

# Chapter 2

## Literature Review

The following literature review aims to contextualize the work described in this dissertation within the realm of hybrid methods for deep-penetration neutron transport. In doing so, the pertinent theoretical information that is relevant to this topic is described. This description is supplemented by a discussion of the various efforts to implement these methods for applied problems, and the degree to which those efforts succeeded. First, a brief overview of variance reduction for Monte Carlo radiation transport is described in Section 2.1. Then, Section 2.1.3 expands on the various efforts to automate variance reduction techniques in Monte Carlo. Section 2.2.1 follows with an introduction of the concept of importance and how that relates to variance reduction. This section also focuses specifically on how the adjoint solution of the neutron transport equation relates to importance.

From this point, the chapter transitions from theory into existing implementations of variance reduction techniques used in modern software in the nuclear engineering community. Beginning in Section 2.3.1, a description of the consistent, adjoint-driven importance sampling method, or CADIS, which has been optimized for variance reduction of local solutions is presented. Next, Section 2.4 discusses the methods implemented to reduce the variance for global solutions. This discussion includes a description of the forward-weighted CADIS (FW-CADIS) method. The last section, 2.5, details the efforts to incorporate angular information into variance reduction methods for Monte Carlo. Sections 2.3.1-2.5 are each concluded with a description of the various software in which these methods have been implemented and the degree to which they improved the variance reduction for their target applications.

### 2.1 Monte Carlo Variance Reduction

Monte Carlo methods for radiation transport are used in the nuclear engineering community for a wide spectrum of application problems. Monte Carlo methods aim to emulate the transport of a particle from birth, through physical interaction, to death by randomly sampling the probabilities of physics that the particle could encounter, e.g. particle produc-

tion, elastic and inelastic scattering, absorption, and so forth. This process of transporting a single particle is repeated many times, to simulate the transport of many particles throughout the problem. When the user achieves a sufficient number of samples—or particles—to reach the desired statistical precision for the region of interest, the simulation is complete. However, this naive approach to simulating each particle—disregarding whether it is likely to contribute to the tallied result—can be extraordinarily computationally inefficient depending on the problem. A code could waste time simulating millions of “unusable” particles and still not reach the desired statistical precision for the tally. Variance reduction techniques were developed to address this issue. In general, these techniques bias the Monte Carlo transport to more effectively contribute to a particular result, while not fundamentally changing the nature of the problem being solved.

### 2.1.1 Statistical Background

Variance reduction techniques are rooted in statistics, so we begin our discussion of variance reduction techniques with a brief primer on the statistical background relevant to Monte Carlo radiation transport. Sections 2.1.1.1 through 2.1.1.3 are summarized from [7] and [8]. Monte Carlo methods transport many randomly sampled particles, and when those particles reach a region of interest, they are scored in a tally. The statistical precision of the tally will reflect the total number of particles that were sampled in a chosen region or at a chosen surface. The reliability of the answer obtained in this region is then dependent on the quantity and the history of the particles sampled.

#### 2.1.1.1 Population Statistics

In radiation transport, one desires to estimate some response in phase-space. This response is the average behavior of the physical interactions in some differential phase-space in energy, space, and time. If the probability density function,  $f(x)$ , for the response is known exactly, then the response in  $dx$  can be calculated exactly by the true mean, or

$$\bar{x} = \int_{-\infty}^{\infty} x f(x) dx. \quad (2.1)$$

Rarely is  $f(x)$  known exactly, so instead it is sampled. Using  $N$  randomly sampled particles, the estimate of the true mean value is given as

$$\hat{x} = \frac{\sum_{i=1}^N x_i}{N}, \quad (2.2)$$

where  $x_i$  is the  $i^{th}$  event.  $\hat{x}$  is the sample mean, or the estimated value of  $\bar{x}$  based on the  $N$  number of samples that were used to calculate  $\hat{x}$ . As  $N \rightarrow \infty$ ,  $\hat{x}$  will  $\rightarrow \bar{x}$ , which is given by the Strong Law of Large Numbers [8].  $\hat{x}$  in itself is a useful measure, but determining the

spread of values about  $\hat{x}$  is also an important measure. This is called the variance. The true variance of the distribution is

$$\sigma^2(x) = \bar{x}^2 - \bar{x}^2, \quad (2.3)$$

and the standard deviation is the square root of the variance

$$\sigma(x) = (\bar{x}^2 - \bar{x}^2)^{1/2}. \quad (2.4)$$

The variance of the sampled distribution differs, as a finite number of samples are used to calculate  $\bar{x}$  and  $\sigma$ . The sample variance is defined by:

$$S^2 = \sum_{i=1}^N \frac{(x_i - \hat{x})^2}{N-1} \cong \hat{x}^2 - \hat{x}^2, \quad (2.5)$$

where

$$\hat{x}^2 = \frac{1}{N} \sum_{i=1}^N x_i^2, \quad (2.6)$$

and the sample standard deviation is given by

$$S = (\hat{x}^2 - \hat{x}^2)^{(1/2)}. \quad (2.7)$$

For (2.5) to hold true, the number of  $N$  samples must be large.  $S^2$  is the sample estimate of the true variance,  $\sigma^2$ . The variance of the estimate of the mean value about  $\bar{x}$  is:

$$S_{\hat{x}}^2 = \frac{S^2}{N}. \quad (2.8)$$

From (2.8), one can see the relationship between the sample standard deviation and the standard error of  $\hat{x}$  about  $\bar{x}$  is

$$S_{\hat{x}} = \sqrt{\frac{S^2}{N}} = \frac{S}{\sqrt{N}}. \quad (2.9)$$

$S_{\hat{x}}$  is the standard error of the estimate of the sample mean. The relative error normalizes the standard error by the estimate of the mean

$$R = \frac{S_{\hat{x}}}{\hat{x}}. \quad (2.10)$$

As a result,  $S$ ,  $R$ , and  $N$  follow the relationship

$$S^2 \propto R^2 \propto \frac{1}{N}. \quad (2.11)$$

### 2.1.1.2 The Central Limit Theorem

Suppose  $\hat{x}$  is calculated from several independent random particles to estimate  $\bar{x}$ . At what point does one conclude that  $\hat{x}$  sufficiently reflects  $\bar{x}$ ? The central limit theorem (CLT) [7, 8] is a very powerful supplement to the quantities described in Section 2.1.1.1. The CLT states that for large  $N$ ,  $\hat{x}$  will have a limiting distribution  $f_N(\hat{x})$ , and that distribution will be a normal distribution

$$f_N(\hat{x}) \approx \frac{1}{\sqrt{2\pi}\sigma(\hat{x})} \exp\left[\frac{-(\hat{x} - \bar{x})^2}{2\sigma^2(\hat{x})}\right], \quad N \rightarrow \infty. \quad (2.12)$$

The standard deviation of  $\hat{x}$  can be related to the standard deviation of the samples by

$$\sigma(\hat{x}) = \frac{\sigma(x)}{\sqrt{N}}. \quad (2.13)$$

Using the definition from Eq. (2.13) in Eq. (2.12) results in

$$f_N(\hat{x}) \approx \sqrt{\frac{N}{2 * \pi \sigma(x)}} \frac{1}{\sigma(x)} \exp\left[\frac{-N(\hat{x} - \bar{x})^2}{2\sigma^2(x)}\right], \quad N \rightarrow \infty. \quad (2.14)$$

This allows us to use known values for  $\hat{x}$  and an approximation of  $\sigma(x)$ —using  $S$ —to determine the probability density function of the sample means  $f_N(\hat{x})$ . Because  $f_N(\hat{x})$  is normally distributed, we can find the probability that  $\hat{x}$  lies in  $\bar{x} \pm \epsilon$  with

$$P\{\bar{x} - \epsilon < \hat{x} \leq \bar{x} + \epsilon\} = \int_{\bar{x}-\epsilon}^{\bar{x}+\epsilon} f_N(\hat{x}) d\hat{x}. \quad (2.15)$$

