-dependence of the moments must be resolved explicitly.

2.5 Extension of the Moment Equations to Weight-Window Variance Reduction

When Booth and Cashwell [1979] developed the history-score moment equations, just derived in the preceding section, the weight window technique had not yet been invented. Therefore, the moment equations that describe the effects of the weight window had not been developed. The following section derives the moment equation for the weight window variance reduction method.

2.5.1 Deduction of the Weight-Window Kernel

Before deduction of the weight window kernel, a couple of definitions are required. Along with ω_l, ω_s, ω_u, and K defined in Section 2.1.4, the following probabilities, kernels, and functions are defined:

- $\mathbf{P}$ = phase space $(\mathbf{r}, \mathbf{\Omega}, E, w, t)$
- $S_k(\mathbf{P})d\mathbf{P}$ = the probability that a particle at $\mathbf{P}$ will undergo a $k:1$ split
- $C(\mathbf{P})d\mathbf{P}$ = the probability that a particle at $\mathbf{P}$ crosses a weight window surface
- $B_{sk}(\mathbf{P}, \mathbf{P}')d\mathbf{P}'$ = the probability that a particle at $\mathbf{P}$ experiencing a $k:1$ split ends up in $d\mathbf{P}'$ about $\mathbf{P}'$
- $B_r(\mathbf{P}, \mathbf{P}'; \alpha)d\mathbf{P}'$ = The probability that a particle at $\mathbf{P}$ will undergo Russian roulette and end up in $d\mathbf{P}'$ about $\mathbf{P}'$ given a survival probability of α
- $\Theta(w)$ = the Heaviside step function such that

$$\Theta(w) = \begin{cases} 0, & w < 0 \\ 1, & w \geq 0 \end{cases}$$

- $p_{ws}(\mathbf{P}, \mathbf{P}', s)ds$ = the probability that a particle contributes a score in ds about s before undergoing the weight window check after crossing a weight window surface
- $p_{wc}(\mathbf{P}, \mathbf{P}', s)ds$ = the probability that a particle contributes a score in ds about s before undergoing a collision

Note that although these probabilities, kernels, and functions are written generally in terms of $\mathbf{P}$, the only variable affected by the weight window kernel is w. Furthermore, ω_l, ω_s, and ω_u generally depend on the phase space location, but this dependence has been omitted for brevity and is implicitly assumed.

When the weight is less than ω_s/K then the particle undergoes Russian roulette with a survival probability of $1/K$. This term may be expressed as

$$[1 - \Theta(w - \omega_s/K)]\, B_r(\mathbf{P}, \mathbf{P}'; 1/K)dw. \tag{2.79}$$

If the weight is greater than ω_s/K and less than ω_l, then the Russian roulette term is

$$\Theta(w - \omega_s/K)\,[1 - \Theta(w - \omega_l)]\, B_r(\mathbf{P}, \mathbf{P}'; w/\omega_s). \tag{2.80}$$

If the particle is inside the window then the weight of the particle remains unchanged. Thus, the term that describes the transition of particles inside the window is

$$\Theta(w - \omega_l)\,[1 - \Theta(w - \omega_u)]\,\delta(\mathbf{P}' - \mathbf{P}). \tag{2.81}$$

For particles with weights greater than the upper window bound but less than K times the same bound, the particles are split $\lceil w/\omega_u \rceil : 1$. Thus, the term for particles having these weights is

$$\Theta(w-\omega_u)\left[1-\Theta(w-K\omega_u)\right]\sum_{k=2}^{K} S_k(\mathbf{P})B_{sk}(\mathbf{P},\mathbf{P}'), \tag{2.82}$$

where

$$S_k(\mathbf{P}) = \begin{cases} 1, & k = \left\lceil \frac{w}{\omega_u} \right\rceil \\ 0, & \text{otherwise} \end{cases} \tag{2.83}$$

Finally, particles having weights in excess of $K\omega_u$ undergo a $K : 1$ split regardless of their weight, and the corresponding term is

$$\Theta(w-K\omega_u)B_{sK}(\mathbf{P},\mathbf{P}'). \tag{2.84}$$

Assembling all of the terms above, one arrives at the weight window kernel, namely

$$\begin{aligned} B_w(\mathbf{P},\mathbf{P}') &= \left[1-\Theta(w-\omega_s/K)\right]B_r(\mathbf{P},\mathbf{P}';1/K) \\ &+ \Theta(w-\omega_s/K)\left[1-\Theta(w-\omega_l)\right]B_r(\mathbf{P},\mathbf{P}';w/\omega_s) \\ &+ \Theta(w-\omega_l)\left[1-\Theta(w-\omega_u)\right]\delta(\mathbf{P}'-\mathbf{P}) + \\ &+ \sum_{k=2}^{K-1}\Theta(w-(k-1)\omega_u)\left[1-\Theta(w-k\omega_u)\right]S_k(\mathbf{P})B_{sk}(\mathbf{P},\mathbf{P}') \\ &+ \Theta(w-K\omega_u)B_{sK}(\mathbf{P},\mathbf{P}'). \end{aligned} \tag{2.85}$$

2.5.2 Derivation of Weight Window Moment Equations

The moment equations that result from the weight-window kernel can be derived. Only a time-independent case, $\mathbf{P} = (\mathbf{r}, \mathbf{\Omega}, E, w)$, without physical multiplication, namely (n,f), is considered for simplicity. Furthermore, all scoring and kernels considered by Booth and Cashwell [1979] are be ignored, and only the effects of weight windows are considered.

First, the history-score distribution function $\psi(\mathbf{P}, s)$ for the weight-windows is derived. Three cases are possible for weight windows because they may be checked at surface crossings or collision:

1. the particle IS NOT crossing a surface where weight windows are checked
2. the particle IS crossing a surface where weight windows are checked
3. the particle collides and weight windows are checked after the collision

The three possibilities above indicate that $\psi(\mathbf{P}, s)ds$ will have three terms and can be written as

$$\begin{aligned} \psi(\mathbf{P},s)ds &= \int d\mathbf{P}^+ T(\mathbf{P},\mathbf{P}^+)[1-C(\mathbf{P}^+)]\psi(\mathbf{P}^+,s)ds \\ &+ \int d\mathbf{P}^+ T(\mathbf{P},\mathbf{P}^+)C(\mathbf{P}^+)\int_{-\infty}^{\infty} ds' p_{ws}(\mathbf{P},\mathbf{P}^+,s')\int d\mathbf{P}' B_w(\mathbf{P}^+,\mathbf{P}')\psi(\mathbf{P}',s-s')ds \end{aligned}$$

$$+ \int d\mathbf{P}^{+} T(\mathbf{P}, \mathbf{P}^{+}) \int d\mathbf{P}' \Sigma(\mathbf{P}^{+}, \mathbf{P}') \int_{-\infty}^{\infty} ds' p_{wc}(\mathbf{P}, \mathbf{P}', s') \int d\mathbf{P}' E(\mathbf{P}', \mathbf{P}'')$$
$$\times \int d\mathbf{P}'' B_w(\mathbf{P}'', \mathbf{P}^c) \psi(\mathbf{P}^c, s - s') ds. \tag{2.86}$$

The first term on the right side of the equation above describes particles that move from $\mathbf{P}$ to $\mathbf{P}^{+}$, do not cross a weight window surface nor collide, and continue from that point. The second term on the right hand side of the equation describes particles that move from $\mathbf{P}$ to $\mathbf{P}^{+}$, cross a weight window surface, potentially contributing a score, and then undergo the weight window game. The third term describes particles that move from $\mathbf{P}$ to $\mathbf{P}^{+}$, collide and exit in $\mathbf{P}'$ and as a result of that collision scatter into $\mathbf{P}''$, potentially contributing a score, and then undergo the weight window game. This expression is not completely accurate because splitting may result from the weight window games. To see the effect of splitting the weight window kernel must be expanded into its five components, so, from this point on, only the term

$$\int d\mathbf{P}' B_w(\mathbf{P}^{+}, \mathbf{P}') \psi(\mathbf{P}', s - s') ds, \tag{2.87}$$

is considered. This term arises in both the surface and collisional weight window terms.

Expansion of the weight window kernel term gives

$$\begin{aligned}
&\int d\mathbf{P}' B_w(\mathbf{P}, \mathbf{P}') \psi(\mathbf{P}', s - s') ds \\
= \;& [1 - \Theta(w - \omega_s/K)] \int d\mathbf{P}' B_r(\mathbf{P}, \mathbf{P}'; 1/K) \psi(\mathbf{P}', s - s') \\
+ \;& \Theta(w - \omega_s/K) [1 - \Theta(w - \omega_l)] \int d\mathbf{P}' B_r(\mathbf{P}, \mathbf{P}'; w/\omega_s) \psi(\mathbf{P}', s - s') \\
+ \;& \Theta(w - \omega_l) [1 - \Theta(w - \omega_u)] \int d\mathbf{P}' \delta(\mathbf{P}' - \mathbf{P}) \psi(\mathbf{P}', s - s') \\
+ \;& \Theta(w - \omega_u) [1 - \Theta(w - K\omega_u)] \sum_{k=2}^{K} S_k(\mathbf{P}) \int d\mathbf{P}' B_{sk}(\mathbf{P}, \mathbf{P}') \\
& \quad \times \prod_{j=1}^{k-1} \int_{-\infty}^{\infty} ds_j \psi(\mathbf{P}', s_j) ds_j \times \psi(\mathbf{P}', s - (s' + s_1 + \ldots + s_{k-1})) ds \\
+ \;& \Theta(w - K\omega_u) \int d\mathbf{P}' B_{sK}(\mathbf{P}, \mathbf{P}') \prod_{j=1}^{k-1} \int_{-\infty}^{\infty} ds_j \psi(\mathbf{P}', s_j) ds_j \\
& \quad \times \psi(\mathbf{P}', s - (s' + s_1 + \ldots + s_{k-1})) ds.
\end{aligned} \tag{2.88}$$

The product terms originate from the fact that of the k particles produced in a split the first $k-1$ of these particles will produces scores in ds_j about s_j where $j = 1, \ldots, k-1$. The kth particle from the split must contribute a score $s - (s' + s_1 + \ldots + s_{k-1})$ for the entire score to sum to s, where s' was the score, if any, generated at the surface crossing or collision where the weight window game is occurring.

Now, the moments of each of these expressions can be determined, where the rth moment M_r is defined to be

$$M_r(\mathbf{P}) = \int_{-\infty}^{\infty} ds \, s^r \psi(\mathbf{P}, s). \tag{2.89}$$

Multiplication of the terms above by s^r and integrating over all s, yields

$$
\begin{aligned}
& [1-\Theta(w-\omega_s/K)] \int d\mathbf{P}' B_r(\mathbf{P},\mathbf{P}';1/K) \int_{-\infty}^{\infty} ds\, s^r \psi(\mathbf{P}', s-s') \\
+ \;& \Theta(w-\omega_s/K)\,[1-\Theta(w-\omega_l)] \int d\mathbf{P}' B_r(\mathbf{P},\mathbf{P}';w/\omega_s) \int_{-\infty}^{\infty} ds\, s^r \psi(\mathbf{P}', s-s') \\
+ \;& \Theta(w-\omega_l)\,[1-\Theta(w-\omega_u)] \int d\mathbf{P}' \delta(\mathbf{P}'-\mathbf{P}) \int_{-\infty}^{\infty} ds\, s^r \psi(\mathbf{P}', s-s') \\
+ \;& \Theta(w-\omega_u)\,[1-\Theta(w-K\omega_u)] \sum_{k=2}^{K} S_k(\mathbf{P}) \int d\mathbf{P}' B_{sk}(\mathbf{P},\mathbf{P}') \\
& \qquad \times \prod_{j=1}^{k-1} \int_{-\infty}^{\infty} ds_j\, \psi(\mathbf{P}', s_j) ds_j \times \int_{-\infty}^{\infty} ds\, s^r \psi(\mathbf{P}', s-(s'+s_1+\ldots+s_{k-1})) \\
+ \;& \Theta(w-K\omega_u) \int d\mathbf{P}' B_{sK}(\mathbf{P},\mathbf{P}') \prod_{j=1}^{K-1} \int_{-\infty}^{\infty} ds_j\, \psi(\mathbf{P}', s_j) ds_j \\
& \qquad \times \int_{-\infty}^{\infty} ds\, s^r \psi(\mathbf{P}', s-(s'+s_1+\ldots+s_{K-1})).
\end{aligned} \tag{2.90}
$$

The first three terms of the above expression can be simplified with the substitution $q = s - s'$, so that

$$
s^r = (q+s')^r = \sum_{n=0}^{r} \binom{r}{n} q^n s'^{r-n}. \tag{2.91}
$$

With this substitution, these three terms become

$$
\begin{aligned}
& [1-\Theta(w-\omega_s/K)] \int d\mathbf{P}' B_r(\mathbf{P},\mathbf{P}';1/K) \int_{-\infty}^{\infty} dq \sum_{n=0}^{r} \binom{r}{n} s'^{r-n} q^n \psi(\mathbf{P}', q) \\
+ \;& \Theta(w-\omega_s/K)\,[1-\Theta(w-\omega_l)] \int d\mathbf{P}' B_r(\mathbf{P},\mathbf{P}';w/\omega_s) \int_{-\infty}^{\infty} dq \sum_{n=0}^{r} \binom{r}{n} s'^{r-n} q^n \psi(\mathbf{P}', q) \\
+ \;& \Theta(w-\omega_l)\,[1-\Theta(w-\omega_u)] \int d\mathbf{P}' \delta(\mathbf{P}'-\mathbf{P}) \int_{-\infty}^{\infty} dq \sum_{n=0}^{r} \binom{r}{n} s'^{r-n} q^n \psi(\mathbf{P}', q),
\end{aligned} \tag{2.92}
$$

or

$$
\begin{aligned}
& [1-\Theta(w-\omega_s/K)] \int d\mathbf{P}' B_r(\mathbf{P},\mathbf{P}';1/K) \sum_{n=0}^{r} \binom{r}{n} s'^{r-n} M_n(\mathbf{P}') \\
+ \;& \Theta(w-\omega_s/K)\,[1-\Theta(w-\omega_l)] \int d\mathbf{P}' B_r(\mathbf{P},\mathbf{P}';w/\omega_s) \sum_{n=0}^{r} \binom{r}{n} s'^{r-n} M_n(\mathbf{P}') \\
+ \;& \Theta(w-\omega_l)\,[1-\Theta(w-\omega_u)] \int d\mathbf{P}' \delta(\mathbf{P}'-\mathbf{P}) \sum_{n=0}^{r} \binom{r}{n} s'^{r-n} M_n \psi(\mathbf{P}').
\end{aligned} \tag{2.93}
$$

The remaining two terms can be simplified by use of $a = s' + s_1 + \ldots + s_{k-1}$ and $q = s - a$, so that

$$\begin{aligned} s^r &= (q+a)^r \\ &= \sum_{n_1=0}^{r} \binom{r}{n_1} q^{n_1} a^{r-n_1}. \end{aligned} \tag{2.94}$$

Insertion of this binomial expansion into the remaining two terms produces

$$\begin{aligned} & \Theta(w-\omega_u)\left[1-\Theta(w-K\omega_u)\right] \sum_{k=2}^{K} S_k(\mathbf{P}) \int d\mathbf{P}' B_{sk}(\mathbf{P},\mathbf{P}') \\ & \quad \times \prod_{j=1}^{k-1} \int_{-\infty}^{\infty} ds_j\, \psi(\mathbf{P}', s_j) ds_j \times \int_{-\infty}^{\infty} dq \sum_{n_1=0}^{r} \binom{r}{n_1} a^{r-n_1} q^{n_1} \psi(\mathbf{P}', q) \\ + \quad & \Theta(w-K\omega_u) \int d\mathbf{P}' B_{sK}(\mathbf{P},\mathbf{P}') \prod_{j=1}^{K-1} \int_{-\infty}^{\infty} ds_j\, \psi(\mathbf{P}', s_j) ds_j \\ & \quad \times \int_{-\infty}^{\infty} dq \sum_{n_1=0}^{r} \binom{r}{n_1} a^{r-n_1} q^{n_1} \psi(\mathbf{P}', q), \end{aligned} \tag{2.95}$$

or

$$\begin{aligned} & \Theta(w-\omega_u)\left[1-\Theta(w-K\omega_u)\right] \sum_{k=2}^{K} S_k(\mathbf{P}) \int d\mathbf{P}' B_{sk}(\mathbf{P},\mathbf{P}') \\ & \quad \times \prod_{j=1}^{k-1} \int_{-\infty}^{\infty} ds_j\, \psi(\mathbf{P}', s_j) ds_j \times \sum_{n_1=0}^{r} \binom{r}{n_1} (s' + s_1 + \ldots + s_{k-1})^{r-n_1} M_{n_1}(\mathbf{P}') \\ + \quad & \Theta(w-K\omega_u) \int d\mathbf{P}' B_{sK}(\mathbf{P},\mathbf{P}') \prod_{j=1}^{K-1} \int_{-\infty}^{\infty} ds_j\, \psi(\mathbf{P}', s_j) ds_j \\ & \quad \times \sum_{n_1=0}^{r} \binom{r}{n_1} (s' + s_1 + \ldots + s_{K-1})^{r-n_1} M_{n_1}(\mathbf{P}'). \end{aligned} \tag{2.96}$$

Next, define $b = s' + s_1 + \ldots + s_{k-2}$, so that

$$(s' + s_1 + \ldots + s_{k-1})^{r-n_1} = (s_{k-1} + b)^{r-n_1} = \sum_{n_2=0}^{r-n_1} \binom{r-n_1}{n_2} s_{k-1}^{n_2} b^{r-n_1-n_2}. \tag{2.97}$$

Insertion of this binomial expansion into the two terms gives

$$\Theta(w-\omega_u)\left[1-\Theta(w-K\omega_u)\right] \sum_{k=2}^{K} S_k(\mathbf{P}) \int d\mathbf{P}' B_{sk}(\mathbf{P},\mathbf{P}')$$

$$\times \prod_{j=1}^{k-2} \int_{-\infty}^{\infty} ds_j \psi(\mathbf{P}', s_j) ds_j \times \sum_{n_1=0}^{r} \binom{r}{n_1} M_{n_1}(\mathbf{P}')$$
$$\times \sum_{n_2=0}^{r-n_1} \binom{r-n_1}{n_2} (s' + s_1 + \ldots + s_{k-2})^{r-n_1-n_2} M_{n_2}(\mathbf{P}')$$
$$+ \quad \Theta(w - K\omega_u) \int d\mathbf{P}' B_{sK}(\mathbf{P}, \mathbf{P}') \prod_{j=1}^{K-2} \int_{-\infty}^{\infty} ds_j \psi(\mathbf{P}', s_j) ds_j$$
$$\times \sum_{n_1=0}^{r} \binom{r}{n_1} M_{n_1}(\mathbf{P}') \sum_{n_2=0}^{r-n_1} \binom{r-n_1}{n_2} (s' + s_1 + \ldots + s_{K-2})^{r-n_1-n_2} M_{n_2}(\mathbf{P}'). \tag{2.98}$$

Repeated application of similar binomial expansions gives, in the end,

$$\Theta(w - \omega_u) [1 - \Theta(w - K\omega_u)] \sum_{k=2}^{K} S_k(\mathbf{P}) \int d\mathbf{P}' B_{sk}(\mathbf{P}, \mathbf{P}')$$
$$\times \sum_{n_1=0}^{r} \binom{r}{n_1} M_{n_1}(\mathbf{P}') \ldots \sum_{n_k=0}^{r-n_1-\ldots-n_{k-1}} \binom{r-n_1-\ldots-n_{k-1}}{n_k} M_{n_k}(\mathbf{P}') s'^{r-n_1-\ldots-n_k}$$
$$+ \quad \Theta(w - K\omega_u) \int d\mathbf{P}' B_{sK}(\mathbf{P}, \mathbf{P}') \sum_{n_1=0}^{r} \binom{r}{n_1} M_{n_1}(\mathbf{P}') \ldots$$
$$\sum_{n_K=0}^{r-n_1-\ldots-n_{K-1}} \binom{r-n_1-\ldots-n_{K-1}}{n_K} M_{n_K}(\mathbf{P}') s'^{r-n_1-\ldots-n_K}. \tag{2.99}$$

Finally, the terms for the weight window moment equation can be combined to obtain

$$[1 - \Theta(w - \omega_s/K)] \int d\mathbf{P}' B_r(\mathbf{P}, \mathbf{P}'; 1/K) \sum_{n=0}^{r} \binom{r}{n} s'^{r-n} M_n(\mathbf{P}')$$
$$+ \quad \Theta(w - \omega_s/K) [1 - \Theta(w - \omega_l)] \int d\mathbf{P}' B_r(\mathbf{P}, \mathbf{P}'; w/\omega_s) \sum_{n=0}^{r} \binom{r}{n} s'^{r-n} M_n(\mathbf{P}')$$
$$+ \quad \Theta(w - \omega_l) [1 - \Theta(w - \omega_u)] \int d\mathbf{P}' \delta(\mathbf{P}' - \mathbf{P}) \sum_{n=0}^{r} \binom{r}{n} s'^{r-n} M_n(\mathbf{P}')$$
$$+ \quad \Theta(w - \omega_u) [1 - \Theta(w - K\omega_u)] \sum_{k=2}^{K} S_k(\mathbf{P}) \int d\mathbf{P}' B_{sk}(\mathbf{P}, \mathbf{P}')$$
$$\times \sum_{n_1=0}^{r} \binom{r}{n_1} M_{n_1}(\mathbf{P}') \ldots \sum_{n_k=0}^{r-n_1-\ldots-n_{k-1}} \binom{r-n_1-\ldots-n_{k-1}}{n_k} M_{n_k}(\mathbf{P}') s'^{r-n_1-\ldots-n_k}$$
$$+ \quad \Theta(w - K\omega_u) \int d\mathbf{P}' B_{sK}(\mathbf{P}, \mathbf{P}') \sum_{n_1=0}^{r} \binom{r}{n_1} M_{n_1}(\mathbf{P}') \ldots$$

$$\sum_{n_K=0}^{r-n_1-\ldots-n_{K-1}} \binom{r-n_1-\ldots-n_{K-1}}{n_K} M_{n_K}(\mathbf{P}') s'^{r-n_1-\ldots-n_K}. \tag{2.100}$$

With the procedure of Booth and Cashwell [1979], the terms involving only the rth moment can be separated from the rest, namely,

$$\begin{aligned}
& [1-\Theta(w-\omega_s/K)] \int d\mathbf{P}' B_r(\mathbf{P},\mathbf{P}';1/K) M_r(\mathbf{P}') \\
+ \;& \Theta(w-\omega_s/K)\,[1-\Theta(w-\omega_l)] \int d\mathbf{P}' B_r(\mathbf{P},\mathbf{P}';w/\omega_s) M_r(\mathbf{P}') \\
+ \;& \Theta(w-\omega_l)\,[1-\Theta(w-\omega_u)] \int d\mathbf{P}' \delta(\mathbf{P}'-\mathbf{P}) M_r(\mathbf{P}') \\
+ \;& \Theta(w-\omega_u)\,[1-\Theta(w-K\omega_u)] \sum_{k=2}^{K} S_k(\mathbf{P}) \int d\mathbf{P}' B_{sk}(\mathbf{P},\mathbf{P}') k M_r(\mathbf{P}') \\
+ \;& \Theta(w-K\omega_u) \int d\mathbf{P}' B_{sK}(\mathbf{P},\mathbf{P}') K M_r(\mathbf{P}'). \\
+ \;& Q_r(\mathbf{P}),
\end{aligned} \tag{2.101}$$

where

$$\begin{aligned}
Q_r(\mathbf{P}) \;=\; & [1-\Theta(w-\omega_s/K)] \int d\mathbf{P}' B_r(\mathbf{P},\mathbf{P}';1/K) \sum_{n=0}^{r-1} \binom{r}{n} s'^{r-n} M_n(\mathbf{P}') \\
+ \;& \Theta(w-\omega_s/K)\,[1-\Theta(w-\omega_l)] \int d\mathbf{P}' B_r(\mathbf{P},\mathbf{P}';w/\omega_s) \sum_{n=0}^{r-1} \binom{r}{n} s'^{r-n} M_n(\mathbf{P}') \\
+ \;& \Theta(w-\omega_l)\,[1-\Theta(w-\omega_u)] \int d\mathbf{P}' \delta(\mathbf{P}'-\mathbf{P}) \sum_{n=0}^{r-1} \binom{r}{n} s'^{r-n} M_n(\mathbf{P}') \\
+ \;& \Theta(w-\omega_u)\,[1-\Theta(w-K\omega_u)] \sum_{k=2}^{K} S_k(\mathbf{P}) \int d\mathbf{P}' B_{sk}(\mathbf{P},\mathbf{P}') \\
& \quad \times \sum_{n_1=0}^{r} \binom{r}{n_1} M_{n_1}(\mathbf{P}') \ldots \sum_{n_k=0}^{r-n_1-\ldots-n_{k-1}} \binom{r-n_1-\ldots-n_{k-1}}{n_k} M_{n_k}(\mathbf{P}') s'^{r-n_1-\ldots-n_k} \\
& \quad \times \prod_{i=1}^{k} (1-\delta_{n_i r}) \\
+ \;& \Theta(w-K\omega_u) \int d\mathbf{P}' B_{sK}(\mathbf{P},\mathbf{P}') \sum_{n_1=0}^{r} \binom{r}{n_1} M_{n_1}(\mathbf{P}') \ldots \\
& \quad \sum_{n_K=0}^{r-n_1-\ldots-n_{K-1}} \binom{r-n_1-\ldots-n_{K-1}}{n_K} M_{n_K}(\mathbf{P}') s'^{r-n_1-\ldots-n_K} \prod_{i=1}^{K} (1-\delta_{n_i r}).
\end{aligned} \tag{2.102}$$

For the special case that $r = 1$, one has

$$\begin{aligned} Q_1(\mathbf{P}') &= [1 - \Theta(w - \omega_s/K)]\, s' \\ &+ \Theta(w - \omega_s/K)\,[1 - \Theta(w - \omega_l)]\, s' \\ &+ \Theta(w - \omega_l)\,[1 - \Theta(w - \omega_u)]\, s' \\ &+ \Theta(w - \omega_u)\,[1 - \Theta(w - K\omega_u)] \sum_{k=2}^{K} S_k(\mathbf{P}) s' \\ &+ \Theta(w - K\omega_u) s' \\ &= s' \end{aligned} \tag{2.103}$$

and for $r = 2$

$$\begin{aligned} Q_2(\mathbf{P}) &= [1 - \Theta(w - \omega_s/K)] \int d\mathbf{P}' B_r(\mathbf{P}, \mathbf{P}'; 1/K) \sum_{n=0}^{1} \binom{2}{n} M_n(\mathbf{P}') s'^{2-n} \\ &+ \Theta(w - \omega_s/K)\,[1 - \Theta(w - \omega_l)] \int d\mathbf{P}' B_r(\mathbf{P}, \mathbf{P}'; w/\omega_s) \sum_{n=0}^{1} \binom{2}{n} M_n(\mathbf{P}') s'^{2-n} \\ &+ \Theta(w - \omega_l)\,[1 - \Theta(w - \omega_u)] \int d\mathbf{P}' \delta(\mathbf{P}' - \mathbf{P}) \sum_{n=0}^{1} \binom{2}{n} M_n(\mathbf{P}') s'^{2-n} \\ &+ \Theta(w - \omega_u)\,[1 - \Theta(w - K\omega_u)] \sum_{k=2}^{K} S_k(\mathbf{P}) \int d\mathbf{P}' B_{sk}(\mathbf{P}, \mathbf{P}') \\ &\qquad \times [s'^2 + 2s'kM_1(\mathbf{P}') + k(k-1)M_1^2(\mathbf{P}')] \\ &+ \Theta(w - K\omega_u) \int d\mathbf{P}' B_{sK}(\mathbf{P}, \mathbf{P}')[s'^2 + 2s'kM_1(\mathbf{P}') + k(k-1)M_1^2(\mathbf{P}')] \end{aligned} \tag{2.104}$$

Now, with insertion of these derived terms into Eq. (2.86), one obtains the full weight-window moment equation

$$\begin{aligned} M_r(\mathbf{P}) &= \int d\mathbf{P}^+ T(\mathbf{P}, \mathbf{P}^+)[1 - C(\mathbf{P}^+)] M_r(\mathbf{P}^+) \\ &+ \int d\mathbf{P}^+ T(\mathbf{P}, \mathbf{P}^+) C(\mathbf{P}^+) \Big\{ [1 - \Theta(w^+ - \omega_s/K)] \int d\mathbf{P}' B_r(\mathbf{P}^+, \mathbf{P}'; 1/K) M_r(\mathbf{P}') \\ &\quad + \Theta(w^+ - \omega_s/K)\,[1 - \Theta(w^+ - \omega_l)] \int d\mathbf{P}' B_r(\mathbf{P}^+, \mathbf{P}'; w^+/\omega_s) M_r(\mathbf{P}') \\ &\quad + \Theta(w^+ - \omega_l)\,[1 - \Theta(w^+ - \omega_u)] \int d\mathbf{P}' \delta(\mathbf{P}' - \mathbf{P}^+) M_r(\mathbf{P}') \\ &\quad + \Theta(w^+ - \omega_u)\,[1 - \Theta(w^+ - K\omega_u)] \sum_{k=2}^{K} S_k(\mathbf{P}) \int d\mathbf{P}' B_{sk}(\mathbf{P}^+, \mathbf{P}') k M_r(\mathbf{P}') \\ &\quad + \Theta(w^+ - K\omega_u) \int d\mathbf{P}' B_{sK}(\mathbf{P}^+, \mathbf{P}') K M_r(\mathbf{P}') \Big\} \end{aligned}$$

$$
\begin{aligned}
&+ \int d\mathbf{P}^+ T(\mathbf{P},\mathbf{P}^+) \int d\mathbf{P}' \Sigma(\mathbf{P}^+,\mathbf{P}') \int d\mathbf{P}'' E(\mathbf{P}',\mathbf{P}'') \\
&\quad \times \Big\{ [1-\Theta(w''-\omega_s/K)] \int d\mathbf{P}^c B_r(\mathbf{P}'',\mathbf{P}^c;1/K) M_r(\mathbf{P}^c) \\
&\qquad + \Theta(w''-\omega_s/K)\,[1-\Theta(w''-\omega_l)] \int d\mathbf{P}^c B_r(\mathbf{P}'',\mathbf{P}^c;w''/\omega_s) M_r(\mathbf{P}^c) \\
&\qquad + \Theta(w''-\omega_l)\,[1-\Theta(w''-\omega_u)] \int d\mathbf{P}^c \delta(\mathbf{P}^c-\mathbf{P}'') M_r(\mathbf{P}^c) \\
&\qquad + \Theta(w''-\omega_u)\,[1-\Theta(w''-K\omega_u)] \sum_{k=2}^{K} S_k(\mathbf{P}) \int d\mathbf{P}^c B_{sk}(\mathbf{P}'',\mathbf{P}^c) k M_r(\mathbf{P}^c) \\
&\qquad + \Theta(w''-K\omega_u) \int d\mathbf{P}^c B_{sK}(\mathbf{P}'',\mathbf{P}^c) K M_r(\mathbf{P}^c) \Big\} \\
&+ Q_r(\mathbf{P}). \qquad (2.105)
\end{aligned}
$$

The full source terms Q_1 and Q_2 are, with the dummy variable s' replaced by s, then

$$
\begin{aligned}
Q_1(\mathbf{P}) &= \int d\mathbf{P}^+ T(\mathbf{P},\mathbf{P}^+) C(\mathbf{P}^+) \int_{-\infty}^{\infty} ds\, p_{ws}(\mathbf{P},\mathbf{P}^+,s) s \\
&+ \int d\mathbf{P}^+ T(\mathbf{P},\mathbf{P}^+) \int d\mathbf{P}' \Sigma(\mathbf{P}^+,\mathbf{P}') \int_{-\infty}^{\infty} ds\, p_{wc}(\mathbf{P},\mathbf{P}',s) \int d\mathbf{P}' E(\mathbf{P}',\mathbf{P}'') s \qquad (2.106)
\end{aligned}
$$

and

$$
\begin{aligned}
Q_2(\mathbf{P}) &= \int d\mathbf{P}^+ T(\mathbf{P},\mathbf{P}^+) C(\mathbf{P}^+) \int_{-\infty}^{\infty} ds\, p_{ws}(\mathbf{P},\mathbf{P}^+,s) \\
&\quad \times \Big\{ [1-\Theta(w^+-\omega_s/K)] \int d\mathbf{P}' B_r(\mathbf{P}^+,\mathbf{P}';1/K) \sum_{n=0}^{1} \binom{2}{n} M_n(\mathbf{P}') s^{2-n} \\
&\qquad + \Theta(w^+-\omega_s/K)\,[1-\Theta(w^+-\omega_l)] \int d\mathbf{P}' B_r(\mathbf{P}^+,\mathbf{P}';w^+/\omega_s) \sum_{n=0}^{1} \binom{2}{n} M_n(\mathbf{P}') s^{2-n} \\
&\qquad + \Theta(w^+-\omega_l)\,[1-\Theta(w^+-\omega_u)] \int d\mathbf{P}' \delta(\mathbf{P}'-\mathbf{P}^+) \sum_{n=0}^{1} \binom{2}{n} M_n(\mathbf{P}') s^{2-n} \\
&\qquad + \Theta(w^+-\omega_u)\,[1-\Theta(w^+-K\omega_u)] \sum_{k=2}^{K} S_k(\mathbf{P}^+) \int d\mathbf{P}' B_{sk}(\mathbf{P}^+,\mathbf{P}') \\
&\qquad\quad \times [s^2 + 2skM_1(\mathbf{P}') + k(k-1)M_1^2(\mathbf{P}')] \\
&\qquad + \Theta(w^+-K\omega_u) \int d\mathbf{P}' B_{sK}(\mathbf{P}^+,\mathbf{P}') [s^2 + 2skM_1(\mathbf{P}') + k(k-1)M_1^2(\mathbf{P}')] \Big\} \\
&+ \int d\mathbf{P}^+ T(\mathbf{P},\mathbf{P}^+) \int d\mathbf{P}' \Sigma(\mathbf{P}^+,\mathbf{P}') \int_{-\infty}^{\infty} ds'\, p_{wc}(\mathbf{P},\mathbf{P}',s') \int d\mathbf{P}' E(\mathbf{P}',\mathbf{P}'') \\
&\quad \times \Big\{ [1-\Theta(w''-\omega_s/K)] \int d\mathbf{P}^c B_r(\mathbf{P}'',\mathbf{P}^c;1/K) \sum_{n=0}^{1} \binom{2}{n} M_n(\mathbf{P}^c) s^{2-n}
\end{aligned}
$$

$$
\begin{aligned}
&+ \quad \Theta(w'' - \omega_s/K)\left[1 - \Theta(w'' - \omega_l)\right] \int d\mathbf{P}^c B_r(\mathbf{P}'', \mathbf{P}^c; w''/\omega_s) \sum_{n=0}^{1} \binom{2}{n} M_n(\mathbf{P}^c) s^{2-n} \\
&+ \quad \Theta(w'' - \omega_l)\left[1 - \Theta(w'' - \omega_u)\right] \int d\mathbf{P}^c \delta(\mathbf{P}^c - \mathbf{P}'') \sum_{n=0}^{1} \binom{2}{n} M_n(\mathbf{P}^c) s^{2-n} \\
&+ \quad \Theta(w'' - \omega_u)\left[1 - \Theta(w'' - K\omega_u)\right] \sum_{k=2}^{K} S_k(\mathbf{P}'') \int d\mathbf{P}^c B_{sk}(\mathbf{P}'', \mathbf{P}^c) \\
&\qquad \times [s^2 + 2skM_1(\mathbf{P}^c) + k(k-1)M_1^2(\mathbf{P}^c)] \\
&+ \quad \Theta(w'' - K\omega_u) \int d\mathbf{P}^c B_{sK}(\mathbf{P}'', \mathbf{P}^c)[s^2 + 2skM_1(\mathbf{P}^c) + k(k-1)M_1^2(\mathbf{P}^c)] \Bigg\} . \quad (2.107)
\end{aligned}
$$

Chapter 3

Discrete Ordinates Solution of the History Score Moment Equations

This chapter describes the method by which the history-score moment equations are solved. Solutions are obtained by using a multigroup adjoint discrete ordinates (S_n) solution with the additional new discretization of the Monte Carlo particle weight. As shown, this new discretization can lead to many difficulties with the solution's accuracy and is probably an area where future work sould be directed.

The following sections provide a brief outline of the basic S_n discretization procedure and the multi-group approximation. Then, the discretization of the Monte Carlo particle weight is discussed. Finally, how different Monte Carlo variance reduction games affect the weight domain of the deterministic calcualation are discussed and shown graphically.

3.1 Discrete-Ordinates Discretization of the History-Score Moment Equations

This discretization begins by considering Eq. (2.71) for the time-independent, non-multiplying case with the addition of the weight-window terms developed in Section 2.5, namely[1]

$$\begin{aligned}
& -\mathbf{\Omega}\cdot\nabla M_r(\mathbf{r},\mathbf{\Omega},E) + \Sigma(\mathbf{r},E)M_r(\mathbf{r},\mathbf{\Omega},E) \\
= & \int dw^+\delta(w^+-\omega_t w)\int d\mathbf{R}^+\delta(\mathbf{R}^+-\mathbf{R})R_o(\mathbf{P}^+)\int d\mathbf{P}''\int_0^1 d\alpha B_o(\alpha,\mathbf{P}^+,\mathbf{P}'')M_r(\mathbf{P}'') \\
+ & \sum_{k=2}^{\infty}\int dw^+\delta(w^+-\omega_t w)\int d\mathbf{R}^+\delta(\mathbf{R}^+-\mathbf{R})S_k(\mathbf{P}^+)\int d\mathbf{P}'' B_{sk}(\mathbf{P}^+,\mathbf{P}'')kM_r(\mathbf{P}'') \\
+ & \int dw^+\delta(w^+-\omega_t w)\int d\mathbf{R}^+\delta(\mathbf{R}^+-\mathbf{R})\int d\mathbf{P}'\Sigma(\mathbf{P}^+,\mathbf{P}')\int d\mathbf{P}'' E(\mathbf{P}',\mathbf{P}'')
\end{aligned}$$

[1]Equation (3.1) is not rigorously correct. Practically, it is possible to use the weight-window method together with importance splitting, rouletting, and weight cutoff variance reduction methods. In practice, however, rarely is the weight window method combined with these other methods because it embodies the others. This equation is therefore written with the assumption that, if splitting, rouletting, and weight cutoff are used, then weight windows are not, and vice versa.

$$
\begin{aligned}
&\times \int d\mathbf{P}^c B_c(\mathbf{P}'', \mathbf{P}^c) M_r(\mathbf{P}^c) \\
+ &\int dw^+ \delta(w^+ - \omega_t w) \int d\mathbf{R}^+ \delta(\mathbf{R}^+ - \mathbf{R}) C(\mathbf{R}^+) \Big\{ \\
&\times \quad [1 - \Theta(w^+ - \omega_s/K)] \int d\mathbf{P}' B_r(\mathbf{P}^+, \mathbf{P}'; 1/K) M_r(\mathbf{P}') \\
&+ \quad \Theta(w^+ - \omega_s/K) [1 - \Theta(w^+ - \omega_l)] \int d\mathbf{P}' B_r(\mathbf{P}^+, \mathbf{P}'; w^+/\omega_s) M_r(\mathbf{P}') \\
&+ \quad \Theta(w^+ - \omega_l) [1 - \Theta(w^+ - \omega_u)] \int d\mathbf{P}' \delta(\mathbf{P}' - \mathbf{P}^+) M_r(\mathbf{P}') \\
&+ \quad \Theta(w^+ - \omega_u) [1 - \Theta(w^+ - K\omega_u)] \sum_{k=2}^{K} S_k(\mathbf{P}) \int d\mathbf{P}' B_{sk}(\mathbf{P}^+, \mathbf{P}') k M_r(\mathbf{P}') \\
&+ \quad \Theta(w^+ - K\omega_u) \int d\mathbf{P}' B_{sK}(\mathbf{P}^+, \mathbf{P}') K M_r(\mathbf{P}') \Big\} \\
+ &\int dw^+ \delta(w^+ - \omega_t w) \int d\mathbf{R}^+ \delta(\mathbf{P}^+ - \mathbf{P}) \int d\mathbf{P}' \Sigma(\mathbf{P}^+, \mathbf{P}') \int d\mathbf{P}'' E(\mathbf{P}', \mathbf{P}'') \\
&\times \Big\{ [1 - \Theta(w'' - \omega_s/K)] \int d\mathbf{P}^c B_r(\mathbf{P}'', \mathbf{P}^c; 1/K) M_r(\mathbf{P}^c) \\
&+ \quad \Theta(w'' - \omega_s/K) [1 - \Theta(w'' - \omega_l)] \int d\mathbf{P}^c B_r(\mathbf{P}'', \mathbf{P}^c; w''/\omega_s) M_r(\mathbf{P}^c) \\
&+ \quad \Theta(w'' - \omega_l) [1 - \Theta(w'' - \omega_u)] \int d\mathbf{P}^c \delta(\mathbf{P}^c - \mathbf{P}'') M_r(\mathbf{P}^c) \\
&+ \quad \Theta(w'' - \omega_u) [1 - \Theta(w'' - K\omega_u)] \sum_{k=2}^{K} S_k(\mathbf{P}) \int d\mathbf{P}^c B_{sk}(\mathbf{P}'', \mathbf{P}^c) k M_r(\mathbf{P}^c) \\
&+ \quad \Theta(w'' - K\omega_u) \int d\mathbf{P}^c B_{sK}(\mathbf{P}'', \mathbf{P}^c) K M_r(\mathbf{P}^c) \Big\} \\
+ &\quad \hat{Q}_r(\mathbf{P}). \qquad (3.1)
\end{aligned}
$$

Here, a more compact operator notation is introduced so as not to carry all the integrals through the discretization procedure. The following operators are defined: the rouletting operator

$$
\mathcal{R}\{\cdot\} = \int dw^+ \delta(w^+ - \omega_t w) \int d\mathbf{R}^+ \delta(\mathbf{R}^+ - \mathbf{R}) R_o(\mathbf{P}^+) \int d\mathbf{P}'' \int_0^1 d\alpha B_o(\alpha, \mathbf{P}^+, \mathbf{P}'') \{\cdot\},
$$

the splitting operator

$$
\mathcal{S}\{\cdot\} = \sum_{k=2}^{\infty} \int dw^+ \delta(w^+ - \omega_t w) \int d\mathbf{R}^+ \delta(\mathbf{R}^+ - \mathbf{R}) S_k(\mathbf{P}^+) \int d\mathbf{P}'' B_{sk}(\mathbf{P}^+, \mathbf{P}'') k \{\cdot\},
$$

the collision operator

$$
\begin{aligned}
C\{\cdot\} = &\int dw^+ \delta(w^+ - \omega_t w) \int d\mathbf{R}^+ \delta(\mathbf{R}^+ - \mathbf{R}) \int d\mathbf{P}' \Sigma(\mathbf{P}^+, \mathbf{P}') \int d\mathbf{P}'' E(\mathbf{P}', \mathbf{P}'') \\
&\times \int d\mathbf{P}^c B_c(\mathbf{P}'', \mathbf{P}^c) \{\cdot\}, \qquad (3.2)
\end{aligned}
$$

the surface-weight-window operator

$$\begin{aligned}
\mathcal{W}_s\{\cdot\} \;=\; & \int dw^+\delta(w^+-\omega_t w)\int d\mathbf{R}^+\delta(\mathbf{R}^+-\mathbf{R})C(\mathbf{R}^+)\Big\{ \\
& \times \quad [1-\Theta(w^+-\omega_s/K)]\int d\mathbf{P}'B_r(\mathbf{P}^+,\mathbf{P}';1/K)\{\cdot\} \\
& + \quad \Theta(w^+-\omega_s/K)\,[1-\Theta(w^+-\omega_l)]\int d\mathbf{P}'B_r(\mathbf{P}^+,\mathbf{P}';w^+/\omega_s)\{\cdot\} \\
& + \quad \Theta(w^+-\omega_l)\,[1-\Theta(w^+-\omega_u)]\int d\mathbf{P}'\delta(\mathbf{P}'-\mathbf{P}^+)\{\cdot\} \\
& + \quad \Theta(w^+-\omega_u)\,[1-\Theta(w^+-K\omega_u)]\sum_{k=2}^{K}S_k(\mathbf{P})\int d\mathbf{P}'B_{sk}(\mathbf{P}^+,\mathbf{P}')k\{\cdot\} \\
& + \quad \Theta(w^+-K\omega_u)\int d\mathbf{P}'B_{sK}(\mathbf{P}^+,\mathbf{P}')K\{\cdot\}\Big\},
\end{aligned} \tag{3.3}$$

and the collision-weight-window operator

$$\begin{aligned}
\mathcal{W}_c\{\cdot\} \;=\; & \int dw^+\delta(w^+-\omega_t w)\int d\mathbf{R}^+\delta(\mathbf{P}^+-\mathbf{P})\int d\mathbf{P}'\Sigma(\mathbf{P}^+,\mathbf{P}')\int d\mathbf{P}''E(\mathbf{P}',\mathbf{P}'') \\
& \times\Big\{[1-\Theta(w''-\omega_s/K)]\int d\mathbf{P}^cB_r(\mathbf{P}'',\mathbf{P}^c;1/K)\{\cdot\} \\
& + \quad \Theta(w''-\omega_s/K)\,[1-\Theta(w''-\omega_l)]\int d\mathbf{P}^cB_r(\mathbf{P}'',\mathbf{P}^c;w''/\omega_s)\{\cdot\} \\
& + \quad \Theta(w''-\omega_l)\,[1-\Theta(w''-\omega_u)]\int d\mathbf{P}^c\delta(\mathbf{P}^c-\mathbf{P}'')\{\cdot\} \\
& + \quad \Theta(w''-\omega_u)\,[1-\Theta(w''-K\omega_u)]\sum_{k=2}^{K}S_k(\mathbf{P})\int d\mathbf{P}^cB_{sk}(\mathbf{P}'',\mathbf{P}^c)k\{\cdot\} \\
& + \quad \Theta(w''-K\omega_u)\int d\mathbf{P}^cB_{sK}(\mathbf{P}'',\mathbf{P}^c)K\{\cdot\}\Big\}.
\end{aligned} \tag{3.4}$$

With these operator definitions, the history-score moment equation becomes

$$\begin{aligned}
-\boldsymbol{\Omega}\cdot\boldsymbol{\nabla}M_r(\mathbf{r},\boldsymbol{\Omega},E,w)+\Sigma(\mathbf{r},E)M_r(\mathbf{r},\boldsymbol{\Omega},E,w) \;=\; & \\
& \mathcal{R}\{M_r(\mathbf{r}'',\boldsymbol{\Omega}'',E'',w'')\}+\mathcal{S}\{M_r(\mathbf{r}'',\boldsymbol{\Omega}'',E'',w'')\} \\
& + \quad C\{M_r(\mathbf{r}^c,\boldsymbol{\Omega}^c,E^c,w^c)\}+\mathcal{W}_s\{M_r(\mathbf{r}',\boldsymbol{\Omega}',E',w')\} \\
& + \quad \mathcal{W}_c\{M_r(\mathbf{r}^c,\boldsymbol{\Omega}^c,E^c,w^c)\}+Q_r(\mathbf{r},\boldsymbol{\Omega},E,w).
\end{aligned} \tag{3.5}$$

3.1.1 Energy Multi-group Discretization

In Eq. (3.5) changes to the energy variable E occur exclusively through operators on the right-hand-side of the equation. Specifically, the collision operator C and the collision-weight-window operator $\mathcal{W}_c$ are the only operators that elicit an energy change, while the splitting operator $\mathcal{S}$, rouletting operator $\mathcal{R}$, and the surface-weight-window operator $\mathcal{W}_s$ leave the energy unchanged.

Within the collision operator and the collision-weight-window operator it is the $E(\mathbf{P}', \mathbf{P}'')$ kernel that causes the energy change. Recalling that this kernel may be explicitly written as

$$E(\mathbf{P}', \mathbf{P}'')d\mathbf{P}'' = P_s(\mathbf{r}'', E'', \mathbf{\Omega}'' \rightarrow E', \mathbf{\Omega}')dE''d\mathbf{\Omega}''\delta(w'' - w')dw''\delta(\mathbf{r}'' - \mathbf{r}')d\mathbf{r}'', \tag{3.6}$$

where $P_s(\mathbf{r}'', E'', \mathbf{\Omega}'' \rightarrow E', \mathbf{\Omega}')$ is the doubly differential scattering probability, the continuous energy transition from E'' to E' is replaced by a discrete transition from energy group $E_{g''}$ to energy group $E_{g'}$. In this way, the integral over the continuous kernel E is replaced by a summation over the number of energy groups G as

$$\int d\mathbf{P}'' \; E(\mathbf{P}', \mathbf{P}'') \Rightarrow \sum_{g''=1}^{G} \int d\mathbf{\Omega}'' \int dw'' \int d\mathbf{r}'' \; P_s^{g'' \rightarrow g'}(\mathbf{r}'', \mathbf{\Omega}'' \rightarrow \mathbf{\Omega}')\delta(w'' - w')\delta(\mathbf{r}'' - \mathbf{r}'). \tag{3.7}$$

For all kernels other than E, the continuous integrations over delta functions of energy may be replaced by summations with Kronecker delta functions of the energy group. Namely energy integrals in other kernels are replaced as

$$\int dE \; \delta(E' \rightarrow E) \Rightarrow \sum_{g'=1}^{G} \delta_{g',g}. \tag{3.8}$$

The energy groups g are formed by dividing the energy domain into a number of bins with widths $E_{g-1/2} - E_{g+1/2} \equiv \Delta E_g$, $g = 1, 2, \ldots, G$. Conventionally, an increase in group number corresponds to a decrease in energy such that $E_1 > E_2 > \ldots > E_G$. With this energy domain discretization, the moment equation can then be expressed as [O'Dell and Alcouffe, 1987]

$$\begin{aligned} -\mathbf{\Omega}\cdot\nabla M_r^g(\mathbf{r}, \mathbf{\Omega}, w) + \Sigma^g(\mathbf{r})M_r^g(\mathbf{r}, \mathbf{\Omega}, w) &= \mathcal{R}\{M_r^{g''}(\mathbf{r}'', \mathbf{\Omega}'', w'')\} + \mathcal{S}\{M_r^{g''}(\mathbf{r}'', \mathbf{\Omega}'', w'')\} + \\ &+ C\{M_r^{g^c}(\mathbf{r}^c, \mathbf{\Omega}^c, w^c)\} + \mathcal{W}_s\{M_r^{g'}(\mathbf{r}', \mathbf{\Omega}', w')\} \\ &+ \mathcal{W}_c\{M_r^{g^c}(\mathbf{r}^c, \mathbf{\Omega}^c, w^c)\} + Q_r^g(\mathbf{r}, \mathbf{\Omega}, w), \end{aligned} \tag{3.9}$$

where $M_r^g(\mathbf{r}, \mathbf{\Omega}, w)$ is the rth moment carried by particles in direction $\mathbf{\Omega}$ at position $\mathbf{r}$ with weight w and within energy group g, and

$$M_r^g(\mathbf{r}, \mathbf{\Omega}, w) = \int_{E_{g+1/2}}^{E_{g-1/2}} dE \; M_r(\mathbf{r}, \mathbf{\Omega}, E, w), \tag{3.10}$$

$$\begin{aligned} \Sigma^g(\mathbf{r}, \mathbf{\Omega}, w) &= \frac{\int_{E_{g+1/2}}^{E_{g-1/2}} dE \; M_r(\mathbf{r}, \mathbf{\Omega}, E, w)\Sigma(\mathbf{r}, E)}{\int_{E_{g+1/2}}^{E_{g-1/2}} dE \; M_r(\mathbf{r}, \mathbf{\Omega}, E, w)} \\ &= \frac{\int_{E_{g+1/2}}^{E_{g-1/2}} dE \; M_r(\mathbf{r}, \mathbf{\Omega}, E, w)\Sigma(\mathbf{r}, E)}{M_r^g(\mathbf{r}, \mathbf{\Omega}, w)}, \end{aligned} \tag{3.11}$$

$$P_s^{g'\to g}(\mathbf{r}',\mathbf{\Omega}'\to\mathbf{\Omega},w) = \frac{\int_{E_{g'+1/2}}^{E_{g'-1/2}} dE' \int_{E_{g+1/2}}^{E_{g-1/2}} dE\; P_s(\mathbf{r}',E',\mathbf{\Omega}'\to E,\mathbf{\Omega},w')M_r(\mathbf{r}',\mathbf{\Omega}',E',w')}{\int_{E_{g'+1/2}}^{E_{g'-1/2}} dE'\; M_r(\mathbf{r}',\mathbf{\Omega}',E',w')}$$
$$= \frac{\int_{E_{g'+1/2}}^{E_{g'-1/2}} dE' \int_{E_{g+1/2}}^{E_{g-1/2}} dE\; P_s(\mathbf{r}',',E',\mathbf{\Omega}'\to E,\mathbf{\Omega},w')M_r(\mathbf{r}',\mathbf{\Omega}',E',w')}{M_r^{g'}(\mathbf{r}',\mathbf{\Omega}',w')}, \quad (3.12)$$

and

$$Q_r^g(\mathbf{r},\mathbf{\Omega},w) = \int_{E_{g+1/2}}^{E_{g-1/2}} dE\; Q_r(\mathbf{r},\mathbf{\Omega},E,w). \quad (3.13)$$

Unfortunately, in order to compute the group-averaged cross sections $\Sigma^g(\mathbf{r},\mathbf{\Omega},w)$ and group-averaged scattering probabilities $P_s^{g'\to g}(\mathbf{r}',\mathbf{\Omega}'\to\mathbf{\Omega},w)$ one must know the group averaged moments, which in turn requires knowing the moments a priori. For standard flux and adjoint flux type deterministic calculations, determining a representative set of group-averaged cross sections involves first assuming a set and solving the moment equations, using the solutions to generate a new set of cross sections, and repeating this process iteratively until the solution converges. This can be a quite tedious and time consuming operation for deterministic calculations that can be avoided in Monte Carlo transport by use of a continuous energy representation of the cross sections. Moreover, the multi-group cross sections for the moment equations could also be dependent on the Monte Carlo particle weight. For the deterministic calculations contained herein, group-averaged cross sections are not generated from realistic continuous energy counterparts. Here, all cross sections are fictitious and specified in a multi-group format to avoid the aforementioned problem. Also, the cross sections are assumed to be independent of particle weight such that $\Sigma^g(\mathbf{r},\mathbf{\Omega},w) = \Sigma^g(\mathbf{r},\mathbf{\Omega})$ and $P_s^{g'\to g}(\mathbf{r}',\mathbf{\Omega}'\to\mathbf{\Omega},w) = P_s^{g'\to g}(\mathbf{r}',\mathbf{\Omega}'\to\mathbf{\Omega})$.

3.1.2 S_n Angular Discretization

Consider a particle at the center of a unit sphere. If all the possible directions that the particle may travel are represented with rays extending from the center of the unit sphere to the surface of the unit sphere, then there are infinitely many rays. The S_n angular discretization method, which is truly an angular quadrature scheme, approximates the infinitely many directions by a discrete number proportional to n. A quadrature weight q_n is associated with each of the discrete directions $\mathbf{\Omega}_n$ such that a specific set of requirements are met by the quadrature scheme. Perhaps, the most standard requirements of an S_n quadrature set are that the quadrature weights sum to unity and exactly integrate a certain number of spherical harmonics. However, these requirements can be relaxed in favor of others, such as having an ordinate that points along a specific direction.

Each discrete direction $\mathbf{\Omega}_n$ represents a ray from the center of the unit sphere to a point on the unit sphere. Each ray has associated with it a unique set of direction cosines μ_n, η_n, and ξ_n given by

$$\mu_n = \mathbf{\Omega}_n\cdot\hat{\mathbf{i}}, \quad \eta_n = \mathbf{\Omega}_n\cdot\hat{\mathbf{j}}, \quad \text{and} \quad \xi_n = \mathbf{\Omega}_n\cdot\hat{\mathbf{k}}, \quad (3.14)$$

where $\hat{\mathbf{i}}$, $\hat{\mathbf{j}}$, and $\hat{\mathbf{k}}$ are unit vectors in the x, y, and z directions, respectively. Figure 3.1 depicts the points on the unit sphere representing quadrature ordinates for a fully-symmetric ($\mu_1 = \eta_1 = \xi_1$, $\mu_2 = \eta_2 = \xi_2$, etc.) quadrature set.

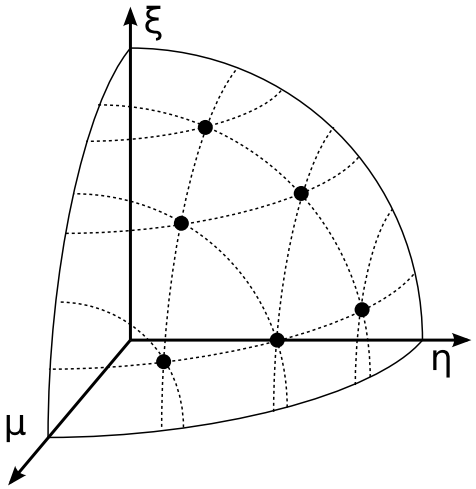


Figure 3.1. A fully-symmetric S_6 quadrature points in the first octant of the unit sphere

With an S_n discretization, integrals over an angular function $f(\mathbf{\Omega})$ become a simple summation, namely

$$\int_{4\pi} d\mathbf{\Omega} f(\mathbf{\Omega}) \Rightarrow \sum_n q_n f(\mathbf{\Omega}_n). \tag{3.15}$$

Furthermore, the discrete ordinates method provides a way of computing the amount of history-score moment changing directions at a scattering event.

For the majority of scattering from materials with random or isotropic structure, the transition probability $P_s(\mathbf{r}', E', \mathbf{\Omega}' \rightarrow E, \mathbf{\Omega})$ depends only on the cosine of the angle between the two directions. In such a case, P_s simplifies to $P_s(\mathbf{r}', E' \rightarrow E, \mathbf{\Omega}'\cdot\mathbf{\Omega})$ or $P_s(\mathbf{r}', E' \rightarrow E, \mu_s)$, where $\mu_s = \mathbf{\Omega}'\cdot\mathbf{\Omega}$ is the cosine of the scattering angle. Legendre moments of the scattering probabilities with respect to a spherical harmonic function $R_m(\mu_s, \psi)$ can be computed. These spherical harmonic functions are dependent on the geometry being considered and are discussed in depth by O'Dell and Alcouffe [1987]. The Legendre moments $P_s^l(\mathbf{r}', E' \rightarrow E)$ are computed as[2]

$$P_s^l(\mathbf{r}', E' \rightarrow E) = \int_{4\pi} d\mathbf{\Omega} P_s(\mathbf{r}', E' \rightarrow E, \mu_s) R_m(\mu_s, \psi). \tag{3.16}$$

Using the S_n discretization, the above integral may be approximated as

$$P_s^l(\mathbf{r}', E' \rightarrow E) = \sum_n q_n P_s(\mathbf{r}', E' \rightarrow E, \mu_n) R_m(\mathbf{\Omega}_n). \tag{3.17}$$

Similarly, the spherical harmonic moments of the rth moment of the history-score moment equa-

[2]Really, the Legendre moments need not be computed with respect to every spherical harmonic function. The R_m are defined such that they represent a spherical harmonic without any imaginary values. However, because the scattering is assumed to be azimuthally symmetric, all moments with respect to R_m that contain azimuthal dependence vanish in the integration leaving only terms that integrate over the Legendre polynomials. For this reason, each m moment has a single corresponding l value, and while not explicitly stated a summation over m implies that any l value within the summation is that corresponding to the m value.

tions $M_r^m(\mathbf{r}', E')$ can be evaluated as

$$M_r^m(\mathbf{r}', E', w') = \int_{4\pi} d\mathbf{\Omega}' M_r(\mathbf{r}', E', \mathbf{\Omega}', w') R_m(\mathbf{\Omega}'), \tag{3.18}$$

which, again by application of the S_n discretization, becomes

$$M_r^m(\mathbf{r}', E', w') = \sum_{n'} q_{n'} M_r^{n'}(\mathbf{r}', E', w') R_m(\mathbf{\Omega}_{n'}). \tag{3.19}$$

Finally, the scattering contributions may be computed by folding the spherical harmonic moments of the history-score moment together with the scattering moments [O'Dell and Alcouffe, 1987]

$$M_r^n(\mathbf{r}, E, w) = \sum_m (2l+1) P_s^l(\mathbf{r}', E' \rightarrow E) M_r^m(\mathbf{r}', E', w') R_m(\mathbf{\Omega}_n). \tag{3.20}$$

The above expression still contains the continuous transition probability from energy E' to E. By applying the multi-group approximation discussed in the previous section, a final expression for the amount of rth history-score moments changing direction and energy at a scattering event is obtained, namely,

$$M_r^{n,g}(\mathbf{r}, w) = \sum_m (2l+1) P_s^{l,g' \rightarrow g}(\mathbf{r}') M_r^{m,g'}(\mathbf{r}', w') R_m(\mathbf{\Omega}_n). \tag{3.21}$$

Application of the multi-group and S_n approximations to the original operator form of the moment equations yields

$$\begin{aligned} -\left[\mathbf{\Omega}\cdot\nabla M_r^g(\mathbf{r}, \mathbf{\Omega}, w)\right]^n + \Sigma^g(\mathbf{r}) M_r^{n,g}(\mathbf{r}, w) &= \mathcal{R}\{M_r^{n'',g''}(\mathbf{r}'', w'')\} + \mathcal{S}\{M_r^{n'',g''}(\mathbf{r}'', w'')\} + \\ &+ C\{M_r^{n^c,g^c}(\mathbf{r}^c, w^c)\} + \mathcal{W}_s\{M_r^{n',g'}(\mathbf{r}', w')\} \\ &+ \mathcal{W}_c\{M_r^{n^c,g^c}(\mathbf{r}^c, w^c)\} + Q_r^{n,g}(\mathbf{r}, w). \end{aligned} \tag{3.22}$$

The term in square brackets, the streaming term, is dependent on the geometry on which the solution is being determined. Until that geometry is specified, the exact form cannot be written, but Duderstadt and Martin [1978] provides the form of the streaming term for various geometries. The form of the operator considered in this work is for Cartesian geometries.

As with the energy variable, the direction of the particles change only at collisions, mathematically represented by the $E(\mathbf{P}, \mathbf{P}')$ kernel contained in the C and $\mathcal{W}_c$ operators. When the moment transitions from one energy and direction to another, the spherical harmonic moments of the moment are computed and folded together with the group scattering probability moments as detailed above. This computation must be performed for all Monte Carlo particle weights being accounted for in the problem.

3.1.3 Monte Carlo Particle Weight Discretization

The above discretizations are standard to solutions of deterministic solutions of transport problems. However, the transport calculations being performed herein are different from standard calculations in that they also require the Monte Carlo particle weight w. In general, the range of weight required

for a problem is unknown a priori, and potentially extends from zero to infinity. However, for practical implementation of a deterministic solution algorithm, the weight domain must be truncated at some lower truncation weight w_{tl} and some upper truncation weight w_{tu}.

For the calculations, the weight domain between w_{tl} and w_{tu} is discretized with a comb discretization. Rather than solving for the rth moment of history-score distribution at all weights between w_{tl} and w_{tu}, the rth moment is only calculated at specific weights w_k such that $w_1 = w_{tl} < w_2 < \ldots < w_{k-1} < w_k = w_{tu}$. It is an important distinction that the weight discretization is not a distribution and has no quadrature weight associated with it. Furthermore, the weight domain is not integrated over, rather specific evaluations of the history-score moments is made at one of the comb weights.

When a Monte Carlo particle of weight w_o is subjected to a variance reduction game, the resulting weight of the particle is a deterministic function of some set of parameters and even potentially the particle's weight itself. Although a particle's weight may change in multiple manners, once the method by which the particle changes weights is known, the new weight is known. For example, in the rouletting of a particle with weight w_o to a survival weight of ω_s the particle survives and is promoted to weight ω_s with a probability of w_o/ω_s, otherwise it is given a weight of zero, i.e., terminated. Even though two possible weight outcomes exist, once it is known if the particle survives or not, the new weight of the particle is known. It is for this reason that the rth moment of the history-score distribution carried by a particle of weight w_o can be thought of as transitioning to another unique weight with a given probability.

Monte Carlo Particle Weight at Tallies

To understand how the rth moment changes as a function of Monte Carlo particle weight, it is necessary to first understand sources of the rth moment. Because the history-score moment equations are adjoint to the transport equation [Amster and Djomehri, 1976], the source for the moment equations originates at the tallies. When a Monte Carlo particle reaches a tally, it does so with a weight w. The score s the particle of weight w contributes to that tally is proportional to the weight, i.e., $s = Aw$, where A is some constant of proportionality. Thus, as the score increases linearly with the particle weight, the first moment source Q_1 does so as well.

For the second moment, the source is slightly more complicated. If the tally is a terminal estimator, that is a particle can contribute only a single score and then is terminated, then the source contribution to the second moment is $s^2 = A^2w^2$. If the tally is non-terminal, then there are additions to this source amount, but this is an unneeded complication to understanding how moment transitions across the weight domain.

For simplicity, consider the case $A = 1$, which is equivalent to a current tally in MCNP. In this case, the first moment source varies as w and the second moment source varies as w^2. If $w = 1$, then the first moment source and second moment source are exactly the same, but for $w < 1$ the second moment source is less than the first moment source and for $w > 1$ the second moment source is greater than the first moment source as shown in Fig. 3.2.

Transition of Moment Resulting from Variance Reduction

With an understanding of how the first and second moments at tallies behave, the transition of moment from one weight to another can be considered. Consider the instance of a n-to-1 split or

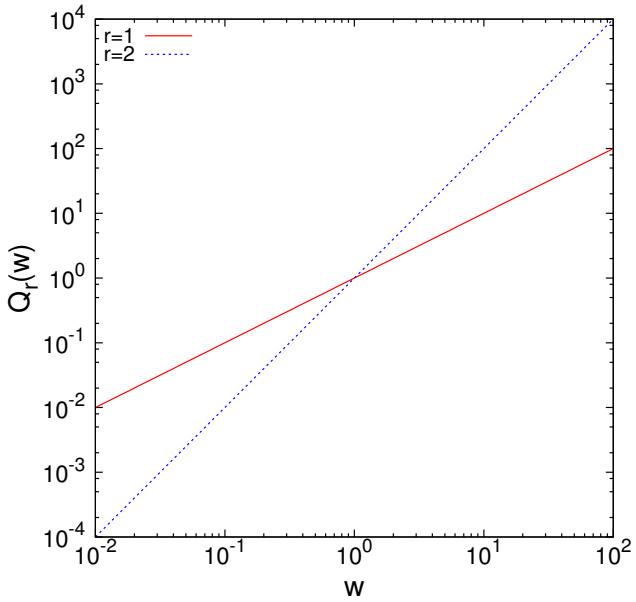


Figure 3.2. Comparison of first and second moment sources for a terminal current tally

a particle with weight w_o. In the forward sense, the particle splits into n particles of weight w_o/n. In the adjoint formulation of the moment equations, the event must be considered in the reverse direction. In this case, the moment carried by a particle of weight w_o/n is multiplied by n, because that is how many particles there were, and this is now the moment carried by a particle of weight w_o.

Consider now the above case applied to the unattenuated source presented in Fig. 3.2 such that $w_o = 1$. Because the score varies linearly with w [Booth and Cashwell, 1979], the first moment carried by a particle of weight w_o/n is $1/n$ times that carried by a particle of weight w_o, or $Q_1(w_o/n) = Q_1(w_o)/n$. When the particles, each carrying a moment of $Q_1(w_o/n) = Q_1(w_o)/n$, split, the resulting first moment for a particle at weight w_o is $Q_1(w_o)/n \times n = Q_1(w_o)$ and remains unchanged. This is exactly as expected because unbiased variance reduction games should not change the first moment of the distribution.

The second moment however does change as a result of this splitting game. For the second moment, the score varies as the weight squared. Thus, the second moment carried by a source particle from the tally of weight w_o/n is $1/n^2$ that carried by a particle of weight w_o, or $Q_2(w_o/n) = Q_2(w_o)/n^2$. In the adjoint process, the n particles, each carrying second moment of $Q_2(w_o/n)$, become the single particle with weight w_o, and the second moment carried by a particle of weight w_o is $nQ_2(w_o/n) = n \times Q_2(w_o)/n^2 = Q_2(w_o)/n$. Therefore the second moment carried by a particle at weight w_o is a factor of n less than without the split, i.e., $n = 1$. Figure 3.3 shows how the source moments presented in Fig. 3.2 would change were they to undergo a 5-to-1 split. Whereas before, particles with unit weight carried the same first and second moments, now particles of unit weight have a lower second moment $Q_2'(w)$ resulting from the split.

Effectively, the second moment weight distribution has shifted to the right. In this case the entire distribution shifts because a splitting game is weight independent. Weight-dependent games, such as weight cutoff, only affect portions of the weight distribution. Because weight-dependent

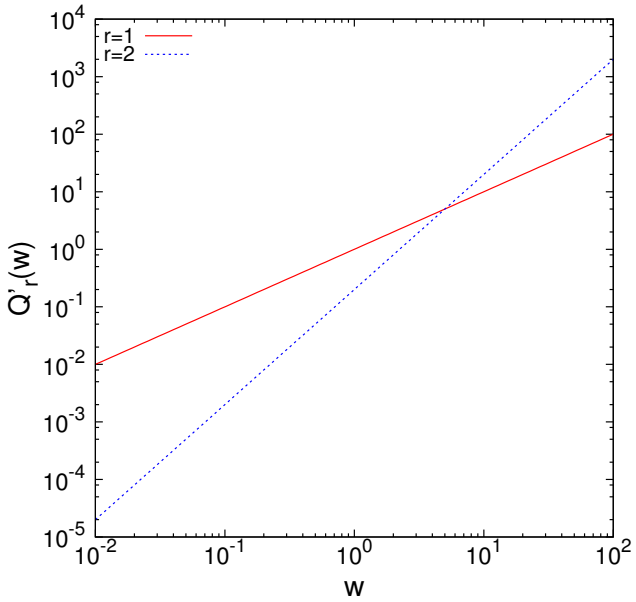


Figure 3.3. Comparison of first and second moment sources for a terminal current tally after a 5-to-1 split

games affect only a portion of the weight domain, they introduce discontinuities into the weight domain. With such discontinuities, the generalization above for the second moment $Q_2(w_o/n) = Q_2(w_o)/n^2$ does not apply and therefore the weight dependence must be tracked explicitly. The effect of different variance reduction games, both weight-dependent and weight-independent are detailed in the Section 3.5.

Transition of Moment with Discretized Weight

The weight domain is discretized by specifying upper and lower truncation weights w_{tu} and w_{tl}, respectively, and the moment for specific weights w_k between w_{tl} and w_{tu} is followed. Because the moment is tracked only at these specific weights, the values of the moment between the discrete weights are not known. For example, in the splitting case above moment transitions from weight w_o/n to w_o. If w_o corresponds to one of the discretized w_k then, in general, it is unlikely that w_o/n also corresponds to one of the w_k.

To solve this problem, a simple interpolation scheme is used. Continuing with the example of the n-to-1 split, is is assumed that each w_k is like the w_o discussed in the previous section. k' is found such that $w_{k'-1} < w_k/n = w_o/n < w_{k'}$. If $w_k/n < w_{tl}$ or $w_k/n > w_{tu}$, then no interpolation is performed and the moment is simply assigned the moment at w_{tl} or w_{tu}, respectively. If $w_k/n < w_{k'-1} + 0.25(w_{k'} - w_{k'-1})$, the interpolated moment is simply set equal to the moment at $w_{k'-1}$, and if $w_k/N > w_{k'-1} + 0.75(w_{k'} - w_{k'-1})$ the interpolated moment is given the value $w_{k'}$, otherwise the interpolated moment is set to $(w_{k'-1} + w_{k'})/2$. Such an interpolation scheme may seem excessively simplistic, but many other more sophisticated interpolation methods were attempted with little or no benefit to the result and with a higher computational cost. Furthermore, no interpolation scheme attempted was able to overcome the problem of interpolation at discontinuities in the weight domain discussed later in this chapter.

With this method of discretizing the weight domain and the ability to appropriately transition

moment from one weight to another, the operator form of the moment equations can be adjusted for the weight discretization as

$$\begin{aligned} -\left[\boldsymbol{\Omega}\cdot\boldsymbol{\nabla} M_r^{g,w}(\mathbf{r},\boldsymbol{\Omega})\right]^n + \Sigma^g(\mathbf{r}) M_r^{n,g,w}(\mathbf{r}) &= \mathcal{R}\{M_r^{n'',g'',w''}(\mathbf{r}'')\} + \mathcal{S}\{M_r^{n'',g'',w''}(\mathbf{r}'')\} + \\ &+ C\{M_r^{n^c,g^c,w^c}(\mathbf{r}^c)\} + \mathcal{W}_s\{M_r^{n',g',w'}(\mathbf{r}')\} \\ &+ \mathcal{W}_c\{M_r^{n^c,g^c,w^c}(\mathbf{r}^c)\} + Q_r^{n,g,w}(\mathbf{r}), \end{aligned} \tag{3.23}$$

where the index w indicates one of the specific weight values between the truncation weights. As with the energy and direction variables, all modifications to the weight occur via the operators on the right-hand-side of this equation.

3.1.4 Cartesian Discretization of the Spatial Domain

For Cartesian geometry, the streaming term may be written [Lewis and Miller, 1993] as

$$-\left[\boldsymbol{\Omega}\cdot\boldsymbol{\nabla} M_r^{g,w}(\mathbf{r},\boldsymbol{\Omega})\right]^n \Rightarrow -\left[\mu_n \frac{\partial M_r^{n,g,w}(x,y,z)}{\partial x} + \eta_n \frac{\partial M_r^{n,g,w}(x,y,z)}{\partial y} + \xi_n \frac{\partial M_r^{n,g,w}(x,y,z)}{\partial z}\right]. \tag{3.24}$$

The Cartesian geometry has the distinct advantage that there are no angular derivatives. Angular derivatives do however arise in cylindrical and spherical geometries and must be treated differently. Such discussion is beyond the scope of this section.

For simplicity, only the 1-D spatial discretization will be derived, and the 2-D discretization will be simply quoted as the derivations are very similar. Also, the notation for the entire right-hand-side of Eq. (3.23) is denoted by $\widetilde{Q}_r^{n,g,w}(x,y,z)$, i.e.,

$$\begin{aligned} \widetilde{Q}_r^{n,g,w}(x,y,z) &= \mathcal{R}\{M_r^{n'',g'',w''}(\mathbf{r}'')\} + \mathcal{S}\{M_r^{n'',g'',w''}(\mathbf{r}'')\} + \\ &+ C\{M_r^{n^c,g^c,w^c}(\mathbf{r}^c)\} + \mathcal{W}_s\{M_r^{n',g',w'}(\mathbf{r}')\} \\ &+ \mathcal{W}_c\{M_r^{n^c,g^c,w^c}(\mathbf{r}^c)\} + Q_r^{n,g,w}(\mathbf{r}). \end{aligned} \tag{3.25}$$

With this simplification and the 1-D form of the streaming operator, Eq. (3.23) becomes

$$-\mu_n \frac{\partial M_r^{n,g,w}(x)}{\partial x} + \Sigma^g(x) M_r^{n,g,w}(x) = \widetilde{Q}_r^{n,g,w}(x). \tag{3.26}$$

Consider a 1-D slab divided into I cells with a *cell-center node* at the center of each cell and *cell-edge nodes* at the border of each cell as shown in Fig. 3.4. The cell-center nodes are labeled with a whole-integer index and the cell-edge nodes are labeled with a half-integer index. The discretization of the x domain is obtained by integrating Eq. (3.26) over a single cell from its left edge to the right edge, namely

$$\int_{x_{i-1/2}}^{x_{i+1/2}} dx\, \mu_n \frac{\partial M_r^{n,g,w}(x)}{\partial x} + \int_{x_{i-1/2}}^{x_{i+1/2}} dx\, \Sigma^g(x) M_r^{n,g,w}(x) = \int_{x_{i-1/2}}^{x_{i+1/2}} \widetilde{Q}_r^{n,g,w}(x). \tag{3.27}$$

The negative sign, present because the history-score moment equation is adjoint to the transport equation, has been dropped as the standard method of solving adjoint transport equations is to solve them as a forward problem and then reverse the direction of the solution. Evaluation of the first integral gives the differences of the moments at the endpoints. The other two integrals are

approximated by the difference of the edge locations times the value at the center, so the above equation becomes

$$\mu_n \left[M_r^{n,g,w}(x_{i+1/2}) - M_r^{n,g,w}(x_{i-1/2}) \right] + \Delta x_i \Sigma^g(x_i) M_r^{n,g,w}(x_i) = \Delta x_i \widetilde{Q}_r^{n,g,w}(x_i), \tag{3.28}$$

where $\Delta x_i = x_{i+1/2} - x_{i-1/2}$.

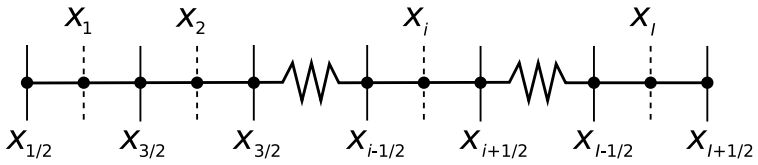


Figure 3.4. 1D slab divided into I cells having I cell-centered nodes and with $I + 1$ cell-edge nodes

Transport-like equations are usually solved deterministically by the method of sweeps. However, only one of the cell-edge values is typically known. Therefore, it is necessary to introduce an auxiliary relationship between $M_r^{n,g,w}(x_{i\pm1/2})$ and $M_r^{n,g,w}(x_i)$. Perhaps the most commonly used relationship is the *diamond difference* scheme, where it is assumed that the cell-center value is the average of the cell-edge values, i.e.,

$$M_r^{n,g,w}(x_i) = \frac{1}{2} \left[M_r^{n,g,w}(x_{i-1/2}) + M_r^{n,g,w}(x_{i+1/2}) \right]. \tag{3.29}$$

With this additional relationship, the 1-D sweep method can be established.

The slab is now assumed to have vacuum boundary conditions. This assumptions requires that

$$M_r^{n,g,w}(x_{1/2}) = 0, \quad \mu_n > 0 \text{ and} \tag{3.30}$$

$$M_r^{n,g,w}(x_{I+1/2}) = 0, \quad \mu_n < 0 \tag{3.31}$$

because there is no way for particles leaving the slab to reenter the slab. Therefore, two similar, yet different, equations are developed to solve for the cases $\mu_n > 0$ and $\mu_n < 0$. For $\mu_n > 0$, Eq. (3.28) and Eq. (3.29) are used to eliminate $M_r^{n,g,w}(x_{i+1/2})$, giving

$$M_r^{n,g,w}(x_i) = \left[\frac{2|\mu_n|}{\Delta x_i} + \Sigma^g(x_i) \right]^{-1} \left[\frac{2|\mu_n|}{\Delta x_i} M_r^{n,g,w}(x_{i-1/2}) + \widetilde{Q}_r^{n,g,w}(x_i) \right], \tag{3.32}$$

and then $M_r^{n,g,w}(x_{i+1/2})$ is obtained from

$$M_r^{n,g,w}(x_{i+1/2}) = 2M_r^{n,g,w}(x_i) - M_r^{n,g,w}(x_{i-1/2}). \tag{3.33}$$

Now, because it is known that $M_r^{n,g,w}(x_{1/2}) = 0$ from the boundary conditions, $M_r^{n,g,w}(x_1)$ can be determined from Eq. (3.32) and then $M_r^{n,g,w}(x_{3/2}$ can be found from Eq. (3.33). These equations can be repeatedly applied is a *sweep* from the left side of the slab to the right side until values of $M_r^{n,g,w}$ have been determined for all cell-center and cell-edge nodes such that $\mu_n > 0$.

In a similar fashion, sweep equations for the right to left sweep can be determined. They are

$$M_r^{n,g,w}(x_i) = \left[\frac{2|\mu_n|}{\Delta x_i} + \Sigma^g(x_i) \right]^{-1} \left[\frac{2|\mu_n|}{\Delta x_i} M_r^{n,g,w}(x_{i+1/2}) + \widetilde{Q}_r^{n,g,w}(x_i) \right], \tag{3.34}$$

and

$$M_r^{n,g,w}(x_{i-1/2}) = 2M_r^{n,g,w}(x_i) - M_r^{n,g,w}(x_{i+1/2}).,\tag{3.35}$$

subject to $\mu_n < 0$. Thus, because $M_r^{n,g,w}(x_{I+1/2}) = 0$ from the boundary conditions, $M_r^{n,g,w}(x_I)$ can be determined by Eq. (3.34) and then $M_r^{n,g,w}(x_{I-1/2})$ can be found by Eq. (3.33). A sweep from the right side of the slab to the left side of the slab then yields values of $M_r^{n,g,w}$ for all cell-center and cell-edge nodes such that $\mu_n < 0$.

For two dimensions, the sweep procedure becomes slightly more complicated. Rather than having simply two sweep directions, there are four sweep directions: $\mu_n > 0$ and $\eta_n > 0$, $\mu_n < 0$ and $\eta_n > 0$, $\mu_n < 0$ and $\eta_n < 0$, and $\mu_n > 0$ and $\eta_n < 0$. For the case $\mu_n > 0$ and $\eta_n > 0$, the sweep equations are

$$\begin{aligned} M_r^{n,g,w}(x_i, y_j) \quad = \quad & \left[\frac{2|\mu_n|}{\Delta x_i} + \frac{2|\eta_n|}{\Delta y_j} + \Sigma^g(x_i, y_j)\right]^{-1} \times \\ & \left[\frac{2|\mu_n|}{\Delta x_i} M_r^{n,g,w}(x_{i-1/2}, y_j) + \frac{2|\eta_n|}{\Delta y_j} M_r^{n,g,w}(x_i, y_{j-1/2}) + \widetilde{Q}_r^{n,g,w}(x_i, y_j)\right], \end{aligned}\tag{3.36}$$

and

$$M_r^{n,g,w}(x_{i+1/2}, y_j) \quad = \quad 2M_r^{n,g,w}(x_i, y_j) - M_r^{n,g,w}(x_{i-1/2}, y_j)\tag{3.37}$$

$$M_r^{n,g,w}(x_i, y_{j+1/2}) \quad = \quad 2M_r^{n,g,w}(x_i, y_j) - M_r^{n,g,w}(x_i, y_{j-1/2}).\tag{3.38}$$

Figure 3.5 shows the order in which the individual nodes are solved. The cell-center nodes are always evaluated using Eq. (3.36) while the cell-edge nodes are evaluated using Eq. (3.37) or Eq. (3.38) for the x or y directions, respectively.

Because of the spatial discretization and how well exponential attenuation is represented across a cell, it is possible for cell-edge values computed during a sweep to be negative, which is not physically reasonable. Many different *negative-flux fixups* have been developed, all introducing varying amounts of error to the solution. For 1-D calculations, negative-flux fixups are unneeded because a positive edge-node flux is guaranteed when [Lewis and Miller, 1993]

$$\Delta x_i < \frac{2\min(|\mu_n|)}{\Sigma^g(x_i)}.\tag{3.39}$$

However, in higher dimensions there is no defined spacing that guarantees positivity of the solution.

This work utilizes the *set to zero and recalculate* negative-flux fixup scheme. For the sweep direction illustrated in Fig. 3.5, the moments $M_r^{n,g,w}(x_{i-1/2}, y_j)$ and $M_r^{n,g,w}(x_i, y_{j-1/2})$ are known when updating cell i, j. The cell-center moment $M_r^{n,g,w}(x_i, y_j)$ is calculated with Eq. (3.36) and is never negative. However, using Eq. (3.37) and Eq. (3.38) it is possible for one or both of $M_r^{n,g,w}(x_{i+1/2}, y_j)$ and $M_r^{n,g,w}(x_i, y_{j+1/2})$ to be negative. When $M_r^{n,g,w}(x_{i+1/2}, y_j)$ or $M_r^{n,g,w}(x_i, y_{j+1/2})$ is found to be negative, then its value is set to zero and the cell-center moment value is recomputed as

$$\begin{aligned} & M_r^{n,g,w}(x_i, y_j) \quad = \\ & \qquad \frac{M_r^{n,g,w}(x_{i-1/2}, y_j) + M_r^{n,g,w}(x_i, y_{j-1/2}) + M_r^{n,g,w}(x_{i+1/2}, y_j) + M_r^{n,g,w}(x_i, y_{j+1/2})}{4}, \end{aligned}\tag{3.40}$$

the average of the cell-edge node values.

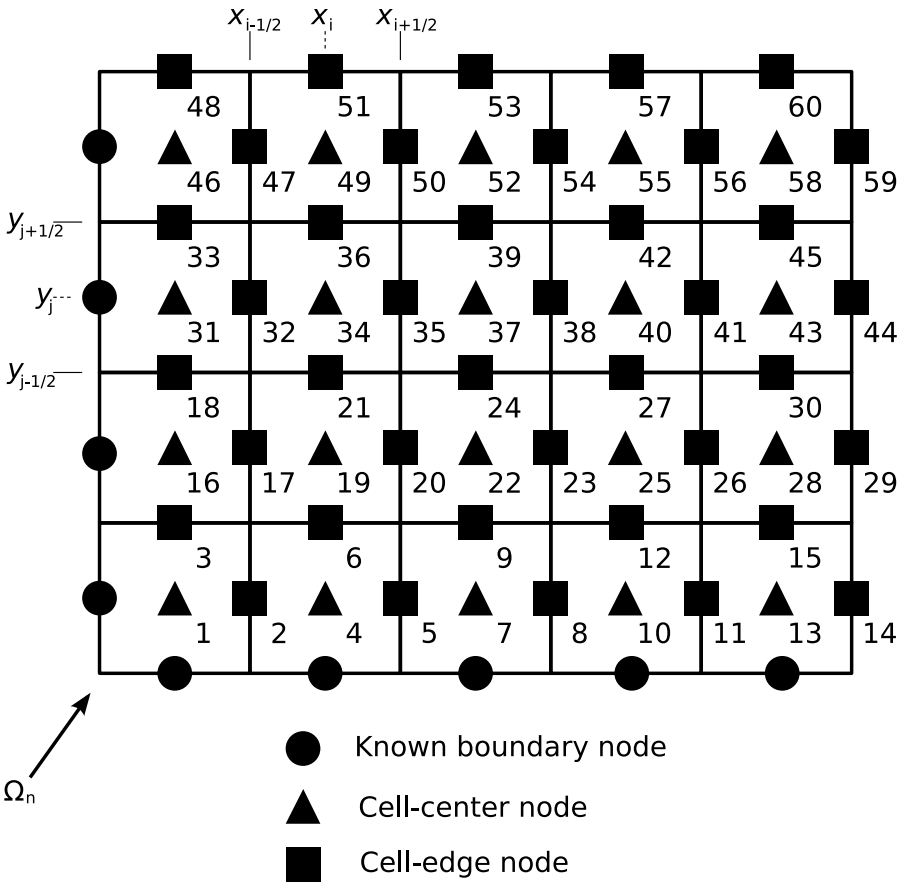


Figure 3.5. 2D solution order of the nodes for the case $\mathbf{\Omega}_n \cdot \hat{\mathbf{i}} = \mu_n > 0$ and $\mathbf{\Omega}_n \cdot \hat{\mathbf{j}} = \eta_n > 0$ (adapted from Lewis and Miller [1993])

3.2 Description of Meshing and Sweeps for History-Score Moment Equation Solutions

While the Monte Carlo method lends itself nicely to complex geometries, the S_n method typically does not. For this reason, all calculations in this work are restricted to geometries that can be defined with planes orthogonal to the x, y, and z axes. Such geometries were chosen also because many variance reduction games occur at the surfaces between cells. In order to accurately represent these types of variance reduction games in the S_n calculation, it is necessary to force a mesh node (either cell-centered or cell-edge) to be located on this surface. Here, cell-edge nodes are forced to be located on surfaces between Monte Carlo cells so that all S_n cells contain the same material. Figure 3.6 shows a 1-D, two-cell geometry and the superimposed S_n mesh. Whenever splitting occurs at a surface or a tally is located on a surface, it is necessary to add source resulting from tallies or splitting on at that surface, and this is why cell-edge nodes corresponding to surfaces are necessary.