Placing our definition for the distribution of  $\hat{x}$ , which is  $f_N(\hat{x})$ , into Eq. (2.15), changing the limits of integration, and changing the variables such that

$$t = \sqrt{N/2}[(\hat{x} - \bar{x})/\sigma(x)],$$

this becomes

$$P\{\bar{x} - \epsilon < \hat{x} \leq \bar{x} + \epsilon\} = \frac{2}{\sqrt{\pi}} \int_0^{(\sqrt{N/2})(\epsilon/\sigma(x))} e^{-t^2} dt. \quad (2.16)$$

Recalling the definition of the error function, Eq. (2.16) becomes

$$P\{\bar{x} - \epsilon < \hat{x} \leq \bar{x} + \epsilon\} = \operatorname{erf}\left[\sqrt{\frac{N}{2}} \frac{\epsilon}{\sigma(x)}\right]. \quad (2.17)$$

Then, using the calculated estimation for  $\sigma(x)$  ( $S$ ), and also recalling that  $S_{\hat{x}} = S/\sqrt{N}$  (Eq. (2.9)), the error function reduces to a function of  $\epsilon$  and  $S_{\hat{x}}$ , or:

$$\operatorname{erf}\left[\sqrt{\frac{N}{2}} \frac{\epsilon}{\sigma(x)}\right] = \operatorname{erf}\left[\sqrt{\frac{1}{2}} \frac{\epsilon}{S_{\hat{x}}}\right]. \quad (2.18)$$

Should  $\varepsilon$  be chosen to be a function of  $S_{\hat{x}}$ , the error function reduces further and becomes merely an evaluation as multiples ( $M$ ) of  $S_{\hat{x}}$  and  $\sqrt{1/2}$ . For the first few multiples of the standard error, this is evaluated as

$$P\{\bar{x} - MS_{\hat{x}} < \hat{x} \leq \bar{x} + MS_{\hat{x}}\} = \begin{cases} .683, & M = 1, \\ .954, & M = 2, \\ .997, & M = 3 \end{cases} \quad (2.19)$$

The central limit theorem tells us that the sample mean follows a normal distribution, regardless of the distribution of the underlying sample, as the number of samples approaches infinity. This means that no matter what distribution is being sampled, the sampled mean will have this expected behavior. As a result, given a calculated value for  $\hat{x}$  and  $S$ , the probability that  $\hat{x}$  is near  $\bar{x}$  is known and calculable. Further, the central limit theorem shows that this distribution is approached very quickly as  $N$  increases, with most problems only requiring  $N > 30$  [7]. Note that  $N$  is not the total number of samples, but the number of samples required to calculate each mean.

However, for the central limit theorem to hold a number of requirements must be satisfied. All of the quantities in Section 2.1.1.1 have the underlying assumption that each  $x_i$  is assumed to be randomly sampled and independent of other  $x_i$ . If some region of phase space is omitted accidentally, these values will not be reflective of the true  $f(x)$ , and so  $\hat{x}$  will not approximate  $\bar{x}$ . Further, for  $S$  to be a good approximation of  $\sigma(x)$ , a large number of  $N$  samples must contribute to the calculation of  $\hat{x}$ . The central limit theorem also assumes that  $f(x)$  is a probability density function that can be sampled and has a variance that exists. As a result, one must be reasonably sure that all of these requirements are satisfied if using Monte Carlo sampling methods.

### 2.1.1.3 The Figure of Merit

The equations in the preceding sections describe how to estimate the statistics of a population given a finite number of samples. In radiation transport, a user seeks to estimate some response, the relative error associated with that response solution, and the time it takes to obtain those values. Equation (2.11) described the relationship between the sample variance, the relative error, and the number of particles as

$$S^2 \propto R^2 \propto \frac{1}{N}.$$

The relationship between the relative error,  $R$ , and the number of particles,  $N$ , (recall that  $R^2 \propto \frac{1}{N}$ ) will be some constant value ( $C$ ):

$$C_1 = R^2 N.$$

As a problem is simulated, the number of particles run,  $N$ , will increase proportionally to the computational transport time,  $T$ . Therefore, the relationship between  $R$  and  $T$  should

also be a constant.

$$C_2 = R^2T$$

The figure of merit (FOM) shown in Eq. (2.20) is the most commonly reported metric using this relationship that is reported. It is widely used in quantifying the effects of variance reduction methods. Because it uses the inverse quantity of the relative error and time, a “good” result would be obtained from a low relative error in a short amount of time, resulting in a FOM with a high numerical value.

$$FOM = \frac{1}{R^2T} \quad (2.20)$$

Further, a user may desire to determine how long a problem must be run to obtain a desired relative error. In that case, Eq. (2.20) can simply be rearranged to

$$R = \frac{1}{(FOM * T)^{1/2}}.$$

The figure of merit is a very useful tool, but it is limited by statistical precision in calculating  $R$ . It is worth noting that early on in a transport simulation, when too few particles have been simulated to effectively capture  $S$  or  $\hat{x}$ , the FOM will oscillate. Eventually, the FOM will converge to a relatively constant value. This behavior can also be used to determine whether one has sufficiently sampled the region in which they are quantifying the response.

## 2.1.2 Variance Reduction Methods for Monte Carlo Radiation Transport

Now that the different parameters that affect the variance in a problem have been introduced, let us transition to different variance reduction techniques that are available in Monte Carlo radiation transport packages. Variance reduction techniques in radiation transport methods fall into four general categories: truncation methods, population control methods, modified sampling methods, and partially-deterministic methods. Of importance for this project are population control methods and modified sampling methods, which are discussed in a number of the papers referenced herein. Truncation methods and partially-deterministic methods do not contribute to and are not the focus of this work, so will only be touched upon briefly.

A later section (3.3) of this dissertation will discuss the choice of software packages used for this project. In particular, our hybrid methods software package is designed to accelerate the Monte Carlo radiation transport package MCNP [10, 9, 8]. Variance reduction methods are available in a number of other Monte Carlo radiation transport packages, and are by no means limited to a particular code. However, the implementation of methods differs between software and the specifics may differ slightly. The discussion for the remainder of this subsection will be centered around the specifics of the code used for this project.

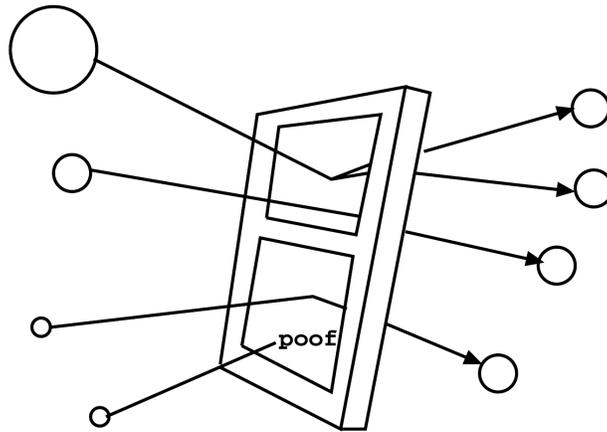


Figure 2.1: Cartoon illustration of a weight window, adapted from [9, 11]

### Population Control Methods

Population control methods adjust the particle population in the problem to obtain better sampling in regions of interest by preferentially increasing or decreasing the particle population. The first two types of population control methods that will be discussed are called splitting and rouletting. Splitting is a method by which the particle population can be increased by splitting a single higher-weight particle into several lower-weight particles. Rouletting, conversely, reduces the particle population by stochastically killing particles. Particles that survive a rouletting routine have their weight adjusted higher, thereby conserving weight in the routine. Both splitting and roulette maintain a fair game by adjusting the particle weights as each routine is performed; statistically, the sum of the child particle weights is the same as the parent weight as it entered the routine.

To use population control methods effectively as a variance reduction technique, splitting is performed in high-importance regions to increase the particle count, and thus the sampling, in important regions. Conversely, rouletting is performed in low-importance regions to reduce the particle population in regions that are unimportant to the tally result. Splitting and rouletting can be applied to include geometry, energy and time.

The weight window combines splitting and rouletting to keep particles within a desired weight range. Figure 2.1 illustrates the different processes a particle may go through when passing through a weight window. The top particle entering the weight window is a single, high-weight particle. The weight of this particle is above the weight window bounds, so as it enters the weight window it is split into multiple particles whose weight is within the window bounds. The second particle entering the window is within the weight window bounds, so it retains its weight and is not split or rouletted. The last two particles entering the window have weights lower than the bound. They undergo a rouletting routine and one particle is killed and the surviving particle is increased in weight. As these particles leave the window, all of them have weights within the range of the window. This will reduce the variance of

the particles contributing to a tally in that region.

While the use of weight windows in a problem helps to keep a more ideal distribution of particle weights, the user is faced with calculating a significant number of parameters to determine weight windows for the entire problem. In the best case with an experienced user, this may just take time. With an inexperienced user or a complex problem this can be insurmountable, and may be too difficult to do without some automated assistance.

It should be noted that while splitting and rouletting can be performed on a single variable—angle, energy, space, or time—the weight windows generally used are either energy-space dependent or space-time dependent. Further, the weight window will split or roulette depending on the particle weight entering the window. Splitting and rouletting on their own either increase or decrease the particle weight proportional to the ratio of cell importances, or  $I'/I$ , no matter what the entering particle weight is. As a result, poorly chosen splitting or rouletting parameters can still have significant tally variance, because particle weights may still span a wide range.

### Modified Sampling Methods

Modified sampling methods adjust transport by sampling from a different probability distribution function than the actual distribution for the problem. This is possible if, as with population control methods, the particle weights are adjusted accordingly. The new probability distribution function should bias particles in regions of high importance to the problem tallies. In MCNP, a number of modified sampling methods exist. These include the exponential transform, implicit capture, forced collisions, source biasing, and neutron-induced photon production biasing.

The exponential transform modifies particle transport from the analog problem by artificially modifying the macroscopic cross section, and thus the distance-to-collision, to move particles in important directions. In directions of higher importance, the cross section is reduced, and particles can flow more freely. In directions of lower importance, the cross section is increased, and particles more frequently interact, thereby increasing their probability of directional change or absorption. The transformed cross section used by the exponential transform is defined by

$$\Sigma_t^* = \Sigma_t(1 - p\mu), \quad (2.21)$$

where  $\Sigma_t^*$  is the transformed total cross section,  $\Sigma_t$  is the true total cross section,  $p$  is the transform parameter, and  $\mu$  is the cosine of the angle between the preferred direction and the particle's transport direction [8, 11, 10].

Because the particle's transport is adjusted in the exponential transform, the particle weight must be adjusted accordingly. This is given by

$$\begin{aligned} w^* &= \frac{\Sigma_t e^{-\Sigma_t s}}{\Sigma_t^* e^{-\Sigma_t^* s}} \\ &= \frac{e^{-\rho \Sigma_t \mu s}}{1 - p\mu}, \end{aligned}$$

where  $s$  is the phase space of particle residence. This weight adjustment ensures that the particle weight is conserved throughout transport, even as the cross section is altered. Because the cross section in the problem is both energy and material dependent (depending on the geometry), the exponential transform will be dependent on space and energy, and particles will be biased in both. While a powerful method, the exponential transform is quite difficult to use and if  $p$  is ill-chosen this method can perform quite poorly. Further, the user has to know quite a bit about the problem physics and material to choose an optimal quantity for  $p$ .

Source biasing, rather than preferentially adjusting particles' directionality by way of adjusting the cross sections, biases particles from their origin. Source biasing has the option to bias particles in energy, direction, and space (if the source is volumetric). This allows the user to choose importance separately for each variable. First, the source variable (let us consider energy for the moment) is defined as a series of bins or a function. Second, the bins are assigned probabilities of occurrence according to their importance. An energy bin with a high importance will be assigned a high probability of occurrence, and a bin with low importance will be assigned a low probability of occurrence. As particles are born in the bins with higher importances, they will have their weights adjusted to the inverse of their probability of occurrence, or  $w^* = p/p^*$ . Here  $p$  refers to the probability density function for the source particles; it bears no relation to the exponential transform factor.

Source biasing is a very simple method that can reduce the solution variance significantly. However, if a user chooses bin sizes or a function that does not properly reflect the particles importances in the problem, the source will be poorly sampled. As a result, sampling may be very inefficient and the figure of merit will decrease. In MCNP, if poor parameters are chosen for this method, the user is given a warning.

## Truncation Methods

Truncation methods stop tracking particles in a region of phase-space that is of low-importance to the tally. These methods can be used in space (a vacuum boundary condition), energy (eliminate particles above or below a specified energy), or time (stop tracking after a given time). To effectively use these methods, the user must be aware of particles' importance to a tally result. If particles that are important to a result are eliminated with a truncation method, the tally will lack the contribution from that particle's phase-space, and will be underestimated as a result. Further, as discussed in Section 2.1.1.2, the central limit theorem only holds assuming that the histories tracked are independent and drawn from identical distributions. Truncating particles of high importance removes the independence from the sampling and modifies the underlying PDF being sampled, so the estimate of the response will be wrong.

It is important in using any variance reduction technique to ensure that a fair game is being played. The user must ensure that the fundamental nature of the problem is not being changed by using a variance reduction technique, or the answer will not be representative of the original problem. Automated variance reduction techniques aim to eliminate this

uncertainty for the user by estimating the importance of particles in some way and then setting up variance reduction parameters automatically. The remainder of this chapter review will focus on efforts to automate population control methods and modified sampling methods for variance reduction.

### 2.1.3 Automated Variance Reduction Methods for Monte Carlo Radiation Transport

Section 2.1.2 described some methods that one may use to reduce the variance in Monte Carlo radiation transport tallies. These methods, if used correctly, can significantly increase the transport efficiency in a Monte Carlo simulation. However, correct use of these methods often requires intelligent selection of variance reduction (VR) parameters, which is a non-trivial task. Users have found themselves often performing several trial runs before choosing final quantities for the VR parameters in their problems, which was computationally inefficient and required significant knowledge of Monte Carlo and variance reduction to execute well [12].

This has been addressed by using Monte Carlo to sample the problem in an initial calculation to determine more favorable variance reduction parameters automatically. Booth and Hendricks, recognizing that choosing optimal weight window values for energy- and space-dependent weight windows was difficult even for experienced users, proposed two tools for Monte Carlo variance reduction in parallel. The first was a Monte Carlo importance generator [12] that could be used to make informed decisions on cell importances throughout the problem. The second method, a Monte Carlo generated weight window generator, calculates the weight window values automatically for a given problem [13]. The importance generator estimates a cell's importance by tracking the weights of the particles in the cell, or

$$Importance = \frac{\text{score of particles leaving the cell}}{\text{weight leaving the cell}}. \quad (2.22)$$

The weight window generator calculates weight window values with

$$W_{i,low} = \frac{1}{kN} (\Sigma W_{i,in} + \Sigma W_{i,out}) \quad (2.23a)$$

$$W_{i,high} = \begin{cases} k * W_{i,low} & \text{if } W_{i,low} \neq 0 \\ \infty & \text{if } W_{i,low} = 0 \end{cases}, \quad (2.23b)$$

where  $W_{i,low}$  and  $W_{i,high}$  are the weight window lower and upper weight bounds respectively,  $W_{i,in}$  and  $W_{i,out}$  are the total weight entering and leaving the cell,  $N$  is the number of source particles, and  $k$  is some weight window width (a constant that Hendricks set to 5).

In his paper, Booth notes that the weight window target value derived from the importance generator was chosen so that the track weight times the expected score in the tally region (for unit track weight) was approximately constant. Booth's importance generator

saw improvements in the FOM between 1.5-8x when compared to the analog run for the test problem presented.

Booth and Hendricks combined these two methods to automate weight window generation based on phase-space importance [14, 15]. They showed that the combination of the importance estimator and the weight window generator was a successful way to perform variance reduction in deep-penetration problems. However, their method depended on several iterations of importance-determining runs to obtain a satisfactory estimation of the importance. For a 300 cm slab problem, the FOM was increased from 1.9 to 75, but took 10 subsequent runs to obtain the FOM of 75, and these runs ranged from 1.2 min (for the analog problem) to 42 minutes (for the 9th run [15]).