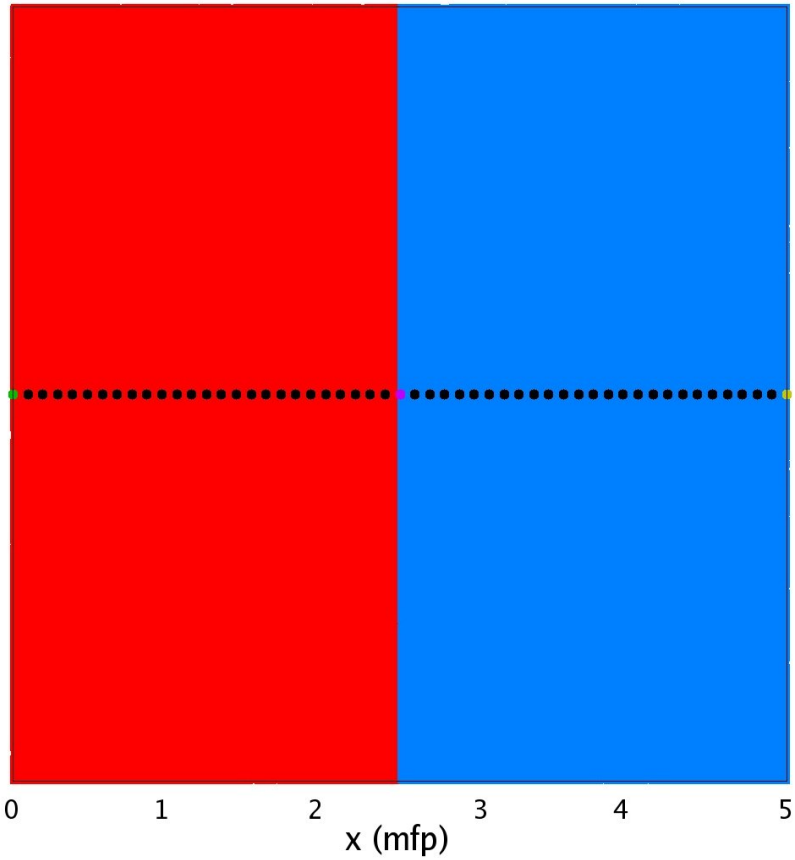


Figure 3.6. 1-D two-cell geometry. Black nodes represent cell-edge and cell-center nodes, the green node represents the source, the purple node represents a cell-edge node on a surface between two Monte Carlo cells and the yellow node represents the tally

The S_n sweep method fundamentally works the same regardless of the moment being calculated or the dimension. For simplicity, only the 1-D algorithm is discussed here. For a 1-D S_n cell,

there are three moments of interest: the cell-center moment $M_r^{n,g,w}(x_i)$ and both cell-edge moments $M_r^{n,g,w}(x_{i\pm1/2})$. In the left-to-right sweep of the slab, $M_r^{n,g,w}(x_{i-1/2})$ is known. If the node corresponding to $i-1/2$ is a tally (this is a source for the adjoint problem) the necessary amount of source is added to the moment before updating the cell-center moment. Next, the new cell-center moment is calculated, and the opposite cell-edge moment is extrapolated, which, if negative, is corrected by the negative-flux fixup scheme. If the new cell-edge node corresponds to a surface where any of splitting, rouletting, or weight windows occurs, then the weight distribution of the moment is modified (these modifications are discussed below) and any additional source added. This process is repeated for all S_n cells to complete the sweep.

Next the collision source is updated at all the cell-center nodes. The weight domain of the moments at the cell-center nodes is modified as the result of any weight cutoff, implicit capture, and weight windows games, and any additional source resulting from splitting is added to the moment. If the cell-center node is contained within a volume for a volume tally, then the expected track length value is added to the moment. Lastly, the scattering moments are computed and the entire sweep process is repeated.

One caveat exists to the description above. It is unnecessary to keep track of the weight domain for the first moment. This is because the first moment, or mean, Eq. (2.77), scales with the particle weight and therefore it is necessary to compute the first moment for a weight of unity and scale it to the other weights as needed. Performing first moment calculations in this manner greatly accelerates the calculation because it entirely removes the necessity of keeping track of the weight domain. Thus, the variance is computed by S_n by performing a weight-independent first moment calculation, computing additions to the second moment resulting from the first moment by scaling the first moment by weight as necessary, and then performing the weight-dependent second moment calculation.

3.3 Problems with Ray Effects

In multiple spatial dimensions, the solution of transport problems is plagued by ray effects. Ray-effects arise because particles are tracked only along the discrete directions of the S_n discretization. Directions not belonging to the discrete set of directions typically experience an underestimation of the flux, or, in this case, first and second moments of the score distribution. Generally, the solution to ray effects is to use a higher S_n discretization order, requiring increased computation times and memory requirements, or apply a first collision source.

In this work, the issue of ray effects is not addressed. The two-dimensional problems were kept small enough such that ray-effects were small, but far from negligible. Furthermore, the two dimensional problems were run with materials with high scattering-to-total ratios such that the isotropic scattering helps alleviate some of the ray effects.

3.4 Calculation of Moments and Variance

Calculating the variance of a specific tally requires that both the first and second moments, each a single scalar value, of the history-score distribution for that tally be evaluated. The history-score moment equations provide a solution for $M_1(\mathbf{r}, \mathbf{\Omega}, E, w)$ and $M_2(\mathbf{r}, \mathbf{\Omega}, E, w)$, functions of the

phase space variables. These functions must be somehow evaluated to the single scalar values corresponding to the first and second moments of the history-score distribution.

In a forward transport problem, a source $S(\mathbf{r}, \mathbf{\Omega}, E)$ emits particles and results in a flux $\phi(\mathbf{r}, \mathbf{\Omega}, E)$. The scalar response $\mathbb{R}$ of the flux to some detector function $D(\mathbf{r}, \mathbf{\Omega}, E)$ is given by

$$\mathbb{R} = \int d\mathbf{r} \int d\mathbf{\Omega} \int dE\ \phi(\mathbf{r}, \mathbf{\Omega}, E) D(\mathbf{r}, \mathbf{\Omega}, E). \tag{3.41}$$

Similarly, for an adjoint transport problem, and adjoint source $S^\dagger(\mathbf{r}, \mathbf{\Omega}, E)$ emits response particles resulting in an adjoint flux $\phi^\dagger(\mathbf{r}, \mathbf{\Omega}, E)$. The adjoint response $\mathbb{R}^\dagger$ is then computed with respect to some adjoint detector function $D^\dagger(\mathbf{r}, \mathbf{\Omega}, E)$ as

$$\mathbb{R}^\dagger = \int d\mathbf{r} \int d\mathbf{\Omega} \int dE\ \phi^\dagger(\mathbf{r}, \mathbf{\Omega}, E) D^\dagger(\mathbf{r}, \mathbf{\Omega}, E). \tag{3.42}$$

It can be shown that $\mathbb{R} = \mathbb{R}^\dagger$ if $S^\dagger(\mathbf{r}, \mathbf{\Omega}, E) = D(\mathbf{r}, \mathbf{\Omega}, E)$ and $D^\dagger(\mathbf{r}, \mathbf{\Omega}, E) = S(\mathbf{r}, \mathbf{\Omega}, E)$ [Lewis and Miller, 1993].

Because the history-score moment equations are adjoint to the forward transport equation, the appropriate detector function is the physical source $S(\mathbf{r}, \mathbf{\Omega}, E)$ just as it was for the adjoint transport equation. With this detector response, the first moment response $\mathbb{R}_{M_1}$ can be calculated as

$$\mathbb{R}_{M_1} = \langle M_1(\mathbf{r}, \mathbf{\Omega}, E), S(\mathbf{r}, \mathbf{\Omega}, E) \rangle = \int d\mathbf{r} \int d\mathbf{\Omega} \int dE\ M_1(\mathbf{r}, \mathbf{\Omega}, E) S(\mathbf{r}, \mathbf{\Omega}, E). \tag{3.43}$$

Similarly, the second moment respons $\mathbb{R}_{M_2}$ is

$$\mathbb{R}_{M_2} = \langle M_2(\mathbf{r}, \mathbf{\Omega}, E), S(\mathbf{r}, \mathbf{\Omega}, E) \rangle = \int d\mathbf{r} \int d\mathbf{\Omega} \int dE\ M_2(\mathbf{r}, \mathbf{\Omega}, E) S(\mathbf{r}, \mathbf{\Omega}, E). \tag{3.44}$$

Finally, the population variance of the score distribution σ^2 is given by

$$\sigma^2 = \mathbb{R}_{M_2} - \mathbb{R}_{M_1}^2 = \langle M_2(\mathbf{r}, \mathbf{\Omega}, E), S(\mathbf{r}, \mathbf{\Omega}, E) \rangle - \langle M_1(\mathbf{r}, \mathbf{\Omega}, E), S(\mathbf{r}, \mathbf{\Omega}, E) \rangle^2 . \tag{3.45}$$

3.5 Effects of Variance Reduction Games on Weight Domain

The following sections describe how various variance reduction games modify the weight domain of the second moment. The first moment scales with the weight of the particle, and, while technically this discussion applies to the first moment as well, the first moment remains independent of variance reduction effects.

3.5.1 Importance Splitting and Implicit Capture

The importance splitting and implicit capture game are discussed together because the effects they have on the weight domain are similar. The importance splitting kernel given in Section 2.2.1 says that, if w is the weight of a particle before a split, the new moment carried by a particle w after a k-to-1 split in the adjoint sense

$$M_r(\mathbf{r}, \mathbf{\Omega}, E, w) = \int dw' \delta(w' - w/k) M_r(\mathbf{r}, \mathbf{\Omega}, E, w') = M_r(\mathbf{r}, \mathbf{\Omega}, E, w/k). \tag{3.46}$$

This equation indicates that the moment carried by a particle of weight w should be the moment at w/k. Essentially, this operations causes a shift in the weight domain such that the moment at w/k becomes the moment at w. From the moment equations the shifted moment also gets multiplied by a factor of k because k particles are making this transition. Thus, because the first moment scales in proportion to weight such that $M_1(w/k) = (w/k)M_1(1)$, when multiplied by this additional factor of k the first moment remains unchanged. However, the second moment, for a weight independent game such as splitting, scales as the weight squared, such that, $M_2(w/k) = (w/k)^2 M_2(1)$, and, when the additional factor of k is accounted for, the second moment is lower $w^2/kM_2(1) < w^2M_2(1)$. Figure 3.7 shows the shift in the second moment resulting from a 2-to-1 split.

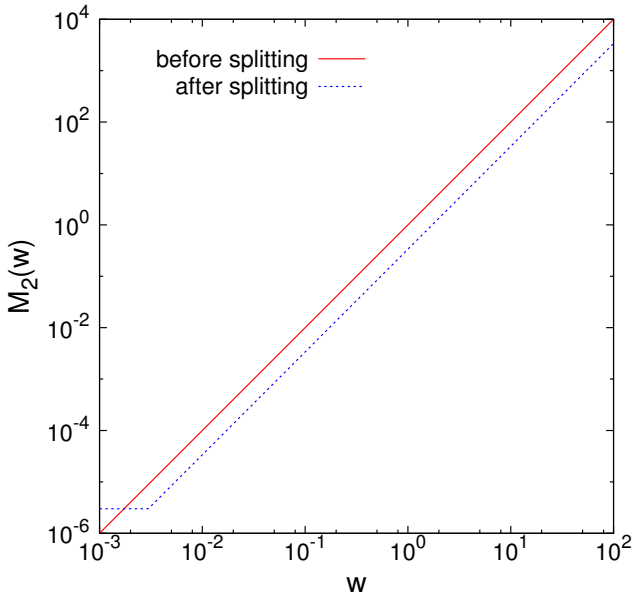


Figure 3.7. The second moment before and after a 2-to-1 splitting event or an implicit capture of survival probability 0.5. The horizontal portion of the after-splitting line is an artifact of the truncation of the weights to w_{tl}

Implicit capture causes a very similar change in the weight distribution as does splitting. One way to write the collision kernel $E(\mathbf{P}, \mathbf{P}')$ is

$$E(\mathbf{P}, \mathbf{P}')d\mathbf{P}' = \frac{\Sigma_s(E')}{\Sigma(E')} P_s(E', \mathbf{\Omega}' \rightarrow E, \mathbf{\Omega})dE'd\mathbf{\Omega}'\delta(\mathbf{r}' - \mathbf{r})d\mathbf{r}'\delta(w' - \omega_{\epsilon_1} w)dw'. \tag{3.47}$$

For normal scattering (without implicit capture) $\omega_{\epsilon_1} = 1$ and the weight of the particle remains unchanged. For implicit capture, the kernel is modified by multiplying by $\Sigma(E')/\Sigma_s(E')$ and setting $\omega_{\epsilon_1} = \Sigma_s(E')/\Sigma(E')$, so that it becomes

$$E(\mathbf{P}, \mathbf{P}')d\mathbf{P}' = P_s(E', \mathbf{\Omega}' \rightarrow E, \mathbf{\Omega})dE'd\mathbf{\Omega}'\delta(\mathbf{r}' - \mathbf{r})d\mathbf{r}'\delta(w' - \omega_{\epsilon_1} w)dw'. \tag{3.48}$$

Here, $\Sigma_s(E')/\Sigma(E')$ is the survival probability of the particle, and in the new kernel all the particles will survive the collision with a modified weight. Considering only the weight portion of the kernel, a particle emerging from the collision will have a weight

$$M_r(\mathbf{r}, \mathbf{\Omega}, E, w) = \int dw' \; \delta(w' - \omega_{\epsilon_1} w) M_r(\mathbf{r}, \mathbf{\Omega}, E, w') = M_r(\mathbf{r}, \mathbf{\Omega}, E, \omega_{\epsilon_1} w), \tag{3.49}$$

indicating that the moment now carried by a particle of weight w should be that of a particle of weight $\omega_{\epsilon_1} w = \Sigma_s(E')w/\Sigma(E')$. For the first moment, again because it scales with weight, $M_1(\mathbf{r}, \mathbf{\Omega}, E, \Sigma_s(E')w/\Sigma(E') = \Sigma_s(E')/\Sigma(E')M_1(\mathbf{r}, \mathbf{\Omega}, E, w)$ and produces exactly what the original kernel would have. For the second moment, because it scales as the square of weight for weight-independent games, $M_2(\mathbf{r}, \mathbf{\Omega}, E, \Sigma_s(E')w/\Sigma(E') = [\Sigma_s(E')/\Sigma(E')]^2 M_2(\mathbf{r}, \mathbf{\Omega}, E, w)$. This value of M_2 is a multiplicative factor of $\Sigma_s(E')/\Sigma(E')$ less than would have been achieved with the original kernel.

3.5.2 Rouletting

The rouletting kernel from Section 2.2.1 is

$$\begin{aligned} B_o(\alpha, \mathbf{P}, \mathbf{P}') &= \delta(\mathbf{r}' - \mathbf{r})d\mathbf{r}'\ \delta(\mathbf{\Omega}' - \mathbf{\Omega})d\mathbf{\Omega}'\ \delta(E' - E)dE' \\ &\times \left[\alpha\delta(w' - w/\alpha) + (1-\alpha)\delta(w')\right] dw'd\alpha. \end{aligned} \tag{3.50}$$

In most practical cases, the survival probability α has a well defined value such that the distribution of α is $\delta(\alpha - \alpha_o)$, where α_o is a defined value. Multiplication of the kernel by $\delta(\alpha - \alpha_o)$ and integration over α gives instead

$$\begin{aligned} B_o(\alpha, \mathbf{P}, \mathbf{P}') &= \delta(\mathbf{r}' - \mathbf{r})d\mathbf{r}'\ \delta(\mathbf{\Omega}' - \mathbf{\Omega})d\mathbf{\Omega}'\ \delta(E' - E)dE' \\ &\times \left[\alpha_o\delta(w' - w/\alpha) + (1-\alpha_o)\delta(w')\right] dw'. \end{aligned} \tag{3.51}$$

Again, only the weight portion of the kernel need be considered because all other variables remain unchanged by this kernel. Thus, the moment emerging from a rouletting event is

$$\begin{aligned} M_r(\mathbf{r}, \mathbf{\Omega}, E, w) &= \int dw'\ [\alpha_o\delta(w' - w/\alpha_o) + (1-\alpha_o)\delta(w')]M_r(\mathbf{r}, \mathbf{\Omega}, E, w') \\ &= \alpha_o M_r(\mathbf{r}, \mathbf{\Omega}, E, w/\alpha_o) + (1-\alpha_o)M_r(\mathbf{r}, \mathbf{\Omega}, E, 0). \end{aligned} \tag{3.52}$$

Because the moment carried by a particle of weight zero is itself zero, the final relationship between the new moment and old moment is

$$M_r(\mathbf{r}, \mathbf{\Omega}, E, w) = \alpha_o M_r(\mathbf{r}, \mathbf{\Omega}, E, w/\alpha_o). \tag{3.53}$$

Because rouletting is a weight independent variance reduction game, the first moment scales with the weight so that $M_1(\mathbf{r}, \mathbf{\Omega}, E, w/\alpha_o) = M_1(\mathbf{r}, \mathbf{\Omega}, E, w)/\alpha_o$ and the first moment remains unchanged. However, the second moment scales with the weight squared, so that $M_2(\mathbf{r}, \mathbf{\Omega}, E, w/\alpha_o) = M_2(\mathbf{r}, \mathbf{\Omega}, E, w)/\alpha_o^2$, and the second moment is a multiplicative factor of $1/\alpha_o$ higher after the rouletting event. Because the second moment is higher, rouletting is a variance increasing event and is used to reduce the time spent tracking unimportant particles. The result of rouletting on the weight domain of the moments is shown in Fig. 3.8.

3.5.3 Weight Cutoff

The weight cutoff game is a weight dependent game and, as such, introduces a discontinuity into the weight domain of the second moment. The kernel for the weight cutoff game is

$$\begin{aligned} B_c(\mathbf{P}, \mathbf{P}')d\mathbf{P}' &= \delta(\mathbf{r}' - \mathbf{r})d\mathbf{r}'\delta(\mathbf{\Omega}' - \delta)d\mathbf{\Omega}'\delta(E' - E)dE' \\ &\times \{\Theta(w - \omega_c)\delta(w' - w) + [1 - \Theta(w - \omega_c)] \\ &\times [(w/\omega_e)\delta(w' - \omega_e) + (1 - w/\omega_e)\delta(w')]\}\, dw'. \end{aligned} \tag{3.54}$$

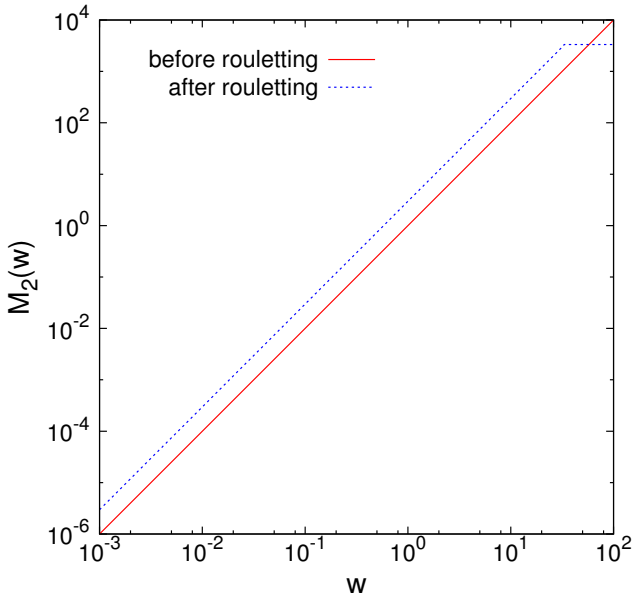


Figure 3.8. The second moment before and after a roulette with survival probability 0.5. The horizontal line in the after-rouletting plot is a result of the upper truncation weight w_{tu}

All other variables other than weight can be ignored because the kernel does not change them. The resulting moment is given by

$$\begin{aligned} M_r(\mathbf{r},\mathbf{\Omega},E,w) &= \int dw' \ \{\Theta(w-\omega_c)\delta(w'-w) + [1-\Theta(w-\omega_c)] \\ &\times \ [(w/\omega_e)\delta(w'-\omega_e) + (1-w/\omega_e)\delta(w')]\}\, M_r(\mathbf{r},\mathbf{\Omega},E,w') \\ &= \Theta(w-\omega_c)M_r(\mathbf{r},\mathbf{\Omega},E,w) + [1-\Theta(w-\omega_c)][(w/\omega_e)M_r(\mathbf{r},\mathbf{\Omega},E,\omega_e) \\ &+ \ (1-w/\omega_e)M_r(\mathbf{r},\mathbf{\Omega},E,0)]. \end{aligned} \tag{3.55}$$

Again, because a moment of weight zero is itself zero, the only remaining terms are

$$M_r(\mathbf{r},\mathbf{\Omega},E,w) = \Theta(w-\omega_c)M_r(\mathbf{r},\mathbf{\Omega},E,w) + [1-\Theta(w-\omega_c)](w/\omega_e)M_r(\mathbf{r},\mathbf{\Omega},E,\omega_e). \tag{3.56}$$

This result indicates, for a weight above the cutoff weight ω_c, the moment remains unchanged. For a weight below the cutoff weight the new moment becomes the ratio of the weight to the survival weight times the moment at the survival weight and, because $\omega_c/\omega_e \neq 1$, a discontinuity is introduced into the weight domain of the second moment.

The first moment always varies in proportion to weight such that for weights below the cutoff weight the first moment is

$$M_1(\mathbf{r},\mathbf{\Omega},E,w) = (w/\omega_e)M_1(\mathbf{r},\mathbf{\Omega},E,\omega_e) = wM_1(\mathbf{r},\mathbf{\Omega}\,E,1) = M_1(\mathbf{r},\mathbf{\Omega},E,w). \tag{3.57}$$

Thus, as expected, the first moment remains unchanged by the weight cutoff game. However, the relationship $M_2(\mathbf{r},\mathbf{\Omega},E,w) = w^2M_2(\mathbf{r},\mathbf{\Omega},E,1)$ no longer holds for this weight dependent game below the weight cutoff. The second moment carried by particles with weights below the weight cutoff is given by

$$M_2(\mathbf{r},\mathbf{\Omega},E,w) = \frac{w}{\omega_e}M_2(\mathbf{r},\mathbf{\Omega},E,\omega_e). \tag{3.58}$$

Interestingly, the second moment now varies as w and not w^2 for weights below the weight cutoff. Furthermore, the second moment is greater than it's previous value making it a variance increasing game. Figure 3.9 illustrates the change in the second moment resulting from the weight cutoff game, and the discontinuity introduced by the weight-dependent game is evident.

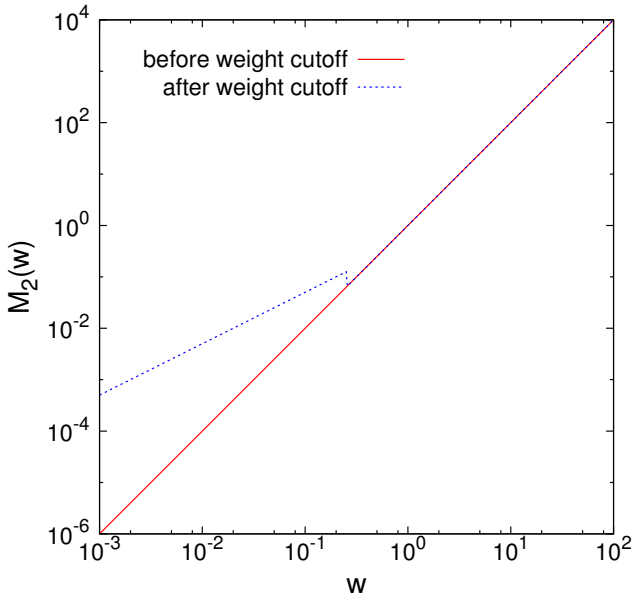


Figure 3.9. The second moment before and after weight cutoff with a cutoff weight of 0.25 and a survival weight of 0.5

3.5.4 Weight Windows

The weight-window game is essentially the combining of a rouletting game, weight-cutoff game, and multiple splitting games done in a cohesive manner. The weight window kernel given by Eq. (2.85) is

$$\begin{aligned} B_w(\mathbf{P},\mathbf{P}') &= [1-\Theta(w-\omega_s/K)]\,B_r(\mathbf{P},\mathbf{P}';1/K) \\ &+ \Theta(w-\omega_s/K)\,[1-\Theta(w-\omega_l)]\,B_r(\mathbf{P},\mathbf{P}';w/\omega_s) \\ &+ \Theta(w-\omega_l)\,[1-\Theta(w-\omega_u)]\,\delta(\mathbf{P}'-\mathbf{P}) + \\ &+ \sum_{k=2}^{K-1}\Theta(w-(k-1)\omega_u)\,[1-\Theta(w-k\omega_u)]\,S_k(\mathbf{P})B_{sk}(\mathbf{P},\mathbf{P}') \\ &+ \Theta(w-K\omega_u)B_{sK}(\mathbf{P},\mathbf{P}'). \end{aligned} \tag{3.59}$$

The first term acts as a 1-to-K rouletting term, where K is the maximum splitting or rouletting parameter. The second term is similarly a rouletting term where the survival probability depends on the weight, thereby making it behave much like the weight cutoff term. The third term handles the splitting of particles by factors of 2 through $K-1$, and the final term handles the splitting of particles by the maximum factor of K.

Each of the kernels behaves as previously described, though it is only applied over the portion of the weight domain permitted by the Heaviside functions. Because all of the kernels contained

in the weight-window kernel leave the first moment unchanged, the kernel as a whole does so as well. However, the second moment is affected by each kernel in the appropriate regime as shown in Fig. 3.10. The portion of the weight domain that is the same before and after the weight-windows game results from the third term of the kernel. The small portion of the plot to the left of this region with with a slope of w rather than w^2 comes from the second term in the kernel and is much like the result of the weight cutoff game. Even further to the left is the $1/K$ maximum rouletting region. To the right of the unchanged region the second moment after the weight windows game decreases in steps with each step corresponding to a different splitting factor k up to the maximum splitting parameter K.

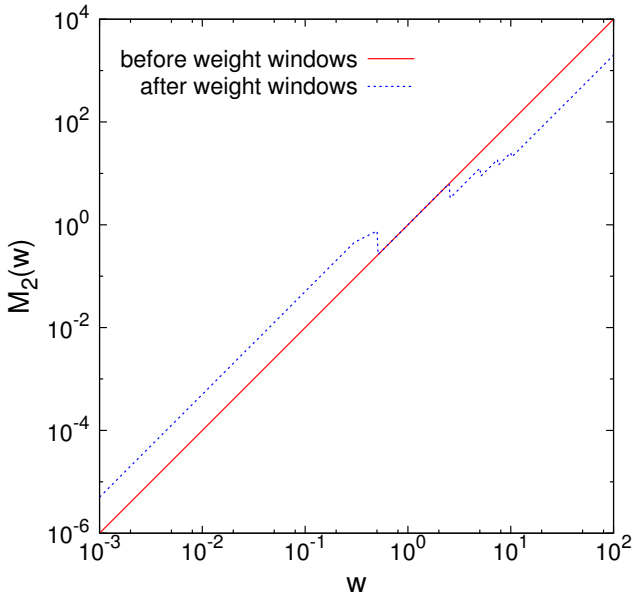


Figure 3.10. The second moment before and after the weight-windows game with a lower bound of 0.5, an upper bound multiplier of 5, a survival multiplier of 3, and a maximum splitting/rouletting parameter of 5

3.5.5 Combined Effect of Multiple Variance Reduction Games

The plots above have shown the effect of only a single variance reduction game on the weight domain. When these games are used in conjunction, the weight domain can become very complex, especially for weight-dependent games. Figure 3.11 shows the weight domain after five different modifications by weight windows with different lower bounds. The number of discontinuities has greatly compounded and in no region does the resulting weight domain resemble that of the original.

3.5.6 Interpolation of Weight Domain with Discontinuities

The method by which values for the moments at a given weight are interpolated during the calculations was previously discussed in Section 3.5.6. While many different interpolation schemes were

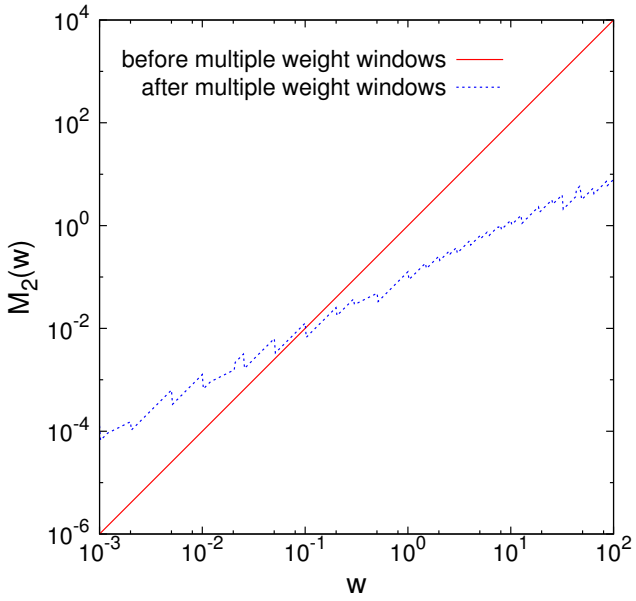


Figure 3.11. The second moment before any and after five applications of the weight-windows game to the weight domain with different weight-window lower bounds, an upper bound multiplier of 5, a survival multiplier of 3, and a maximum splitting/rouletting parameter of 5

attempted, the method described in that section was in the end chosen for its simplicity as none of the other interpolation methods were able to overcome problems interpolating at discontinuities in the weight domain. The problem is that the weight domain is tracked at a finite number of weights w_k and, if a true discontinuity exists between w_k and w_{k+1}, the exact location of that discontinuity is lost by the discretization. Because the second moment can vary substantially on both sides of a discontinuity, the interpolation method used can occasionally produce an interpolated value from the incorrect side of the discontinuity. In general, the errors introduced by incorrect interpolations are somewhat dispersed over a large problem. Even so, when obtaining the final value of the second moment it was found that a wrong interpolation can cause substantial errors (10–20%) in the predicted variance.

3.6 A Deterministic Expected Track Length to Next Event Estimator

A common method of calculating particle fluence in a volume for a Monte Carlo particle transport simulation is to determine the expected particle track length in the volume and divide by the volume itself. This quantity is known as the *track-length fluence estimator*. Such an estimator is a nondeterministic function of phase space $\mathbf{R} = (\mathbf{r}, \mathbf{\Omega}, E, t)$ because the distance to particle collision is a random variable. Other estimators, such as the surface fluence estimator, are deterministic functions of phase space. Here, an expected track-length estimator is developed that is a deterministic function of phase space and can therefore be used as a source for the deterministic S_n

calculations.

To calculate the variance of a track-length estimator using a deterministic S_n approach to solve the history-score moment equations, the score in the Monte Carlo calculation is the source term for the S_n calculation. S_n calculations usually have deterministic source terms, and typical S_n calculations cannot use the stochastic track-length estimator as the source. Thus, an approximate method for deterministically representing track-length estimators is needed, and, in this work, an expected-track-length-to-next-event estimate is used, where an event is defined as a collision or surface crossing.

3.6.1 Derivation of the Expected Track Length to Next Event

Consider a transport problem where a particle located at $\mathbf{r}$ is traveling in direction $\mathbf{\Omega}$ toward a cell boundary surface at $\mathbf{r}_s$, as illustrated in Fig. 3.12. The probability $p(s)ds$ that the particle collides in ds about s[3] is known to be

$$p(s)ds = \Sigma e^{-\Sigma s} ds, \tag{3.60}$$

where Σ is the total macroscopic cross section. If the particle is allowed to free flight forever $s \to \infty$, then it is well known that the expected distance to collision is $1/\Sigma$.

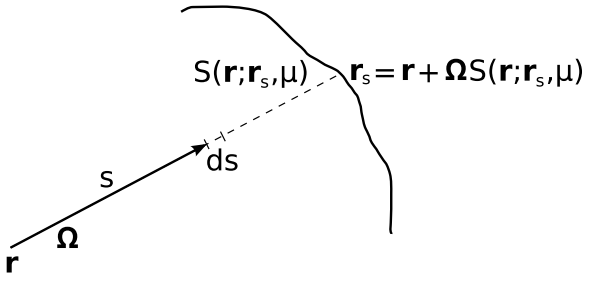


Figure 3.12. Expected track length to next event example

Here, the quantity of interest is not the distance to collision but the expected track length from $\mathbf{r}$ to the boundary at $\mathbf{r}_s$. Therefore the expected distance to collision from the current location at $\mathbf{r}$ to $\mathbf{r}_s$, not ∞, is required. This value is determined by integrating $sp(s)ds$ from 0 to the distance to the boundary $S(\mathbf{r};\mathbf{r}_s,\mathbf{\Omega})$, where the distance is a function of $\mathbf{r}$ and parameterized in $\mathbf{r}_s$ and $\mathbf{\Omega}$, normalized to the interaction probability from $\mathbf{r}$ to $\mathbf{r}_s$, namely

$$\langle s \rangle = \frac{\int_0^{S(\mathbf{r};\mathbf{r}_s,\mathbf{\Omega})} ds\, s\Sigma e^{-\Sigma s}}{\int_0^{S(\mathbf{r};\mathbf{r}_s,\mathbf{\Omega})} ds\, \Sigma e^{-\Sigma s}}. \tag{3.61}$$

The denominator can be directly integrated, and integrating the numerator by parts yields

$$\langle s \rangle = \frac{\left[-se^{-\Sigma s} - \frac{1}{\Sigma}e^{-\Sigma s}\right]_0^{S(\mathbf{r};\mathbf{r}_s,\mathbf{\Omega})}}{1 - e^{-\Sigma S(\mathbf{r};\mathbf{r}_s,\mathbf{\Omega})}} = \frac{\frac{1}{\Sigma} - \left[\frac{1}{\Sigma} + S(\mathbf{r};\mathbf{r}_s,\mathbf{\Omega})\right] e^{-\Sigma S(\mathbf{r};\mathbf{r}_s,\mathbf{\Omega})}}{1 - e^{-\Sigma S(\mathbf{r};\mathbf{r}_s,\mathbf{\Omega})}}. \tag{3.62}$$

In the limit that $S(\mathbf{r};\mathbf{r}_s,\mathbf{\Omega}) \to \infty$, the expected distance to collision returns the expected $1/\Sigma$, and, in the limit that $S(\mathbf{r};\mathbf{r}_s,\mathbf{\Omega}) \to 0$, by application of l'Hôpital's rule, $\langle s \rangle = 0$, also as expected. Thus,

[3]Here, the symbol s temporarily refers to a distance and not a score

the asymptotic behavior of the expression for $\langle s \rangle$ is as expected, but this is simply the expected distance to collision up to $\mathbf{r}_s$ and not the expected track length to next event.

The expected track length to next event (either collision or surface crossing) can now be found by evaluating a weighted average of the free flight distance to the surface and the expected distance to collision up to the surface. Thus, the expected track length to next event $\langle T_l \rangle$ is given by

$$\begin{aligned} \langle T_l \rangle &= \boxed{\text{free-flight distance of } S(\mathbf{r};\mathbf{r}_s,\mathbf{\Omega})} \times \boxed{\text{probability of free-flight of } S(\mathbf{r};\mathbf{r}_s,\mathbf{\Omega})} \\ &+ \boxed{\text{expected collision distance in } S(\mathbf{r};\mathbf{r}_s,\mathbf{\Omega})} \times \boxed{\text{probability of collision in } S(\mathbf{r};\mathbf{r}_s,\mathbf{\Omega})} . \end{aligned} \tag{3.63}$$

This evaluates to

$$\begin{aligned} \langle T_l \rangle &= S(\mathbf{r};\mathbf{r}_s,\mathbf{\Omega}) e^{-\Sigma S(\mathbf{r};\mathbf{r}_s,\mathbf{\Omega})} + \frac{\frac{1}{\Sigma} - \left[\frac{1}{\Sigma} + S(\mathbf{r};\mathbf{r}_s,\mathbf{\Omega})\right] e^{-\Sigma S(\mathbf{r};\mathbf{r}_s,\mathbf{\Omega})}}{1 - e^{-\Sigma S(\mathbf{r};\mathbf{r}_s,\mathbf{\Omega})}} \left(1 - e^{-\Sigma S(\mathbf{r};\mathbf{r}_s,\mathbf{\Omega})}\right) \\ &= S(x;\mathbf{r},\mathbf{\Omega}) e^{-\Sigma S(\mathbf{r};\mathbf{r}_s,\mathbf{\Omega})} + \frac{1}{\Sigma} - \left[\frac{1}{\Sigma} + S(\mathbf{r};\mathbf{r},\mathbf{\Omega})\right] e^{-\Sigma S(\mathbf{r};\mathbf{r}_s,\mathbf{\Omega})} \\ &= \frac{1}{\Sigma}\left(1 - e^{-\Sigma S(\mathbf{r};\mathbf{r}_s,\mathbf{\Omega})}\right). \end{aligned} \tag{3.64}$$

As $S(\mathbf{r};\mathbf{r}_s,\mathbf{\Omega}) \to \infty$ the expression above converges to the expected $1/\Sigma$, and as $S(\mathbf{r};\mathbf{r}_s,\mathbf{\Omega}) \to 0$ it converges to zero, as expected. More interestingly, as $\Sigma \to \infty$ (a perfect absorber) the expected track length to next event becomes zero, and as $\Sigma \to 0$, by application of l'Hôpital's rule, the expected track length to next event becomes $S(\mathbf{r};\mathbf{r}_s,\mathbf{\Omega})$, the distance to the surface. The track length to next event is plotted in Fig. 3.13 as a function of the distance for $\Sigma = 1$ cm^{-1}. Note that the track length to next event asymptotically approaches the mean free path length as S increases and the slope asymptotically approaches unity as S goes to zero, indicating that for small distances the expected track length to next event is that small distance.

For a single dimension, Eq. (3.64) becomes

$$\langle T_l \rangle = \begin{cases} \frac{1}{\Sigma}\left(1 - e^{-\Sigma(x_s - x)/\mu}\right), & (x_s - x) > 0 \text{ and } \mu > 0 \\ \frac{1}{\Sigma}\left(1 - e^{-\Sigma(x_s - x)/\mu}\right), & (x_s - x) < 0 \text{ and } \mu < 0 \\ 0 \quad , & \text{otherwise} \end{cases} , \tag{3.65}$$

where the fact that

$$S(\mathbf{r};\mathbf{r}_s,\mathbf{\Omega}) = \begin{cases} (x_s - x)/\mu, & (x_s - x) > 0 \text{ and } \mu > 0 \\ (x_s - x)/\mu, & (x_s - x) < 0 \text{ and } \mu < 0 \\ 0 \quad , & \text{otherwise} \end{cases} , \tag{3.66}$$

has been used. Here, μ is direction cosine in the x direction and x_s is the x position of the surface, as shown in Fig. 3.14.

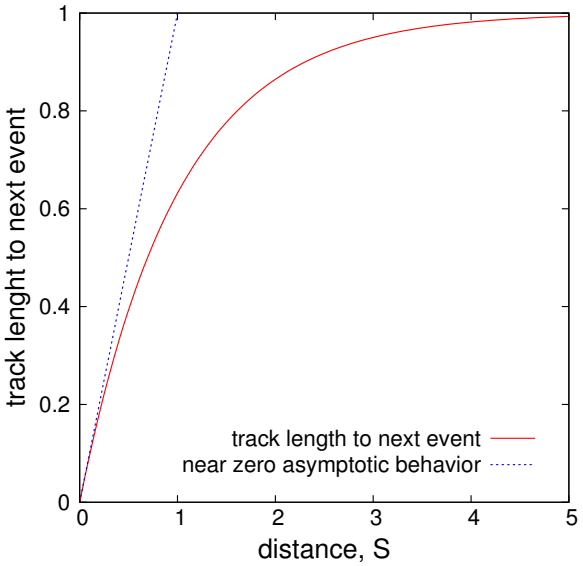


Figure 3.13. Track length to next event as a function of distance to surface S for $\Sigma = 1$ cm^{-1}

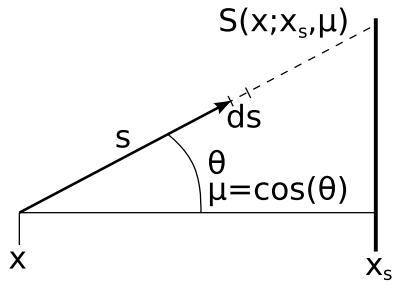


Figure 3.14. 1-D expected track length to next event

3.6.2 Representing Expected Track-Length To Next-Event as an Adjoint S_n Source

To rationalize how the expected track length to the next event is represented as an S_n adjoint source, one first must consider the possible outcomes of track-length measurements for the Monte Carlo simulation. Consider the two cases demonstrated in Fig. 3.15. In case one, the particle flies directly through the tally cell and the entire path length from boundary to boundary is the track-length score. In the second case, the particle enters one boundary and collides. From the collision point on, the particle generates a second track-length score. Thus, track-length scores originate in two locations: (1) from the point the particle enters the tally volume and (2) from collision locations inside the tally volume.

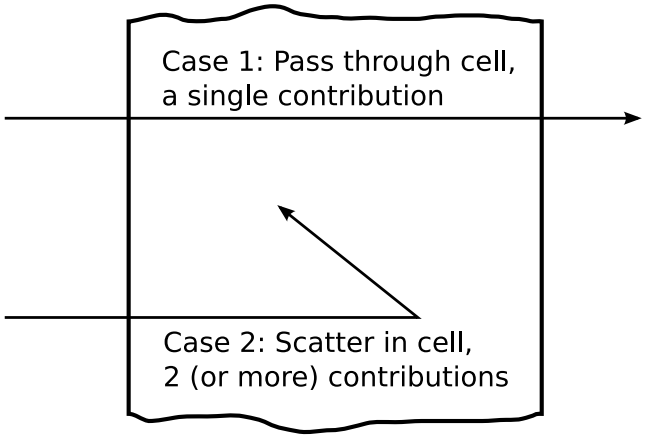


Figure 3.15. Two cases of track length estimation

To represent the track-length scores as the S_n source requires that both types of score contributions, particles entering the tally volume and collisions within the tally volume, be counted. Thus, S_n cell-edge boundaries are forced to coincide with the tally volume boundaries, as shown in Fig. 3.16. Also, as already mentioned, the S_n method cannot model a sampled track length, so the expected track length to next event must be used in the sampled track-length's place. In the forward sense, the expected track length to next event generated at the cell boundary edge nodes is the expected track length to next event across the entire distance of the tally volume. This amount of score accounts for all first entries of particles into the tally volume from regions external to the tally volume. Importantly, this amount of source is only generated in the direction of particles entering the tally volume.

In the forward sense, once inside the tally volume the particles may collide. If they collide, they may exit the collision in any direction and either collide again or exit the tally volume. Thus, the score generated at the cell-center nodes, where collision source is processed, is the expected track length to the next event along the distance from the cell-center node to the boundary of the tally volume in the direction of the discrete ordinate of travel, as indicated in Fig. 3.16. Cell center nodes nearer one boundary will generate smaller expected track lengths to next event along ordinates pointing toward that boundary and progressively larger values as the node moves further from the boundary.

To convert these forward transport scores into a source for the adjoint calculation, one must

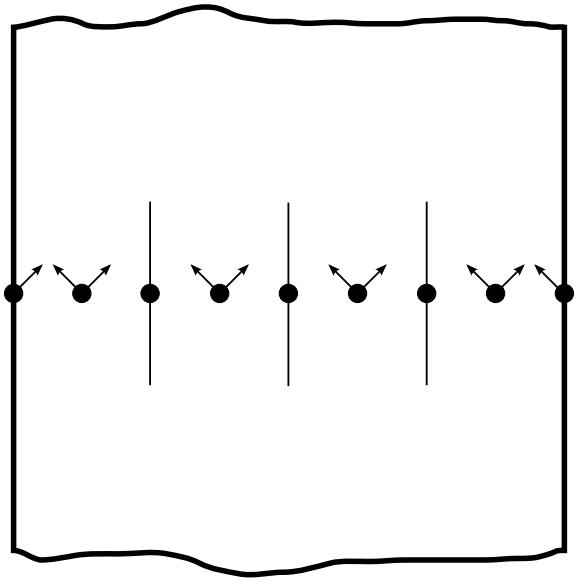


Figure 3.16. Locations and directions of expected track-length scores. The tally cell shown is divided into four S_n cells where collisional contributions to the tally are added at all cell-center nodes and the contribution from particles entering the cell are only for the cell-edge nodes corresponding to the tally cell boundary.

recall that the direction of adjoint particle flight is opposite the direction of forward particle flight. Thus, the scores generated at the tally volume boundaries are converted to adjoint source by reversing the direction of the source contribution, and, therefore, the adjoint source is contributed as the adjoint particle leaves the tally volume. Similarly, the expected-track-length-to-next-event adjoint source contributed at a collision is for the direction opposite that which the adjoint particle enters the collision, because, in the forward sense, the score is generated after the forward particle leaves the collision. The directions of adjoint source contribution are indicated in Fig. 3.17.

3.6.3 Shortcomings of an Expected Track Length Estimator

The expected-track-length estimator works well when the mean-free-path length $1/\Sigma$ is less than or equal to the maximum distance across the cell along one of the discrete ordinates. For example, if x_1 represents the location of the left side of a 1-D cell, x_2 represents the location of the right side of a 1-D cell, and μ_n represent the discrete ordinates, then the expected-track-length estimator works if

$$\frac{1}{\Sigma} \leq \frac{|x_2 - x_1|}{\min(|\mu_n|)}. \tag{3.67}$$

In one- and two-dimensional calculations, the Monte Carlo cells forming the geometry are infinite in two and one directions, respectively. Particles traveling in one of the infinite directions stream until they collide. If the total cross section in that cell is small, then the streaming path is very long, and, in the limit $\Sigma \to 0$, the expected track length is $S(\mathbf{r}, \mathbf{r}_s, \mathbf{\Omega})$. Because there is no surface to bound $S(\mathbf{r}, \mathbf{r}_s, \mathbf{\Omega})$, it is infinite along ordinates that point in the infinite directions and the estimator is infinite. Representing this in S_n requires there be an ordinate that points in the

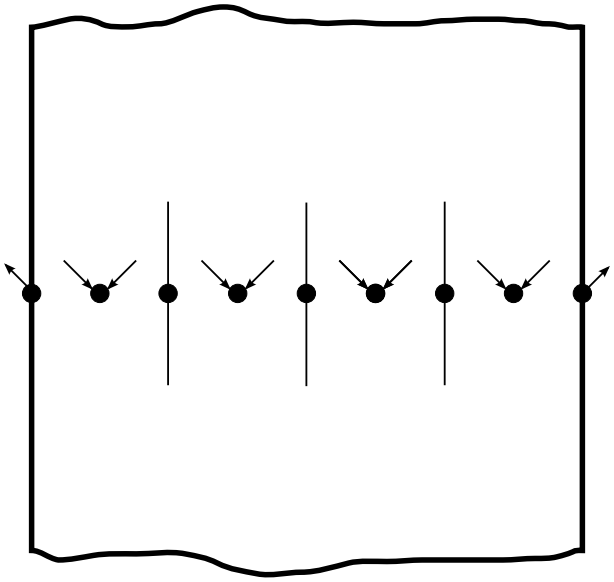


Figure 3.17. Contributions of the adjoint S_n expected track-length source. In contrast to Fig. 3.16, the source at the cell-center nodes, where collisions are processed, is added as adjoint particles enter the collision and similarly for the cell-edge nodes corresponding to the cell boundaries adjoint source is added as the particles leave the tally volume.

direction that could contribute this infinite path length, which is generally not possible. Thus, for the adjoint moment calculations, the source is truncated to values less than or equal to

$$\frac{|x_2 - x_1|}{\min(|\mu_n|)}, \tag{3.68}$$

and, provided the mean free path is less than this value, the expected track length estimator works successfully.

Chapter 4

Estimation of MCNP Calculation Times

To obtain a set of variance reduction parameters that minimizes the cost of a Monte Carlo problem, the average calculation time per history must be known or calculated. This work attempts to calculate the average time per history τ by solving an equation similar to the history score moment equation. Previous works have assumed that the expected number of collisions per history is a descent approximation to the expected time. If the problem is collision dominated, then the expected number of collisions per history is a sufficient approximation to the the expected time, but if scattering is only a small portion of the time required per history then the expected number of collisions per history is a poor approximation to the expected time. This chapter derives the *expected future time equation*, explains how specific timing parameters are extracted from the MCNP transport code, and demonstrates how the future time transitions through phase space.

The future time equation describes the expected future amount of time that a particle at $\mathbf{P}$ expects to experience. This is determined from an adjoint equation very similar to the moment equations, but with the source distributed thoughout the domain of the problem not simply at the tallies. With knowledge of the future time of a particle at $\mathbf{P}$, the expected time per history $\tau(\mathbf{P})$ resulting from a source distribution $S(\mathbf{P})$ may be calculated as

$$\tau = \int d\mathbf{P}\, S(\mathbf{P})\tau(\mathbf{P}). \tag{4.1}$$

Generally, the future time is dependent on the weight of the particle as it transitions through its history. For this reason, the future time and the second moment have much in common, and may be solved with the same S_n sweep. The primary difference however, is that the sources of time in the future-time equation are assumed to be independent of weight, whereas the sources of second moment scale with w^2.

4.1 Derivation of the Future-Time Equation

For this derivation, a time-independent calculation is assumed. The computation time required for a Monte Carlo calculation to process a single history $\tau(\mathbf{P}) = \tau(\mathbf{r}, \mathbf{\Omega}, E, w)$ is sought in terms of the times required to process specific events. The times considered are

$$\tau_t(\mathbf{P}) \quad = \quad \boxed{\begin{array}{l}\text{the time required to process a tally event of a particle} \\ \text{at } \mathbf{P}\end{array}} \tag{4.2}$$

$$\tau_c(\mathbf{P}) = \boxed{\text{the time required to process a collison event of a particle at } \mathbf{P}} \tag{4.3}$$

$$\tau_w(\mathbf{P}) = \boxed{\text{the time required to process a weight-window event of a particle at } \mathbf{P}} \tag{4.4}$$

$$\tau_x(\mathbf{P}) = \boxed{\text{the time required to process a cross-section lookup event of a particle at } \mathbf{P}} \tag{4.5}$$

$$\tau_g(\mathbf{P}) = \boxed{\text{the time required to process a geometry tracking event of a particle at } \mathbf{P}} \tag{4.6}$$

$$\tau_s(\mathbf{P}) = \boxed{\text{the time required to process a surface-crossing event of a particle at } \mathbf{P}} \tag{4.7}$$

$$\tau_b(\mathbf{P}) = \boxed{\text{the time required to process a banking event of a particle at } \mathbf{P}} \tag{4.8}$$

$$\tau_{src}(\mathbf{P}) = \boxed{\text{the time required to process a source event of a particle at } \mathbf{P}}. \tag{4.9}$$

The transport kernels defined in Section 2.2.1 are used in this derivation along with the weight-window kernel in Eq. (2.85). Furthermore, the following probabilities and kernels are also defined

$$S_s(\mathbf{P}^+)d\mathbf{P}^+ = \boxed{\text{The probability of crossing a surface in the Monte Carlo geometry at } d\mathbf{P}^+ \text{ about } \mathbf{P}^+} \tag{4.10}$$

$$W_s(\mathbf{P}^+)d\mathbf{P}^+ = \boxed{\text{The probability of weight window game at surface crossing for a particle at } d\mathbf{P}^+ \text{ about } \mathbf{P}^+} \tag{4.11}$$

$$W_c(\mathbf{P}'')d\mathbf{P}'' = \boxed{\text{The probability of weight window game at a collision for a particle at } d\mathbf{P}'' \text{ about } \mathbf{P}''} \tag{4.12}$$

$$C_c(\mathbf{P}'')d\mathbf{P}'' = \boxed{\text{The probability of weight-cutoff game at a collision for a particle at } d\mathbf{P}'' \text{ about } \mathbf{P}''} \tag{4.13}$$

$$F(\mathbf{P}^+)d\mathbf{P}^+ = \boxed{\text{The probability of a tally in } d\mathbf{P}^+ \text{ about } \mathbf{P}^+} \tag{4.14}$$

4.1.1 Total Future Time from Surface Crossing Event

The time accrued from a surface event of a particle at $\mathbf{P}$ is the probability of transitioning from $\mathbf{P}$ to $\mathbf{P}^+$, times the probability of a surface crossing at $\mathbf{P}^+$, times the calculation time for that particle plus any progeny produced in the crossing. If there is a surface crossing, then the time for the history increases by the amount of time to process the surface crossing $\tau_s(\mathbf{P}^+)$. For rouletting events at the surface crossing, it is assumed that the time required to process the rouletting is a small fraction of the surface crossing time and is negligible. Thus, for a rouletting event, the additional time is given by

$$t_{\text{roulette}} = R_o(\mathbf{P}^+)\int_o^1 d\alpha \int d\mathbf{P}'' B_o(\alpha, \mathbf{P}^+, \mathbf{P}'')[\tau(\mathbf{P}'') + \tau_g(\mathbf{P}'') + \tau_x(\mathbf{P}'')], \tag{4.15}$$

the probability of rouletting, times the transition kernel for rouletted particles, times the future time for the rouletted particle plus the time to track the particle in the geometry and look up the cross sections. Similarly, the splitting of particles is assumed not to increase the time to process the surface crossing substantially beyond the base surface crossing time, so for a splitting event the time is given by

$$t_{\text{split}} = S_k(\mathbf{P}^+) \int d\mathbf{P}'' B_{sk}(\mathbf{P}^+, \mathbf{P}'') k[\tau_b(\mathbf{P}'') + \tau(\mathbf{P}'') + \tau_g(\mathbf{P}'') + \tau_x(\mathbf{P}'')], \tag{4.16}$$

the probability of a split, times the splitting kernel, times the number of particles resulting from the split times the sum of the times each split particle will experience. Each split particle will be banked and removed from the bank giving a time of τ_b, have its geometry tracked for a time of τ_g, have cross sections looked up yielding a time τ_x, and be followed to future events for a time τ.

The weight-windows game is assumed to increase the surface crossing time beyond the base time τ_s by an amount τ_w. However, the time for the weight window is applied only to the particle before it is split, not to progeny. Therefore, the expression for time resulting from weight windows at surface crossing is

$$t_{\text{ww}} = W_s(\mathbf{P}^+) \left(\tau_w(\mathbf{P}^+) + \int d\mathbf{P}^+ B_w(\mathbf{P}^+, \mathbf{P}'') [\tau(\mathbf{P}'') + \tau_g(\mathbf{P}'') + \tau_x(\mathbf{P}'')] \right), \tag{4.17}$$

the probability of weight windows at a surface times the time to process weight windows plus the times from all progeny of the weight windows events. The banking time for the particles resulting from weight window splitting is hidden within the weight window kernel.

The total time from a surface-crossing event forward is the sum of the times for splitting, rouletting, and weight windows plus the surface crossing itself all multiplied by the probability of transitioning to the location of the surface crossing. This whole time is $t_{\text{surface}} = t_{\text{roulette}} + t_{\text{split}} + t_{\text{ww}}$, namely

$$\begin{aligned} t_{\text{surface}} &= \int d\mathbf{P}^+ \, T(\mathbf{P}, \mathbf{P}^+) S_s(\mathbf{P}^+) \Bigg[\tau_s(\mathbf{P}^+) \\ &+ R_o(\mathbf{P}^+) \int_o^1 d\alpha \int d\mathbf{P}'' B_o(\alpha, \mathbf{P}^+, \mathbf{P}'') [\tau(\mathbf{P}'') + \tau_g(\mathbf{P}'') + \tau_x(\mathbf{P}'')] \\ &+ S_k(\mathbf{P}^+) \int d\mathbf{P}'' B_{sk}(\mathbf{P}^+, \mathbf{P}'') k[\tau_b(\mathbf{P}'') + \tau(\mathbf{P}'') + \tau_g(\mathbf{P}'') + \tau_x(\mathbf{P}'')] \\ &+ W_s(\mathbf{P}^+) \left(\tau_w(\mathbf{P}^+) + \int d\mathbf{P}^+ B_w(\mathbf{P}^+, \mathbf{P}'') [\tau(\mathbf{P}'') + \tau_g(\mathbf{P}'') + \tau_x(\mathbf{P}'')] \right) \Bigg]. \end{aligned} \tag{4.18}$$

4.1.2 Total Future Time from Collision Event

Collision events my be broken into the collisions that produce zero, one, or more progeny. The probability that a particle collides is $\Sigma(\mathbf{P}^+, \mathbf{P}')$. A particle producing zero progeny is absorbed such that the only time accrued is the time to process the collision

$$t_{\text{absorption}} = \int d\mathbf{P}' \Sigma(\mathbf{P}^+, \mathbf{P}') A(\mathbf{P}') \tau_c(\mathbf{P}'). \tag{4.19}$$

Particles that scatter (producing a single progeny) produce a total time of the time required to process the collision plus any additional time introduced by weight-cutoff games or weight window games and tracking the scattered particle. The time for scattering events can be expressed as

$$\begin{aligned} t_{\text{scatter}} &= \int \mathbf{P}'\Sigma(\mathbf{P}^+,\mathbf{P}') \int d\mathbf{P}'' E(\mathbf{P}',\mathbf{P}'') \Big\{ \tau_c(\mathbf{P}') + \Big[[1 - C_c(\mathbf{P}'') - W_c(\mathbf{P}'')] \\ &\qquad \times [\tau(\mathbf{P}'') + \tau_g(\mathbf{P}'') + \tau_x(\mathbf{P}'')] \\ &+ C_c(\mathbf{P}'') \int d\mathbf{P}^c \; B_c(\mathbf{P}'',\mathbf{P}^c)[\tau(\mathbf{P}^c) + \tau_g(\mathbf{P}^c) + \tau_x(\mathbf{P}^c)] \\ &+ W_c(\mathbf{P}'') \Big(\tau_w(\mathbf{P}'') + \int d\mathbf{P}^c \; B_w(\mathbf{P}'',\mathbf{P}^c)[\tau(\mathbf{P}^c) + \tau_g(\mathbf{P}^c) + \tau_x(\mathbf{P}^c)] \Big) \Big] \Big\}. \end{aligned} \tag{4.20}$$

Here it has been assumed that the weight-cutoff game adds a negligible amount of time in comparison to the base collision time; however the weight-windows time is non-negligible.

For a collision event yielding multiple progeny, the total time is the time required to process the collision plus all time associated with each of the progeny produced. Each of the progeny may or may not undergo weight cutoff or weight windows games, thus the time for these events is

$$\begin{aligned} t_{\text{mult}} &= \int \mathbf{P}'\Sigma(\mathbf{P}^+,\mathbf{P}') \int d\mathbf{P}_1'' \ldots \int d\mathbf{P}_K'' \epsilon_k(\mathbf{P}',\mathbf{P}_1'',\ldots,\mathbf{P}_n'') \Big\{ \tau_c(\mathbf{P}') \\ &\qquad + \sum_{n=2}^{k} \Big[[1 - C_c(\mathbf{P}_n'') - W_c(\mathbf{P}_n'')][\tau(\mathbf{P}_n'') + \tau_g(\mathbf{P}_n'') + \tau_x(\mathbf{P}_n'')] \\ &+ C_c(\mathbf{P}_n'') \int d\mathbf{P}_n^c \; B_c(\mathbf{P}_n'',\mathbf{P}_n^c)[\tau(\mathbf{P}_n^c) + \tau_g(\mathbf{P}_n^c) + \tau_x(\mathbf{P}_n^c)] \\ &+ W_c(\mathbf{P}_n'') \Big(\tau_w(\mathbf{P}'') + \int d\mathbf{P}_n^c \; B_w(\mathbf{P}_n'',\mathbf{P}_n^c)[\tau(\mathbf{P}_n^c) + \tau_g(\mathbf{P}_n^c) + \tau_x(\mathbf{P}_n^c)] \Big) \Big] \Big\}. \end{aligned} \tag{4.21}$$

Combining the absorption, scattering, and multiplication collision terms and multiplying by the probability of a transition to the collision location gives the time to process a collision and all subsequent time $t_{\text{collision}} = t_{\text{absorption}} + t_{\text{scatter}} + t_{\text{mult}}$, namely

$$\begin{aligned} &\int d\mathbf{P}^+ T(\mathbf{P},\mathbf{P}^+) \int d\mathbf{P}'\Sigma(\mathbf{P}^+,\mathbf{P}') \Big\{ \tau_c(\mathbf{P}') \\ &+ \int d\mathbf{P}'' E(\mathbf{P}',\mathbf{P}'') \Big[[1 - C_c(\mathbf{P}'') - W_c(\mathbf{P}'')][\tau(\mathbf{P}'') + \tau_g(\mathbf{P}'') + \tau_x(\mathbf{P}'')] \\ &+ C_c(\mathbf{P}'') \int d\mathbf{P}^c \; B_c(\mathbf{P}'',\mathbf{P}^c)[\tau(\mathbf{P}^c) + \tau_g(\mathbf{P}^c) + \tau_x(\mathbf{P}^c)] \\ &+ W_c(\mathbf{P}'') \Big(\tau_w(\mathbf{P}'') + \int d\mathbf{P}^c \; B_w(\mathbf{P}'',\mathbf{P}^c)[\tau(\mathbf{P}^c) + \tau_g(\mathbf{P}^c) + \tau_x(\mathbf{P}^c)] \Big) \Big] \\ &+ \int d\mathbf{P}'' \epsilon_k(\mathbf{P}',\mathbf{P}_1'',\ldots,\mathbf{P}_n'') \sum_{n=2}^{k} \Big[[1 - C_c(\mathbf{P}_n'') - W_c(\mathbf{P}_n'')][\tau(\mathbf{P}_n'') + \tau_g(\mathbf{P}_n'') + \tau_x(\mathbf{P}_n'')] \\ &+ C_c(\mathbf{P}_n'') \int d\mathbf{P}_n^c \; B_c(\mathbf{P}_n'',\mathbf{P}_n^c)[\tau(\mathbf{P}_n^c) + \tau_g(\mathbf{P}_n^c) + \tau_x(\mathbf{P}_n^c)] \\ &+ W_c(\mathbf{P}_n'') \Big(\tau_w(\mathbf{P}'') + \int d\mathbf{P}_n^c \; B_w(\mathbf{P}_n'',\mathbf{P}_n^c)[\tau(\mathbf{P}_n^c) + \tau_g(\mathbf{P}_n^c) + \tau_x(\mathbf{P}_n^c)] \Big) \Big] \Big\}, \end{aligned} \tag{4.22}$$

Here the property

$$A(\mathbf{P}') + \int d\mathbf{P}''\ E(\mathbf{P}', \mathbf{P}'') + \int d\mathbf{P}_1'' \ldots \int d\mathbf{P}_n'' \epsilon_K(\mathbf{P}', \mathbf{P}_1'', \ldots, \mathbf{P}_n'') = 1, \tag{4.23}$$

indicating that all colliding particles undergo absorption, scattering, or multiplication, has been used.

4.1.3 Total Future Time Equation

The total future time for a given history is the combination of sum of the time for surface crossings and collisions. This combination gives the equation for the expected future time

$$\begin{aligned}
\tau(\mathbf{P}) \;=\; & \int d\mathbf{P}^+\ T(\mathbf{P}, \mathbf{P}^+) S_{\,s}(\mathbf{P}^+) \Bigg[\tau_s(\mathbf{P}^+) \\
& + \quad R_o(\mathbf{P}^+) \int_o^1 d\alpha \int d\mathbf{P}'' B_o(\alpha, \mathbf{P}^+, \mathbf{P}'')[\tau(\mathbf{P}'') + \tau_g(\mathbf{P}'') + \tau_x(\mathbf{P}'')] \\
& + \quad S_{\,k}(\mathbf{P}^+) \int d\mathbf{P}'' B_{sk}(\mathbf{P}^+, \mathbf{P}'') k[\tau_b(\mathbf{P}'') + \tau(\mathbf{P}'') + \tau_g(\mathbf{P}'') + \tau_x(\mathbf{P}'')] \\
& + \quad W_s(\mathbf{P}^+) \left(\tau_w(\mathbf{P}^+) + \int d\mathbf{P}^+ B_w(\mathbf{P}^+, \mathbf{P}'')[\tau(\mathbf{P}'') + \tau_g(\mathbf{P}'') + \tau_x(\mathbf{P}'')] \right) \Bigg] \\
+ \quad & \int d\mathbf{P}^+ T(\mathbf{P}, \mathbf{P}^+) \int d\mathbf{P}' \Sigma(\mathbf{P}^+, \mathbf{P}') \Bigg\{ \tau_c(\mathbf{P}') \\
& + \quad \int d\mathbf{P}'' E(\mathbf{P}', \mathbf{P}'') \Bigg[[1 - C_c(\mathbf{P}'') - W_c(\mathbf{P}'')][\tau(\mathbf{P}'') + \tau_g(\mathbf{P}'') + \tau_x(\mathbf{P}'')] \\
& + \quad C_c(\mathbf{P}'') \int d\mathbf{P}^c\ B_c(\mathbf{P}'', \mathbf{P}^c)[\tau(\mathbf{P}^c) + \tau_g(\mathbf{P}^c) + \tau_x(\mathbf{P}^c)] \\
& + \quad W_c(\mathbf{P}'') \left(\tau_w(\mathbf{P}'') + \int d\mathbf{P}^c\ B_w(\mathbf{P}'', \mathbf{P}^c)[\tau(\mathbf{P}^c) + \tau_g(\mathbf{P}^c) + \tau_x(\mathbf{P}^c)] \right) \Bigg] \\
& + \quad \int d\mathbf{P}'' \epsilon(\mathbf{P}', \mathbf{P}_1'', \ldots, \mathbf{P}_K'') \sum_{k=2}^{K} \Bigg[[1 - C_c(\mathbf{P}_n'') - W_c(\mathbf{P}_n'')] \\
& \quad \times \quad [\tau(\mathbf{P}_n'') + \tau_g(\mathbf{P}_n'') + \tau_x(\mathbf{P}_n'')] \\
& + \quad C_c(\mathbf{P}_n'') \int d\mathbf{P}_n^c\ B_c(\mathbf{P}_n'', \mathbf{P}_n^c)[\tau(\mathbf{P}_n^c) + \tau_g(\mathbf{P}_n^c) + \tau_x(\mathbf{P}_n^c)] \\
& + \quad W_c(\mathbf{P}_n'') \left(\tau_w(\mathbf{P}'') + \int d\mathbf{P}_n^c\ B_w(\mathbf{P}_n'', \mathbf{P}_n^c)[\tau(\mathbf{P}_n^c) + \tau_g(\mathbf{P}_n^c) + \tau_x(\mathbf{P}_n^c)] \right) \Bigg] \Bigg\} \\
+ \quad & \int d\mathbf{P}^+\ T(\mathbf{P}, \mathbf{P}^+) F(\mathbf{P}^+) \tau_t(\mathbf{P}^+),
\end{aligned} \tag{4.24}$$

where the final term accounts for time required to process tally events at $\mathbf{P}^+$.

4.2 Computing the Expected Future Time

The equation derived in the previous section gives an expression for the total future time of a particle in phase space $\mathbf{P}$. In order to obtain an estimate of the expected future time, the above equation must be integrated over the physical source $S(\mathbf{P})$, just as the first and second moments were, to obtain the time response τ

$$\tau = \int d\mathbf{P} S(\mathbf{P}) [\tau_{src}(\mathbf{P}) + \tau(\mathbf{P})] \tag{4.25}$$

One caveat exists to the above expression. If collision based weight windows are utilized in MCNP, then the source is subject to splitting or rouletting if it falls outside the window. For this reason, the above expression must be modified to account for this possibility as follows

$$\begin{aligned} \tau &= \int d\mathbf{P} S(\mathbf{P}) \Bigg\{ [1 - W_c(\mathbf{P})] [\tau_{src}(\mathbf{P}) + \tau(\mathbf{P})] \\ &+ W_c(\mathbf{P}) \left[\tau_{src}(\mathbf{P}) + \int d\mathbf{P}' \, B_w(\mathbf{P}, \mathbf{P}') \tau(\mathbf{P}') \right] \Bigg\}. \end{aligned} \tag{4.26}$$

The expression above contains an increase in time resulting from both the splitting or rouletting of particles born into the weight window and the addition to the time from determining the initial source particle parameters from the Monte Carlo source routine.

4.3 Determination of MCNP Routine Times

To approximate the future time, it is necessary to obtain from MCNP the times for each type of event $\tau_t(\mathbf{P}), \tau_c(\mathbf{P}), \ldots$. The times that were deemed important to the determination of the future time are those called by MCNP's HSTORY routine. Nearly 95% of the calculation time for the problems investigated is spent in the HSTORY routine or a routine called by HSTORY. Also, the times for specific routines are probably somewhat dependent on the state of the particle $\mathbf{P} = (\mathbf{r}, \mathbf{\Omega}, E, t, w)$, but for this work all the times are considered to be independent of the particle state, i.e., $\tau_t(\mathbf{P}) = \tau_t, \tau_c(\mathbf{P}) = \tau_c, \ldots$, etc.

The times for each subroutine call were determined by profiling the MCNP code. Profiling is typically used to determine where bottlenecks in the source code occur, but can also be used to approximate the times required to process each subroutine. Two different profiling methods were attempted. The first utilized the GNU profiling utility GPROF. Compiling the source code with GPROF capabilities augments the source code with additional instructions that, as the program runs, writes a binary data file with code performance information. After the program has completed, the data file can be analyzed with the GPROF utility providing the total time spent in each routine and the number of times the routine was called. The quotient of these two values provides and estimate of the time per routine. Because the source code is augmented with the additional instructions, the routines themselves are fundamentally altered potentially altering the time required for each call. Initially, it was thought that the times would be reasonable approximations, but results obtained using the profile times obtained from GPROF were not as accurate as hoped.

The second method to obtain times for each event was to use the VALGRIND utility. VALGRIND is typically thought of as a memory debugging tool used to find memory leaks, but it

also contains a profiling utility that works differently than GPROF. VALGRIND's profiling utility CALLGRIND runs the actual program (not an augmented version) under a virtual machine. Because it runs under a virtual machine, the runtime is exceptionally slower than when not running the program under VALGRIND. However, the information CALLGRIND produces is the total number of instructions executed by each subroutine and total, so, if the MCNP problem is run without CALLGRIND, then the total runtime is known. Dividing the total runtime by the total number of instructions, assuming a constant time per instructions, yields the time per instruction from which the time per subroutine can be determined. Using CALLGRIND without an augmented version of the source code and a finer granularity of timing information (total instructions rather than total time) seemed to produce estimates of times adequately, though nowhere near the actual calculation times.

For calculations of the times performed using both GPROF and CALLGRIND it is necessary to subtract the time spent in routines of interest from the calling routine. For example, MCNP's SURFAC routine processes particle's across surfaces of the geometry and, if surface weight windows are used, calls the WTWNDO routine. Because the timing of both SURFAC and WTWNDO are of interest, the time spent processing WTWNDO must be subtracted from the time spent processing SURFAC. Table 4.1 shows which MCNP routines correspond to the times of interest.

Table 4.1. Correspondence between times of interest and MCNP routines

Time	τ_s	τ_c	τ_g	τ_x	τ_t	τ_w	τ_b	τ_{src}
MCNP Routine	SURFAC[a]	COLIDN	TRACK	ACETOT	TALLY	WTWNDO[b]	BANKIT	STARTP[c] + RN_INIT_PARTICLE

[a] less WTWNDO and TALLY
[b] less BANKIT
[c] less WTWNDO

An additional complication arises from the fact that most modern computers do not simply process a single task. The operating system must balance the requirements of the user with the requirements of the system. When running a calculation, if suddenly the system needs processor time, the timing of the profiled calculation can be affected. Using CALLGRIND to track the total number of instructions used helps to mediate this problem, but the calculation to obtain the true runtime is subject to this type of multitasking problem. For this reason, the estimates of the routine times are subject to some fluctuation. However, no other method of obtaining the necessary timing information was attempted.

Three one-dimensional problems were profiled to obtain a set of times for use in the S_n code to predict the MCNP problem runtime. Because the goal is the optimization of weight-window games, the timing information was obtained with specifically the weight-window game in mind. The first problem uses only weight window variance reduction for a 1-group problem, the second problem uses weight window and implicit capture variance reduction for a 1-group problem, and the third problem uses only weight window variance reduction in a 2-group problem. The times obtained from the three profiles are presented in Table 4.2, Table 4.3, and Table 4.4 respectively.

The results of the three profiles agree reasonably well considering. Most of the profiled times agree within 5–10%, which, given all the assumptions, seems like decent agreement. The times

Table 4.2. CALLGRIND time profiling information for only weight windows in a 1-group problem on an AMD Opteron (2.2 GHz) Linux system. Times are in seconds.

MCNP Routine	# Instructions	Time in Routine[a]	Calls	Time Per Call
SURFAC	1.26062E+09[b]	5.89E-01	1.45247E+06	4.05E-07
BANKIT	3.27671E+08	1.53E-01	6.89335E+05	2.22E-07
ACETOT	7.08281E+08	3.31E-01	4.11791E+06	8.04E-08
WTWNDO	3.98725E+08[c]	1.86E-01	3.80505E+06	4.89E-08
TRACK	8.91833E+08	4.16E-01	4.11791E+06	1.01E-07
COLIDN	1.80121E+09	8.42E-01	2.66544E+06	3.15E-07
TALLY	5.34787E+07	2.50E-02	7.28920E+04	3.43E-07
STARTP	1.98047E+09[d]	9.26E-01	1.00000E+06	9.26E-07
RN_INIT_PARTICLE	7.17371E+08	3.35E-01	1.00000E+06	3.35E-07

[a] obtained from a total of 10693583913 instructions in a 5 minute run
[b] less WTWNDO and TALLY
[c] less BANKIT
[d] less WTWNDO

Table 4.3. CALLGRIND time profiling information of weight windows with implicit capture in a 1-group problem on an AMD Opteron (2.2 GHz) Linux System. Times are in seconds.

MCNP Routine	# Instructions	Time in Routine[a]	Calls	Time Per Call
SURFAC	1.46832E+09[b]	6.54E-01	1.70405E+06	3.84E-07
BANKIT	2.64695E+08	1.18E-01	5.60036E+05	2.10E-07
ACETOT	8.31591E+08	3.70E-01	4.83483E+06	7.67E-08
WTWNDO	5.37541E+08[c]	2.39E-01	4.96282E+06	4.83E-08
TRACK	1.05411E+09	4.70E-01	4.83483E+06	9.72E-08
COLIDN	2.92458E+09	1.30E+00	3.13078E+06	4.16E-07
TALLY	8.80225E+07	3.92E-02	1.20069E+05	3.26E-07
STARTP	1.98047E+09[d]	8.83E-01	1.00000E+06	8.83E-07
RN_INIT_PARTICLE	7.17371E+08	3.19E-01	1.00000E+06	3.19E-07

[a] obtained from a total of 12825918309 instructions in a 5.72 minute run
[b] less WTWNDO and TALLY
[c] less BANKIT
[d] less WTWNDO

Table 4.4. CALLGRIND time profiling information of weight windows in a 2-group problem on an AMD Opteron (2.2 GHz) Linux System. Times are in seconds.

MCNP Routine	# Instructions	Time in Routine[a]	Calls	Time Per Call
SURFAC	9.75566E+08[b]	4.79E-01	1.09154E+06	4.39E-07
BANKIT	7.11044E+07	3.49E-02	2.79823E+05	1.24E-07
ACETOT	2.77869E+08	1.36E-01	1.61552E+06	8.45E-08
WTWNDO	1.78849E+08[c]	8.78E-02	1.43465E+06	6.12E-08
TRACK	3.47610E+08	1.70E-01	1.61552E+06	1.05E-07
COLIDN	1.63078E+08	8.01E-02	5.23983E+05	1.53E-07
TALLY	1.14646E+08	5.63E-02	1.30528E+05	4.31E-07
STARTP	2.01149E+09[d]	9.88E-01	1.00000E+06	9.88E-07
RN_INIT_PARTICLE	7.17371E+08	3.52E-01	1.00000E+06	3.52E-07

[a] obtained from a total of 6045414993 instructions in a 2.97 minute run
[b] less WTWNDO and TALLY
[c] less BANKIT
[d] less WTWNDO

that do differ substantially are the COLIDN time, the WTWNDO time, and the BANKIT time. The time to process COLIDN sees a 33% increase in time for the implicit capture case between Table 4.2 and Table 4.3, and is substantially less in Table 4.4, probably a result of the lower cross sections used for this profiling. The difference between the first two COLIDN times results from the additional time of processing the implicit capture and is to be expected. Given these profiles, the times used for calculations presented in this work are those given in Table 4.4 with the exception that the COLLIDN time τ_c is increased by 30%, as indicated by Table 4.2 and Table 4.3, if implicit capture is used.

The profiled times are sure to be problem dependent. However, not every possible problem can be profiled to find the correct set of times. Rather, these timing studies were performed in the expectation that a representative time could be established that might extend to similar problems. In reality, a set of times, dependent on the state of the particle and the variance reduction techniques being used, could probably be developed. A more realistic approach would be to have the S_n code call the Monte Carlo routines directly with the appropriate input parameters and obtain the timings directly as needed. However, such refinements are beyond this "proof of concept" study.

4.4 Solving for the Expected Future Time

As with the moment equations, the future-time equation can be converted into an integro-differential form and discretized using the discrete ordinates method. It is important to note that the future-time equation is dependent on the Monte Carlo particle weight as particles with low weights will be preferentially killed to reduce the cost of a calculation. The discretized future-time equation can be written

$$[-\mathbf{\Omega}\cdot\nabla\tau^{g,w}(\mathbf{r})]^n + \Sigma^g\tau^{g,w,n}(\mathbf{r}) = \mathcal{S}[\tau^{g'',w'',n''}(\mathbf{r}'')] + C[\tau^{g^c,w^c,n^c}(\mathbf{r}^c)] + Q^{g,w,n}(\mathbf{r}) \tag{4.27}$$

where $\mathcal{S}$ and C are the surface-crossing and collision operators given by

$$\begin{aligned}
\mathcal{S}[\tau(\mathbf{P}'')] \;=\; & \tau_s(\mathbf{P}^+) \\
& + \; R_o(\mathbf{P}^+)\int_o^1 d\alpha \int d\mathbf{P}'' B_o(\alpha,\mathbf{P}^+,\mathbf{P}'')[\tau(\mathbf{P}'')+\tau_g(\mathbf{P}'')+\tau_x(\mathbf{P}'')] \\
& + \; S_k(\mathbf{P}^+)\int d\mathbf{P}'' B_{sk}(\mathbf{P}^+,\mathbf{P}'')k[\tau_b(\mathbf{P}'')+\tau(\mathbf{P}'')+\tau_g(\mathbf{P}'')+\tau_x(\mathbf{P}'')] \\
& + \; W_s(\mathbf{P}^+)\left(\tau_w(\mathbf{P}^+)+\int d\mathbf{P}^+ B_w(\mathbf{P}^+,\mathbf{P}'')[\tau(\mathbf{P}'')+\tau_g(\mathbf{P}'')+\tau_x(\mathbf{P}'')]\right)
\end{aligned} \tag{4.28}$$

and

$$\begin{aligned}
C[\tau(\mathbf{P}^c)] \;=\; & \tau_c(\mathbf{P}') \\
& + \int d\mathbf{P}'' E(\mathbf{P}',\mathbf{P}'')\Big[[1-C_c(\mathbf{P}'')-W_c(\mathbf{P}'')][\tau(\mathbf{P}'')+\tau_g(\mathbf{P}'')+\tau_x(\mathbf{P}'')] \\
& + \; C_c(\mathbf{P}'')\int d\mathbf{P}^c\; B_c(\mathbf{P}'',\mathbf{P}^c)[\tau(\mathbf{P}^c)+\tau_g(\mathbf{P}^c)+\tau_x(\mathbf{P}^c)] \\
& + \; W_c(\mathbf{P}'')\left(\tau_w(\mathbf{P}'')+\int d\mathbf{P}^c\; B_w(\mathbf{P}'',\mathbf{P}^c)[\tau(\mathbf{P}^c)+\tau_g(\mathbf{P}^c)+\tau_x(\mathbf{P}^c)]\right)\Big]
\end{aligned}$$

$$
\begin{aligned}
&+ \quad \int d\mathbf{P}''\epsilon(\mathbf{P}', \mathbf{P}_1'', \ldots, \mathbf{P}_K'') \sum_{k=2}^{K} \Bigg[[1 - C_c(\mathbf{P}_n'') - W_c(\mathbf{P}_n'')] \\
&\quad \times \quad [\tau(\mathbf{P}_n'') + \tau_g(\mathbf{P}_n'') + \tau_x(\mathbf{P}_n'')] \\
&+ \quad C_c(\mathbf{P}_n'') \int d\mathbf{P}_n^c \, B_c(\mathbf{P}_n'', \mathbf{P}_n^c)[\tau(\mathbf{P}_n^c) + \tau_g(\mathbf{P}_n^c) + \tau_x(\mathbf{P}_n^c)] \\
&+ \quad W_c(\mathbf{P}_n'') \Bigg(\tau_w(\mathbf{P}'') + \int d\mathbf{P}_n^c \, B_w(\mathbf{P}_n'', \mathbf{P}_n^c)[\tau(\mathbf{P}_n^c) + \tau_g(\mathbf{P}_n^c) + \tau_x(\mathbf{P}_n^c)] \Bigg) \Bigg]
\end{aligned} \tag{4.29}
$$

This integro-differential future-time equation is also adjoint to the transport equation. As such, the source term Q contains the time required to process a tally, although additional sources of time are contained within the $\mathcal{S}$ and C operators. The most fundamental difference between the history-score moment equations and the future-time equation is that the sources are independent of the Monte Carlo particle weight, whereas the sources for the moment equations scaled as w for M_1 and w^2 for M_2. The effects that the weight transitions kernels have on the future-time are detailed in the next section.

4.5 Effects of the Weight Kernels on the Future Time

Just as with the history-score moment equation, the future-time equation is also affected by the weight transition kernels. Because a split of k-to-1 requires approximately k-times as long to process from that point forward and a roulette of 1-for-k requires $1/k$-times as long to process from that point forward, the future-time is dependent on how the weight changes.

An important distinction exists between the sources for the history-score moment equations and the future-time equations. The tallies, which are the source for the history-score moment equations, vary as w for the first moment and w^2 for the second moment, whereas the time to process events is assumed to be independent of particle weight. Because the sources of time are considered to be independent of particle weight, the weight kernels have different effects on the future-time weight distribution.

The following sections describe how each of the weight transition kernels affects the future-time weight distribution. For each case the initial future-time weight distribution is assumed to be weigh independent with a value of unity. The effect of each weight-transition kernel on the future-time weight distribution is shown mathematically as well as graphically.

4.5.1 Splitting and Implicit Capture

The splitting kernel, given in Section 2.2.1, indicates that

$$
\tau(\mathbf{r}, \mathbf{\Omega}, E, w) = \int dw' \delta(w' - w/k)\tau(\mathbf{r}, \mathbf{\Omega}, E, w') = \tau(\mathbf{r}, \mathbf{\Omega}, E, w/k). \tag{4.30}
$$

This result indicates that the time carried by a particle of weight w should now be the time carried by a particle of weight w/k. This kernel omits the fact that there are now k particle to be tracked, but, after accounting for these k particles, the correct time is

$$
\tau(\mathbf{r}, \mathbf{\Omega}, E, w) = k\tau(\mathbf{r}, \mathbf{\Omega}, E, w/k). \tag{4.31}
$$

Because the original time is weight independent, $\tau(\mathbf{r}, \mathbf{\Omega}, E, w) = \tau(\mathbf{r}, \mathbf{\Omega}, E, w/k)$ and the entire future-time weight domain increases by a factor of k. This behavior is illustrated in Fig. 4.1.

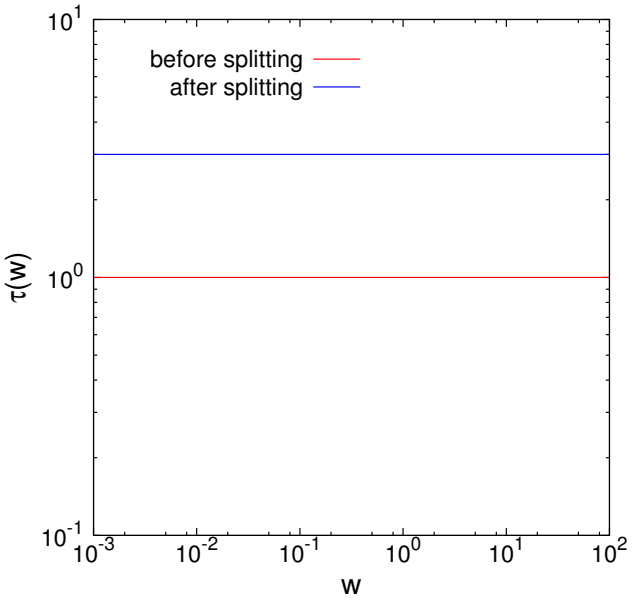


Figure 4.1. The effect of a 3-to-1 split on the weight domain of the future time

Implicit capture has no effect on the future-time of a weight-independent distribution. It was previously established that the weight-only portion of the implicit capture kernel modifies the weight domain as

$$\begin{aligned} \tau(\mathbf{r}, \mathbf{\Omega}, E, w) &= \int dw' \; \delta(w' - \omega_{\epsilon_1} w) \tau(\mathbf{r}, \mathbf{\Omega}, E, w') \\ &= \tau(\mathbf{r}, \mathbf{\Omega}, E, \omega_{\epsilon_1} w), \end{aligned} \tag{4.32}$$

where ω_{ϵ_1} is the scattering probability $\Sigma_s(E)/\Sigma(E)$. However, because the initial future-time distribution is weight-independent, $\tau(\mathbf{r}, \mathbf{\Omega}, E, \omega_{\epsilon_1} w) = \tau(\mathbf{r}, \mathbf{\Omega}, E, w)$ and the future-time distribution remains unchanged. Were the original distribution weight-dependent, then the time required to track a particle of weight w would now be the time required to track a particle of weight $\Sigma_s(E)w/\Sigma$.

4.5.2 Rouletting

In the case of rouletting, the future-time is affected in the opposite manner as splitting. The rouletting kernel for a specific rouletting probability α_o is

$$\begin{aligned} B_o(\alpha_o, \mathbf{P}, \mathbf{P}') &= \delta(\mathbf{r}' - \mathbf{r}) d\mathbf{r}' \; \delta(\mathbf{\Omega}' - \mathbf{\Omega}) d\mathbf{\Omega}' \; \delta(E' - E) dE' \\ &\times \; [\alpha_o \delta(w' - w/\alpha_o) + (1 - \alpha_o)\delta(w')] \, dw'. \end{aligned} \tag{4.33}$$

Only the weight variable is changed by this kernel, so the effect on the future-time is

$$\begin{aligned} \tau(\mathbf{r}, \mathbf{\Omega}, E, w) &= \int dw' \; [\alpha_o \delta('-w/\alpha_o) + (1 - \alpha_o)\delta(w')] \tau(\mathbf{r}, \mathbf{\Omega}, E, w') \\ &= \alpha_o \tau(\mathbf{r}, \mathbf{\Omega}, E, w/\alpha_o) + (1 - \alpha_o)\tau(\mathbf{r}, \mathbf{\Omega}, E, 0). \end{aligned} \tag{4.34}$$

Because the time required to track particles of weight zero is itself zero (the particles are terminated), the resulting value for $\tau(\mathbf{r}, \mathbf{\Omega}, E, w)$ is $\alpha_o \tau(\mathbf{r}, \mathbf{\Omega}, E, w/\alpha_o)$. Again, because the initial future-time distribution is weight independent such that $\tau(\mathbf{r}, \mathbf{\Omega}, E, w/\alpha_o) = \tau(\mathbf{r}, \mathbf{\Omega}, E, w)$, the resulting future-time is simply

$$\tau(\mathbf{r}, \mathbf{\Omega}, E, w) = \alpha_o \tau(\mathbf{r}, \mathbf{\Omega}, E, w) \tag{4.35}$$

Thus, the expected time to track particles from this point forward in the calculation is a factor of $\alpha_o < 1$ less after the rouletting event as shown in Fig. 4.2.

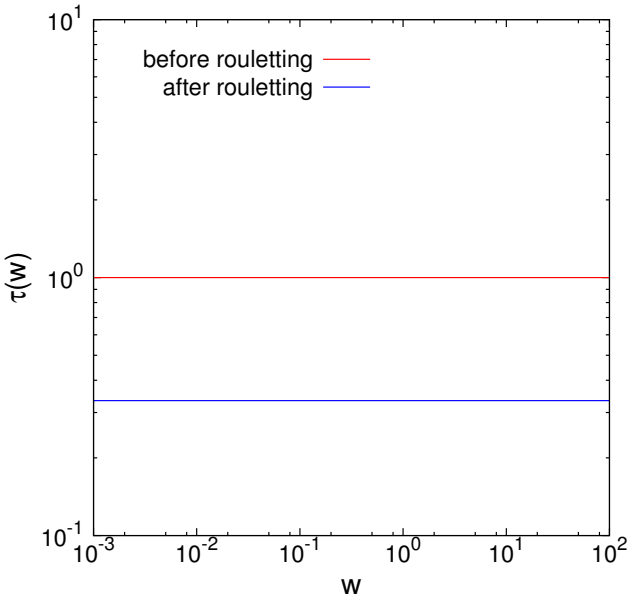


Figure 4.2. The effect of rouletting with a probability of 1/3 on the weight domain of the future time

4.5.3 Weight Cutoff

The weight cutoff kernel introduces a weight dependence into a weight-independent future-time distribution. The weight-cutoff kernel is

$$\begin{aligned} B_c(\mathbf{P}, \mathbf{P}')d\mathbf{P}' &= \delta(\mathbf{r}' - \mathbf{r})d\mathbf{r}'\delta(\mathbf{\Omega}' - \delta)d\mathbf{\Omega}'\delta(E' - E)dE' \\ &\times \{\Theta(w - \omega_c)\delta(w' - w) + [1 - \Theta(w - \omega_c)] \\ &\times [(w/\omega_e)\delta(w' - \omega_e) + (1 - w/\omega_e)\delta(w')]\}\, dw', \end{aligned} \tag{4.36}$$

where, ω_c and ω_s are the cutoff weights and survival weights, respectively. Application of this kernel to the future-time, ignoring all other variables than weight, gives

$$\begin{aligned} \tau(\mathbf{r}, \mathbf{\Omega}, E, w) &= \int dw' \ \{\Theta(w - \omega_c)\delta(w' - w) + [1 - \Theta(w - \omega_c)] \\ &\times [(w/\omega_e)\delta(w' - \omega_e) + (1 - w/\omega_e)\delta(w')]\}\, \tau(\mathbf{r}, \mathbf{\Omega}, E, w') \\ &= \Theta(w - \omega_c)\tau(\mathbf{r}, \mathbf{\Omega}, E, w) + [1 - \Theta(w - \omega_c)][(w/\omega_e)\tau(\mathbf{r}, \mathbf{\Omega}, E, \omega_e) \\ &+ (1 - w/\omega_e)\tau(\mathbf{r}, \mathbf{\Omega}, E, 0)]. \end{aligned} \tag{4.37}$$

Again, because the time required to track particles with weight zero is zero, the only terms that need be considered are

$$\tau(\mathbf{r}, \mathbf{\Omega}, E, w) = \Theta(w - \omega_c)\tau(\mathbf{r}, \mathbf{\Omega}, E, w) + [1 - \Theta(w - \omega_c)](w/\omega_e)\tau(\mathbf{r}, \mathbf{\Omega}, E, \omega_e). \tag{4.38}$$

The time required to track particles above the weight cutoff remains unchanged. However, the time required to track particles below the weight cutoff decreases as the weight decreases as the weight decreases and the probability of surviving the weight cutoff decreases. The effect the weight cutoff kernel has on an initially weight-independent future-time is presented in Fig. 4.3.