It should be noted that both the importance generator and the weight window generator use a lower-fidelity Monte Carlo run to gain an initial estimate for a cell's importance and generate variance reduction parameters from them to bias a more computationally-intensive and higher-fidelity run. Naturally, the variance reduction parameters generated by using these techniques are limited by the statistical precision in the regions used to generate them. Hendricks also pointed out that the weight window generator tended to populate all regions of phase space equally, which he conceded was not ideal for all problems [13]. Furthermore, for deep-penetration particle transport, the variance reduction parameters for low flux regions are exceedingly difficult to generate, resulting in unfavorable VR parameters.

The MCNP [8, 9] weight window generator has been extended beyond the initial space- and energy-implementation described in Booth's paper. It now has the ability to automatically generate space- energy- and angle-weight windows. The importance generator in MCNP also has been extended to time-importance; the values of which can be used for splitting or rouletting parameters [9], and can be optimized on a grid independent from the MCNP geometry [16].

As with Booth and Hendricks' original implementations, this updated weight window generator still relies on adequate sampling to obtain sufficient weight window parameters. When additional degrees of freedom, like angle-dependence, are added, the time to converge on those parameters takes even longer. The weight window generator also only allows for a single tally to be optimized at once, so multiple tallies cannot be optimized simultaneously. Finally, the weight window generator still requires user input and updating to seed the weight window solution. The user must choose the meshing of the problem and have some intuition as to how the problem should be subdivided. In the paper by Van Riper et al, it was found that depending on user experience, the weight window generator can have differences in the FOM from 2 to 10 times [17] for the problems that they investigated.

## 2.2 Importance Functions for Variance Reduction

The effective use of variance reduction techniques can lead to a faster time to a desired solution and a reduced variance in the specified tally. However, specifying variance reduction parameters is not always a straightforward procedure. In simple geometries, a user might

intuitively understand which regions of a problem may contribute more to a desired solution, but for more complex geometries, this may not be so obvious. In the following subsections, the theory in determining which regions of a problem are important to eliciting a tally response will be described. The first topic discussed will be the concept of importance and obtaining a measure of importance with Monte Carlo sampling. Second, the adjoint equation and its relation to importance will be introduced. Last, the contribution solution and how its relation to tally responses is reviewed.

### 2.2.1 The Concept of Importance

The concept of importance is, in essence, a means of defining which regions of a problem that are likely to contribute to a response and which are less likely to contribute to a response. The regions that are more likely to generate a response will have a higher importance than those that do not. If an importance function for a system can be obtained computationally, that importance function can be strategically used in variance reduction techniques to speed up the Monte Carlo calculations.

As described in Section 2.1.3, Booth [12] proposed a method to quantify a cell's importance within a Monte Carlo simulation (Eq. (2.22)). In this method, Booth suggested estimating the cell's importance using Monte Carlo transport as:

$$Importance = \frac{\text{score of particles leaving the cell}}{\text{weight leaving the cell}}.$$

This particular calculation of importance follows from the intuitive explanation for importance in the preceding paragraph. Recall from Section 2.1.2 that in variance reduction methods, the population of particles is increased in important regions such that the number of samples or particles contributing to a tally increases, but the total problem weight is conserved. More important regions should have many lower-weight particles to reduce the tally variance. Using Booth's bookkeeping method for estimating regional importance, if a cell has a greater weight leaving the cell than the number of particles, that means that the relative contribution of that cell to the tally is likely to be lower than other regions. If, instead, the number of particles leaving the cell is greater than the weight leaving the cell, then that region is more important to the tally response, because that particle population is higher than other cells.

While this estimation of the importance requires only a Monte Carlo forward calculation of the problem, it was referred to as the forward-adjoint importance generator [12, 14, 15] because the bookkeeping tracked by Eq. (2.22) was a forward-approximation of the adjoint. Adjoint theory and how it relates to importance will be discussed in Section 2.2.2. Booth's estimation of importance was used to generate weight window target values inversely proportional to the importance. In this case, the track weight times the expected score is approximately constant in the problem. Choosing this inverse relationship between the weight window and importance is common practice in variance reduction, and is often a good choice because it is nearly optimal over a broad range of a problem phase-space [18].

It should be noted that Booth's method is reliant on the statistical precision of the cells sampled to generate their importances. For deep-penetration problems, obtaining a "good" estimate of the cell importances many mean free paths from the forward source takes several iterations. With large fluctuations between iterations, this has the potential to be a very slow and computationally inefficient way to calculate importance in a problem. Using a solution of the adjoint that is equally valid across all of the problem space is more ideal for deep-penetration problems.

## 2.2.2 The Adjoint Solution for Importance

Using the solution of the adjoint formulation of the neutron transport equation is one of the most widely recognized methods for generating an importance function. This subsection will begin with a brief summary of adjoint theory. A discussion on how the adjoint solution differs physically from the forward solution for a particular problem follows. Last, some early investigations on how the adjoint and importance are related are summarized.

### 2.2.2.1 Theory

In previous sections we have reviewed the statistical precision of Monte Carlo-based methods, and how sampled results in Monte Carlo can be obtained in less time with variance reduction methods. We have also briefly addressed the forward and the adjoint solutions for a particular problem. In neutron transport, the integral form of the forward, steady-state, particle transport equation can be defined as:

$$\hat{\Omega} \cdot \nabla \psi(\vec{r}, E, \hat{\Omega}) + \Sigma_t(\vec{r}, E) \psi(\vec{r}, E, \hat{\Omega}) = \int_{4\pi} \int_0^\infty \Sigma_s(E' \rightarrow E, \hat{\Omega}' \rightarrow \hat{\Omega}) \psi(\vec{r}, E', \hat{\Omega}') dE' d\hat{\Omega}' + q_e(\vec{r}, E, \hat{\Omega}), \quad (2.24)$$

where  $\vec{r}$ ,  $E$ , and  $\hat{\Omega}$ , are direction, energy, and angle, respectively, giving six dimensions of phase-space in total.  $\psi$  is the neutron flux,  $\Sigma$  is the neutron interaction (scattering, absorption, total) cross section, and  $q_e$  is the external fixed source. Alternatively, this can be written in operator form,

$$H\psi = q_e, \quad (2.25)$$

where  $H$  represents the streaming, scattering, and absorptive terms from Eq. (2.24),  $\psi$  is the angular flux as it is in Eq. (2.24), and  $q_e$  is the source term.

The forward transport equation tells us where particles are moving throughout the system. Of note: the particles move in the scattering term from  $E'$  into  $E$ , and from  $\hat{\Omega}'$  into  $\hat{\Omega}$ . Therefore, for a particular problem with a given  $q_e$ , particles start at  $q_e$  and move throughout the system, either downscattering in energy, streaming out of the problem, absorbed by the problem materials, or induce a response at the tally location.

The adjoint equation of the same form, in contrast, can be expressed as:

$$-\hat{\Omega} \cdot \nabla \psi^\dagger(\vec{r}, E, \hat{\Omega}) + \Sigma_t(\vec{r}, E) \psi^\dagger(\vec{r}, E, \hat{\Omega}) = \int_{4\pi} \int_0^\infty \Sigma_s(E \rightarrow E', \hat{\Omega} \rightarrow \hat{\Omega}') \psi^\dagger(\vec{r}, E', \hat{\Omega}') dE' d\hat{\Omega}' + q_e^\dagger(\vec{r}, E, \hat{\Omega}), \quad (2.26)$$

or in operator form as

$$H^\dagger \psi^\dagger = q_e^\dagger, \quad (2.27)$$

where the variables with  $\dagger$  signify the adjoint-specific variables for the problem: the adjoint flux  $\psi^\dagger$  and the adjoint source  $q_e^\dagger$ . Note here that the particles in the adjoint equation move from  $E$  into  $E'$ , and from  $\hat{\Omega}$  into  $\hat{\Omega}'$ , which indicates an upscattering in energy and a reversal of direction when compared to the forward problem. The external source, too, is different, changing from  $q_e$  to  $q_e^\dagger$ .