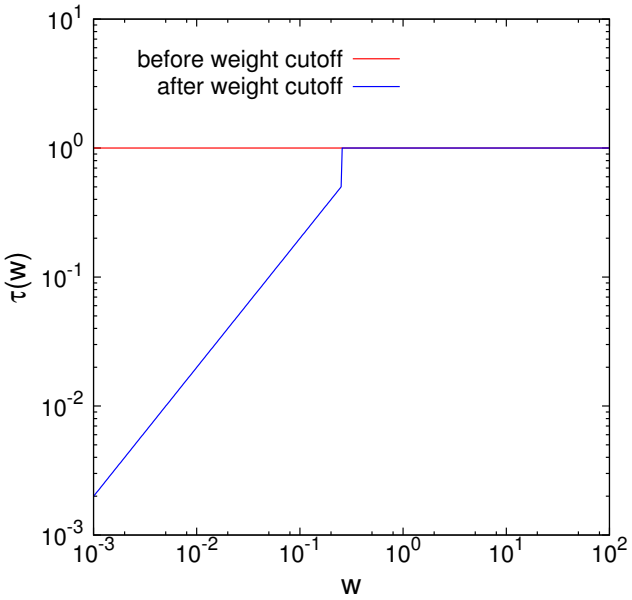


Figure 4.3. The effect of a weight cutoff on the weight domain of the future-time for a cut weight of 0.25 and a survival weight of 0.5

4.5.4 Weight Windows

The weight-window game combines the rouletting, weight cutoff, and splitting games into a single weight dependent game. The weight-window kernel is

$$\begin{aligned} B_w(\mathbf{P}, \mathbf{P}') &= [1 - \Theta(w - \omega_s/K)]\, B_r(\mathbf{P}, \mathbf{P}'; 1/K) \\ &+ \Theta(w - \omega_s/K)\,[1 - \Theta(w - \omega_l)]\, B_r(\mathbf{P}, \mathbf{P}'; w/\omega_s) \\ &+ \Theta(w - \omega_l)\,[1 - \Theta(w - \omega_u)]\,\delta(\mathbf{P}' - \mathbf{P}) + \\ &+ \sum_{k=2}^{K-1} \Theta(w - (k-1)\omega_u)\,[1 - \Theta(w - k\omega_u)]\, S_k(\mathbf{P}) B_{sk}(\mathbf{P}, \mathbf{P}') \\ &+ \Theta(w - K\omega_u) B_{sK}(\mathbf{P}, \mathbf{P}'), \end{aligned} \tag{4.39}$$

where ω_s is the survival weight, ω_l is the lower window weight, ω_u is the upper window weight, and K is the maximum splitting/rouletting parameter. For the future-time, each of the splitting

terms must also be multiplied by the number of particles produced in the split, namely

$$\begin{aligned} B_w(\mathbf{P},\mathbf{P}') &= [1-\Theta(w-\omega_s/K)]\, B_r(\mathbf{P},\mathbf{P}';1/K) \\ &+ \Theta(w-\omega_s/K)\,[1-\Theta(w-\omega_l)]\, B_r(\mathbf{P},\mathbf{P}';w/\omega_s) \\ &+ \Theta(w-\omega_l)\,[1-\Theta(w-\omega_u)]\,\delta(\mathbf{P}'-\mathbf{P}) + \\ &+ \sum_{k=2}^{K-1} \Theta(w-(k-1)\omega_u)\,[1-\Theta(w-k\omega_u)]\, S_k(\mathbf{P}) B_{sk}(\mathbf{P},\mathbf{P}')k \\ &+ \Theta(w-K\omega_u) B_{sK}(\mathbf{P},\mathbf{P}')K. \end{aligned} \tag{4.40}$$

The resulting kernel modifies weights below ω_s/K by a 1-for-K rouletting and weights between ω_s/K and ω_l by a weight-cutoff with survival weight ω_s. Both of these games, as was seen above, reduce the future-time. The future time for particles with weights between ω_l and ω_u is left unchanged, while for particles with weights greater than ω_u are split such that their weight is in the window and the future time is multiplied by the appropriate splitting value k. Figure 4.4 shows how an initially weight-independent future-time function is modified by the weight-windows game.

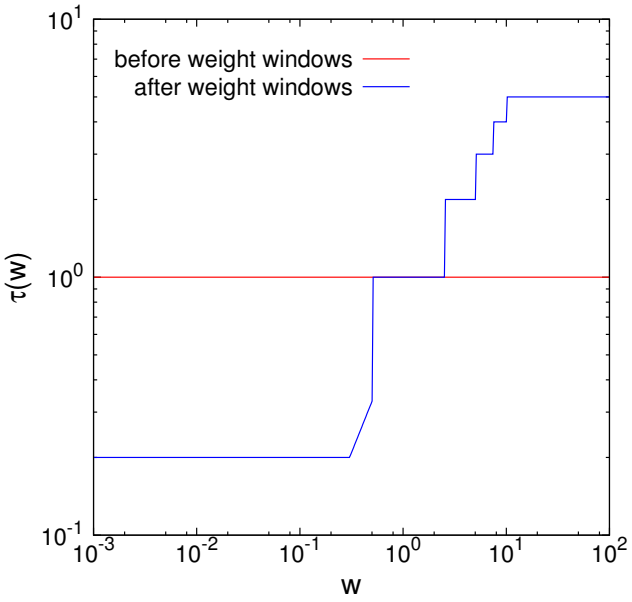


Figure 4.4. The effect of weight windows on the weight domain of the future time for a lower bound of 0.5, a upper bound multiplier of 5, a survival multiplier of 3, and a maximum splitting/rouletting parameter of 5

As soon as the future-time function experiences a single weight windows game, the weight independence of the distribution is lost. Combined effects of multiple weight windows games introduce more discontinuities to the weight domain of the future time as shown in Fig. 4.5.

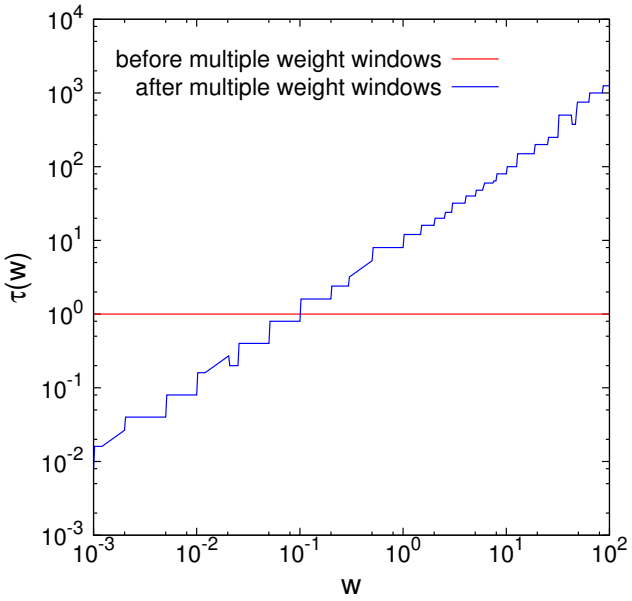


Figure 4.5. The effect of five weight window kernels on the weight domain of the future time

Chapter 5

Cost Optimization of Transport Problems

The methods for calculating population variances of Monte Carlo particle transport problems and estimating the times of those calculations have been developed for the goal of minimizing the computational cost of Monte Carlo transport problems. This chapter presents the optimization method used to minimize the cost function and presents results for 1-D and 2-D transport problems.

5.1 Optimization Method

In the optimization calculations a set of variance reduction parameters $\{V\}$ that minimize the functional cost

$$C(\{V\}) = \sigma^2(\{V\})\tau(\{V\}) \tag{5.1}$$

is sought. Here, the population variance of the tally score distribution σ^2 and the expected time per particle τ have been expressed as functions of the set of variance reduction parameters $\{V\}$. In general, the variance reduction techniques, such as splitting or implicit capture, seek to reduce σ^2 while at the same time to modestly increase τ. Alternatively, variance increasing techniques, such as rouletting or weight cutoff, seek to increase modestly σ^2 while decreasing τ.

The set of variance reduction parameters $\{V\}$ could potentially contain any of the variance-reduction parameters or subset of the parameters in the problem. Realistically, an attempt to optimize the problem with every possible parameter becomes unwieldy, and this work has restricted itself to optimization of the weight-window method because it embodies many of the other variance reduction games.

A general optimization of the weight-windows technique poses an interesting issue. For a general problem, the number of cells can be quite large and each cell requires a weight-window lower bound that must be optimized. As the number of cells grows, the number of S_n calculations that must be performed for the optimization grows with it, and, because each S_n calculation is somewhat computationally expensive, the optimization becomes unwieldy for gradient based optimization method. To overcome this problem, a basis function method that reduces the number of S_n calculations is adopted.

5.1.1 Basis Functions

Use of the adjoint function as an importance biasing function or a method of producing weight windows has proven to be a quite successful means of biasing forward Monte Carlo calculations. As such, the adjoint function presents itself as a reasonable candidate for an initial guess, upon which to improve. For the calculation contained here, the adjoint function is computed, converted into a set of weight window lower bounds for each cell, and then those lower bounds are used as a basis to be optimized.

Having computed an initial value for the weight window lower bound in each cell, a method of improving that basis without computing the gradient with respect to every weight-window lower bound is required. To this end, the problem domain is sub-divided into a courser *optimization mesh*, and on each of the optimization-mesh elements a scaling parameter is applied. Figure 5.1 shows how such a mesh is applied.

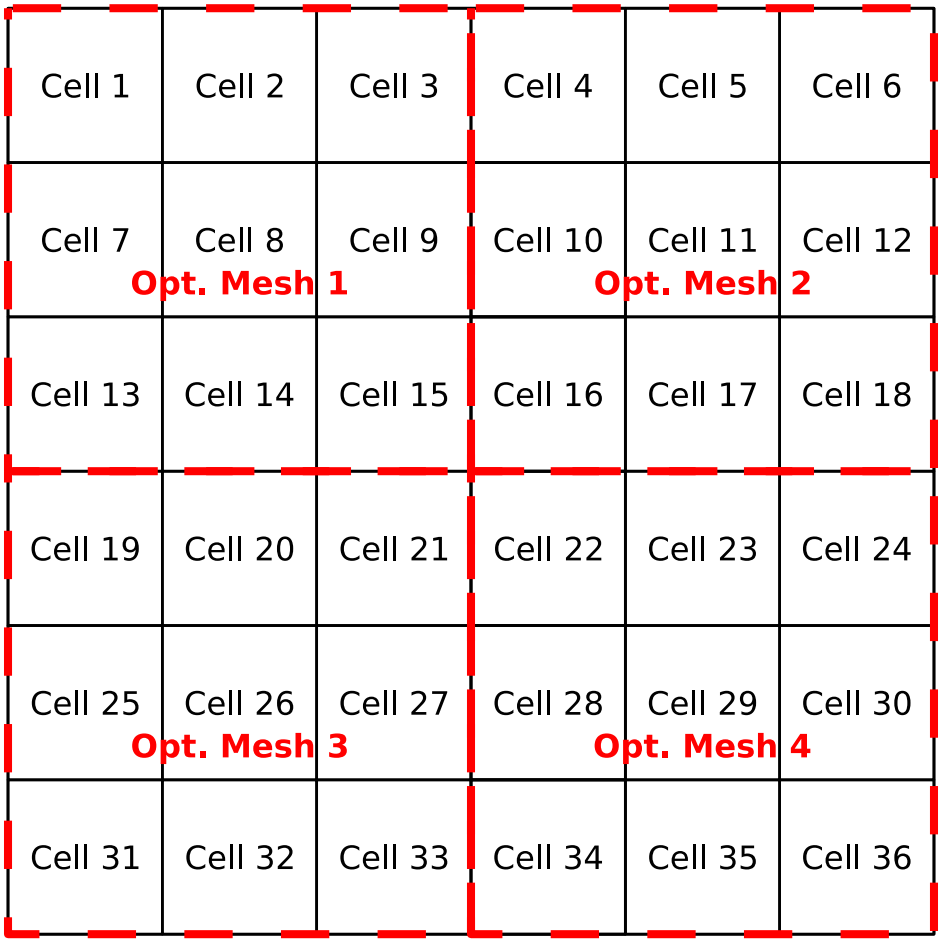


Figure 5.1. Optimization mesh over the cells in the problem. Each cell has its own weight-window lower bound and each optimization-mesh element has its own parameter by which to scale the weight-window lower bounds of cells contained within the element. Heavy red dashed lines indicate the optimization mesh elements.

Each element of the optimization mesh contains within it a number of cells, each with its own weight-window lower bound ww_c^g for each energy group g, where c is the cell index. A

parameter A_o^g is assigned to each optimization-mesh element, where o is the optimization-mesh element index, such that all weight windows lower bounds of cells within the optimization-mesh element are multiplied by A_o^g. Thus, the modified weight-window lower bounds $ww_c'^g$ are given by

$$ww_c'^g = A_o^g ww_c^g, \tag{5.2}$$

subject to cell c being contained within optimization-mesh element o.

Rather than optimizing the weight-window lower bounds of each cell, the optimization-mesh scaling parameters are optimized to produce a minimum cost calculation. For the example geometry in Fig. 5.1, a 36-cell problem, originally requiring 36 calculations to compute the gradient with respect to each weight window lower bound, is converted into a 4-parameter optimization problem.

5.1.2 Optimization Procedure

Using the weight-window lower bounds as a basis function with an optimization mesh parameter A_o^g applied over a portion of the function, the cost of the calculation can now be written

$$C(\{A_o^g\}) = \sigma^2(\{A_o^g\})\tau(\{A_o^g\}), \tag{5.3}$$

where $\{A_o^g\}$ represents the set of optimization-mesh parameters. The set $\{A_o^g\}$ represents a $g \times o$ number of optimization parameters. The direction of steepest descent of the cost function $C(\{A_o^g\})$ can now be calculated with respect to $\{A_o^g\}$ as

$$-\boldsymbol{\nabla}_{\{A_o^g\}} C(\{A_o^g\}) = -\left(\frac{\partial C(\{A_o^g\})}{\partial A_1^1}\hat{\mathbf{e}}_{A_1^1} + \frac{\partial C(\{A_o^g\})}{\partial A_2^1}\hat{\mathbf{e}}_{A_2^1} + \ldots + \frac{\partial C(\{A_o^g\})}{\partial A_O^G}\hat{\mathbf{e}}_{A_O^G}\right), \tag{5.4}$$

where G and O are the number of groups and number of optimization-mesh elements, respectively, and $\hat{\mathbf{e}}_{A_o^g}$ is a unit vector in the optimization hyperspace in the direction of A_o^g.

Each of the partial derivatives in Eq. (5.4) requires separate S_n calculation to evaluate. Numerically, these partial derivatives are evaluated by increasing a specific A_o^g corresponding to $\partial C(\{A_o^g\})/\partial A_o^g$ by a user specified amount ΔA and computing the new cost C'. The partial derivative is then numerically estimated by $(C' - C)/\Delta A$, and the process is repeated for the next partial derivative. In actuality, the A_o^g values are computed logarithmically such that the small-change optimization-mesh parameter $A_o'^g$ is

$$A_o'^g = 10^{(\log(A_o^g)+\Delta A)}, \tag{5.5}$$

and the partial derivatives are computed as $(C' - C)/(A_o'^g - A_o^g)$.

Once all partial derivatives have been calculated such that a numerical estimate of $-\boldsymbol{\nabla}_{\{A_o^g\}} C(\{A_o^g\})$ has been determined, a new set of parameters $\{A_o'^g\}$ is obtained by moving in the direction of the steepest descent. The new parameter is calculated by normalizing the gradient vector $-\boldsymbol{\nabla}_{\{A_o^g\}} C(\{A_o^g\})$ and updating each A_o^g as

$$A_o'^g = A_o^g - \frac{\partial C(\{A_o^g\})}{\partial A_o^g}\frac{\Delta A}{\left|\boldsymbol{\nabla}_{\{A_o^g\}} C(\{A_o^g\})\right|}. \tag{5.6}$$

The entire optimization procedure is performed as illustrated in Fig. 5.2. First, an initial first-moment calculation is performed followed by computing the initial set of weight windows for each

cell and in each energy group ww_c^g by

$$ww_c^g = \left[\int_{E_{g+1/2}}^{E_{g-1/2}} \int d\mathbf{r} \int d\mathbf{\Omega} M_1(\mathbf{r}, \mathbf{\Omega}, E) \right], \quad \mathbf{r} \in V_c. \tag{5.7}$$

Numerically, the weight-windows are computed by

$$ww_c^g = \left[\sum_i \sum_j \sum_k \sum_n w_n M_1^{g,n}(x_i, y_j, z_k) \Delta x_i \Delta y_j \Delta z_k \right]^{-1}, \quad (x_i, y_j, z_k) \in V_c. \tag{5.8}$$

Moreover, the weight-window lower bounds are normalized to a user-specified reference cell c_r and divided by two to obtain the lower bounds. Given this initial set of weight-window lower bounds, an initial cost is calculated by performing a second-moment and future-time calculation.

Next, each of the optimization parameters A_o^g is dithered to compute the gradient and a new set of A_o^g's. The cost is evaluated at the new A_o^g's, and, if the new cost is lower than the previous cost, the process of computing the gradient and stepping toward a lower cost continues. If the new cost is greater, then the calculation is assumed to have reached the minimum point at the previous set of A_o^g's and the calculation ceases. As with all gradient methods, it is possible that the calculation finds a local minimum rather than a global minimum. Potentially better optimization methods are discussed in the following chapter.

5.2 Optimization Test Problems

The optimization method described above has been applied to a series of 1-D and 2-D problems. These problems are chosen because they tend to over split particles causing an increase in computational time without a commensurate reduction in the variance.

5.2.1 Weight-Window Surface and Lower Bound

As an initial test of the optimization capabilities, a 1-D one-group problem was developed to optimize the location of the boundary between two weight-window cells and the lower bound of the right cell. The geometry is shown in Fig. 5.3. The slab is 5 mfp thick and has a scattering-to-total cross section fraction of $c = 0.25$.

For this problem, no optimization mesh is applied because the position of the surface between the two cells is being optimized not only the weight-window lower bounds. To optimize the problem, the location of the intermediate surface P_1 was varied from 0.25 to 4.75 mfp. The weight window lower bound of the right cell P_2 was varied logarithmically from 10^{-4} to 0.5. For each case, a separate S_n calculation was performed and an estimate of the Monte Carlo cost obtained. The cost surface for the S_n calculations is shown in Fig. 5.4(a).

The entire calculation was repeated using MCNP rather than the S_n code. For each iterate of P_1 and P_2, the MCNP calculation was run and the figure of merit extracted from the output file and inverted to obtain a quantity proportional to cost. The optimization surface obtained with MCNP is shown in Fig. 5.4(b), where the cost values have been scaled by a constant factor to the same magnitude as the S_n results.

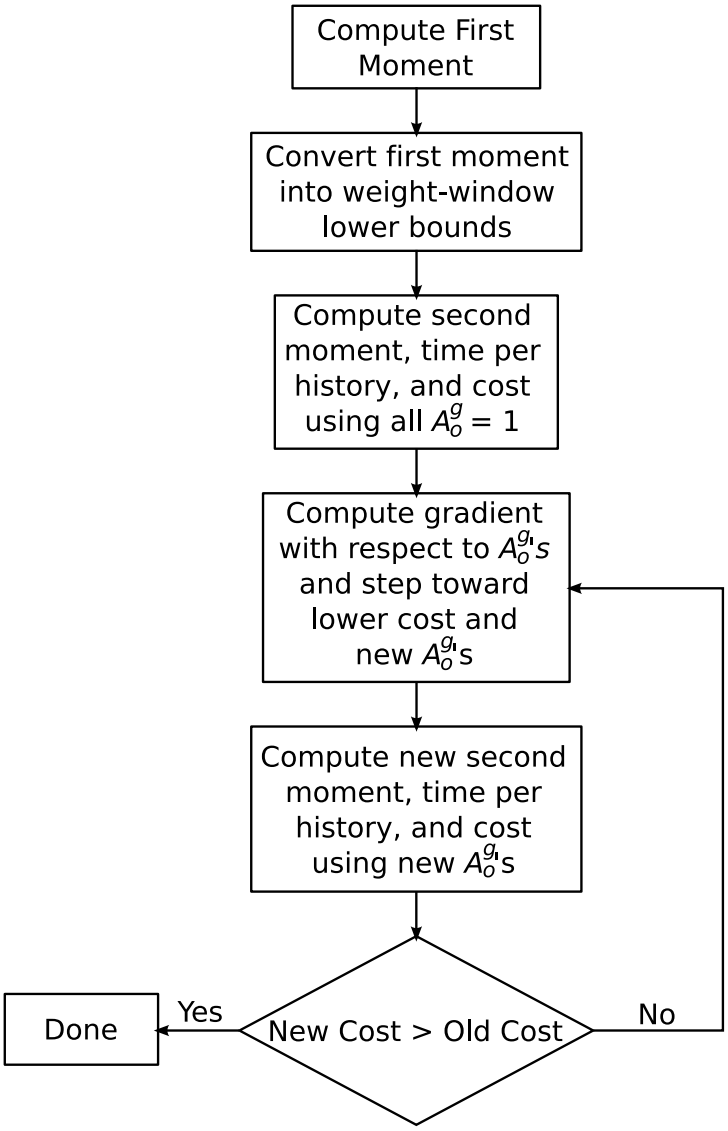


Figure 5.2. Flow of the optimization algorithm. Each instance that the gradient of the cost is computed with respect to the A_o^g's requires a separate S_n calculation for each A_o^g

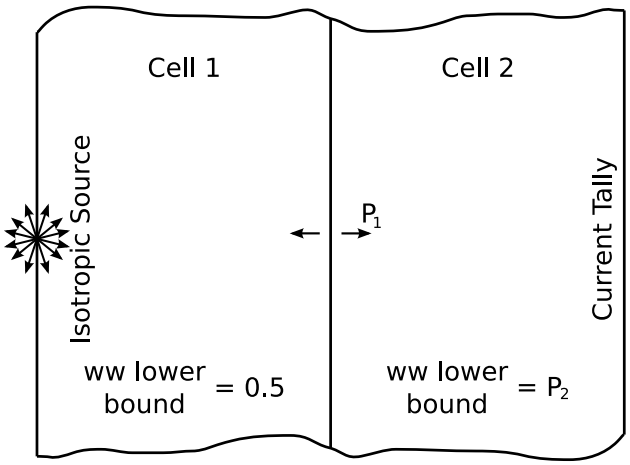


Figure 5.3. Two-cell slab geometry optimizing the location of the separating surface P_1 and the weight window lower bound of the right cell P_2.

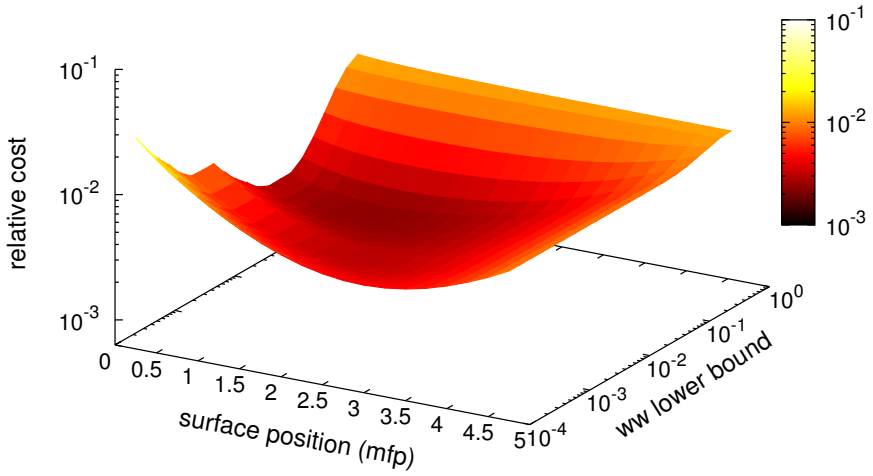
Figure 5.4(a) compares quite well with Fig. 5.4(b). Both methods clearly locate a minimum and that minimum is in approximately the same location. The S_n calculation finds the optimum surface location at $P_1 = 1.75$ mfp and the optimum weight-window lower bound for the right cell at $P_2 = 1.66 \times 10^{-2}$. The MCNP calculation finds the optimum surface location at $P_1 = 1.75$ mfp and the optimum weight-window lower bound at 2.54×10^{-2}. In either calculation, the difference between a weight-window lower bound of 1.66×10^{-2} and 2.54×10^{-2} produces a cost difference of less than 1%.

The results obtained from this calculation were compared to the findings of Juzaitis [1982], where optimum locations of importance splitting surfaces are sought. Interestingly, Juzaitis found the minimum cost location of the surface to be around 3 mfp for a similar problem. One possibility for the discrepancy in results is that Juzaitis used the importance splitting technique while the weight-windows technique is used here. A more likely explanation is that the Monte Carlo code used for each comparison was different, and the S_n code developed by Juzaitis to solve the moment equations catered to the specific Monte Carlo code being used.

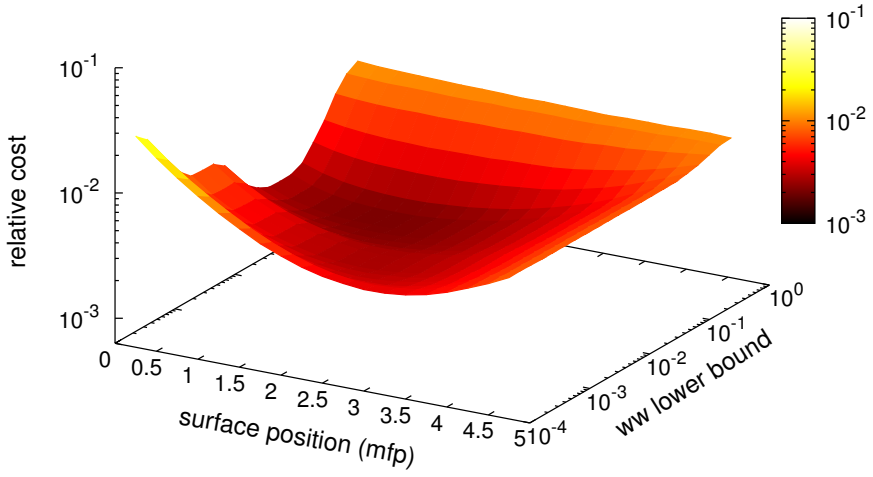
5.2.2 1-Group Slab

The one-group 1-D slab problem was an initial investigation of the weight-window lower bound optimization capabilities of the moments-equation method. The slab is 5 mean-free-paths thick with a scattering ratio of $c = 0.85$. An isotropic source is incident on the left side of the slab and the desired tally is the current out the right side of the slab. As illustrated in Fig. 5.5, the slab is divided into four cells.

For this problem, the optimization mesh corresponds to the cell boundaries. The optimization is performed with an S_{32} angular discretization, 501 Monte Carlo particle weight bins, and a steepest descent step size of 0.2. Table 5.1 presents the predicted gain in efficiency from the S_n optimization and the realized gain in efficiency from MCNP when the optimized set of weight windows are used. The results are normalized to the efficiency of the calculation run with the set of weight windows computed directly from the adjoint function. Table 5.2 shows how the means, variances,



(a)



(b)

Figure 5.4. Optimization surface for the location of the weight window cell boundary and weight window lower bound of the right cell from (a) the S_n optimization and (b) the MCNP optimization (results are multiplied by a proportionality constant to present on the same scale)

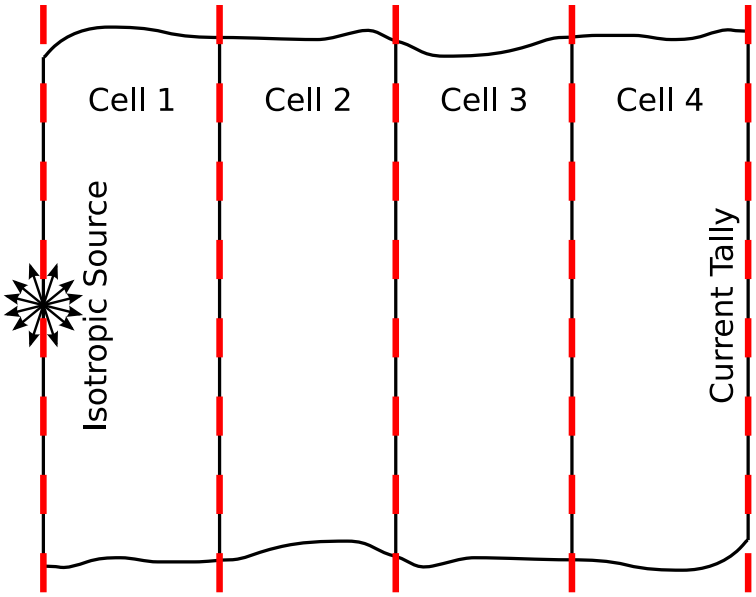


Figure 5.5. 1-group slab problem geometry. Red dashed lines represent the optimization mesh locations

and predicted times-per-history compare.

Table 5.1. Predicted and realized efficiency gains for the 1-group slab problem

Method	Predicted Cost	Predicted Gain	MCNP FOM	Realized Gain
Adjoint WW	1.04214E-04	1	4.32902E+05	1
Optimization	9.01291E-05	1.16	5.26203E+05	1.22

It is clear from Table 5.2 that the time per history is not computed very accurately. However, more important to the estimation of the cost is the ratios of the times, which for the S_n calculation is 1.37 and for the MCNP calculation is 1.29. It is the difference in calculation time ratios that causes the differences in predicted and realized efficiency gains. Despite the absolute differences in the predicted time per history being different, the ratio is maintained and still provides a reasonably adequate prediction of the decreased cost. The final sets of weight window lower bounds are given in Table 5.3.

5.2.3 Iron Window

This test problem is designed to demonstrate a common problem with importance-based biasing resulting from substantial changes in cross sections. Such problems arise when a particle transitions from one energy to another and the new energy has a substantially lower cross section, caused, for example, by scattering into an anti-resonance like that of iron (at $\sim$ 24 keV). At the energy of the substantially lower cross section, the particle is more likely to stream farther and to

Table 5.2. Comparison of means, variances, and time-per-history between the S_n and MCNP calculations

Method	S_n mean	MCNP mean	S_n σ^2	MCNP σ^2	S_n τ (sec./hist.)	MCNP τ (sec./hist.)
Adjoint WW	9.01766E-03	9.09488E-03	2.28695E-03	2.26035E-03	3.70559E-06	5.04000E-06
Optimization	9.01766E-03	9.10943E-03	1.44492E-03	1.45568E-03	5.07236E-06	6.48000E-06

Table 5.3. Adjoint and Optimized weight window lower bounds for the 1-group slab problem

Method	Cell 1	Cell 2	Cell 3	Cell 4
Adjoint WW	5.00000E-01	1.78612E-01	7.03716E-02	2.64944E-02
Optimized	4.27562E-01	6.65740E-02	2.59931E-02	2.01596E-02

contribute to tallies than at energies with higher cross sections. With importance-based biasing, the tendency is to over split a particle into many particles that behave in the same manner. Such over splitting increases the computation time without an equivalent reduction of variance.

The iron-window problem is simulated here with a two-group slab transport problem. The geometry is similar to that of the previous problem except that only one optimization mesh element over the entire problem is utilized as illustrated in Fig. 5.6. The multi-group cross sections used for the problem are presented in Table 5.4. The uniform source is isotropically incident on the left side of the slab in group 1, and the total current is tallied on the right side of the slab.

Table 5.4. Two group cross sections used to represent the iron-window effect

g	Σ	Σ_c
1	1 cm^{-1}	0.9 cm^{-1}
2	1×10^{-5} cm^{-1}	7×10^{-6} cm^{-1}

g	$\Sigma_{gg'}$	
	$g' = 1$	$g' = 2$
1	0.05 cm^{-1}	0.05 cm^{-1}
2	0	3×10^{-6} cm^{-1}

With importance-based biasing computed directly from the adjoint function, importances for the two groups are computed as shown in Fig. 5.7. A particle that scatters from group 1 to group 2 experiences an approximately 20-to-1 split. Each of the progeny resulting from the split will behave in a similar manner, simply streaming through the slab. Because all the progeny from the split behave the same, the reduction in variance from splitting the particles is minute compared to the additional computational time required to track them.

This problem was optimized on the Monte Carlo cost by placing a single optimization mesh element over the entire problem. With this optimization mesh, two scaling parameters A_o^g are opti-

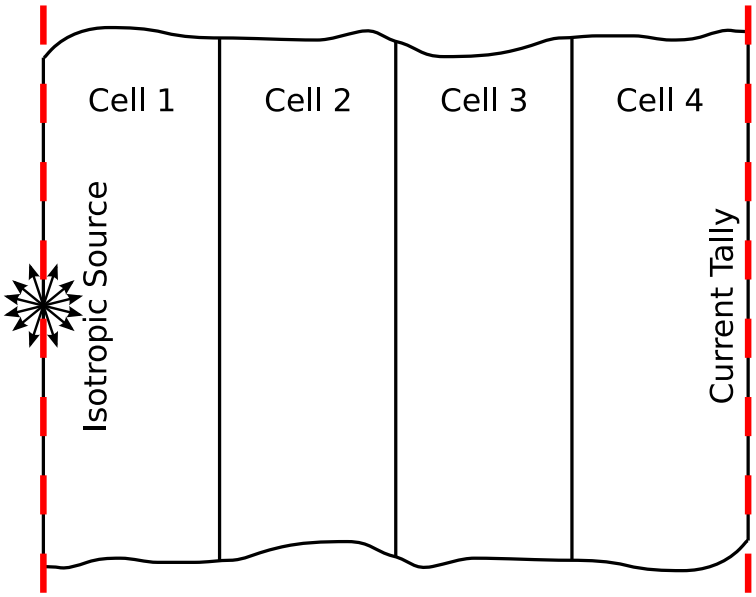


Figure 5.6. Iron window like problem geometry. Thick red dashed lines indicate the optimization mesh

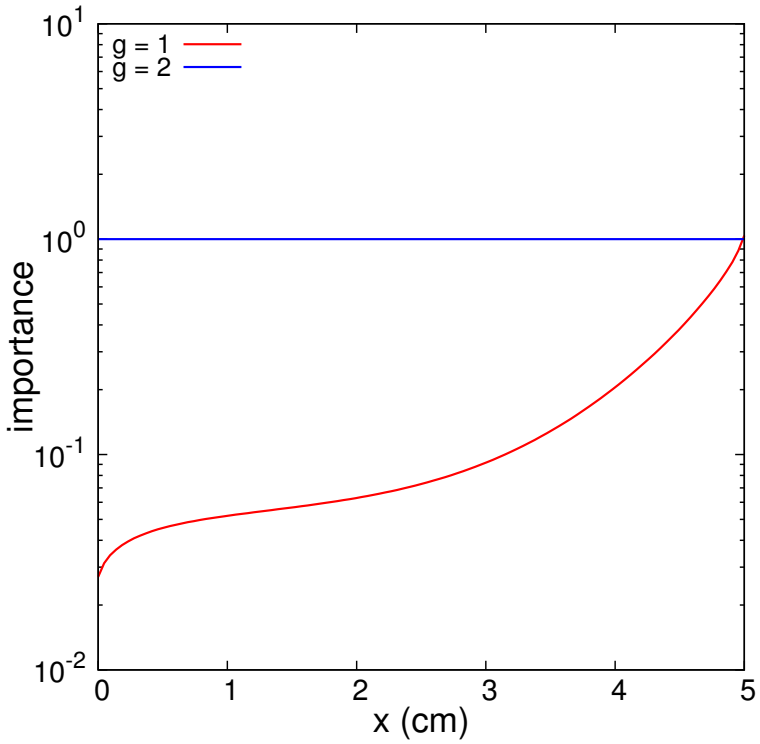


Figure 5.7. Importances for the two group iron-window problem. If a particle scatters from group 1 to group 2 near the source, an approximately 20-to-1 split where all the progeny will behave similarly will ensue

mized, one for each energy group. The calculation was performed with an S_{32} angular quadrature, 501 weight bins, and a gradient step size of 0.2. The resulting predicted and realized efficiency gains over the adjoint generated weight windows are presented in Table 5.5. With the optimized set of weight windows, a cost reduction of a factor of 2.34 is predicted and reduction of 2.17 is realized. From Table 5.6 it is clear that the discrepency between the predicted and realized gains is again due to the poor estimation of the time per history. Nonetheless, the ratio between the predicted 2.17 and realized 2.37 times remains reasonably close to approximate the minimum cost. The optimized and adjoint-generated weight windows are presented in Table 5.7.

Table 5.5. Predicted and realized efficiency gains for the iron window like problem

Method	Predicted Cost	Predicted Gain	MCNP FOM	Realized Gain
Adjoint WW	2.30024E-04	1	2.86922E+05	1
Optimization	9.83604E-05	2.34	6.23205E+05	2.17

Table 5.6. Comparison of means, variances, and time-per-history between the S_n and MCNP calculations for the iron-window problem

Method	S_n mean	MCNP mean	S_n σ^2	MCNP σ^2	S_n τ (sec./hist.)	MCNP τ (sec./hist.)
Adjoint WW	1.35174E-02	1.34915E-02	1.29472E-02	1.29820E-02	3.24628E-06	2.88000E-06
Optimization	1.35174E-02	1.34835E-02	2.54747E-03	2.55824E-03	7.05502E-06	6.84000E-06

5.2.4 2-D 1-Group Block

The 2-D block problem was a first test of the optimization of a two-dimensional problem. The geometry, pictured in Fig. 5.8, consists of a single material with a volume flux tally in the upper right hand corner. The source is a uniformly distributed isotropic volume source in the lower left hand corner of the block. The remainder of the domain is divided into 92 1 cm by 1 cm cells each having its own weight-window lower bound. The total cross section in the block is 1 cm^{-1} with a scattering ratio of $c = 0.8$.

The optimization mesh for the 2-D block problem is a uniform 5-by-5 element mesh. Each mesh element has a physical size of 2 cm by 2 cm and contains four of the geometry cells, except for the source and tally cells that are each uniquely covered by an optimization mesh element. The optimization calculation was performed with an S_8 triangular Gauss-Chebyshev angular quadrature and 501 weight bins spanning the weight range 10^{-9} to 10^{1}. The results of the optimization are presented in Table 5.8.

The predicted gain from the S_n optimization is about 5% different from the realized gain. However, considering the approximations made in calculating the time per history this seems reason-

Table 5.7. Optimized and adjoint-generated weight windows for the iron window like problem in cell *c* and group *g*

g \ *c*	1	2	3	4
adjoint-generated weight windows				
1	5.00000E-01	3.70177E-01	2.19849E-01	5.83791E-02
2	2.29280E-02	2.29267E-02	2.29255E-02	2.29243E-02
optimized weight windows				
1	2.90346E-02	2.14959E-02	1.27664E-02	3.39002E-03
2	5.26638E-02	5.26609E-02	5.26581E-02	5.26553E-02

Table 5.8. Predicted and realized efficiency gains for the 2-D block problem

Method	**Predicted Cost**	**Predicted Gain**	**MCNP FOM**	**Realized Gain**
Adjoint WW	6.03494E-04	1	6.87040E+04	1
Optimization	5.05767E-04	1.19	8.55240E+04	1.25

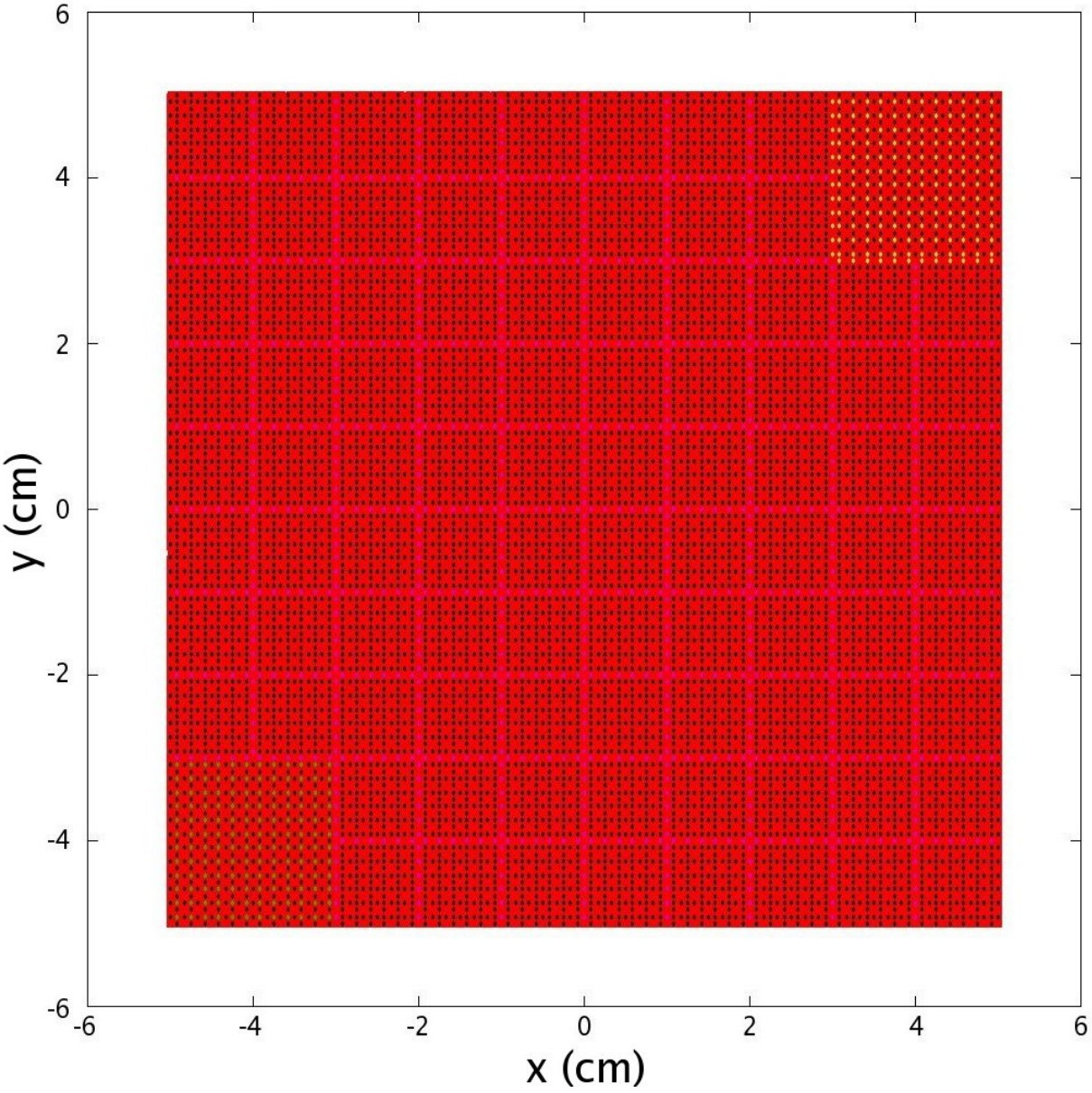


Figure 5.8. 2-D, 1-group block geometry with a source in lower left hand corner and tally in upper right hand corner . The S_8 mesh is shown superimposed on the geometry, where black nodes represent cell-center or cell-edge nodes, purple nodes represent cell-edge nodes that are also Monte Carlo cell boundaries, green nodes represent the tally, and dark red nodes represent the source.

able. The mean, variance, and time per history actually calculated by the S_n method and MCNP for the adjoint generated weight windows and optimized weight windows are presented in Table 5.9. The first moment is underestimated by about 3%, which would lead to some error in the second moment calculation because the first moment factors into the source for the second moment. The variance is overestimated by about 3.5% for the adjoint generated weight windows and 7% for the optimized case. Calculation of the second moments from the variances and first moments indicates that the second moment is also overestimated by about 3.5% and 7% for each case, respectively. Some of the error in the calculations can be attributed to ray effects. Nonetheless, the ratio of the adjoint generated weight window variance to the optimized weight window is 1.41 for the S_n calculation and 1.46 for the MCNP calculation. Furthermore, the ratio of the adjoint generated weight window and optimized weight window time per history for both the S_n calculation and the MCNP calculation are 0.844 and 0.853 respectively. For this calculation, there is about 5% error in the ratios of the variances and 1% error in the ratios of the times, and even with these errors the efficiency gain is predicted adequately.

Table 5.9. Comparison of means, variances, and time-per-history between the S_n and MCNP calculations for the 2-D block problem

Method	S_n mean	MCNP mean	S_n σ^2	MCNP σ^2	S_n τ (sec./hist.)	MCNP τ (sec./hist.)
Adjoint WW	8.05420E-04	8.29434E-04	8.10688E-05	7.82905E-05	4.82908E-06	7.68000E-06
Optimization	8.05420E-04	8.29957E-04	5.73463E-05	5.35638E-05	5.72124E-06	9.00000E-06

The weight windows obtained through adjoint generation and optimization are presented in Fig. 5.9(a) and Fig. 5.9(b), respectively. The differences between the two sets of weight windows are subtle, but present. Most noticeable are the changes along the diagonal of the block from the source to the tally where the optimized weight window is lower than the adjoint-generated weight window. Fig. 5.9(c) shows the fractional difference

$$\text{fractional difference} = \frac{|ww_{\text{opt}} - ww_{\text{adjoint}}|}{ww_{\text{adjoint}}} \tag{5.9}$$

between the optimized weight windows and the adjoint-generated weight windows, where, if negative, the optimized weight window is less than the adjoint-generated weight window.

5.2.5 Top Hat

The geometry of the 2-D top hat problem [Booth and Burn, 1993] geometry is presented in Fig. 5.10. Appropriately named, the top hat problem consists of a long vertical section of material terminating in a current tally representing the hat and a horizontal offshoot on each side representing the brim. The top hat problem is composed of a single material at different densities. The volume source is located at the bottom of the hat at the central axis. The problem is slightly simplified from that presented by Booth and Burn [1993] so that the S_n code could accommodate

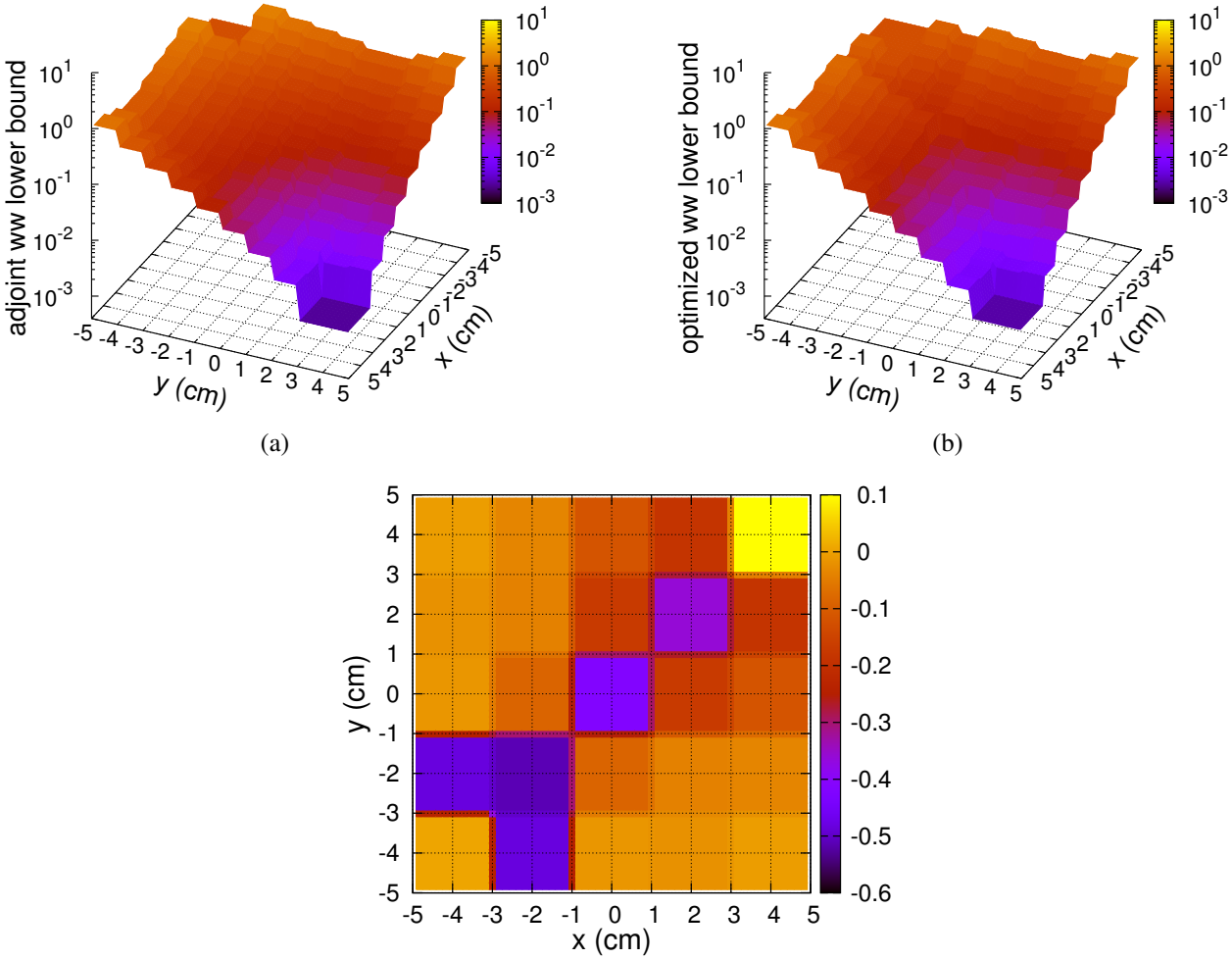


Figure 5.9. 2-D block geometry weight window lower bound for the (a) adjoint generated weigh windows and (b) optimized weight windows. (c) the fractional difference between the optimized weight windows and the adjoint generated weight windows

the problem. Specifically, the problem was executed with a single energy group, an S_{12} trianglar Gauss-Chebyshev quadrature, and a 501 weight bin discretization. The scattering fraction for the single group is $c = 0.8$.

The top hat problem is noted for its tendency to over split particles. From the source, the particles can either traverse the distance of the hat and contribute to the tally or enter the brim of the hat, scatter toward the top of the hat, reenter the hat, and contribute to the tally. Using importance-based biasing, the importance is much higher at the top of the hat than at the brim. However, a particle leaving the brim and reentering the hat near the top splits extensively, but all split particles are likely to contribute to the tally. Such over splitting increases the calculation time without a significant decrease in the variance.

The S_n optimization of the top hat problem was run with a steepest descent step size of 0.2. The predicted efficiency gain from the optimization and its comparison to adjoint-generated weight windows are presented in Table 5.10. A predicted gain of 1.34 is expected from the S_n calculation and a gain of 1.27 is realized when the adjoint-generated and optimized weight windows are utilized within MCNP. The difference between the predicted gain and realized gain is approximately 5%.

Table 5.10. Predicted and realized efficiency gains for the top hat problem

Method	Predicted Cost	Predicted Gain	MCNP FOM	Realized Gain
Adjoint WW	1.31147E-03	1	2.82637E+04	1
Optimization	9.81829E-04	1.34	3.59361E+04	1.27

The means, variances, and times per history for the top hat problem are presented in Table 5.11. None of these three values compare well to the actual values computed by MCNP. The means calculated by the S_n are approximately 20% different from those calculated by MCNP. The variances calculated by the S_n are 20% different in the adjoint-generated weight window case and nearly 30% different in the optimized weight window case. Again, much of the error in the estimation of the variances can probably be attributed to ray effects. The times per history are 27% different in the adjoint-generated case and 30% in the optimized case. Even with these poor approximations to the mean and variances, the overall estimate of the efficiency gain maintains only 5% deviation from the actual. This maintenance is a result of the ratios of the variances and time per history between the adjoint-generated and optimized cases being approximately maintained. For the S_n calculation the variance ratio of the adjoint-generated to the optimized case is 1.60 while for the same MCNP calculation the ratio is 1.48. Similarly the time per history ratio for the adjoint-generated to the optimized case is 0.832 for the S_n and 0.879 for MCNP. Because the ratios of these two values is approximately maintained, the predicted gain is nearly correct.

The weight windows for the adjoint generated case are presented in Fig. 5.11(a), and the weight windows for the optimized case are given in Fig. 5.11(b). The most notable differences are in the outer brim of the hat and the path down the center of the hat. The weight window lower bound in the brim is lower for the optimized case and suggests a greater amount of splitting for particles entering the brim. A greater level of splitting in the brim causes the correlation between the split particles to be lost there. Then, when the particles traverse to the top of the hat, the weight window

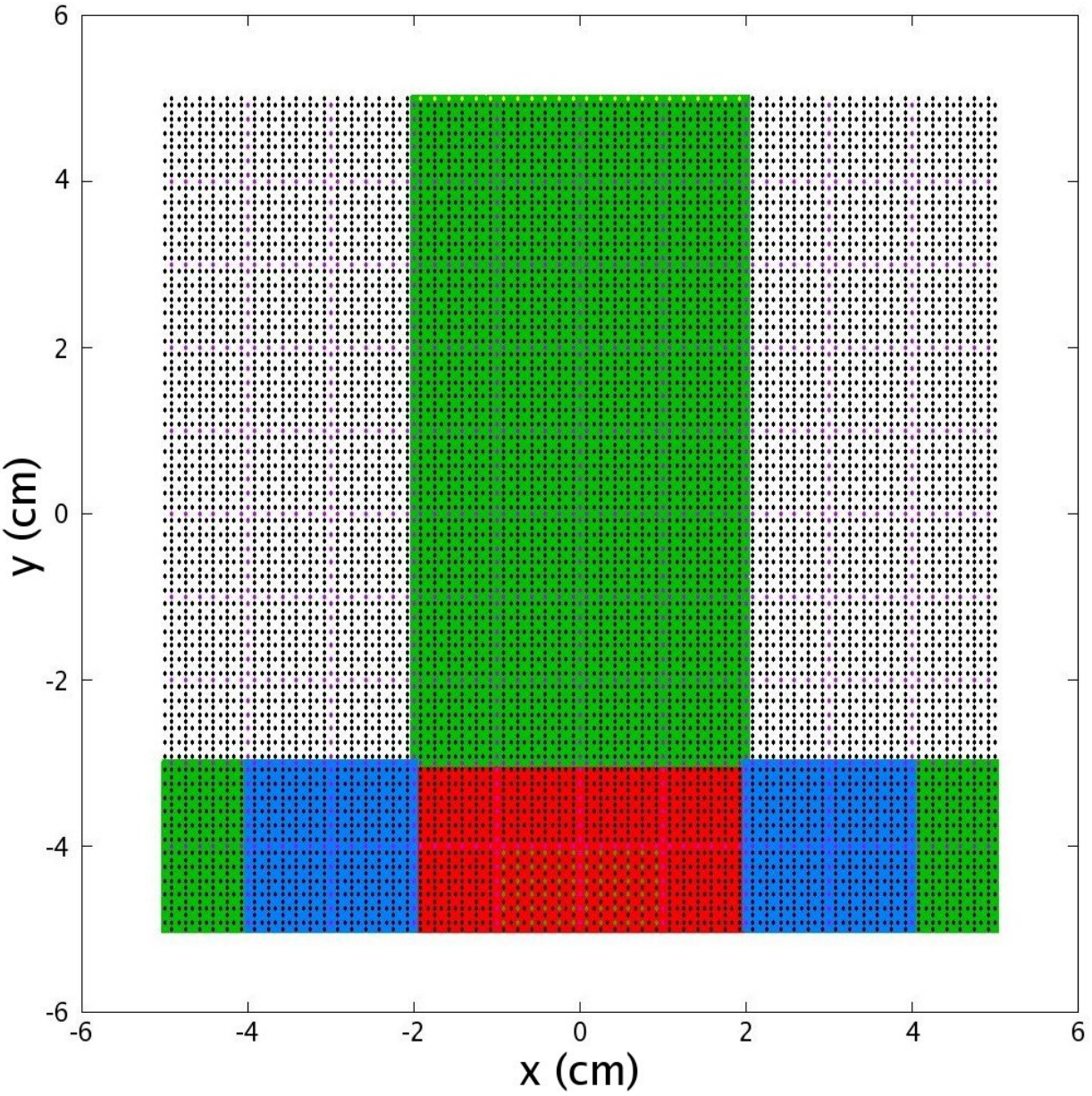


Figure 5.10. The top hat problem geometry. The different colors represent different material atomic densities: red = 2 cm^{-1}, blue = 0.2 cm^{-1}, and green = 1 cm^{-1}

Table 5.11. Comparison of means, variances, and time-per-history between the S_n and MCNP calculations for the top hat problem

Method	**S_n mean**	**MCNP mean**	**S_n σ^2**	**MCNP σ^2**	**S_n τ (sec./hist.)**	**MCNP τ (sec./hist.)**
Adjoint WW	1.51247E-04	1.25620E-04	2.48926E-06	1.92792E-06	1.20520E-05	1.74000E-05
Optimization	1.51247E-04	1.24232E-04	1.55142E-06	1.30144E-06	1.44770E-05	1.98000E-05

value is nearly the same and the particle do not split there. Similarly, particles that do not enter the brim and transition directly down the hat encounter lower weight window bounds in the optimized case than in the adjoint-generated case. Again, this causes the particles to be split to a greater extent earlier thereby losing the correlation between the split particles earlier.

Figure 5.11(c) shows the fractional differences between the optimized weight windows and the adjoint weight windows. Clearly from Fig. 5.11(b) the greatest difference is in the brim of the hat where the optimized weight window is 60% less than the adjoint-generated window. Also interesting is that the weight window lower bounds on the edge of the hat near the tally have been increased such that there is a greater overlap in the weight windows of particles in the brim and entering these regions. Thus, particles leaving the brim and reentering the hat near the tally are not split as much as with just the adjoint-generated weight windows.

5.2.6 Two-legged Duct

A two legged duct problem with the source at one opening of the duct and a current tally at the other opening of the duct was also optimized. A general difficulty with duct-type problems based on importance biasing in Monte Carlo is that particles over split. Typically, this over splitting is a result of particles streaming down a duct leg and transitioning from a region of low importance to one of much higher importance. The over splitting, as with the top hat problem, requires additional time to track the particles from the split without much of a decrease in variance.

The two-legged duct problem geometry is presented in Fig. 5.12. The duct is void and the surrounding material has a one-group cross section of 1 cm^{-1}. The 10 cm-by-10 cm geometry is broken up into 100 1 cm-by-1 cm cells, each having its own weight window lower bound. The duct has a width of 2 cm. The optimization mesh has a nonuniform element size, easiest inferred from Fig. 5.13(c) where regions of like color represent an optimization mesh element.

The S_n optimization calculation was performed with an S_8 triangular Gauss-Chebyshev angular discretization, 501 weight bins, and a steepest descent step size of 0.2. The optimized weight windows predict an efficiency gain of 1.17 over the adjoint-generated weight windows. When the weight windows are used in MCNP, the realized gain is 1.18. Table 5.12 presents the costs and figures of merit obtained using the optimized and adjoint-generated weight windows. The means, variances, and times per history are given in Table 5.13. As with the top hat problem, the S_n estimate of the means, variances, and times per history are not exact and vary between 14–30% error, much of which can probably be attributed to ray effects for the mean and variance. However, the variance ratio of the adjoint-generated case to the optimized case is 1.40 for the S_n calculation and 1.39 for the MCNP calculation. Furthermore, the time per history ratio of the adjoint-generated case to the optimized case is .832 for the S_n calculation and .859 for the MCNP calculation. Even though the variances and times per history are calculated only approximately, the ratios are again maintained and the estimated efficiency gain is reasonable.

The values of adjoint-generated weight windows and the optimized weight windows are shown in Fig. 5.13(a) and Fig. 5.13(b), respectively. The fractional difference between the adjoint-generated and optimized weight windows is shown in Fig. 5.13(c). Figure 5.13(c) shows that the primary modifications to the adjoint-generated weight windows occur towards the end of the first leg of the duct, where the optimized weight window is 40% lower than the adjoint generated weight window. The weight-window lower bounds at the end of the first leg are adjusted by the optimization such that the weight window differs little down the second leg of the duct.

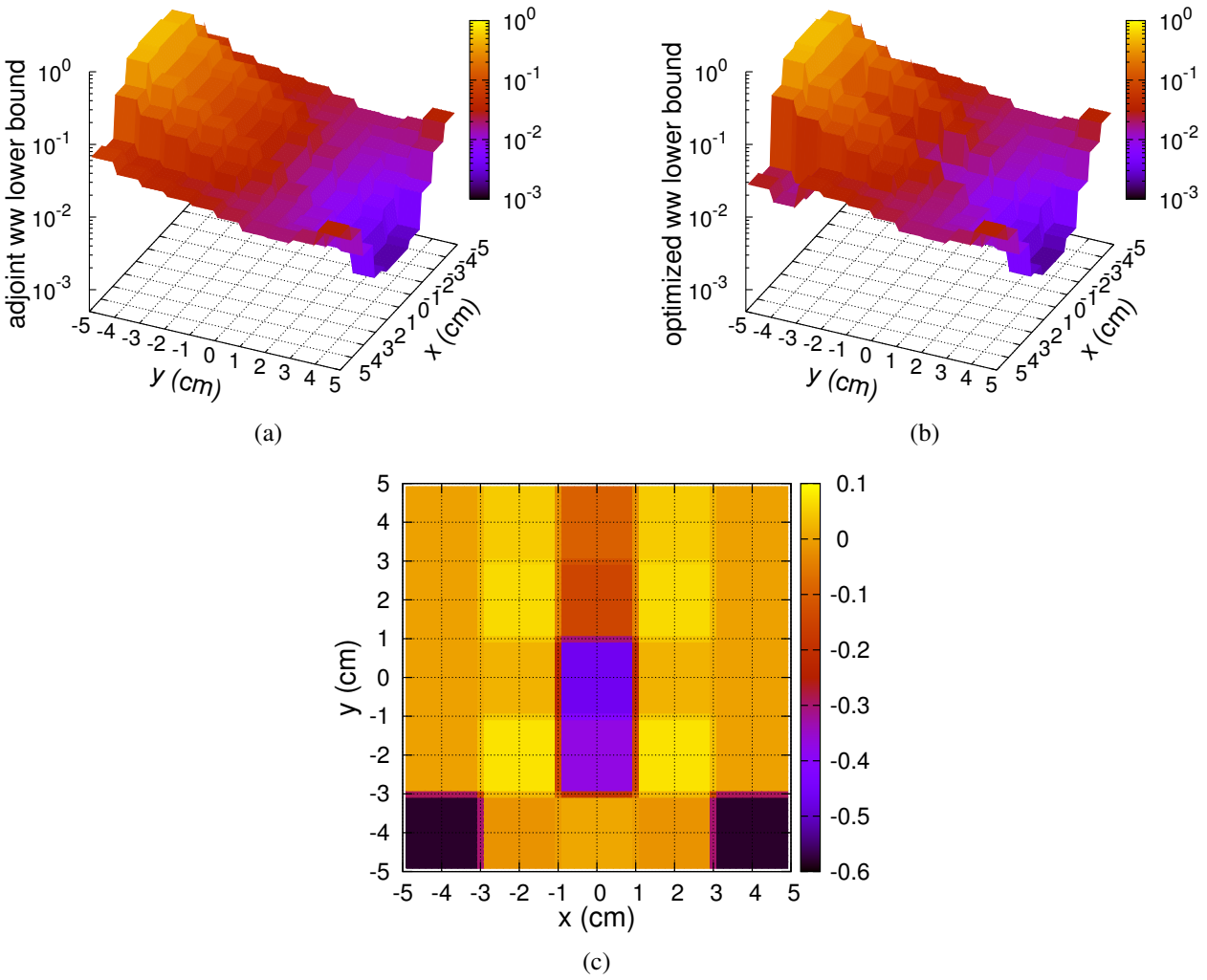


Figure 5.11. Top hat problem geometry weight window lower bound for the (a) adjoint generated weigh windows and (b) optimized weight windows. (c) the fractional difference between the optimized weight windows and the adjoint generated weight windows

Table 5.12. Predicted and realized efficiency gains for the wide two-legged duct problem

Method	Predicted Cost	Predicted Gain	MCNP FOM	Realized Gain
Adjoint WW	3.17398E-04	1	1.19256E+05	1
Optimization	2.71285E-04	1.17	1.40493E+05	1.18

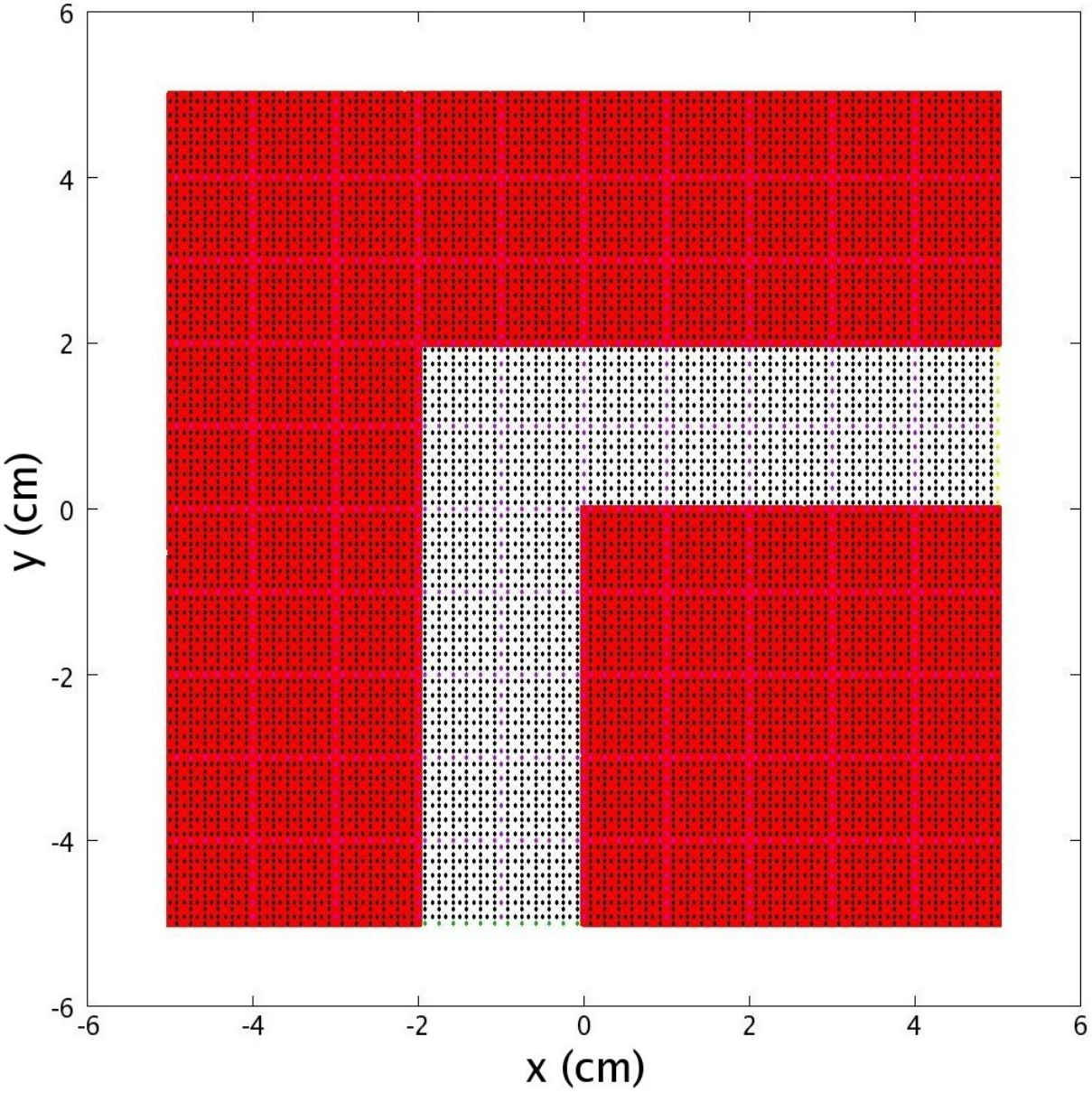


Figure 5.12. Wide two-legged duct problem geometry

Table 5.13. Comparison of means, variances, and time-per-history between the S_n and MCNP calculations for the wide two-legged duct problem

Method	S_n mean	MCNP mean	S_n σ^2	MCNP σ^2	S_n τ (sec./hist.)	MCNP τ (sec./hist.)
Adjoint WW	6.25891E-03	5.47763E-03	1.84946E-03	1.59846E-03	6.72288E-06	9.48000E-06
Optimization	6.25891E-03	5.43904E-03	1.31662E-03	1.15043E-03	8.07168E-06	1.10400E-05

Most interesting for the two-legged duct problem is that the weight window lower bound on a direct path from the source to the tally through the material are reduced. This indicates that some fraction of the variance for this problem is from particles moving through the material rather than scattering through the duct. From the results of the three-legged duct problem in the following section, it is likely this pathway rather than scattering down the duct that provides the optimized benefit as the problem is simple and the adjoint-generated windows are probably sufficient for biasing through the duct.

5.2.7 Three-legged Duct

A three-legged duct problem was also optimized. As with the two-legged duct, the three-legged duct geometry is 10 cm by 10 cm and divided into 100 1 cm-by-1 cm cells each with its own weight-window lower bound. The geometry is shown in Fig. 5.14. The surrounding material has a total cross section of $\Sigma = 1$ cm^{-1} and a scattering ratio of $c = 0.8$.

The optimization mesh is a 5-by-5 uniform mesh with each mesh element having a dimension of 2 cm by 2 cm. The optimization was run using an S_8 triangular Gauss-Chebyshev angular discretization, 501 weight bins spanning the weight domain 10^{-9} to 10^{1}, and a steepest descent step size of 0.2. The S_n and MCNP results using the adjoint-generated and optimized sets of weight windows are presented in Fig. 5.14. For this calculation, only a small increase in efficiency by a factor of 1.04 is predicted, and an increase of 1.08 is realized in the MCNP calculation.

Table 5.14. Predicted and realized efficiency gains for the wide three-legged duct problem

Method	Predicted Cost	Predicted Gain	MCNP FOM	Realized Gain
Adjoint WW	8.93642E-04	1	8.04780E+04	1
Optimization	8.57150E-04	1.04	8.65300E+04	1.08

The means, variances, and times per history calculated by both the S_n and MCNP are shown in Table 5.15. The error in calculating the mean is about 30%, and the error in calculating the variance is around 50%, both probably mostly due to ray effects. The times per history are calculated with between 30% and 40% error. However, despite the poor absolute calculations of the means, variances, and times-per-history, the ratios of the values remain approximately. The variance ratio of the adjoint-generated weight window case to the optimized case is 1.36 for the S_n and 1.34 for MCNP. Similarly, the time-per-history ratio of the adjoint-generated weight window case to the optimized case is 7.66 for the S_n calculation and 8.07 for MCNP.

The changes to the values of the weight windows from the adjoint-generated case to the optimized case are seen by contrasting Fig. 5.15(a) to Fig. 5.15(b). The fractional changes in each optimization region are given in Fig. 5.15(c). The most notable change is reduction of weight window lower bounds at the corners of the duct bends. The corner nearest the source is reduced by about 20% while the corner nearest the tally is reduced by nearly 60%. Nonetheless, the benefit observed in this problem is small and is likely due to the overall simplicity of the problem. For such a simple problem the importance function likely works well and the additional cost introduced by splitting is small.

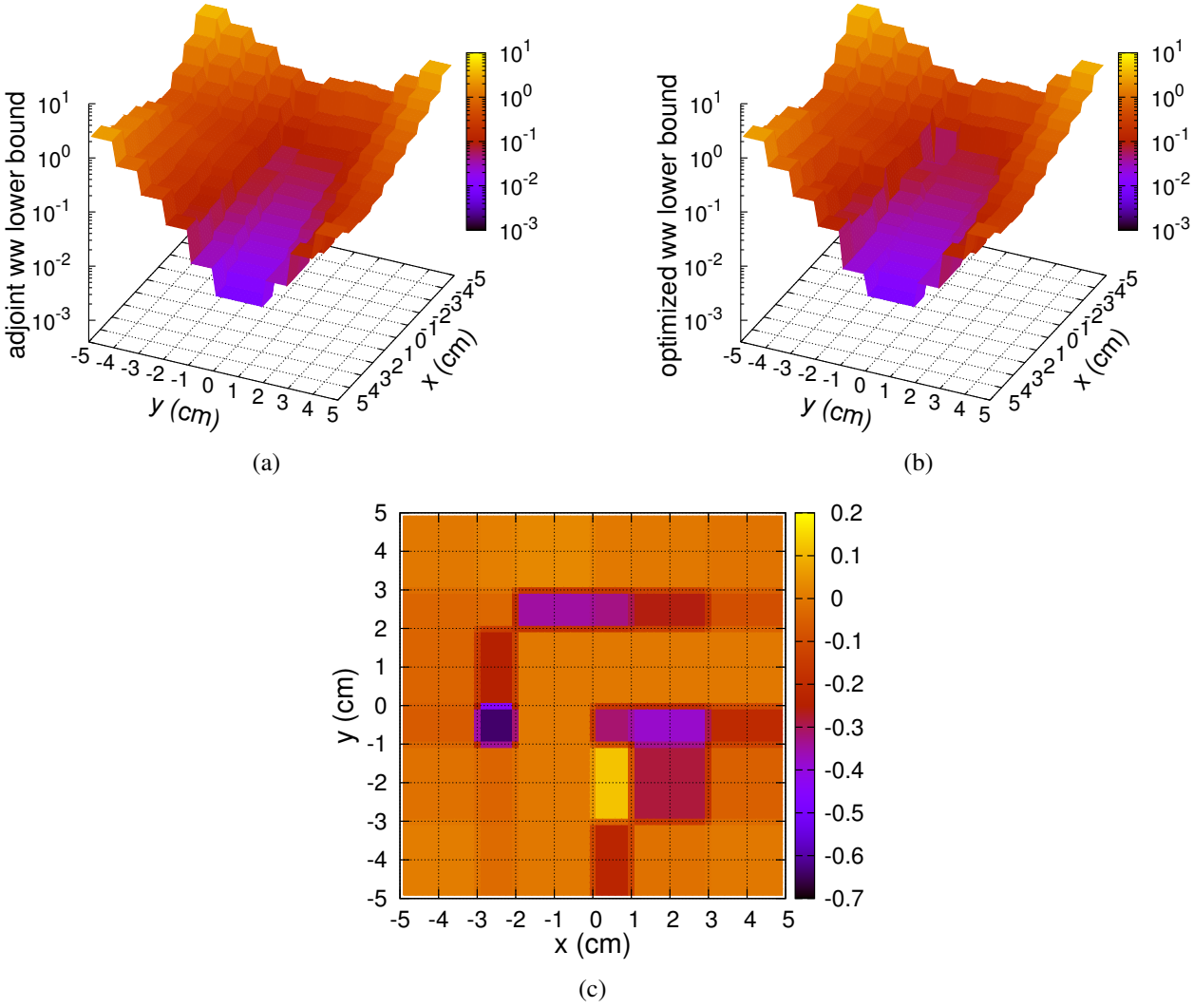


Figure 5.13. Top hat problem geometry weight window lower bound for the (a) adjoint generated weigh windows and (b) optimized weight windows. (c) the fractional difference between the optimized weight windows and the adjoint generated weight windows

Table 5.15. Comparison of means, variances, and time-per-history between the S_n and MCNP calculations for the wide three-legged duct problem

Method	**S_n mean**	**MCNP mean**	**S_n σ^2**	**MCNP σ^2**	**S_n τ (sec./hist.)**	**MCNP τ (sec./hist.)**
Adjoint WW	1.78015E-03	1.37034E-03	1.68914E-04	1.10883E-04	1.67653E-05	1.26000E-05
Optimization	1.78015E-03	1.36546E-03	1.24187E-04	8.26195E-05	2.18724E-05	1.56000E-05

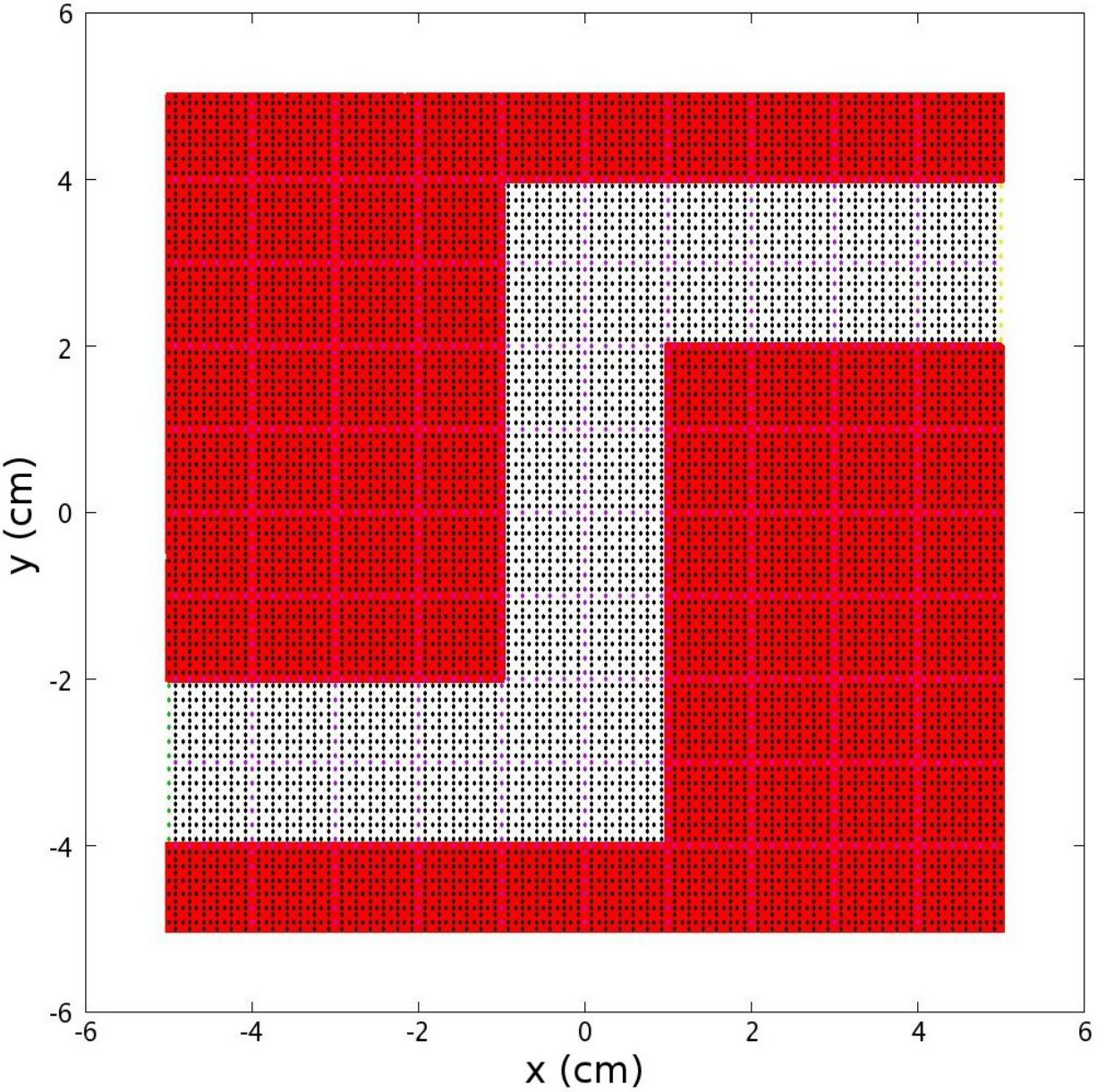


Figure 5.14. Three-legged duct problem geometry

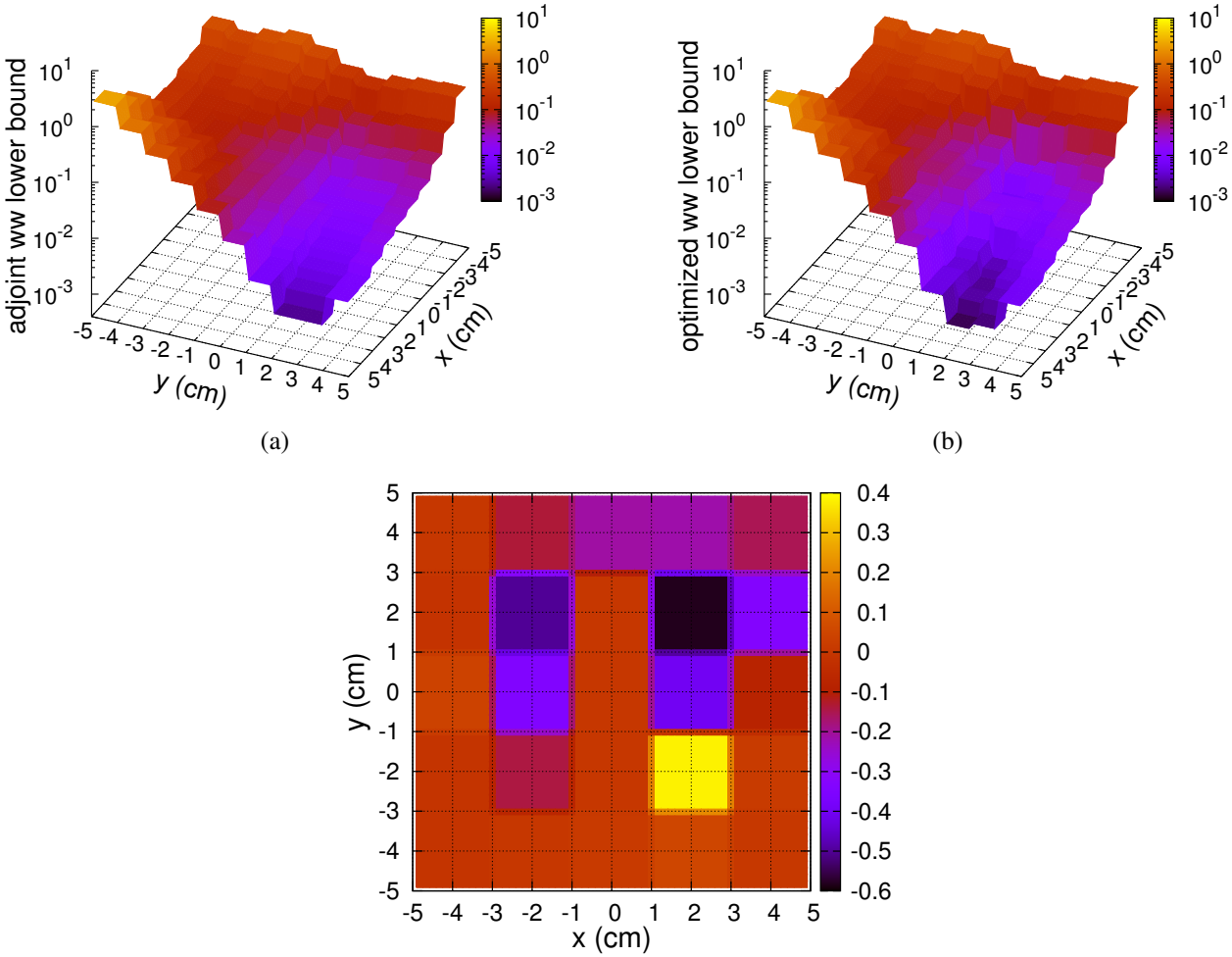


Figure 5.15. Top hat problem geometry weight window lower bound for the (a) adjoint generated weigh windows and (b) optimized weight windows. (c) the fractional difference between the optimized weight windows and the adjoint generated weight windows

Chapter 6

Conclusions and Future Work

The preceeding chapters present work that optimizes Monte Carlo calculations with deterministic methods. Specifically, the weight-window lower bounds for both one- and two-dimensional calculations have been optimized. These calculations extend previous work by considering weight-dependent variance reduction techniques and attempting to estimate the expected time-per-hisory of MCNP, the Monte Carlo code used for comparison. To perform the general weight-dependent calculations, the Monte Carlo particle weight must be included as a parameter of the phase space and is another required discretization.

Using the adjoint importance calculation as an initial guess and a simple gradient-descent optimization algorithm, optimized sets of weight-windows were determined. In most cases, the optimizations increased the efficiencies of the calculations by 20-30%, but for the iron-window-like problem the efficiency was increased by over 200%.

6.1 Limitations of the Method

The technique for optimizing Monte Carlo calculations with general weight-dependent variance reduction has some limitations. Aside from standard problems encountered with the discrete ordinates method, such as ray effects and limited geometries, the additional Monte Carlo particle weight domain introduces additional complications. The two largest concerns are the computation time required to perform the optimizations and the significant increase in the amount of memory required to store the problem introduced by the addition of the weight domain discretization.

The S_n code developed for this work was incredibly simple. Although it was parallelized using OpenMP threading over the directional ordinates, the two dimensional calculations still required up to a day to complete the optimizations and the problem sizes were generally small. Were this method implemented into a production-level discrete ordinates code, such as PARTISN [Alcouffe et al., 2008], with sophisticated parallel sweep routines and domain decompositions, it is not unrealistic to expect significant calculation speed increases and the ability solve more complex problems.

The primary limitation of the method is the additional computer memory required to store the Monte Carlo particle weight dimension. For the majority of the optimization calculations, a weight domain of 10^{-9} to 10^{1} was spanned with 501 weight bins and produced reasonable enough estimates of the variances and expected time per history to calculate the optimization. With this

number of weight bins, the second moment calculation requires approximately 501 times as much memory as the weight independent first moment calculation. The memory requirements were exceedingly large and an alternative method of discretizing the weight domain likely will be required if general weight-dependent calculations are to be performed in future work.

6.2 Future Work

Even with the practical limitations presented above, this work has provided a useful tool for the study of both weight-independent and weight-dependent variance reduction. Furthermore, it has demonstrated extensibility to two-dimensional deterministic transport problems, and it is reasonable to conclude that the calculations could also be performed on a three dimensional problem if the memory and runtime problems were resolved.

While the study of the weight-dependent games provides useful insight into how variances are being reduced, the full weight dependent treatment may not always be required. Booth and Burn [1993] demonstrated that the Direct Statistical Approach (DSA) method could significantly out perform importance based biasing methods. In Booth and Burn's work, the DSA technique was used to optimize splitting parameters and the weight-window generator was used to generate weight windows, and, for the problems investigated, the DSA technique tended to produce more efficient calculations even though it employed weight-independent variance reduction techniques.

One possibility for improving on the results from the DSA method might be to convert the splitting parameters into a weight window. This concept is very similar to how MCNP's weight-window generator actually estimates an importance and then converts the importance into a weight window. Similarly, were deterministic methods used only to optimize splitting/rouletting parameters, then likely those splitting/rouletting parameters could also be converted into a weight window. Because splitting and rouletting are weight independent variance reduction there is no need for the additional weight domain discretization.

Another exciting possibility is the deterministic optimization of weight-independent variance parameters. For weight independent variance reduction, the weight dependence of the solution does not need to be accounted for explicitly. Therefore, the variances and time can be calculated much more quickly than with the weight-dependence. This is similar in concept to the work of Juzaitis [1982], but for a much larger number of cells. Also, the deterministically optimized importances can potentially be converted into a set of weight windows, much like the proposed discussion for the DSA technique above.

One of the largest approximations in this work is the calculation of the expected MCNP time per history by the S_n code. The effort required to profile MCNP for specific problems makes the estimation of the calculation times an exceedingly time consuming task. It has been demonstrated in this work that even a potentially poor estimation of the calculation times works well provided the ratios of the estimated times is equivalent to the ratio of the realized times. If the Monte Carlo code and S_n code were linked, another possibility would be for the S_n code to make direct calls to the Monte Carlo code for timing information when needed.

Bibliography

R.E. Alcouffe, R.S. Baker, J.A. Dahl, S.A. Turner, and R.C. Ward. Partisn: A time-dependent, parllel neutral particle transport code system. Technical Report LA-UR-08-07258, Los Alamos National Laboratory, 2008.

H.J. Amster and M.J. Djomehri. Prediction of statistical error in monte carlo transport caclulations. *Nucl. Sci. Engg.*, 60:131–142, 1976.

G.I. Bell and S. Glasstone. *Nuclear Reactor Theory*. Robert E. Kreiger Publishing Co., 1970.

T.E. Booth. Automatic importance estimation in forward monte calro calculations. *Tran. Amer. Nucl. Soc.*, 41:308, 1982.

T.E. Booth and K.W. Burn. Some sample problem comparisons between the dsa cell model and the quasi-deteriministic method. *Ann. Nucl. Energy*, 20:733–763, 1993.

T.E. Booth and E.D. Cashwell. Analysis of errors in monte carlo transport calculations. *Nucl. Sci. Engg.*, 71:128–142, 1979.

T.E. Booth and J.S. Hendricks. Importance estimation in forward monte carlo calculations. *Nuclear Technology and Fusion*, 5:90–100, 1984.

K.W. Burn. Complete optimization of space/energy cell importances with the dsa cell importance model. *Ann. Nucl. Energy*, 19:65–98, 1992.

K.W. Burn. Extending the direct statistical approach to include particle bifurcation between the splitting surfaces. *Nucl. Sci. Engg.*, 119:44–79, 1995.

K.W. Burn. A new weight-dependent direct statistical approach model. *Nucl. Sci. Engg.*, 125: 128–170, 1997.

L.L. Carter and E.D. Cashwell. Particle-transport simulation with the monte carlo method. Technical Report TID-26607, Technical Information Center Energy Research and Development Administration, 1975.

A. Dubi. A general model for geometrical splitting in monte carlo–i. *Transport Theory and Statistical Physics*, 14:195, 1985a.

A. Dubi. A general model for geometrical splitting in monte carlo–ii. *Transport Theory and Statistical Physics*, 14:195, 1985b.