To solve the adjoint problem the adjoint source,  $q_e^\dagger$ , can be chosen such that it has the potential to reveal information about the forward problem. In MC variance reduction, we seek to obtain information on the detector or tally response for the system. The response for the forward problem given a defined source distribution  $q(\vec{r}, E, \hat{\Omega})$  is

$$R_{tally} = \int_{4\pi} \int_V \int_E \psi(\vec{r}, E, \hat{\Omega}) \Sigma_{tally}(\vec{r}, E, \hat{\Omega}) dE dV d\hat{\Omega}, \quad (2.28a)$$

where  $dE$   $dV$  and  $d\Omega$  are the differential spaces of energy, volume, and angle in the tally region. This can be simplified using bracket notation, where the brackets indicate an integration over all phase-space,

$$R_{tally} = \langle \psi \Sigma_{tally} \rangle. \quad (2.28b)$$

$\psi$  is the angular flux and  $\Sigma_{tally}$  is the effective tally response function.

For a simple source-detector problem, we choose  $q_e^\dagger$  to be  $\Sigma_{tally}$ ; or the adjoint source is the tally/detector response function for the system. Therefore, the adjoint particles start at low energy at the detector location, move away from the adjoint source (the detector location), and scatter up in energy. By making the choice that  $q_e^\dagger = \Sigma_{tally}$ , the response function can be written as a product for the forward flux and the adjoint source

$$R_{tally} = \langle \psi q^\dagger \rangle. \quad (2.29)$$

By using the adjoint identity and the same operators  $H$  from Eqs. (2.25) and (2.27)

$$\langle \psi, H^\dagger \psi^\dagger \rangle = \langle \psi^\dagger, H \psi \rangle. \quad (2.30)$$

Eq. (2.29) can be written as a function of the adjoint flux and the forward source distribution

$$R = \langle \psi^\dagger q \rangle. \quad (2.31)$$

At this point, we know that the solution to the adjoint problem transports particles from the adjoint source (which is the detector or tally location) into the problem phase-space. The

adjoint particles are upscattered in energy and are transported in  $-\Omega$  relative to the forward problem. However, it may not be immediately obvious how this adjoint solution relates to importance for the forward solution. Let us start with a simple illustrative example: a monoenergetic, monodirectional, point source. The forward source takes the form of a delta function:

$$q(\vec{r}, E, \hat{\Omega}) = \delta(\vec{r} - \vec{r}_0)\delta(E - E_0)\delta(\hat{\Omega} - \hat{\Omega}_0).$$

Using this definition of the forward source and evaluating Eq. (2.31), we obtain

$$\begin{aligned} R &= \langle \psi^\dagger q \rangle \\ &= \int_V \int_E \int_\Omega \psi^\dagger(\vec{r}, E, \hat{\Omega}) q(\vec{r}, E, \hat{\Omega}) d\hat{\Omega} dE dV \\ &= \psi^\dagger(\vec{r}_0, E_0, \hat{\Omega}_0). \end{aligned}$$

This result shows that the solution to the adjoint equation is the detector response for the forward problem. As a result, the adjoint flux can be used as an indicator of a particle produced in  $\vec{r}, E, \hat{\Omega}$  contributing to a response in the system. This indicator can be thought of as the particle's importance to achieving the tally or response objective. Consequently, it is often said that the adjoint is the importance function for the problem.

The adjoint solution is used in nuclear engineering for a number of applications, including reactor physics and perturbation theory [19, 21, 20, 22]. However, Goertzel and Kalos' early work recognized its application for deep-penetration radiation shielding. Goertzel and Kalos [23] showed analytically that the exact adjoint solution, if used as an importance or weighting function for the forward Monte Carlo calculation, will result in a zero variance solution for the forward Monte Carlo problem. Further, Kalos [24] showed in a 1D infinite hydrogen slab problem that an analytically-derived adjoint importance function significantly improved the speed to convergence for neutron transport in deep-penetration problems.

Goertzel and Kalos' finding that an exact adjoint can lead to a zero variance solution means that if a single particle is transported with the adjoint weighting function, its score will be the same as the total system response. Only a single particle is required to get an exact solution for the forward problem. This is prohibitive because obtaining an exact adjoint solution is just as computationally expensive as getting an exact forward solution. Instead, one seeks to obtain a good, but fairly inexpensive, estimate of the adjoint solution based on computational limitations. A good importance estimate should help reduce the variance in a reasonable amount of time and be relatively computationally inexpensive. A Monte Carlo solution can provide a continuous solution over the problem phase-space. However, as discussed in Section 2.1.2, the quality of this adjoint solution relies on the number of samples used to calculate it and that may take a significant amount of time. A deterministic solution has the potential to offer equal or better solution confidence across the entire problem. However, it is discretized in space, energy, and angle. For deep-penetration importance functions, we opt for deterministically-obtained solutions due to the solution's equally distributed validity.

### 2.2.2.2 Implementation

Coveyou, Cain, and Yost [25] expanded on Goertzel and Kalos' work by interpreting in which ways the adjoint solution could be adapted for Monte Carlo variance reduction. In particular, they investigated the choice of biasing schemes and how effective they were at variance reduction for a simple one-dimensional problem. They reiterated that the adjoint solution is a good estimate for importance, but should not be calculated explicitly, and rather estimated by a simpler model. The adjoint function is not necessarily the most optimal importance function; however, it is likely the closest and most obtainable estimate of importance that can be calculated [25]. They concluded that source biasing by the solution to the adjoint equation or by the expected response is the best choice for Monte Carlo variance reduction, as it can be used independently from any other variance reduction technique, and provides good results.

Tang and Hoffman [26] built off of the parameters derived by Coveyou et al. [25] to generate variance reduction parameters automatically for fuel cask dose rate analyses. In their work, Tang and Hoffman used the 1D discrete ordinates code XSDRNPM-S to calculate the adjoint fluxes for their shielding problems. The results from this calculation were then used to generate biasing parameters for Monte Carlo; specifically, they aimed at generating parameters for energy biasing, source biasing, Russian roulette and splitting, and next event estimation probabilities. They implemented their work in the SAS4 module in SCALE [27]; it was one of the earlier implementations of a fully-automated deterministic biasing procedure for Monte Carlo.

### 2.2.3 The Contributon Solution for Importance

Contributon theory is another useful concept that can be used as a measure of importance [28, 29, 30]. However, contributon theory quantifies importance differently than adjoint theory. In contributon transport, a pseudo-particle, the *contributon*, is defined. The contributon carries response in the problem system from the radiation source to a detector. The total number of contributons in a system are conserved by the *contributon conservation principle*: all contributons that are emitted from the source eventually arrive at the detector. Much of the work in this realm has been done by Williams and collaborators [28, 29, 30].

The contributon transport equation can be derived in a form analogous to the forward (Eq. (2.24)) and adjoint (Eq. (2.26)) equations. The derivation of Eq. (2.33) and its corresponding variables is available in a number of the sources referenced in this section, so we will abstain from re-deriving it here. The angular contributon flux is defined as the product of the forward and adjoint angular fluxes:

$$\Psi(\vec{r}, E, \hat{\Omega}) = \psi^\dagger(\vec{r}, E, \hat{\Omega})\psi(\vec{r}, E, \hat{\Omega}). \quad (2.32)$$

The contributon transport equation is:

$$\begin{aligned} \hat{\Omega} \cdot \nabla \Psi(\vec{r}, E, \hat{\Omega}) + \tilde{\Sigma}_t(\vec{r}, E, \hat{\Omega}) \Psi(\vec{r}, E, \hat{\Omega}) = \\ \int_{4\pi} \int_0^\infty \tilde{p}(\vec{r}, \hat{\Omega}' \rightarrow \hat{\Omega}, E' \rightarrow E) \tilde{P}(\vec{r}, \hat{\Omega}', E') \tilde{\Sigma}_t(\vec{r}, E', \hat{\Omega}') \Psi(\vec{r}, E', \hat{\Omega}') dE' d\hat{\Omega}' + \hat{p}(\vec{r}, E, \hat{\Omega}) R. \end{aligned} \quad (2.33)$$

The units of phase-space are the same as observed in the forward and adjoint transport equations. The symbols decorated with tildes denote variables that are unique to the contributon equation;  $\tilde{p}$  and  $\tilde{P}$  are both probability functions related to scattering and  $\tilde{\Sigma}$  are effective cross sections. The effective cross sections are given by:

$$\begin{aligned} \tilde{\Sigma}_t(\vec{r}, E, \hat{\Omega}) &= \tilde{\Sigma}_s(\vec{r}, E, \hat{\Omega}) + \tilde{\Sigma}_a(\vec{r}, E, \hat{\Omega}) \\ &= \frac{\iint \Sigma_s(\vec{r}, \hat{\Omega}'' \cdot \hat{\Omega}, E \rightarrow E'') \psi^\dagger(\vec{r}, \hat{\Omega}'', E'') d\hat{\Omega}'' dE''}{\psi^\dagger(\vec{r}, E, \hat{\Omega})} + \frac{Q^\dagger(\vec{r}, E, \hat{\Omega})}{\psi^\dagger(\vec{r}, E, \hat{\Omega})}. \end{aligned} \quad (2.34)$$

Note here that the effective scattering and absorption cross sections are adjoint flux-dependent. Where the adjoint flux becomes small, the interaction probabilities will become large. As a result, regions where the adjoint flux is high interaction probabilities become low, causing fewer interactions and more streaming. Conversely, regions with low adjoint fluxes—like the problem boundary—will have a very high cross section, thus encouraging particle transport back towards the adjoint source. This increased probability of interaction in low flux regions encourages response particle (contributon) transport towards the detector or tally, thus contributing to a response.

The scattering probability of a contributon at position  $\vec{r}$ ,  $E'$ , and  $\hat{\Omega}'$  is:

$$\tilde{P}(\vec{r}, \hat{\Omega}', E') = \frac{\tilde{\Sigma}_s(\vec{r}, E', \hat{\Omega}')}{\tilde{\Sigma}_t(\vec{r}, E', \hat{\Omega}')}, \quad (2.35)$$

and the probability that a contributon scattering at  $\vec{r}$ ,  $E'$ , and  $\hat{\Omega}'$  will scatter into  $d\hat{\Omega} dE$  is

$$\tilde{p}(\vec{r}, \hat{\Omega}' \rightarrow \hat{\Omega}, E' \rightarrow E) = \frac{\Sigma_s(\vec{r}, \hat{\Omega}' \cdot \hat{\Omega}, E' \rightarrow E) \psi^\dagger(\vec{r}, E, \hat{\Omega})}{\iint \Sigma_s(\vec{r}, \hat{\Omega}'' \cdot \hat{\Omega}'', E' \rightarrow E'') \psi^\dagger(\vec{r}, E'', \hat{\Omega}'') d\hat{\Omega}'' dE''}. \quad (2.36)$$

The distribution function governing the contributon source is

$$\hat{p}(\vec{r}, E, \hat{\Omega}) = \frac{\psi^\dagger(\vec{r}, E, \hat{\Omega}) Q(\vec{r}, E, \hat{\Omega})}{\iiint \psi^\dagger(\vec{r}', E', \hat{\Omega}') Q(\vec{r}', E', \hat{\Omega}') d\hat{\Omega}' dE' dV'}, \quad (2.37)$$

note that the contributon source is actually defined in Eq. (2.33) by the product of  $\hat{p}$  and  $R$ .  $R$  is contributon production rate; it is given by integral of the adjoint flux and the forward source

$$\begin{aligned} R &= \int \int \int \psi^\dagger(\vec{r}, E, \hat{\Omega}) Q(\vec{r}, E, \hat{\Omega}) d\hat{\Omega} dE dV, \\ &= \langle \psi^\dagger Q \rangle \end{aligned} \quad (2.38)$$

which is recognizable as the system response described in Section 2.2.2. It can also be shown by integrating Eq. (2.33) over all phase space and ensuring that the function  $\hat{p}$  is normalized, that

$$R = \langle \widetilde{\Sigma}_a \Psi \rangle, \quad (2.39)$$

or the rate at which contributons die in the detector is the same as the rate at which they are produced by the contributon source. Knowing that  $R$  is the contributon production rate, let us consider the probability that a particle will be absorbed in the detector, or  $P$ , given by

$$P = \langle \Sigma_a \psi \rangle. \quad (2.40)$$

Adding a factor of  $\psi^\dagger/\psi^\dagger$  to the terms on the right hand side, this becomes

$$P = \left\langle \frac{\Sigma_a}{\psi^\dagger} \psi \psi^\dagger \right\rangle. \quad (2.41)$$

By using the identities from the contributon equation, this is also

$$P = \langle \widetilde{\Sigma}_a \Psi \rangle. \quad (2.42)$$

Next, substituting the definition from Eq. (2.39) into this equation, it follows that

$$P = R. \quad (2.43)$$

This is the same *contributon conservation principle* introduced at the beginning of this section. Williams noted that one could go so far as to transport contributons rather than real particles with Monte Carlo. Because every particle transported would eventually reach the detector and give an exact value for  $R$  (as shown by Eq. (2.43)), this would lead to a zero variance solution. However, the nature of solving the contributon equation with Monte Carlo (or any other transport mechanism) involves knowing the exact solution to the adjoint equation, and so relies on the same computational obstacles as solving the adjoint transport equation.

As mentioned in the previous section, the adjoint flux is an indicator of a particle's importance to inducing a response. Conversely, the contributon flux describes the importance of a particle to the solution. Becker's thesis [31] aptly points out that this is illustrated most dramatically in a source-detector problem, where the forward source has little importance to the adjoint source, but does have importance to the problem solution. As a result, both the contributon solution and the adjoint solution can be considered importance functions for a problem, but the importance that they describe differs.

Williams recognized the applications of contributons to shield design and optimization in an extension of contribution theory called spatial channel theory. In particular, Williams noted that variables relevant to contributon response were useful in determining transport paths through media [30, 32]. A study of different contributon values throughout the system could enlighten users on regions with higher response potential. This could then be used to

intelligently choose regions for detector locations or add to shielding. The contribution values in this theory include the contribution flux, the contribution density, the contribution current, or the contribution velocity [33]. In this way, the user could find the particles most influential to the response of the system. A region with high response potential is the most important to a detector tally. The variables of response described by Williams are the response potential, the response current, and the response vorticity [29].

Contribution theory and spatial channel theory have been applied successfully to shielding analyses [34, 32] due to their ability to show particle flow between a source and response effectively. Williams and Engle showed that spatial channel theory can be used in reactor shielding analyses. In their work, they used contribution currents to determine preferential flow paths through the Fast Flux Test Facility (FFTF) [32]. Seydaliev [34] used angle-dependent forward and adjoint fluxes and currents to visualize the contribution flux for simple source-detector problems. In this work, he showed that contribution flow in the system behaves much like a fluid between the source and detector, following preferential flow paths more densely. Seydaliev also observed ray effects in the contribution flux for high energy photons, and traditional methods like using a first collision source, did not remedy the issue. The contribution formulation of particle transport can show important particle flow paths between a source and a detector, but it is still not immune to computational obstacles that exist for standard forward- and adjoint- transport.

The past few subsections have described the different means by which importance can be defined or quantified for a problem. As discussed in Section 2.2.1, generating an importance function with Monte Carlo is limited in that the quality of the importance map is only as good as the regions that are sampled. For deep-penetration problems, it may be prohibitively difficult to obtain adequate importance sampling with traditional Monte Carlo methods.

Deterministically-obtained importance functions, however, offer the benefit of a solution that is equally valid across all of the problem solution-space. This is because the deterministic solution's precision is limited to convergence criteria, not sampling of individual particles. Using a deterministic solution is often faster and much less computationally-intensive than Monte Carlo for importance quantification. As a result, many hybrid methods opt to use deterministically-obtained importance functions to generate variance reduction parameters for Monte Carlo transport.

## 2.3 Automated Variance Reduction Methods for Local Solutions

The next several sections (2.3 through 2.5) describe different ways that deterministically-obtained importance functions can be applied to variance reduction methods in practice. Local variance reduction methods are those that optimize a tally response in a localized region of the problem phase-space. These types of problems may be the most immediately physically intuitive to a user, where a person standing  $x$  meters away from a source may wish

to know their personal dose rate. In this section, notable automated deterministically-driven variance reduction methods that have been designed for such localized response optimization are described. Recall that Booth's importance generator (Section 2.1.3) was also designed for localized tally results, but will not be elaborated upon here.