A. Dubi, A. Goldfeld, and K. Burn. Application of the single surface extended model of geometrical splitting in monte-carlo. *Nucl. Sci. Engg.*, 91:470–480, 1985.

A. Dubi, A. Goldfeld, and K. Burn. Application of the direct statistical approach on a multisurface splitting sroblem in monte-carlo calculations. *Nucl. Sci. Engg.*, 93:204–213, 1986.

J.J. Duderstadt and W.R. Martin. *Transport Theory*. John Wiley & Sons, 1978.

W.L. Dunn and J.K. Shultis. *Exploring Monte Carlo Methods (2008 Draft)*. Elsevier, 2008.

R.J. Juzaitis. Predicting the cost of splitting in monte carlo particle transport. *Nucl. Sci. Engg.*, 80: 424–447, 1982.

E.E. Lewis and W.F. Miller. *Computational Methods of Neutron Transport*. American Nuclear Society, Inc., 1993.

I. Lux. Variance versus efficiency in transport monte carlo. *Nucl. Sci. Engg.*, 73:66–75, 1980.

R.D. O'Dell and R.E. Alcouffe. Transport calculations for nuclear analyses: Theory and guidlines for effective use of transport codes. Technical Report LA-10983-MS, Los Alamos National Laboratory, 1987.

P.K. Sarkar and M.A. Prasad. Prediction of statistical error and optimization of biased monte carlo transport caclculations. *Nucl. Sci. Engg.*, 70:243–261, 1979.

A.J. van Wijk and J.E. Hoogenboom. A priori efficiency calculations for monte carlo applications in neutron transport. In *Joint International Conference on Supercomputin in Nuclear Applications and Monte Carlo*, Hitotsubashi Memorial Hall, Tokyo, Japan, October 17–21 2010.

J.C. Wagner and A. Haghighat. Automated variance reduction of monte carlo shielding calculations using the discrete ordinates adjoint function. *Nucl. Sci. Engg.*, 128:186–208, 1998.

X-5 Monte Carlo Team. Mcnp–a general monte carlo n-particle transport code, version 5. Technical Report LA-UR-03-1987, Los Alamos National Laboratory, 2003.

Appendix A

Verification of History Score Moment Equations Solutions

A discrete ordinates transport code was written to calculate the first and second moments of a history-score distribution for a tally in a Monte Carlo transport problem. Much effort has been devoted to ensuring that the discrete ordinates solution method described in Chapter 3 can produce reasonable approximations to the variances of the Monte Carlo transport problem. This chapter summarizes a substantial amount of the comparison between the discrete ordinates transport code and the Monte Carlo transport code MCNP.

A.1 The MCNP Particle Transport Code

The Monte Carlo N-Particle transport code is a neutron, photon, and electron Monte Carlo code developed at Los Alamos National Laboratory [X-5 Monte Carlo Team, 2003]. MCNP allows the user to specify a geometry, materials, radiation source, and tallies and performs a Monte Carlo radiation transport calculation. The result from the code is typically the mean of a tally and the uncertainty in that mean.

To perform the discrete ordinates calculations so that they are comparable to the results from MCNP5 (RSICC version 1.51), it is necessary to understand not only how the variance reduction techniques in MCNP work in theory but also how they are implemented. It is of utmost importance that the order and manner by which variance reduction games are executed in the MCNP code is reflected in the discrete ordinance code. If the variance reduction games are not correctly reflected, then, while the first moments may be computed correctly, the second moments most assuredly are not.

The method by which certain variance reduction techniques and combinations thereof merits some discussion. Non-integer importance splitting in MCNP always splits the particles to the same weight. For example, a particle with unit weight leaving an importance 1 region and entering an importance 2.5 region might be split 2-to-1 with each particle having weight $1/2$ 50% of the time and 3-to-1 with each particle having weight $1/3$ the other 50%. In MCNP the particle is split 2-to-1 50% of the time and 3-to-1 the other 50%, but each particle has weight $1/2.5$ regardless of the splitting magnitude. The first method preserves weight at every split. MCNP's method preserves weight on average and minimizes the distribution of particle weights.

MCNP has the ability to play the weight window game at surface crossing and/or collisions. However, as of MCNP5 version 1.51, if the weight window game is played only at surface crossings, the implicit capture and weight cutoff games are deactivated. Furthermore, when a particle is sourced into a weight window, if only surface weight windows are employed, the weight window game is ignored on the sourced particle. These quirks in the implementation of MCNP's weight-window method had to be considered in writing the S_n code.

A.2 Semianalytic Pure-Absorber Importance-Splitting Comparisons

In this section, the moment equations are solved analytically for a one-group purely absorbing slab of thickness T. Internal to the slab at x_s is a k-to-1 splitting surface. The physical source is isotropic on the left side of the slab, and the tally is a current tally on the right side of the slab. The geometry is presented in Fig. A.1.

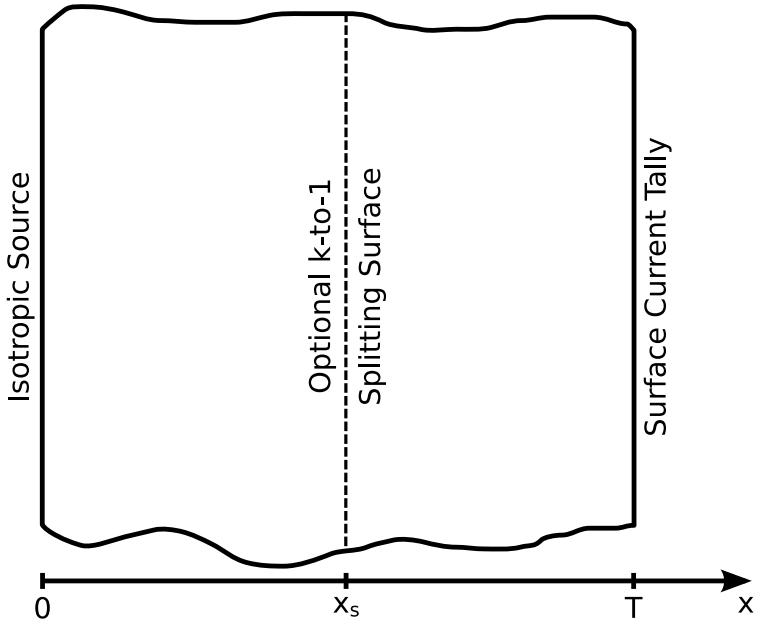


Figure A.1. Purely absorbing slab geometry with optional k-to-1 splitting surface

For a Monte Carlo surface tally, the rth moment detector response is the number of particles crossing the surface at $\mathbf{r}_s$ in specific direction times the score for particles crossing in that direction to the rth power. For a current tally (MCNP F1 tally) this can be expressed as

$$\mathbb{R}_{F1} = w^r |\mathbf{\Omega}\cdot\hat{\mathbf{n}}| \delta(\mathbf{r} - \mathbf{r}_s), \tag{A.1}$$

where $\hat{\mathbf{n}}$ is the unit normal for the surface and the weight w is the score. Similarly for a surface-flux tally (MCNP F2 tally), the response is

$$\mathbb{R}_{F2} = \left(\frac{w}{\mathbf{\Omega}\cdot\hat{\mathbf{n}}A}\right)^r |\mathbf{\Omega}\cdot\hat{\mathbf{n}}| \delta(\mathbf{r} - \mathbf{r}_s). \tag{A.2}$$

The response function is the source for adjoint equations such as the moment equations. Therefore, for the 1-D purely absorbing slab problem in Fig. A.1 the moment equation can be expressed

$$-\mu\frac{dM_r}{dx} + \Sigma M_r(x,\mu,w) = w^r|\mu|\delta(x-T). \tag{A.3}$$

Dividing by $-\mu$, the equation above becomes

$$\frac{dM_r}{dx} - \frac{\Sigma}{\mu}M_r(x,\mu,w) = -w^r\frac{|\mu|}{\mu}\delta(x-T), \tag{A.4}$$

and multiplication by the integrating factor $\exp(-\Sigma/\mu(x-x_o))$, where x_o is a point in the slab less than x, gives

$$\frac{dM_r}{dx}e^{-\frac{\Sigma}{\mu}(x-x_o)} - \frac{\Sigma}{\mu}M_r(x,\mu,w)e^{-\frac{\Sigma}{\mu}(x-x_o)} = -w^r\frac{|\mu|}{\mu}\delta(x-T)e^{-\frac{\Sigma}{\mu}(x-x_o)}. \tag{A.5}$$

For $x < T$, μ must be greater than zero for the integrating factor to cause a decrease, and the derivation is restricted to $\mu > 0$ with the Heaviside function $\Theta(\mu)$

$$\left[\frac{dM_r}{dx}e^{-\frac{\Sigma}{\mu}(x-x_o)} - \frac{\Sigma}{\mu}M_r(x,\mu,w)e^{-\frac{\Sigma}{\mu}(x-x_o)}\right]\Theta(\mu) = -w^r\delta(x-T)e^{-\frac{\Sigma}{\mu}(x-x_o)}\Theta(\mu). \tag{A.6}$$

The left hand side of the above equation can be written as a derivative with respect to x as

$$\frac{d}{dx}\left[M_r(x,\mu,w)e^{-\frac{\Sigma}{\mu}(x-x_o)}\right]\Theta(\mu) = -w^r\delta(x-T)e^{-\frac{\Sigma}{\mu}(x-x_o)}\Theta(\mu), \tag{A.7}$$

and integration from (x,T) gives

$$\int_x^T dx'\ \frac{d}{dx'}\left[M_r(x',\mu,w)e^{-\frac{\Sigma}{\mu}(x'-x_o)}\right]\Theta(\mu) = -\int_x^T dx'\ w^r\delta(x'-T)e^{-\frac{\Sigma}{\mu}(x'-x_o)}\Theta(\mu). \tag{A.8}$$

Evaluation of the integrals gives

$$\left[M_r(T,\mu,w)e^{-\frac{\Sigma}{\mu}(T-x_o)} - M_r(x,\mu,w)e^{-\frac{\Sigma}{\mu}(x-x_o)}\right]\Theta(\mu) = -w^r e^{-\frac{\Sigma}{\mu}(T-x_o)}\Theta(mu). \tag{A.9}$$

From the adjoint boundary condition that nothing exits the slab, i.e. $M_r(T,\mu,w) = 0,\ \mu > 0$, the expression above simplifies to

$$-M_r(x,\mu,w)e^{-\frac{\Sigma}{\mu}(x-x_o)}\Theta(\mu) = -w^r e^{-\frac{\Sigma}{\mu}(T-x_o)}\Theta(mu), \tag{A.10}$$

and, solving for $M_r(x,\mu,w)$ gives

$$M_r(x,\mu,w)\Theta(\mu) = w^r e^{-\frac{\Sigma}{\mu}(T-x)}\Theta(\mu). \tag{A.11}$$

Ignoring for a moment the internal splitting surface, Eq. (A.11) applies throughout the slab for the rth moment. Assuming that all Monte Carlo particles are born isotropically with weight w_o

at $x = 0$, the physical source distribution $S(x,\mu,w) = 1/2\delta(x)\delta(w - w_o)$ and the first and second moment responses are given by

$$\begin{aligned}
\mathbb{R}_{M_1} &= \int_0^1 d\mu \int_0^T dx \int_0^\infty dw S(x,\mu,w) M_1(x,\mu,w) \\
&= \frac{1}{2}\int_0^1 d\mu \int_0^T dx \int_0^\infty dw \delta(x)\delta(w - w_o) w e^{-\frac{\Sigma}{\mu}(T-x)} \\
&= \frac{w_o}{2}\int_0^1 d\mu\, e^{-\frac{\Sigma}{\mu}T},
\end{aligned} \tag{A.12}$$

and

$$\begin{aligned}
\mathbb{R}_{M_2} &= \int_0^1 d\mu \int_0^T dx \int_0^\infty dw S(x,\mu,w) M_2(x,\mu,w) \\
&= \frac{1}{2}\int_0^1 d\mu \int_0^T dx \int_0^\infty dw \delta(x)\delta(w - w_o) w^2 e^{-\frac{\Sigma}{\mu}(T-x)} \\
&= \frac{w_o^2}{2}\int_0^1 d\mu\, e^{-\frac{\Sigma}{\mu}T},
\end{aligned} \tag{A.13}$$

respectively. The integrals over μ can be evaluated numerically to obtain the moments.

If the splitting surface at x_s is considered, then it divides slab into a left cell and a right cell. In the right cell, Eq. (A.11) still applies, namely

$$M_{r,R} = w^r e^{-\frac{\Sigma}{\mu}(T-x)}\Theta(\mu), \tag{A.14}$$

but the moment in the left cell $M_{r,L}$ is modified by the splitting. For the first moment, the value just to the left of the splitting surface $M_{1,L}(x_s^-,\mu,w)$ is given by

$$\begin{aligned}
M_{1,L}(x_s^-,\mu,w) &= k\int_0^\infty dw'\ \delta(w' - w/k) M_{r,R}(x_s^+,\mu,w') \\
&= k\int_0^\infty dw'\ \delta(w' - w/k) w' e^{-\frac{\Sigma}{\mu}(T-x_s)} \\
&= k\frac{w}{k}e^{-\frac{\Sigma}{\mu}(T-x_s)} \\
&= we^{-\frac{\Sigma}{\mu}(T-x_s)},
\end{aligned} \tag{A.15}$$

and the first moment remains unchanged by the splitting. Applying the exponential attenuation from x_s to zero, one obtains

$$\begin{aligned}
M_{1,L}(x,\mu,w) &= we^{-\frac{\Sigma}{\mu}(T-x_s)} \times e^{-\frac{\Sigma}{\mu}(x_s-x)} \\
&= we^{-\frac{\Sigma}{\mu}(T-x)},
\end{aligned} \tag{A.16}$$

exactly as without the splitting surface.

The second moment behaves in a similar manner with the addition of source from the first moment given by Eq. (2.67). Thus, the second moment just to the left of the splitting surface is

$$M_{2,L}(x_s^-,\mu,w) = \int_0^\infty dw'\ \delta(w' - w/k)\left[kM_{2,R}(x_s^+,\mu,w') + k(k-1)M_{1,R}^2(x_s^+,\mu,w')\right]$$

$$= \int_0^\infty dw'\ \delta(w' - w/k)\left[kw'^2 e^{-\frac{\Sigma}{\mu}(T-x_s)} + k(k-1)w'^2 e^{-\frac{2\Sigma}{\mu}(T-x_s)}\right]$$
$$= \frac{w^2}{k} e^{-\frac{\Sigma}{\mu}(T-x_s)} + \frac{(k-1)w^2}{k} e^{-\frac{2\Sigma}{\mu}(T-x_s)}. \tag{A.17}$$

Again, applying exponential attenuation from x_s to zero, the second moment in the left cell is given by

$$M_{2,L}(x_s^-, \mu, w) = \frac{w^2}{k} e^{-\frac{\Sigma}{\mu}(T-x)} + \frac{(k-1)w^2}{k} e^{-\frac{2\Sigma}{\mu}(T-x_s)} e^{-\frac{\Sigma}{\mu}(x_s-x)}. \tag{A.18}$$

Interestingly, as $k \to \infty$ the first term, which originated from the second moment in the right cell, vanishes leaving only the term $w^2 \exp(-2\Sigma/\mu(T - x_s)) \exp(-\Sigma/\mu(x_s - s))$, which originated from the addtional source from the first moment.

Considering the same source distribution given before $S(x, \mu, w) = 1/2\delta(x)\delta(w - w_o)$, the first and second moment responses with splitting are

$$\mathbb{R}_{M_1} = \frac{w_o}{2} \int_0^1 d\mu\ e^{-\frac{\Sigma}{\mu}T} \tag{A.19}$$

and

$$\mathbb{R}_{M_2} = \frac{w_o^2}{2k} \int_0^1 d\mu\ \left[e^{-\frac{\Sigma}{\mu}T} + (k-1)e^{-\frac{2\Sigma}{\mu}(T-x_s)} e^{-\frac{\Sigma}{\mu}x_s}\right]. \tag{A.20}$$

Again, both these responses can be numerically integrated over μ. Table A.1 presents a direct comparison of MCNP calculations, the S_n code, and numerically integrated semi-analytic results for various splitting cases.

Table A.1. Comparison of MCNP, S_n, and semianalytic results for a single k-to-1 splitting surface at $x_s = T/2$ for $\Sigma = 1$ cm^{-1} and $T = 5$ cm

k	MCNP		S_n[a]		Semianalytic[b]	
	M_1	σ^2[c]	M_1	σ^2	M_1	σ^2
1	4.96380E-04	4.96134E-04	4.97456E-04	4.97209E-04	4.98235E-04	4.97986E-04
2	4.97900E-04	2.63656E-04	4.97456E-04	2.63325E-04	4.98235E-04	2.63707E-04
5	4.98116E-04	1.23109E-04	4.97456E-04	1.22920E-04	4.98235E-04	1.23140E-04
10	4.98062E-04	7.62872E-05	4.97456E-04	7.61526E-05	4.98235E-04	7.62839E-05
15	4.98117E-04	6.06588E-05	4.97456E-04	6.05835E-05	4.98235E-04	6.06653E-05
25	4.98420E-04	4.82101E-05	4.97456E-04	4.80688E-05	4.98235E-04	4.81704E-05
50	4.98364E-04	3.88170E-05	4.97456E-04	3.87200E-05	4.98235E-04	3.87993E-05
100	4.98463E-04	3.41392E-05	4.97456E-04	3.40469E-05	4.98235E-04	3.41137E-05

[a] S_n calculations used 3000 weight bins spanning 10^{-2} through 10^{1}

[b] Semi-analytic results obtained by a 96-ordinate Gauss-Legendre quadrature integration of Eq. (A.19) and Eq. (A.20)

[c] MCNP variances are multiplied by the number of histories to obtain the population variances

A.3 1-D Verification of S_n and MCNP Calculations for Different Tallies and Combinations of Variance Reduction

The S_n code developed to calculate variances of the history-score moment equations was extensively compared to one-group MCNP (RSICC version 5 1.51) calculations. The tables contained in this section present the results of these comparisons. Each table presents the results for an S_n quadrature order n and the number of weight bins w. The extent of the weight domain spanned by the upper and lower truncation weights is given next to the number of weight bins as $[w_{lt}, w_{ut}]$. Any variance reduction parameters necessary are either given in the table body or the caption.

All calculations presented below were computed using MCNP's multigroup-adjoint capabilities. One-group cross-section sets were generated corresponding to the desired scattering and capture ratios using the MAKEMG utility. The population variances of the MCNP calculations were estimated from the sample variances by

$$\sigma^2_{\text{approx}} = \left[\overline{x}R(\overline{x})\right]^2 \times N, \tag{A.21}$$

where σ^2_{approx} is the approximated population variance, $\overline{x}$ is the tally mean, $R(\overline{x})$ is the tally relative error, and N is the number of histories used in the computation.

For all the results, the percent error is reported between the population variance computed by MCNP and the population variance computed with the S_n code. The population variance computed with MCNP is assumed to be the actual population variance, so the percent errors are computed as

$$\%\ \text{error} = \frac{\left|\sigma^2_{S_n} - \sigma^2_{\text{MCNP}}\right|}{\sigma^2_{\text{MCNP}}} \times 100. \tag{A.22}$$

A.3.1 Surface Current Tally

Analog

Table A.2. Analog 1-D slab problem with slab thickness of 5 cm

$n, w\ [w_{lt}, w_{ut}]$	Cell 1	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 32$ $w = 501, \left[10^{-8}, 10^{1}\right]$	$c = 0$	4.96380E-04	4.96134E-04	4.97456E-04	4.99934E-04	0.77
$n = 32$ $w = 501, \left[10^{-8}, 10^{1}\right]$	$c = 0.85$	1.37294E-02	1.35408E-02	1.37041E-02	1.35914E-02	0.37

Importance Splitting/Rouletting

Table A.3. Importance Splitting/Rouletting in 1-D slab problem with two equally sized cells for slab thickness of 5 cm

$n, w\ [w_{lt}, w_{ut}]$	Cell 1	Cell 2	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 32$ $w = 501, \left[10^{-8}, 10^{1}\right]$	$c = 0$ $I = 1$	$c = 0$ $I = 2$	5.04450E-04	2.67501E-04	4.97456E-04	2.69928E-04	0.91

n, w $[w_{lt}, w_{ut}]$	Cell 1	Cell 2	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 32$ $w = 501, [10^{-8}, 10^{1}]$	$c = 0$ $I = 1$	$c = 0$ $I = 4$	5.01665E-04	1.48120E-04	4.97456E-04	1.46621E-04	1.01
$n = 32$ $w = 501, [10^{-8}, 10^{1}]$	$c = 0$ $I = 1$	$c = 0$ $I = 8$	4.99862E-04	8.85937E-05	4.97456E-04	8.96007E-05	1.14
$n = 32$ $w = 1501, [10^{-8}, 10^{1}]$	$c = 0$ $I = 1$	$c = 0$ $I = 8$	4.99862E-04	8.85937E-05	4.97456E-04	8.83088E-05	0.32
$n = 32$ $w = 501, [10^{-8}, 10^{1}]$	$c = 0.85$ $I = 1$	$c = 0.85$ $I = 2$	1.37299E-02	8.34531E-03	1.37041E-02	8.54496E-03	2.39
$n = 32$ $w = 1501, [10^{-8}, 10^{1}]$	$c = 0.85$ $I = 1$	$c = 0.85$ $I = 2$	1.37299E-02	8.34531E-03	1.37041E-02	8.40883E-03	0.76
$n = 32$ $w = 1501, [10^{-8}, 10^{1}]$	$c = 0.85$ $I = 1$	$c = 0.85$ $I = 4$	1.37246E-02	5.74289E-03	1.37041E-02	5.74745E-03	0.08
$n = 32$ $w = 501, [10^{-8}, 10^{1}]$	$c = 0.85$ $I = 1$	$c = 0.85$ $I = 8$	1.37125E-02	4.43685E-03	1.37041E-02	4.52905E-03	2.08
$n = 32$ $w = 1501, [10^{-8}, 10^{1}]$	$c = 0.85$ $I = 1$	$c = 0.85$ $I = 8$	1.37125E-02	4.43685E-03	1.37041E-02	4.46136E-03	0.55
$n = 32$ $w = 501, [10^{-8}, 10^{1}]$	$c = 0$ $I = 1$	$c = 0.85$ $I = 2$	1.94183E-03	1.16512E-03	1.93401E-03	1.18615E-03	1.81
$n = 32$ $w = 1501, [10^{-8}, 10^{1}]$	$c = 0$ $I = 1$	$c = 0.85$ $I = 2$	1.94183E-03	1.16512E-03	1.93401E-03	1.16792E-03	0.24
$n = 32$ $w = 501, [10^{-8}, 10^{1}]$	$c = 0$ $I = 1$	$c = 0.85$ $I = 4$	1.93867E-03	7.72388E-04	1.93401E-03	7.71729E-04	0.09
$n = 32$ $w = 501, [10^{-8}, 10^{1}]$	$c = 0$ $I = 1$	$c = 0.85$ $I = 8$	1.94163E-03	5.80368E-04	1.93401E-03	5.88381E-04	1.38
$n = 32$ $w = 1501, [10^{-8}, 10^{1}]$	$c = 0$ $I = 1$	$c = 0.85$ $I = 8$	1.94163E-03	5.80368E-04	1.93401E-03	5.79867E-04	0.09

Table A.4. Importance Splitting/Rouletting in 1-D slab problem with four equally sized cells for slab thickness of 5 cm

n, w $[w_{lt}, w_{ut}]$	Cell 1	Cell 2	Cell 3	Cell 4	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 32$ $w = 501, [10^{-8}, 10^{1}]$	$c = 0$ $I = 1$	$c = 0$ $I = 2$	$c = 0$ $I = 4$	$c = 0$ $I = 8$	4.99005E-04	8.85005E-05	4.97456E-04	9.35489E-05	5.70
$n = 32$ $w = 1501, [10^{-8}, 10^{1}]$	$c = 0$ $I = 1$	$c = 0$ $I = 2$	$c = 0$ $I = 4$	$c = 0$ $I = 8$	4.99005E-04	8.85005E-05	4.97456E-04	8.96592E-05	1.30
$n = 32$ $w = 4501, [10^{-8}, 10^{1}]$	$c = 0$ $I = 1$	$c = 0$ $I = 2$	$c = 0$ $I = 4$	$c = 0$ $I = 8$	4.99005E-04	8.85005E-05	4.97456E-04	8.80807E-05	0.47
$n = 32$ $w = 501, [10^{-8}, 10^{1}]$	$c = 0$ $I = 1$	$c = 0$ $I = 4$	$c = 0$ $I = 16$	$c = 0$ $I = 64$	4.98327E-04	2.43321E-05	4.97456E-04	2.42294E-05	0.42
$n = 32$ $w = 501, [10^{-8}, 10^{1}]$	$c = 0$ $I = 1$	$c = 0$ $I = 2$	$c = 0$ $I = 3$	$c = 0$ $I = 4$	4.98225E-04	1.49135E-04	4.97456E-04	1.49820E-04	0.45
$n = 32$ $w = 501, [10^{-8}, 10^{1}]$	$c = 0.85$ $I = 1$	$c = 0.85$ $I = 2$	$c = 0.85$ $I = 4$	$c = 0.85$ $I = 8$	1.37170E-02	3.98013E-03	1.37041E-02	4.20861E-03	5.74
$n = 32$ $w = 1501, [10^{-8}, 10^{1}]$	$c = 0.85$ $I = 1$	$c = 0.85$ $I = 2$	$c = 0.85$ $I = 4$	$c = 0.85$ $I = 8$	1.37170E-02	3.98013E-03	1.37041E-02	4.04637E-03	1.66
$n = 32$ $w = 4501, [10^{-8}, 10^{1}]$	$c = 0.85$ $I = 1$	$c = 0.85$ $I = 2$	$c = 0.85$ $I = 4$	$c = 0.85$ $I = 8$	1.37170E-02	3.98013E-03	1.37041E-02	3.97927E-03	0.02

n, w $[w_{lt}, w_{ut}]$	Cell 1	Cell 2	Cell 3	Cell 4	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 32$ $w = 501, \left[10^{-8}, 10^{1}\right]$	$c = 0.85$ $I = 1$	$c = 0.85$ $I = 4$	$c = 0.85$ $I = 16$	$c = 0.85$ $I = 64$	1.37202E-02	2.07298E-03	1.37041E-02	2.06917E-03	0.18
$n = 32$ $w = 501, \left[10^{-8}, 10^{1}\right]$	$c = 0.85$ $I = 1$	$c = 0.85$ $I = 2$	$c = 0.85$ $I = 3$	$c = 0.85$ $I = 4$	1.37215E-02	5.60326E-03	1.37041E-02	5.63164E-03	0.51
$n = 32$ $w = 501, \left[10^{-8}, 10^{1}\right]$	$c = 0$ $I = 1$	$c = 0.85$ $I = 2$	$c = 0$ $I = 4$	$c = 0.85$ $I = 8$	1.69685E-03	3.77551E-04	1.69189E-03	3.98819E-04	5.63
$n = 32$ $w = 1501, \left[10^{-8}, 10^{1}\right]$	$c = 0$ $I = 1$	$c = 0.85$ $I = 2$	$c = 0$ $I = 4$	$c = 0.85$ $I = 8$	1.69685E-03	3.77551E-04	1.69189E-03	3.82768E-04	1.38
$n = 32$ $w = 4501, \left[10^{-8}, 10^{1}\right]$	$c = 0$ $I = 1$	$c = 0.85$ $I = 2$	$c = 0$ $I = 4$	$c = 0.85$ $I = 8$	1.69685E-03	3.77551E-04	1.69189E-03	3.76188E-04	0.36
$n = 32$ $w = 501, \left[10^{-8}, 10^{1}\right]$	$c = 0$ $I = 1$	$c = 0.85$ $I = 4$	$c = 0$ $I = 16$	$c = 0.85$ $I = 64$	1.69612E-03	1.37924E-04	1.69189E-03	1.37217E-04	0.51
$n = 32$ $w = 501, \left[10^{-8}, 10^{1}\right]$	$c = 0$ $I = 1$	$c = 0.85$ $I = 2$	$c = 0$ $I = 3$	$c = 0.85$ $I = 4$	1.69684E-03	5.84059E-04	1.69189E-03	5.84827E-04	0.13

Table A.5. Importance Splitting/Rouletting in 1-D slab problem with ten equally sized cells for slab thickness of 5 cm

n, w $[w_{lt}, w_{ut}]$	Cell i	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 32$ $w = 501, \left[10^{-8}, 10^{1}\right]$	$c_i = 0.85$ $I_i = 2^{i-1}$	1.37081E-02	1.18067E-03	1.37045E-02	1.27539E-03	8.02
$n = 32$ $w = 1001, \left[10^{-8}, 10^{1}\right]$	$c_i = 0.85$ $I_i = 2^{i-1}$	1.37081E-02	1.18067E-03	1.37045E-02	1.17862E-03	0.17

Implicit Capture and Weight Cutoff

Table A.6. Implicit capture and weight cutoff in 1-D slab problem with slab thickness of 5 cm, cutoff weight of 0.25 and survival weight of 0.5

n, w $[w_{lt}, w_{ut}]$	Cell 1	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 32$ $w = 501, \left[10^{-9}, 10^{1}\right]$	$c = 0.5$	1.83570E-03	1.16267E-03	1.84161E-03	1.17190E-03	0.79
$n = 32$ $w = 501, \left[10^{-9}, 10^{1}\right]$	$c = 0.85$	1.37138E-02	6.43122E-03	1.37041E-02	6.57490E-03	2.23
$n = 32$ $w = 1501, \left[10^{-9}, 10^{1}\right]$	$c = 0.85$	1.37138E-02	6.43122E-03	1.37041E-02	6.51459E-03	1.30
$n = 32$ $w = 3001, \left[10^{-9}, 10^{1}\right]$	$c = 0.85$	1.37138E-02	6.43122E-03	1.37041E-02	6.39667E-03	0.54

Table A.7. Importance Splitting/Rouletting, implicit capture, and weight cutoff in 1-D slab problem with four equally sized cells for slab thickness of 5 cm, cutoff weight of 0.25, and survival weight 0.5

n, w $[w_{lt}, w_{ut}]$	Cell 1	Cell 2	Cell 3	Cell 4	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 32$ $w = 501, \left[10^{-9}, 10^{1}\right]$	$c = 0.5$ $I = 1$	$c = 0.5$ $I = 2$	$c = 0.5$ $I = 4$	$c = 0.5$ $I = 8$	1.84621E-03	2.43778E-04	1.84161E-03	2.47111E-04	1.37
$n = 32$ $w = 1501, \left[10^{-10}, 10^{1}\right]$	$c = 0.5$ $I = 1$	$c = 0.5$ $I = 2$	$c = 0.5$ $I = 4$	$c = 0.5$ $I = 8$	1.84621E-03	2.43778E-04	1.84161E-03	2.43943E-04	0.07
$n = 32$ $w = 501, \left[10^{-9}, 10^{1}\right]$	$c = 0.5$ $I = 1$	$c = 0.5$ $I = 2$	$c = 0.5$ $I = 3$	$c = 0.5$ $I = 4$	1.84263E-03	3.86265E-04	1.84161E-03	3.74992E-04	2.92
$n = 32$ $w = 1501, \left[10^{-10}, 10^{1}\right]$	$c = 0.5$ $I = 1$	$c = 0.5$ $I = 2$	$c = 0.5$ $I = 3$	$c = 0.5$ $I = 4$	1.84263E-03	3.86265E-04	1.84161E-03	3.86757E-04	0.13
$n = 32$ $w = 501, \left[10^{-9}, 10^{1}\right]$	$c = 0.5$ $I = 1$	$c = 0.5$ $I = 4$	$c = 0.5$ $I = 16$	$c = 0.5$ $I = 64$	1.84461E-03	8.68005E-05	1.84161E-03	8.86535E-05	2.13
$n = 32$ $w = 1501, \left[10^{-10}, 10^{1}\right]$	$c = 0.5$ $I = 1$	$c = 0.5$ $I = 4$	$c = 0.5$ $I = 16$	$c = 0.5$ $I = 64$	1.84461E-03	8.68005E-05	1.84161E-03	8.70745E-05	0.32
$n = 32$ $w = 501, \left[10^{-9}, 10^{1}\right]$	$c = 0.85$ $I = 1$	$c = 0.85$ $I = 2$	$c = 0.85$ $I = 4$	$c = 0.85$ $I = 8$	1.37103E-02	2.19687E-03	1.37041E-02	2.25667E-03	2.72
$n = 32$ $w = 3001, \left[10^{-9}, 10^{1}\right]$	$c = 0.85$ $I = 1$	$c = 0.85$ $I = 2$	$c = 0.85$ $I = 4$	$c = 0.85$ $I = 8$	1.37103E-02	2.19687E-03	1.37041E-02	2.20107E-03	0.19
$n = 32$ $w = 501, \left[10^{-9}, 10^{1}\right]$	$c = 0.85$ $I = 1$	$c = 0.85$ $I = 2$	$c = 0.85$ $I = 3$	$c = 0.85$ $I = 4$	1.37058E-02	2.96377E-03	1.37041E-02	2.95580E-03	0.27
$n = 32$ $w = 3001, \left[10^{-9}, 10^{1}\right]$	$c = 0.85$ $I = 1$	$c = 0.85$ $I = 2$	$c = 0.85$ $I = 3$	$c = 0.85$ $I = 4$	1.37058E-02	2.96377E-03	1.37041E-02	2.98078E-03	0.57
$n = 32$ $w = 501, \left[10^{-9}, 10^{1}\right]$	$c = 0.85$ $I = 1$	$c = 0.85$ $I = 4$	$c = 0.85$ $I = 16$	$c = 0.85$ $I = 64$	1.37121E-02	1.31371E-03	1.37041E-02	1.35068E-03	2.81
$n = 32$ $w = 1501, \left[10^{-9}, 10^{1}\right]$	$c = 0.85$ $I = 1$	$c = 0.85$ $I = 4$	$c = 0.85$ $I = 16$	$c = 0.85$ $I = 64$	1.37121E-02	1.31371E-03	1.37041E-02	1.30845E-03	0.40

Weight Windows

Table A.8. Surface weight windows in a 1-D slab problem with four equally sized cells for slab thickness of 5 cm, upper window multiplier of 5, survival multiplier of 3, and maximum splitting/rouletting parameter of 5

n, w $[w_{lt}, w_{ut}]$	Cell 1	Cell 2	Cell 3	Cell 4	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 32$ $w = 501, \left[10^{-8}, 10^{1}\right]$	$c = 0$ $\omega_l = 0.5$	$c = 0$ $\omega_l = 0.25$	$c = 0$ $\omega_l = 0.125$	$c = 0$ $\omega_l = 0.0625$	4.99215E-04	1.69775E-04	4.97456E-04	1.76231E-04	3.80
$n = 32$ $w = 1001, \left[10^{-8}, 10^{1}\right]$	$c = 0$ $\omega_l = 0.5$	$c = 0$ $\omega_l = 0.25$	$c = 0$ $\omega_l = 0.125$	$c = 0$ $\omega_l = 0.0625$	4.99215E-04	1.69775E-04	4.97456E-04	1.69016E-04	0.45
$n = 32$ $w = 501, \left[10^{-8}, 10^{1}\right]$	$c = 0$ $\omega_l = 0.5$	$c = 0$ $\omega_l = 0.125$	$c = 0$ $\omega_l = 0.03125$	$c = 0$ $\omega_l = 0.0078125$	4.98182E-04	4.14158E-05	4.97456E-04	4.26430E-05	2.96
$n = 32$ $w = 1001, \left[10^{-8}, 10^{1}\right]$	$c = 0$ $\omega_l = 0.5$	$c = 0$ $\omega_l = 0.125$	$c = 0$ $\omega_l = 0.03125$	$c = 0$ $\omega_l = 0.0078125$	4.98182E-04	4.14158E-05	4.97456E-04	4.11972E-05	0.53
$n = 32$ $w = 501, \left[10^{-8}, 10^{1}\right]$	$c = 0.85$ $\omega_l = 0.5$	$c = 0.85$ $\omega_l = 0.25$	$c = 0.85$ $\omega_l = 0.125$	$c = 0.85$ $\omega_l = 0.0625$	1.37045E-02	6.28237E-03	1.37041E-02	6.57299E-03	4.63
$n = 32$ $w = 1001, \left[10^{-8}, 10^{1}\right]$	$c = 0.85$ $\omega_l = 0.5$	$c = 0.85$ $\omega_l = 0.25$	$c = 0.85$ $\omega_l = 0.125$	$c = 0.85$ $\omega_l = 0.0625$	1.37045E-02	6.28237E-03	1.37041E-02	6.31932E-03	0.59

n, w $[w_{lt}, w_{ut}]$	Cell 1	Cell 2	Cell 3	Cell 4	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 32$ $w = 501, \left[10^{-8}, 10^{1}\right]$	$c = 0.85$ $\omega_l = 0.5$	$c = 0.85$ $\omega_l = 0.125$	$c = 0.85$ $\omega_l = 0.03125$	$c = 0.85$ $\omega_l = 0.0078125$	1.37205E-02	2.60569E-03	1.37041E-02	2.86428E-03	9.92
$n = 32$ $w = 1501, \left[10^{-8}, 10^{1}\right]$	$c = 0.85$ $\omega_l = 0.5$	$c = 0.85$ $\omega_l = 0.125$	$c = 0.85$ $\omega_l = 0.03125$	$c = 0.85$ $\omega_l = 0.0078125$	1.37205E-02	2.60569E-03	1.37041E-02	2.59453E-03	0.43
$n = 32$ $w = 501, \left[10^{-8}, 10^{1}\right]$	$c = 0.85$ $\omega_l = 0.4$	$c = 0.85$ $\omega_l = 0.16$	$c = 0.85$ $\omega_l = 0.064$	$c = 0.85$ $\omega_l = 0.0256$	1.37191E-02	3.74125E-03	1.37041E-02	3.76557E-03	0.65

Table A.9. Collision weight windows in a 1-D slab problem with four equally sized cells for slab thickness of 5 cm, upper window multiplier of 5, survival multiplier of 3, and maximum splitting/rouletting parameter of 5

n, w $[w_{lt}, w_{ut}]$	Cell 1	Cell 2	Cell 3	Cell 4	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 32$ $w = 501, \left[10^{-8}, 10^{1}\right]$	$c = 0.85$ $\omega_l = 0.5$	$c = 0.85$ $\omega_l = 0.25$	$c = 0.85$ $\omega_l = 0.125$	$c = 0.85$ $\omega_l = 0.0625$	1.37201E-02	8.58340E-03	1.37041E-02	8.80590E-03	2.59
$n = 32$ $w = 6001, \left[10^{-8}, 10^{1}\right]$	$c = 0.85$ $\omega_l = 0.5$	$c = 0.85$ $\omega_l = 0.25$	$c = 0.85$ $\omega_l = 0.125$	$c = 0.85$ $\omega_l = 0.0625$	1.37201E-02	8.58340E-03	1.37041E-02	8.65455E-03	0.83
$n = 32$ $w = 501, \left[10^{-8}, 10^{1}\right]$	$c = 0.85$ $\omega_l = 0.5$	$c = 0.85$ $\omega_l = 0.125$	$c = 0.85$ $\omega_l = 0.03125$	$c = 0.85$ $\omega_l = 0.0078125$	1.37167E-02	5.24747E-03	1.37041E-02	5.39854E-03	2.88
$n = 32$ $w = 1501, \left[10^{-8}, 10^{1}\right]$	$c = 0.85$ $\omega_l = 0.5$	$c = 0.85$ $\omega_l = 0.125$	$c = 0.85$ $\omega_l = 0.03125$	$c = 0.85$ $\omega_l = 0.0078125$	1.37167E-02	5.24747E-03	1.37041E-02	5.27452E-03	0.52
$n = 32$ $w = 501, \left[10^{-8}, 10^{1}\right]$	$c = 0.85$ $\omega_l = 0.4$	$c = 0.85$ $\omega_l = 0.16$	$c = 0.85$ $\omega_l = 0.064$	$c = 0.85$ $\omega_l = 0.0256$	1.37171E-02	6.00866E-03	1.37041E-02	6.06768E-03	0.98

Table A.10. Surface and collision weight windows in a 1-D slab problem with four equally sized cells for slab thickness of 5 cm, upper window multiplier of 5, survival multiplier of 3, and maximum splitting/rouletting parameter of 5

n, w $[w_{lt}, w_{ut}]$	Cell 1	Cell 2	Cell 3	Cell 4	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 32$ $w = 501, \left[10^{-8}, 10^{1}\right]$	$c = 0.85$ $\omega_l = 0.5$	$c = 0.85$ $\omega_l = 0.25$	$c = 0.85$ $\omega_l = 0.125$	$c = 0.85$ $\omega_l = 0.0625$	1.37045E-02	6.28237E-03	1.37041E-02	6.30069E-03	0.29
$n = 32$ $w = 501, \left[10^{-8}, 10^{1}\right]$	$c = 0.85$ $\omega_l = 0.5$	$c = 0.85$ $\omega_l = 0.125$	$c = 0.85$ $\omega_l = 0.03125$	$c = 0.85$ $\omega_l = 0.0078125$	1.37076E-02	2.60149E-03	1.37041E-02	2.86430E-03	10.10
$n = 32$ $w = 1501, \left[10^{-8}, 10^{1}\right]$	$c = 0.85$ $\omega_l = 0.5$	$c = 0.85$ $\omega_l = 0.125$	$c = 0.85$ $\omega_l = 0.03125$	$c = 0.85$ $\omega_l = 0.0078125$	1.37076E-02	2.60149E-03	1.37041E-02	2.59446E-03	0.27
$n = 32$ $w = 501, \left[10^{-8}, 10^{1}\right]$	$c = 0.85$ $\omega_l = 0.4$	$c = 0.85$ $\omega_l = 0.16$	$c = 0.85$ $\omega_l = 0.064$	$c = 0.85$ $\omega_l = 0.0256$	1.37191E-02	3.74125E-03	1.37041E-02	3.76557E-03	0.65

Table A.11. Surface and collision weight windows with implicit capture in a 1-D slab problem with four equally sized cells for slab thickness of 5 cm, upper window multiplier of 5, survival multiplier of 3, and a maximum splitting/rouletting parameter of 5

$n, w\ [w_{lt}, w_{ut}]$	Cell 1	Cell 2	Cell 3	Cell 4	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 32$ $w = 501, \left[10^{-8}, 10^{1}\right]$	$c = 0.85$ $\omega_l = 0.4$	$c = 0.85$ $\omega_l = 0.16$	$c = 0.85$ $\omega_l = 0.064$	$c = 0.85$ $\omega_l = 0.0256$	1.84475E-03	3.00133E-04	1.84161E-03	3.04408E-04	1.42
$n = 32$ $w = 1001, \left[10^{-8}, 10^{1}\right]$	$c = 0.85$ $\omega_l = 0.4$	$c = 0.85$ $\omega_l = 0.16$	$c = 0.85$ $\omega_l = 0.064$	$c = 0.85$ $\omega_l = 0.0256$	1.84475E-03	3.00133E-04	1.84161E-03	2.99912E-04	0.07
$n = 32$ $w = 501, \left[10^{-8}, 10^{1}\right]$	$c = 0.5$ $\omega_l = 0.4$	$c = 0.5$ $\omega_l = 0.16$	$c = 0.5$ $\omega_l = 0.064$	$c = 0.5$ $\omega_l = 0.0256$	1.37117E-02	2.74257E-03	1.37041E-02	2.86651E-03	4.52
$n = 32$ $w = 1001, \left[10^{-8}, 10^{1}\right]$	$c = 0.5$ $\omega_l = 0.4$	$c = 0.5$ $\omega_l = 0.16$	$c = 0.5$ $\omega_l = 0.064$	$c = 0.5$ $\omega_l = 0.0256$	1.37117E-02	2.74257E-03	1.37041E-02	2.71389E-03	1.05

Exponential Transform

Table A.12. Exponential transform slab problem with slab thickness of 5 cm and a stretching parameter of 0.5

$n, w\ [w_{lt}, w_{ut}]$	Cell 1	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 32$ $w = 3001, \left[10^{-3}, 10^{3}\right]$	$c = 0$	4.98920E-04	4.07527E-05	4.96457E-04	4.04216E-05	0.81
$n = 32$ $w = 3001, \left[10^{-4}, 10^{4}\right]$	$c = 0$	4.98920E-04	4.07527E-05	4.98371E-04	4.08921E-05	0.34
$n = 32$ $w = 3001, \left[10^{-3}, 10^{3}\right]$	$c = 0.85$	1.37095E-02	6.76340E-03	1.37135E-02	6.82815E-03	0.96

Table A.13. Exponential transform with implicit capture and weight cutoff slab problem with slab thickness of 5 cm, a stretching parameter of 0.5, a cutoff weight of 0.25, and a survival weight of 0.5

$n, w\ [w_{lt}, w_{ut}]$	Cell 1	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 32$ $w = 4001, \left[10^{-4}, 10^{4}\right]$	$c = 0.85$	1.37113E-02	3.23602E-03	1.36478E-02	3.22238E-03	0.42