### 2.3.1 CADIS

In 1997, Haghghat and Wagner introduced the Consistent Adjoint-Driven Importance Sampling method (CADIS) [1, 3, 2] as a tool for automatic variance reduction for local tallies in Monte Carlo. CADIS was unique in that it used the adjoint solution from a deterministic simulation to consistently bias the particle distribution and particle weights. Earlier methods had not ensured the consistency between source biasing and particle birth weights. CADIS was applied to a large number of application problems and performed well in reducing the variance for local tallies [35].

The next several paragraphs present and discuss the theory supporting CADIS. Note that the theory presented is specific to space-energy CADIS, which is what is currently implemented in existing software. The original CADIS equations are based on space and energy  $(\vec{r}, E)$  dependence, but not angle, so  $\phi^\dagger$  can be used rather than  $\psi^\dagger$ . This does not mean that CADIS is not applicable to angle. This is merely a choice made by the software and method developers given the computational resources required to calculate and store full angular flux datasets, and the inefficiency that using angular fluxes might pose for problems where angle dependence is not paramount.

In trying to reduce the variance for a local tally, we aim to encourage particle movement towards the tally or detector location. In other words, we seek to encourage particles to induce a detector response while discouraging them from moving through unimportant regions in the problem. Recall from Eqs. (2.29) and (2.31) that the total system response can be expressed as either an integral of  $\psi^\dagger q_e$  (the adjoint flux and the forward source), or  $\psi q_e^\dagger$  (the forward flux and the adjoint source). Also recall that the adjoint solution is a measure for response importance.

To generate the biased source distribution for the Monte Carlo calculation,  $\hat{q}$ , should be related to its contribution to inducing a response in the tally or detector. It follows, then, that the biased source distribution is the ratio of the contribution of a cell to a tally response to the tally response induced from the entire problem. Thus, the biased source distribution for CADIS is a function of the adjoint scalar flux and the forward source distribution  $q$  in region  $\vec{r}, E$ , and the total response  $R$

$$\begin{aligned}\hat{q} &= \frac{\phi^\dagger(\vec{r}, E)q(\vec{r}, E)}{\iint \phi^\dagger(\vec{r}, E)q(\vec{r}, E)dEd\vec{r}} \\ &= \frac{\phi^\dagger(\vec{r}, E)q(\vec{r}, E)}{R}.\end{aligned}\tag{2.44a}$$

The starting weights of the particles sampled from the biased source distribution ( $\hat{q}$ ) must be adjusted to account for the biased source distribution. As a result, the starting weights

are a function of the biased source distribution and the original forward source distribution:

$$\begin{aligned} w_0 &= \frac{q}{\hat{q}} \\ &= \frac{R}{\phi^\dagger(\vec{r}, E)}. \end{aligned} \quad (2.44b)$$

Note that when Eq. (2.44a) is placed into Eq. (2.44b), the starting weight is a function of the total problem response and the adjoint scalar flux in  $\vec{r}, E$ . The target weights for the biased particles are given by

$$\hat{w} = \frac{R}{\phi^\dagger(\vec{r}, E)}, \quad (2.44c)$$

where the target weight  $\hat{w}$  is also a function of the total response and the adjoint scalar flux in region  $\vec{r}, E$ . The equations for  $\hat{w}$  and  $w_0$  match; particles are born at the same weight of the region they are born into. Consequently, the problem limits excessive splitting and roulette at the particle births, in addition to consistently biasing the particle source distribution and weights. This is the “consistent” feature of the CADIS method.

CADIS supports adjoint theory by showing that using the adjoint solution ( $\phi^\dagger$ ) for variance reduction parameter generation successfully improves Monte Carlo calculation runtime. CADIS showed improvements in the FOM when compared to analog Monte Carlo on the order of  $10^2$  to  $10^3$ , and on the order of five when compared to “expert” determined or automatically-generated weight windows [3, 4] for simple shielding problems. For more complex shielding problems, improvements in the FOM were on the order of  $10^1$  [1, 3]. Note that CADIS improvement is dependent on the nature of the problem and that these are merely illustrative examples.

### 2.3.2 Becker’s Local Weight Windows

Becker’s work in the mid- 2000s also explored generating biasing parameters for local source-detector problems [31]. Becker noted that in traditional weight window generating methods, some estimation of the adjoint flux is used to bias a forward Monte Carlo calculation. The product of this weight window biasing and the forward Monte Carlo transport ultimately distributed particles in the problem similarly to the contribution flux. In his work, Becker used a formulation of the contribution flux, as described in Eq. (2.32) to optimize the flux at the forward detector location. The relevant equations are given by Eqs. (2.45a) - (2.45f).

First, the scalar contribution flux  $\phi^c$ , which is a function of space and energy is calculated with a product of the deterministically-calculated forward and adjoint fluxes, where

$$\phi^c(\vec{r}, E) = \phi(\vec{r}, E)\phi^\dagger(\vec{r}, E). \quad (2.45a)$$

This is then integrated over all energy to obtain a spatially-dependent contribution flux

$$\tilde{\phi}^c(\vec{r}) = C_{norm} \int_0^\infty \phi^c(\vec{r}, E)dE, \quad (2.45b)$$

where the normalization constant,  $C_{norm}$ , for a given detector volume,  $V_D$ , is:

$$C_{norm} = \frac{V_D}{\int_{V_D} \int_0^\infty \phi^c(\vec{r}, E) dE dV}. \quad (2.45c)$$

The space- and energy-dependent weight windows are given by:

$$\bar{w}(\vec{r}, E) = \frac{B(\vec{r})}{\phi^\dagger(\vec{r}, E)}, \quad (2.45d)$$

where

$$B(\vec{r}) = \alpha(\vec{r})\tilde{\phi}^c(\vec{r}) + 1 - \alpha(\vec{r}), \quad (2.45e)$$

and

$$\alpha(\vec{r}) = \left[ 1 + \exp\left(\frac{\tilde{\phi}_{max}^c}{\tilde{\phi}^c(\vec{r})} - \frac{\tilde{\phi}^c(\vec{r})}{\tilde{\phi}_{max}^c}\right) \right]^{-1}. \quad (2.45f)$$

Becker found that this methodology compared similarly to CADIS for local solution variance reduction for a large challenge problem comprised of nested cubes. The particle density throughout the problem was similar between CADIS and Becker's local weight window. The FOMs were also relatively similar, but were reported only with Monte Carlo calculation runtimes (meaning that the deterministic runtimes were excluded). Note that Becker's method requires both a forward and an adjoint calculation to calculate the contribution fluxes, while CADIS requires only an adjoint calculation.

## 2.4 Automated Variance Reduction Methods for Global Solutions

Variance reduction methods for global solutions are designed to obtain an even distribution of error across several tallies or a tally map that spans the entire problem phase-space. The previous section detailed several methods that automate variance reduction for localized tallies. However, for global solutions these methods do not work well. The global tally suffers from a large range in variance across the physical problem space, and the solution is dependent on the flux distribution throughout the problem.

This section describes several methods that provide automated variance reduction for global solutions or multiple tallies. The general principle that these methods follow is that by distributing particles evenly throughout the Monte Carlo problem, a global tally will have approximately the same sample size in each region, resulting in a uniform variance across the tally. This often requires a forward deterministic solution to determine the density of forward particles throughout the problem, and subsequently using that forward distribution to aid in generating an importance map. This section summarizes the theory behind a number of existing global variance reduction methods. The section is concluded with a summary of how the methods performed and in which problems they performed well.

### 2.4.1 Cooper’s Isotropic Weight Windows

Cooper and Larsen developed a weight window technique to reduce the variance of Monte Carlo global solutions [36] using a calculation of the forward flux from solutions obtained from diffusion, quasidiffusion [37], or pure Monte Carlo. In their work, Cooper and Larsen utilized a forward solution to the transport equation to generate weight window values to uniformly distribute particles throughout the problem. By doing so, the variance in the scalar flux remained relatively constant throughout the problem for a problem-wide tally, rather than rising significantly with increasing distance from the forward source. Cooper’s “isotropic” weight windows (named because they were not dependent on  $\hat{\Omega}$ ) dependent on  $\vec{r}$  are given by:

$$\bar{w}w(\vec{r}) = \frac{\phi(\vec{r})}{\max \phi(\vec{r})}, \quad (2.46a)$$

$$ww(\vec{r})_{top} = \rho \bar{w}w(\vec{r}), \quad (2.46b)$$

and

$$ww(\vec{r})_{bottom} = \frac{\bar{w}w(\vec{r})}{\rho}, \quad (2.46c)$$

where  $\rho$  is the weight window scaling factor. Note that by setting the weight window target value to be inversely proportional to the total flux in the cell, the density of particles throughout the problem ends up as roughly constant. Also note from Eq. (2.46a) that the weight windows are depend on space only.