Table A.14. Exponential transform, importance Splitting/Rouletting, implicit capture, and weight cutoff in 1-D slab problem with four equally sized cells for slab thickness of 5 cm, stretching parameter of 0.5, cutoff weight of 0.25, and survival weight 0.5

$n, w\ [w_{lt}, w_{ut}]$	Cell 1	Cell 2	Cell 3	Cell 4	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 32$ $w = 5001, \left[10^{-5}, 10^{5}\right]$	$c = 0.85$ $I = 1$	$c = 0.85$ $I = 5$	$c = 0.85$ $I = 50$	$c = 0.85$ $I = 2500$	1.37057E-02	7.30012E-04	1.36513E-02	7.24719E-04	0.73

Table A.15. Exponential transform, surface and collision weight windows, and implicit capture slab problem with four equally sized cells for slab thickness of 5 cm, stretching parameter of 0.5, upper window multiplier of 5, survival multiplier of 3, and maximum splitting/rouletting parameter of 5

n, w $[w_{lt}, w_{ut}]$	Cell 1	Cell 2	Cell 3	Cell 4	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 32$ $w = 5001, \left[10^{-5}, 10^{5}\right]$	$c = 0.85$ $\omega_l = 0.5$	$c = 0.85$ $\omega_l = 0.05$	$c = 0.85$ $\omega_l = 0.005$	$c = 0.85$ $\omega_l = 0.0005$	1.37093E-02	8.91142E-04	1.36657E-02	8.87626E-04	0.40

A.3.2 Surface Flux Tally

Analog

Table A.16. Analog 1-D slab problem with slab thickness of 5 cm

$n, w\ [w_{lt}, w_{ut}]$	Cell 1	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 32$ $w = 501, \left[10^{-8}, 10^{1}\right]$	$c = 0$	5.72008E-04	6.69853E-04	5.73188E-04	6.75346E-04	0.82
$n = 32$ $w = 501, \left[10^{-8}, 10^{1}\right]$	$c = 0.5$	2.59750E-03	5.73572E-03	2.58352E-03	5.58311E-03	2.66
$n = 64$ $w = 1001, \left[10^{-8}, 10^{1}\right]$	$c = 0.5$	2.59750E-03	5.73572E-03	2.58693E-03	5.64266E-03	1.62
$n = 96$ $w = 1001, \left[10^{-8}, 10^{1}\right]$	$c = 0.5$	2.59750E-03	5.73572E-03	2.58765E-03	5.65771E-03	1.36
$n = 32$ $w = 501, \left[10^{-8}, 10^{1}\right]$	$c = 0.85$	2.23978E-02	6.53713E-02	2.23062E-02	6.32958E-02	3.17
$n = 96$ $w = 1001, \left[10^{-8}, 10^{1}\right]$	$c = 0.85$	2.23978E-02	6.53713E-02	2.23452E-02	6.45113E-02	1.32

Importance Splitting/Rouletting

Table A.17. Importance Splitting/Rouletting in 1-D slab problem with four equally sized cells for slab thickness of 5 cm

$n, w\ [w_{lt}, w_{ut}]$	Cell 1	Cell 2	Cell 3	Cell 4	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 32$ $w = 501, \left[10^{-8}, 10^{1}\right]$	$c = 0$ $I = 1$	$c = 0$ $I = 2$	$c = 0$ $I = 4$	$c = 0$ $I = 8$	5.75114E-04	1.17457E-04	5.73188E-04	1.24052E-04	5.61
$n = 32$ $w = 4501, \left[10^{-8}, 10^{1}\right]$	$c = 0$ $I = 1$	$c = 0$ $I = 2$	$c = 0$ $I = 4$	$c = 0$ $I = 8$	5.75114E-04	1.17457E-04	5.73188E-04	1.16773E-04	0.58
$n = 32$ $w = 501, \left[10^{-8}, 10^{1}\right]$	$c = 0.85$ $I = 1$	$c = 0.85$ $I = 2$	$c = 0.85$ $I = 4$	$c = 0.85$ $I = 8$	2.24062E-02	1.42504E-02	2.23062E-02	1.47305E-02	3.37
$n = 32$ $w = 1501, \left[10^{-8}, 10^{1}\right]$	$c = 0.85$ $I = 1$	$c = 0.85$ $I = 2$	$c = 0.85$ $I = 4$	$c = 0.85$ $I = 8$	2.24062E-02	1.42504E-02	2.23062E-02	1.41414E-02	0.76

Implicit Capture and Weight Cutoff

Table A.18. Implicit capture and weight cutoff slab problem with slab thickness of 5 cm, cutoff weight of 0.25, and survival weight of 0.5

$n, w\ [w_{lt}, w_{ut}]$	Cell 1	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 32$ $w = 501, \left[10^{-9}, 10^{1}\right]$	$c = 0.5$	2.60308E-03	3.21507E-03	2.58352E-03	3.13227E-03	2.58
$n = 96$ $w = 1501, \left[10^{-9}, 10^{1}\right]$	$c = 0.5$	2.60308E-03	3.21507E-03	2.58765E-03	3.17271E-03	1.32

n, w $[w_{lt}, w_{ut}]$	Cell 1	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 32$ $w = 501, [10^{-9}, 10^{1}]$	$c = 0.85$	2.23988E-02	2.88868E-02	2.23062E-02	2.85991E-02	1.00
$n = 96$ $w = 1501, [10^{-9}, 10^{1}]$	$c = 0.85$	2.23988E-02	2.88868E-02	2.23452E-02	2.88098E-02	0.27

Table A.19. Importance splitting/rouletting, implicit capture, and weight cutoff slab problem with slab thickness of 5 cm, cutoff weight of 0.25, and survival weight of 0.5

n, w $[w_{lt}, w_{ut}]$	Cell 1	Cell 2	Cell 3	Cell 4	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 32$ $w = 501, [10^{-9}, 10^{1}]$	$c = 0.5$ $I = 1$	$c = 0.5$ $I = 2$	$c = 0.5$ $I = 4$	$c = 0.5$ $I = 8$	2.59344E-03	5.82107E-04	2.58352E-03	5.81949E-04	0.03
$n = 96$ $w = 1501, [10^{-9}, 10^{1}]$	$c = 0.5$ $I = 1$	$c = 0.5$ $I = 2$	$c = 0.5$ $I = 4$	$c = 0.5$ $I = 8$	2.59344E-03	5.82107E-04	2.58352E-03	5.82031E-04	0.01
$n = 32$ $w = 501, [10^{-9}, 10^{1}]$	$c = 0.85$ $I = 1$	$c = 0.85$ $I = 2$	$c = 0.85$ $I = 4$	$c = 0.85$ $I = 8$	2.23792E-02	7.25540E-03	2.23062E-02	7.35207E-03	1.33
$n = 32$ $w = 1501, [10^{-9}, 10^{1}]$	$c = 0.85$ $I = 1$	$c = 0.85$ $I = 2$	$c = 0.85$ $I = 4$	$c = 0.85$ $I = 8$	2.23792E-02	7.25540E-03	2.23062E-02	7.29140E-03	0.50

Weight Windows

Table A.20. Surface and collision weight windows in a 1-D slab problem with four equally sized cells for slab thickness of 5 cm, upper window multiplier of 5, survival multiplier of 3, and a maximum splitting/rouletting parameter of 5

n, w $[w_{lt}, w_{ut}]$	Cell 1	Cell 2	Cell 3	Cell 4	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 32$ $w = 501, [10^{-9}, 10^{1}]$	$c = 0.85$ $\omega_l = 0.5$	$c = 0.85$ $\omega_l = 0.25$	$c = 0.85$ $\omega_l = 0.125$	$c = 0.85$ $\omega_l = 0.0625$	2.23712E-02	2.39186E-02	2.23062E-02	2.35880E-02	1.38
$n = 64$ $w = 501, [10^{-9}, 10^{1}]$	$c = 0.85$ $\omega_l = 0.5$	$c = 0.85$ $\omega_l = 0.25$	$c = 0.85$ $\omega_l = 0.125$	$c = 0.85$ $\omega_l = 0.0625$	2.23712E-02	2.39186E-02	2.23379E-02	2.38551E-02	0.27
$n = 32$ $w = 501, [10^{-9}, 10^{1}]$	$c = 0.85$ $\omega_l = 0.4$	$c = 0.85$ $\omega_l = 0.16$	$c = 0.85$ $\omega_l = 0.064$	$c = 0.85$ $\omega_l = 0.0256$	2.23908E-02	1.35256E-02	2.23062E-02	1.34027E-02	0.91
$n = 64$ $w = 501, [10^{-9}, 10^{1}]$	$c = 0.85$ $\omega_l = 0.4$	$c = 0.85$ $\omega_l = 0.16$	$c = 0.85$ $\omega_l = 0.064$	$c = 0.85$ $\omega_l = 0.0256$	2.23908E-02	1.35256E-02	2.23379E-02	1.35553E-02	0.22

Table A.21. Surface and collision weight windows with implicit capture in a 1-D slab problem with four equally sized cells for slab thickness of 5 cm, upper window multiplier of 5, survival multiplier of 3, and a maximum splitting/rouletting parameter of 5

n, w $[w_{lt}, w_{ut}]$	Cell 1	Cell 2	Cell 3	Cell 4	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 32$ $w = 501, [10^{-9}, 10^{1}]$	$c = 0.85$ $\omega_l = 0.5$	$c = 0.85$ $\omega_l = 0.25$	$c = 0.85$ $\omega_l = 0.125$	$c = 0.85$ $\omega_l = 0.0625$	2.23835E-02	1.52763E-02	2.23062E-02	1.56355E-02	2.35
$n = 32$ $w = 1501, [10^{-9}, 10^{1}]$	$c = 0.85$ $\omega_l = 0.5$	$c = 0.85$ $\omega_l = 0.25$	$c = 0.85$ $\omega_l = 0.125$	$c = 0.85$ $\omega_l = 0.0625$	2.23835E-02	1.52763E-02	2.23062E-02	1.53230E-02	0.31

n, w $[w_{lt}, w_{ut}]$	Cell 1	Cell 2	Cell 3	Cell 4	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 32$ $w = 501, \left[10^{-9}, 10^{1}\right]$	$c = 0.85$ $\omega_l = 0.4$	$c = 0.85$ $\omega_l = 0.16$	$c = 0.85$ $\omega_l = 0.064$	$c = 0.85$ $\omega_l = 0.0256$	2.23893E-02	8.95764E-03	2.23062E-02	9.34977E-03	4.38
$n = 32$ $w = 2001, \left[10^{-9}, 10^{1}\right]$	$c = 0.85$ $\omega_l = 0.4$	$c = 0.85$ $\omega_l = 0.16$	$c = 0.85$ $\omega_l = 0.064$	$c = 0.85$ $\omega_l = 0.0256$	2.23893E-02	8.95764E-03	2.23062E-02	9.01506E-03	0.64

A.3.3 Volume Flux Tally

Analog

Table A.22. Analog 1-D slab problem with slab thickness of 5 cm

$n, w\ [w_{lt}, w_{ut}]$	Cell 1	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 32$ $w = 501, \left[10^{-9}, 10^{1}\right]$	$c = 0$	1.32689E-03	8.16888E-04	1.32053E-03	8.12895E-04	0.49
$n = 32$ $w = 501, \left[10^{-9}, 10^{1}\right]$	$c = 0.5$	6.24843E-03	8.22030E-03	6.22975E-03	8.19295E-03	0.33
$n = 32$ $w = 501, \left[10^{-9}, 10^{1}\right]$	$c = 0.85$	5.24347E-02	1.47982E-01	5.24037E-02	1.47702E-01	0.19

Importance Splitting/Rouletting

Table A.23. Importance Splitting/Rouletting in 1-D slab problem with four equally sized cells for slab thickness of 5 cm

$n, w\ [w_{lt}, w_{ut}]$	Cell 1	Cell 2	Cell 3	Cell 4	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 32$ $w = 501, \left[10^{-8}, 10^{1}\right]$	$c = 0$ $I = 1$	$c = 0$ $I = 2$	$c = 0$ $I = 4$	$c = 0$ $I = 8$	1.32249E-03	2.69843E-04	1.32053E-03	2.72150E-04	0.86
$n = 32$ $w = 501, \left[10^{-8}, 10^{1}\right]$	$c = 0.85$ $I = 1$	$c = 0.85$ $I = 2$	$c = 0.85$ $I = 4$	$c = 0.85$ $I = 8$	5.24289E-02	5.18854E-02	5.24037E-02	5.24549E-02	1.10
$n = 32$ $w = 3001, \left[10^{-8}, 10^{1}\right]$	$c = 0.85$ $I = 1$	$c = 0.85$ $I = 2$	$c = 0.85$ $I = 4$	$c = 0.85$ $I = 8$	5.24289E-02	5.18854E-02	5.24037E-02	5.15297E-02	0.69

Implicit Capture and Weight Cutoff

Table A.24. Implicit capture and weight cutoff slab problem with slab thickness of 5 cm, cutoff weight of 0.25, and survival weight of 0.5

$n, w\ [w_{lt}, w_{ut}]$	Cell 1	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 32$ $w = 501, \left[10^{-9}, 10^{1}\right]$	$c = 0.5$	6.24793E-03	4.93331E-03	6.22975E-03	4.94054E-03	0.15
$n = 32$ $w = 6001, \left[10^{-9}, 10^{1}\right]$	$c = 0.5$	6.24793E-03	4.93331E-03	6.22975E-03	4.90991E-03	0.47
$n = 32$ $w = 501, \left[10^{-9}, 10^{1}\right]$	$c = 0.85$	5.24636E-02	6.80883E-02	5.24037E-02	6.88635E-02	1.14
$n = 32$ $w = 6001, \left[10^{-9}, 10^{1}\right]$	$c = 0.85$	5.24636E-02	6.80883E-02	5.24037E-02	6.75722E-02	0.76

Table A.25. Importance splitting/rouletting, implicit capture, and weight cutoff slab problem with slab thickness of 5 cm, cutoff weight of 0.25, and survival weight of 0.5

n, w $[w_{lt}, w_{ut}]$	Cell 1	Cell 2	Cell 3	Cell 4	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 32$ $w = 501, [10^{-9}, 10^{1}]$	$c = 0.5$ $I = 1$	$c = 0.5$ $I = 2$	$c = 0.5$ $I = 4$	$c = 0.5$ $I = 8$	6.23863E-03	1.62617E-03	6.22975E-03	1.64903E-03	1.41
$n = 96$ $w = 7501, [10^{-9}, 10^{1}]$	$c = 0.5$ $I = 1$	$c = 0.5$ $I = 2$	$c = 0.5$ $I = 4$	$c = 0.5$ $I = 8$	6.23863E-03	1.62617E-03	6.22975E-03	1.61950E-03	0.41
$n = 32$ $w = 501, [10^{-9}, 10^{1}]$	$c = 0.85$ $I = 1$	$c = 0.85$ $I = 2$	$c = 0.85$ $I = 4$	$c = 0.85$ $I = 8$	5.24363E-02	2.84398E-02	5.24037E-02	2.90021E-02	1.98
$n = 32$ $w = 1501, [10^{-9}, 10^{1}]$	$c = 0.85$ $I = 1$	$c = 0.85$ $I = 2$	$c = 0.85$ $I = 4$	$c = 0.85$ $I = 8$	5.24363E-02	2.84398E-02	5.24037E-02	2.83556E-02	0.30

Weight Windows

Table A.26. Surface and collision weight windows in a 1-D slab problem with four equally sized cells for slab thickness of 5 cm, upper window multiplier of 5, survival multiplier of 3, and a maximum splitting/rouletting parameter of 5

n, w $[w_{lt}, w_{ut}]$	Cell 1	Cell 2	Cell 3	Cell 4	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 32$ $w = 501, [10^{-9}, 10^{1}]$	$c = 0.85$ $\omega_l = 0.5$	$c = 0.85$ $\omega_l = 0.25$	$c = 0.85$ $\omega_l = 0.125$	$c = 0.85$ $\omega_l = 0.0625$	5.24069E-02	7.87319E-02	5.24037E-02	7.99182E-02	1.51
$n = 32$ $w = 6001, [10^{-9}, 10^{1}]$	$c = 0.85$ $\omega_l = 0.5$	$c = 0.85$ $\omega_l = 0.25$	$c = 0.85$ $\omega_l = 0.125$	$c = 0.85$ $\omega_l = 0.0625$	5.24069E-02	7.87319E-02	5.24069E-02	7.93027E-02	0.72
$n = 32$ $w = 501, [10^{-9}, 10^{1}]$	$c = 0.85$ $\omega_l = 0.4$	$c = 0.85$ $\omega_l = 0.16$	$c = 0.85$ $\omega_l = 0.064$	$c = 0.85$ $\omega_l = 0.0256$	5.24379E-02	4.79950E-02	5.24379E-02	4.85364E-02	1.13
$n = 32$ $w = 6001, [10^{-9}, 10^{1}]$	$c = 0.85$ $\omega_l = 0.4$	$c = 0.85$ $\omega_l = 0.16$	$c = 0.85$ $\omega_l = 0.064$	$c = 0.85$ $\omega_l = 0.0256$	5.24379E-02	4.79950E-02	5.24379E-02	4.80725E-02	0.16

Table A.27. Surface and collision weight windows with implicit capture in a 1-D slab problem with four equally sized cells for slab thickness of 5 cm, upper window multiplier of 5, survival multiplier of 3, and a maximum splitting/rouletting parameter of 5

n, w $[w_{lt}, w_{ut}]$	Cell 1	Cell 2	Cell 3	Cell 4	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 32$ $w = 501, [10^{-9}, 10^{1}]$	$c = 0.85$ $\omega_l = 0.5$	$c = 0.85$ $\omega_l = 0.25$	$c = 0.85$ $\omega_l = 0.125$	$c = 0.85$ $\omega_l = 0.0625$	5.24430E-02	5.17237E-02	5.24037E-02	5.28183E-02	2.12
$n = 32$ $w = 6001, [10^{-9}, 10^{1}]$	$c = 0.85$ $\omega_l = 0.5$	$c = 0.85$ $\omega_l = 0.25$	$c = 0.85$ $\omega_l = 0.125$	$c = 0.85$ $\omega_l = 0.0625$	5.24430E-02	5.17237E-02	5.24037E-02	5.15423E-02	0.35
$n = 32$ $w = 501, [10^{-9}, 10^{1}]$	$c = 0.85$ $\omega_l = 0.4$	$c = 0.85$ $\omega_l = 0.16$	$c = 0.85$ $\omega_l = 0.064$	$c = 0.85$ $\omega_l = 0.0256$	5.24526E-02	3.51895E-02	5.24037E-02	3.62229E-02	2.94
$n = 32$ $w = 6001, [10^{-9}, 10^{1}]$	$c = 0.85$ $\omega_l = 0.4$	$c = 0.85$ $\omega_l = 0.16$	$c = 0.85$ $\omega_l = 0.064$	$c = 0.85$ $\omega_l = 0.0256$	5.24526E-02	3.51895E-02	5.24037E-02	3.52318E-02	0.12

A.4 2-D Verification of the S_n and MCNP Calculations

Two-dimensional verification calculations were also performed for the geometry presented in Fig. A.2. The capabilities of the S_n code were limited, and ray effects and available computer

memory limited the spatial size of the verification problems. For these two-dimensional problems, the total cross section is 1 cm^{-1} with a scattering ratio of $c = 0.85$.

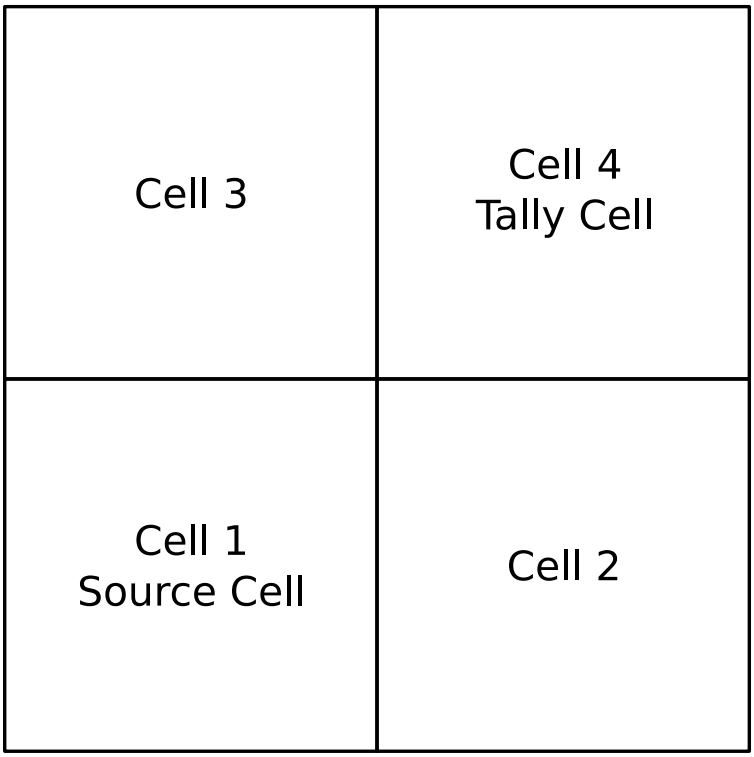


Figure A.2. 2-D verification problem geometry. The source is isotropic in cell 1, and the tally is a volume flux tally in cell 4.

Table A.28. Importance splitting/rouletting, implicit capture, and weight cutoff slab problem with slab thickness of 5 cm, cutoff weight of 0.25, and survival weight of 0.5

n, w $[w_{lt}, w_{ut}]$	Cell 1	Cell 2	Cell 3	Cell 4	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 8$ $w = 501, \left[10^{-4}, 10^{2}\right]$	$c = 0.85$ $I = 1$	$c = 0.85$ $I = 2$	$c = 0.85$ $I = 2$	$c = 0.85$ $I = 3$	1.26653E-01	1.00829E-01	1.30531E-01	1.00537E-01	0.29

Table A.29. Surface and collision weight windows with implicit capture using an upper window multiplier of 5, survival multiplier of 3, and a maximum splitting/rouletting parameter of 5

n, w $[w_{lt}, w_{ut}]$	Cell 1	Cell 2	Cell 3	Cell 4	MCNP M_1	MCNP σ^2	S_n M_1	S_n σ^2	σ^2 %error
$n = 16$ $w = 501, \left[10^{-8}, 10^{1}\right]$	$c = 0.85$ $\omega_l = 0.035$	$c = 0.85$ $\omega_l = 0.0175$	$c = 0.85$ $\omega_l = 0.0175$	$c = 0.85$ $\omega_l = 0.00875$	1.26688E-01	4.21023E-02	1.27767E-01	4.21888E-02	0.21
$n = 16$ $w = 1001, \left[10^{-8}, 10^{1}\right]$	$c = 0.85$ $\omega_l = 0.035$	$c = 0.85$ $\omega_l = 0.0175$	$c = 0.85$ $\omega_l = 0.0175$	$c = 0.85$ $\omega_l = 0.00875$	1.26688E-01	4.21023E-02	1.27767E-01	4.26931E-02	1.40
$n = 16$ $w = 501, \left[10^{-9}, 10^{1}\right]$	$c = 0.85$ $\omega_l = 0.5$	$c = 0.85$ $\omega_l = 0.05$	$c = 0.85$ $\omega_l = 0.05$	$c = 0.85$ $\omega_l = 0.005$	1.26670E-01	7.79515E-02	1.27767E-01	8.07974E-02	3.65
$n = 16$ $w = 1501, \left[10^{-9}, 10^{1}\right]$	$c = 0.85$ $\omega_l = 0.5$	$c = 0.85$ $\omega_l = 0.05$	$c = 0.85$ $\omega_l = 0.05$	$c = 0.85$ $\omega_l = 0.005$	1.26670E-01	7.79515E-02	1.27767E-01	7.91639E-02	1.56

Appendix B

Proof that the Integro-Differential Streaming Operator is Inverse to the Transition Kernel Operator

This appendix summarizes a derivation contained in Dunn and Shultis [2008] that shows the relationship between the integro-differential form of the transport equation and the integral form. It is augmented slightly by details from a similar derivation by Bell and Glasstone [1970]. The goal is to show the inverse relationship between the integral transition operator $\int d\mathbf{P}\, T(\mathbf{P}', \mathbf{P})$ and the differential operator $(1/v)(\partial\phi/\partial t) + \mathbf{\Omega}\cdot\phi(\mathbf{r}, \mathbf{\Omega}, E, t)$, subject to the boundary condition that nothing enters the domain.

The integro-differential form of the linearized Boltzmann transport equation is

$$\frac{1}{v}\frac{\partial\phi}{\partial t} + \mathbf{\Omega}\cdot\nabla\phi(\mathbf{r}, \mathbf{\Omega}, E, t) + \Sigma(\mathbf{r}, E)\phi(\mathbf{r}, \mathbf{\Omega}, E, t) = \int_\Omega d\mathbf{\Omega}' \int_0^\infty dE'\, \Sigma_s(\mathbf{r}, \mathbf{\Omega}', E' \to \mathbf{\Omega}, E)\phi(\mathbf{r}, \mathbf{\Omega}', E', t) + q(\mathbf{r}, \mathbf{\Omega}, E, t), \tag{B.1}$$

where $\mathbf{r}$ is the particle's position, $\mathbf{\Omega}$ is the particle's direction, E is the particle's energy, t is time, Σ is the total macroscopic cross section, $\Sigma_s(\mathbf{\Omega}', E' \to \mathbf{\Omega}, E)d\mathbf{\Omega}dE$ is the probability that a particle scatters from $d\mathbf{\Omega}'$ about $\mathbf{\Omega}'$ and dE' about E' into $\mathbf{\Omega}$ and E, and q is an external source. Defining the *emission rate density* χ as

$$\chi(\mathbf{r}, \mathbf{\Omega}, E, t) = \int_\Omega d\mathbf{\Omega}' \int_0^\infty dE'\, \Sigma_s(\mathbf{r}, \mathbf{\Omega}', E' \to \mathbf{\Omega}, E)\phi(\mathbf{r}, \mathbf{\Omega}', E', t) + q(\mathbf{r}, \mathbf{\Omega}, E, t), \tag{B.2}$$

Eq. (B.1) may be rewritten as

$$\frac{1}{v}\frac{\partial\phi}{\partial t} + \mathbf{\Omega}\cdot\nabla\phi(\mathbf{r}, \mathbf{\Omega}, E, t) + \Sigma(\mathbf{r}, E)\phi(\mathbf{r}, \mathbf{\Omega}, E, t) = \chi(\mathbf{r}, \mathbf{\Omega}, E, t). \tag{B.3}$$

Now, considering the total derivative of ϕ with respect to distance s directed in $\mathbf{\Omega}$, one has

$$\frac{d\phi}{ds} = \frac{\partial\phi}{\partial t}\frac{dt}{ds} + \frac{\partial\phi}{\partial x}\frac{dx}{ds} + \frac{\partial\phi}{\partial y}\frac{dy}{ds} + \frac{\partial\phi}{\partial z}\frac{dz}{ds}. \tag{B.4}$$

Consider a particle transitioning from $\mathbf{r}' = (x', y', z')$ to $\mathbf{r} = (x, y, z)$ as illustrated in Fig. B.1. Here, it is clear that

$$t = t' + \frac{s}{v} \tag{B.5}$$
$$x = x' + \Omega_x s \tag{B.6}$$
$$y = y' + \Omega_y s \tag{B.7}$$
$$z = z' + \Omega_z s, \tag{B.8}$$

which indicates that

$$\frac{dt}{ds} = \frac{1}{v} \tag{B.9}$$
$$\frac{dx}{ds} = \Omega_x \tag{B.10}$$
$$\frac{dy}{ds} = \Omega_y \tag{B.11}$$
$$\frac{dz}{ds} = \Omega_z. \tag{B.12}$$

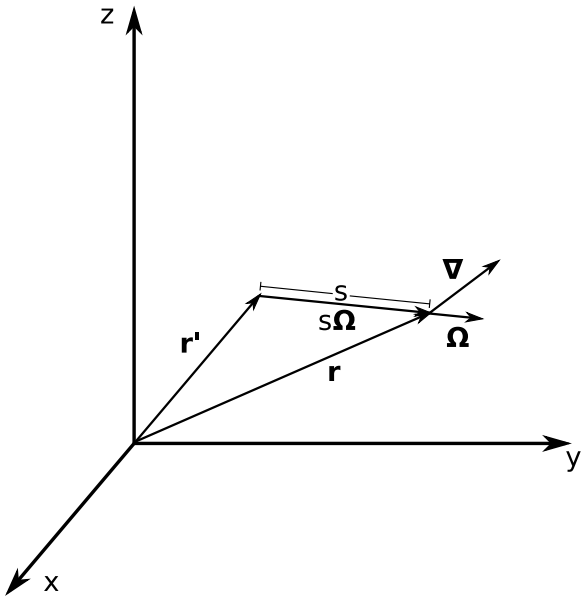


Figure B.1. A particle at energy E transitioning along direction $\mathbf{\Omega}$ from $\mathbf{r}'$ at time t' to $\mathbf{r}$ at time t

Substituting the expressions for the derivatives above into Eq. (B.4), one finds that

$$\begin{aligned} \frac{d\phi}{ds} &= \frac{\partial\phi}{\partial t}\frac{1}{v} + \frac{\partial\phi}{\partial x}\Omega_x + \frac{\partial\phi}{\partial y}\Omega_y + \frac{\partial\phi}{\partial z}\Omega_z \\ &= \frac{1}{v}\frac{\partial\phi}{\partial t} + \mathbf{\Omega}\cdot\nabla\phi, \end{aligned} \tag{B.13}$$

and Eq. (B.3) may be rewritten as

$$\frac{d\phi}{ds} + \Sigma(\mathbf{r}, E)\phi(\mathbf{r}, \mathbf{\Omega}, E, t) = \chi(\mathbf{r}, \mathbf{\Omega}, E, t). \tag{B.14}$$

Now, allowing $\mathbf{r} = \mathbf{r}' + s\mathbf{\Omega}$ and $t = t' + s/v$ one can re-express the above as

$$\frac{d\phi}{ds} + \Sigma(\mathbf{r}' + s\mathbf{\Omega}, E)\phi\left(\mathbf{r}' + s\mathbf{\Omega}, \mathbf{\Omega}, E, t' + \frac{s}{v}\right) = \chi\left(\mathbf{r}' + s\mathbf{\Omega}, \mathbf{\Omega}, E, t' + \frac{s}{v}\right). \tag{B.15}$$

Multiplying by the integrating factor

$$\exp\left[\int_0^s ds''\, \Sigma(\mathbf{r}' + s''\mathbf{\Omega}, E)\right],$$

having the property that

$$\frac{d}{ds}\left\{\exp\left[\int_0^s ds''\, \Sigma(\mathbf{r}' + s''\mathbf{\Omega}, E)\right]\right\} = \Sigma(\mathbf{r}' + s\Omega, E)\exp\left[\int_0^s ds''\, \Sigma(\mathbf{r}' + s''\mathbf{\Omega}, E)\right],$$

one obtains

$$\begin{aligned}
&\frac{d\phi}{ds}\exp\left[\int_0^s ds''\, \Sigma(\mathbf{r}' + s''\mathbf{\Omega}, E)\right] \\
&\qquad + \Sigma(\mathbf{r}' + s\mathbf{\Omega}, E)\phi\left(\mathbf{r}' + s\mathbf{\Omega}, \mathbf{\Omega}, E, t' + \frac{s}{v}\right)\exp\left[\int_0^s ds''\, \Sigma(\mathbf{r}' + s''\mathbf{\Omega}, E)\right] \\
&\qquad = \chi\left(\mathbf{r}' + s\mathbf{\Omega}, \mathbf{\Omega}, E, t' + \frac{s'}{v}\right)\exp\left[\int_0^s ds''\, \Sigma(\mathbf{r}' + s''\mathbf{\Omega}, E)\right].
\end{aligned} \tag{B.16}$$

Thus,

$$\begin{aligned}
&\frac{d}{ds}\left\{\phi\left(\mathbf{r}' + s\Omega, \mathbf{\Omega}, E, t' + \frac{s}{v}\right)\exp\left[\int_0^s ds''\, \Sigma(\mathbf{r}' + s''\mathbf{\Omega}, E)\right]\right\} \\
&\qquad = \chi\left(\mathbf{r}' + s\mathbf{\Omega}, \mathbf{\Omega}, E, t' + \frac{s'}{v}\right)\exp\left[\int_0^s ds''\, \Sigma(\mathbf{r}' + s''\mathbf{\Omega}, E)\right].
\end{aligned} \tag{B.17}$$

Now, integrating the above expression from $-\infty$ to s, one has

$$\begin{aligned}
&\int_{-\infty}^s ds' \frac{d}{ds'}\left\{\phi\left(\mathbf{r}' + s'\Omega, \mathbf{\Omega}, E, t' + \frac{s}{v}\right)\exp\left[\int_0^{s'} ds''\, \Sigma(\mathbf{r}' + s''\mathbf{\Omega}, E)\right]\right\} \\
&\qquad = \int_{-\infty}^s ds'\chi\left(\mathbf{r}' + s'\mathbf{\Omega}, \mathbf{\Omega}, E, t' + \frac{s'}{v}\right)\exp\left[\int_0^{s'} ds''\, \Sigma(\mathbf{r}' + s''\mathbf{\Omega}, E)\right],
\end{aligned} \tag{B.18}$$

which, upon integrating the left side, gives

$$\begin{aligned}
&\left\{\phi\left(\mathbf{r}' + s\Omega, \mathbf{\Omega}, E, t' + \frac{s}{v}\right)\exp\left[\int_0^s ds''\, \Sigma(\mathbf{r}' + s''\mathbf{\Omega}, E)\right]\right. \\
&\qquad \left. - \phi\left(\mathbf{r}' + (-\infty)\Omega, \mathbf{\Omega}, E, t' + \frac{s}{v}\right)\exp\left[\int_0^{-\infty} ds''\, \Sigma(\mathbf{r}' + s''\mathbf{\Omega}, E)\right]\right\} \\
&\qquad = \int_{-\infty}^s ds'\chi\left(\mathbf{r}' + s'\mathbf{\Omega}, \mathbf{\Omega}, E, t' + \frac{s'}{v}\right)\exp\left[\int_0^{s'} ds''\, \Sigma(\mathbf{r}' + s''\mathbf{\Omega}, E)\right].
\end{aligned} \tag{B.19}$$

The exponential in the second term causes the entire term to vanish leaving only

$$\phi\left(\mathbf{r}' + s\Omega, \mathbf{\Omega}, E, t' + \frac{s}{v}\right)\exp\left[\int_0^s ds''\,\Sigma(\mathbf{r}' + s''\mathbf{\Omega}, E)\right]$$
$$= \int_{-\infty}^{s} ds'\chi\left(\mathbf{r}' + s'\mathbf{\Omega}, \mathbf{\Omega}, E, t' + \frac{s'}{v}\right)\exp\left[\int_0^{s'} ds''\,\Sigma(\mathbf{r}' + s''\mathbf{\Omega}, E)\right]. \quad \text{(B.20)}$$

Multiplying both sides by $\exp\left[-\int_0^s ds''\,\Sigma(\mathbf{r}' + s''\mathbf{\Omega}, E)\right]$ and realizing s is a constant with respect to the integral over ds' one obtains the *integral form* of the transport equation

$$\phi\left(\mathbf{r}' + s\Omega, \mathbf{\Omega}, E, t' + \frac{s}{v}\right)$$
$$= \int_{-\infty}^{s} ds'\chi\left(\mathbf{r}' + s'\mathbf{\Omega}, \mathbf{\Omega}, E, t' + \frac{s'}{v}\right)\exp\left[-\int_{s'}^{s} ds''\,\Sigma(\mathbf{r}' + s''\mathbf{\Omega}, E)\right]. \quad \text{(B.21)}$$

Making the substitutions $\mathbf{r} = \mathbf{r}' + s\mathbf{\Omega}$ and $t = t' + s/v$ one obtains

$$\phi(\mathbf{r}, \mathbf{\Omega}, E, t)$$
$$= \int_{-\infty}^{s} ds'\chi\left(\mathbf{r} - (s - s')\mathbf{\Omega}, \mathbf{\Omega}, E, t - \frac{s - s'}{v}\right)$$
$$\times\exp\left[-\int_{s'}^{s} ds''\,\Sigma(\mathbf{r} - (s - s'')\mathbf{\Omega}, E)\right]. \quad \text{(B.22)}$$

Now, defining $a = s - s'$ and $b = s - s''$ such that $da = -ds'$ and $db = -ds''$ one gets

$$\phi(\mathbf{r}, \mathbf{\Omega}, E, t)$$
$$= \int_0^{\infty} da\chi\left(\mathbf{r} - a\mathbf{\Omega}, \mathbf{\Omega}, E, t - \frac{a}{v}\right)\exp\left[-\int_0^a db\,\Sigma(\mathbf{r} - b\mathbf{\Omega}, E)\right], \quad \text{(B.23)}$$

and, allowing the dummy variables of integration a and b go to s' and s'' respectively, one obtains

$$\phi(\mathbf{r}, \mathbf{\Omega}, E, t)$$
$$= \int_0^{\infty} ds'\chi\left(\mathbf{r} - s'\mathbf{\Omega}, \mathbf{\Omega}, E, t - \frac{s'}{v}\right)\exp\left[-\int_0^{s'} ds''\,\Sigma(\mathbf{r} - s''\mathbf{\Omega}, E)\right]. \quad \text{(B.24)}$$

As one final step, the integral over ds' can be converted to a volume integral over $d\mathbf{r}'$ by realizing that $d\mathbf{r}' = -s'^2 ds' d\mathbf{\Omega}' = R^2 dR d\mathbf{\Omega}'$, where $R = -s'$ and $s'^2 = R^2 = |\mathbf{r} - \mathbf{r}'|^2$. Defining

$$\tau(\mathbf{r}', \mathbf{r}) = \int_0^{|\mathbf{r}-\mathbf{r}'|} ds''\,\Sigma(\mathbf{r} - s''\mathbf{\Omega}, E), \quad \text{(B.25)}$$

Eq. (B.24) becomes

$$\begin{aligned}\phi(\mathbf{r}, \mathbf{\Omega}, E, t) &= \int_0^t dt' \int_0^{\infty} R^2 dR \int_{\mathbf{\Omega}'} d\mathbf{\Omega}'\chi(\mathbf{r}', \mathbf{\Omega}', E, t')\frac{\exp[-\tau(\mathbf{r}, \mathbf{r}')]}{R^2} \\ &\quad \times\delta(\mathbf{\Omega}' - \mathbf{\Omega})\delta\left[t' - \left(t - \frac{|\mathbf{r} - \mathbf{r}'|}{v}\right)\right] \\ &= \int_0^t dt' \int d\mathbf{r}'\chi(\mathbf{r}', \mathbf{\Omega}', E, t')\frac{\exp[-\tau(\mathbf{r}', \mathbf{r})]}{|\mathbf{r} - \mathbf{r}'|^2}\delta(\mathbf{\Omega}' - \mathbf{\Omega}) \\ &\quad \times \delta\left[t' - \left(t - \frac{|\mathbf{r} - \mathbf{r}'|}{v}\right)\right]. \quad \text{(B.26)}\end{aligned}$$

The above expression is commonly written as

$$\phi(\mathbf{P}) = \int d\mathbf{P}' T(\mathbf{P}', \mathbf{P}) \phi(\mathbf{P}'), \tag{B.27}$$

where $\mathbf{P} = (\mathbf{r}, \mathbf{\Omega}, E, t)$, $\mathbf{P}' = (\mathbf{r}', \mathbf{\Omega}', E', t')$, and

$$T(\mathbf{P}', \mathbf{P}) = \frac{\exp[-\tau(\mathbf{r}', \mathbf{r})]}{|\mathbf{r} - \mathbf{r}'|^2} \delta(E' - E) \delta(\mathbf{\Omega}' - \mathbf{\Omega}) \delta\left[t' - \left(t - \frac{|\mathbf{r} - \mathbf{r}'|}{v}\right)\right].$$

It is clear from the derivation above that Eq. (B.27) is equivalent to Eq. (B.21), and, by reversing the procedure above, one could obtain Eq. (B.21) from Eq. (B.27).

The integro-differential form can be recovered by operating on both sides of Eq. (B.21) with $(1/v)\partial/\partial t + \mathbf{\Omega}\cdot\nabla + \Sigma = d/ds + \Sigma$, namely

$$\begin{aligned}
&\frac{1}{v}\frac{\partial \phi}{\partial t} + \mathbf{\Omega}\cdot\nabla\phi + \Sigma(\mathbf{r}' + s\mathbf{\Omega}, E)\phi\left(\mathbf{r}' + s\mathbf{\Omega}, \mathbf{\Omega}, E, t + \frac{s}{v}\right) \\
&\qquad = \left(\frac{d}{ds} + \Sigma\right) \int_{-\infty}^{s} ds' \chi\left(\mathbf{r}' + s'\mathbf{\Omega}, \mathbf{\Omega}, E, t' + \frac{s'}{v}\right) \\
&\qquad\quad \times \quad \exp\left[-\int_{s'}^{s} ds'' \, \Sigma(\mathbf{r}' + s''\mathbf{\Omega}, E)\right].
\end{aligned} \tag{B.28}$$

Application of Leibniz's rule gives

$$\begin{aligned}
&\frac{1}{v}\frac{\partial \phi}{\partial t} + \mathbf{\Omega}\cdot\nabla\phi + \Sigma(\mathbf{r}' + s\mathbf{\Omega}, E)\phi\left(\mathbf{r}' + s\mathbf{\Omega}, \mathbf{\Omega}, E, t' + \frac{s}{v}\right) \\
&\qquad = \int_{-\infty}^{s} ds' \frac{\partial}{\partial s}\left\{\chi\left(\mathbf{r}' + s'\mathbf{\Omega}, \mathbf{\Omega}, E, t' + \frac{s'}{v}\right)\right. \\
&\qquad\qquad \left. \times \exp\left[-\int_{s'}^{s} ds'' \, \Sigma(\mathbf{r}' + s''\mathbf{\Omega}, E)\right]\right\} \\
&\qquad + \chi\left(\mathbf{r}' + s\mathbf{\Omega}, \mathbf{\Omega}, E, t' + \frac{s}{v}\right) \exp\left[-\int_{s}^{s} ds'' \, \Sigma(\mathbf{r}' + s''\mathbf{\Omega}, E)\right] \frac{d}{ds}(s) \\
&\qquad + \Sigma(\mathbf{r}' + s\mathbf{\Omega}, E) \int_{-\infty}^{s} ds' \chi\left(\mathbf{r}' + s'\mathbf{\Omega}, \mathbf{\Omega}, E, t' + \frac{s'}{v}\right) \\
&\qquad\qquad \times \exp\left[-\int_{s'}^{s} ds'' \, \Sigma(\mathbf{r}' + s''\mathbf{\Omega}, E)\right],
\end{aligned} \tag{B.29}$$

and simplification yields

$$\begin{aligned}
&\frac{1}{v}\frac{\partial \phi}{\partial t} + \mathbf{\Omega}\cdot\nabla\phi + \Sigma(\mathbf{r}' + s\mathbf{\Omega}, E)\phi\left(\mathbf{r}' + s\mathbf{\Omega}, \mathbf{\Omega}, E, t' + \frac{s}{v}\right) \\
&\qquad = -\Sigma(\mathbf{r}' + s\mathbf{\Omega}, E) \int_{-\infty}^{s} ds' \chi\left(\mathbf{r}' + s'\mathbf{\Omega}, \mathbf{\Omega}, E, t' + \frac{s'}{v}\right) \\
&\qquad\qquad \times \exp\left[-\int_{s'}^{s} ds'' \, \Sigma(\mathbf{r}' + s''\mathbf{\Omega}, E)\right] \\
&\qquad + \chi\left(\mathbf{r}' + s\mathbf{\Omega}, \mathbf{\Omega}, E, t' + \frac{s}{v}\right)
\end{aligned}$$

$$+\Sigma(\mathbf{r}' + s\mathbf{\Omega}, E) \int_{-\infty}^{s} ds' \chi\left(\mathbf{r}' + s'\mathbf{\Omega}, \mathbf{\Omega}, E, t' + \frac{s'}{v}\right)$$
$$\times \exp\left[-\int_{s'}^{s} ds'' \, \Sigma(\mathbf{r}' + s''\mathbf{\Omega}, E)\right], \tag{B.30}$$

or simply

$$\frac{1}{v}\frac{\partial \phi}{\partial t} + \mathbf{\Omega}\cdot\nabla\phi + \Sigma(\mathbf{r}' + s\mathbf{\Omega}, E)\phi\left(\mathbf{r}' + s\mathbf{\Omega}, \mathbf{\Omega}, E, t' + \frac{s}{v}\right) = \chi\left(\mathbf{r}' + s\mathbf{\Omega}, \mathbf{\Omega}, E, t' + \frac{s}{v}\right). \tag{B.31}$$

Thus, one may conclude that, for an arbitrary function of $F(\mathbf{P})$,

$$\left(\frac{1}{v}\frac{\partial \phi}{\partial t} + \mathbf{\Omega}\cdot\nabla + \Sigma\right) \int d\mathbf{P}' T(\mathbf{P}', \mathbf{P}) F(\mathbf{P}') = F(\mathbf{P}). \tag{B.32}$$