In practice, Cooper’s algorithm iteratively switches between solving the diffusion equation with transport correctors (Eddington factors described by [38]), and Monte Carlo solutions; this process is known as quasidiffusion [38, 37]. An initial quasidiffusion solution is used to generate weight windows, and then after a relatively short runtime, the Monte Carlo solution is used to generate updated Eddington factors for the quasidiffusion solution.

Because Cooper’s method depends on Monte Carlo to generate the Eddington factors for the quasidiffusion problem, this method is limited by the iterative switch between the quasidiffusion solution and the Monte Carlo solution. The frequency with which this switching happens is entirely up to the user, but may drastically affect the efficiency of the method. Further, Cooper notes that we do not know at what point in time (for which number of N particles) the Monte Carlo solution becomes more accurate than the quasidiffusion solution, which is an important issue in choosing solution parameters.

### 2.4.2 Becker’s Global Weight Windows

Becker, in addition to developing the local VR method discussed in Section 2.3.2, developed a global space-energy weight correction method both with (Section 2.5) and without directional biasing [39, 31]. Becker’s global method uses a formulation of the space-dependent contribution flux, as with the local weight windows described in Section 2.3.2. For reference, those are defined in Eqs. (2.45a) and (2.45b).

Becker defines the spatially-dependent contribution flux parameter as  $B(\vec{r})$ , where

$$B(\vec{r}) = \tilde{\phi}^c(\vec{r}). \quad (2.47)$$

Becker's method defines a different adjoint source distribution depending on the response desired for the calculation. To optimize the flux the adjoint source is defined as:

$$q^\dagger(\vec{r}, E) = \frac{1}{\phi(\vec{r}, E)}. \quad (2.48a)$$

If the detector response is desired then

$$q^\dagger(\vec{r}, E) = \frac{\sigma_d(\vec{r}, E)}{\int_0^\infty \phi(\vec{r}, E) \sigma_d(\vec{r}, E) dE}, \quad (2.48b)$$

can be used instead. The space- and energy-dependent weight windows are then a function of the contribution flux, where

$$\bar{w}(\vec{r}, E) = \frac{B(\vec{r})}{\phi^\dagger(\vec{r}, E)}. \quad (2.49)$$

The process followed by Becker's global method uses two deterministic calculations to generate weight windows for the Monte Carlo calculation. First, the forward flux is calculated deterministically and used to construct the adjoint source distribution. After the adjoint solution is run, the contribution flux is calculated. The contribution flux and the adjoint flux are then used to construct the weight windows.

Becker's method aims to distribute response evenly throughout the problem. However, like FW-CADIS (discussed below in Section 2.4.3), the global response weight windows are proportional to the forward response,

$$\bar{w}(\vec{r}, E) \propto \frac{\int \sigma(\vec{r}, E) \phi(\vec{r}, E) dE}{\sigma(\vec{r}, E)} \quad (2.50)$$

rather than the forward flux as in Cooper's method, where  $\bar{w}(\vec{r}, E) \propto \phi(\vec{r}, E)$ .

In implementation, both Becker and Cooper's global methods undersampled the source (in comparison to FW-CADIS, which will be described in Section 2.4.3) for a specified calculation time. However, Becker's method sampled  $\sim 1/3$  the number of particles that Cooper's method did. Notably, Becker's method did obtain better relative uncertainties for low flux-regions in the problem.

### 2.4.3 FW-CADIS

In 2007, Peplow, Blakeman, and Wagner [40] proposed three methods by which variance reduction could be decreased in global mesh tallies in deep-penetration radiation transport problems. The first method, using a CADIS calculation where the adjoint source is defined at the problem boundary, aimed at moving particles outward to the problem edges. The second

method used standard CADIS, but instead defined each cell as equally important, so the adjoint source was defined equally throughout the problem phase-space. The last method, called Forward-Weighted CADIS (FW-CADIS), distributed the adjoint source across mesh cells as an inverse relation to the forward response of the cell. In their work, Peplow et al. found that the first method had large uncertainties in areas of the problem far from the boundary; the second method performed slightly, but not significantly, better than analog; and the third method had the most uniform uncertainty distribution.

FW-CADIS [5, 6, 41] built off of the work by Cooper and the CADIS method. Like Becker's method, FW-CADIS uses a forward deterministic calculation to determine the source distribution for the adjoint calculation. Unlike Becker's method, which used contribution fluxes to construct weight windows, the CADIS method uses adjoint fluxes as the basis of the weight window values. Similar to Cooper's method, however, FW-CADIS uses the forward calculation to determine how to evenly distribute particles throughout the problem. Like CADIS, FW-CADIS uses the adjoint solution from the deterministic calculation to generate consistent source biasing, weight windows, and particle birth weights.

The adjoint source for the adjoint calculation is dependent on the desired response for the system. The generic description for the adjoint source is given by Eq. (2.51) and more specific parameters are given by Eqs. (2.52a)-(2.52c). First, we can describe a general form of the adjoint source definition for all phase-space,  $P$ , as:

$$q^\dagger(P) = \frac{\sigma_d(P)}{R}. \quad (2.51)$$

Thus the adjoint source is dependent on the detector (or tally) cross-section and whatever response is being calculated in the system. Depending on whether the response is a flux or a dose rate, the adjoint source will differ. For example, the adjoint source for the spatially dependent global dose,  $\int \phi(\vec{r}, E)\sigma_d(\vec{r}, E)dE$  is:

$$q^\dagger(\vec{r}, E) = \frac{\sigma_d(\vec{r}, E)}{\int \sigma_d(\vec{r}, E)\phi(\vec{r}, E)dE}. \quad (2.52a)$$

The adjoint source for the spatially dependent total flux  $\int \phi(\vec{r}, E)dE$  is:

$$q^\dagger(\vec{r}) = \frac{1}{\int \phi(\vec{r}, E)dE}. \quad (2.52b)$$

Last, the adjoint source for the energy- and spatially- dependent flux  $\phi(\vec{r}, E)$  is:

$$q^\dagger(\vec{r}, E) = \frac{1}{\phi(\vec{r}, E)}. \quad (2.52c)$$

The process followed by FW-CADIS is to initially run a deterministic forward calculation to obtain the forward response. This solution is used to create the source distribution for the adjoint problem. A second deterministic calculation is run to obtain the adjoint solution. The adjoint solution is then used to generate variance reduction parameters in the same manner as CADIS.

### 2.4.4 Other Notable Methods

Baker and Larsen showed that the exponential transform can be used to generate VR parameters for global low-variance solutions in Monte Carlo [42]. In this work, Baker used a forward diffusion solution to generate parameters for a combination of VR techniques: implicit capture and weight cutoff, geometry splitting / rouletting with implicit capture and weight cutoff, and the exponential transform combined with implicit capture and a weight cutoff. The exponential transform method was then compared to the other combinations of VR techniques to quantify its success. In their work, Baker and Larsen found that the exponential transform approach did not work well for highly scattering problems, where geometry splitting and Russian roulette were generally better options. Their work did not focus on generating weight window values, nor was it tested on deep-penetration shielding problems.

While the aforementioned methods in this and the previous sections use deterministically-obtained solutions to generate importance maps, it should be noted that not all methods use this approach. Booth and Hendricks' methods used initial Monte Carlo calculations to reduce the relative error in tallies. Two methods in the global variance reduction realm are notable in that they too use Monte Carlo estimates of the flux to generate variance reduction parameters [44, 43]. Van Wijk et al. [44] developed an automated weight window generator that used a Monte Carlo calculation of the forward flux to generate weight window values. The weight window target values could be determined based on either a flux-centered scheme like Cooper's (Eq. (2.46a)) or by using a ratio of the square roots of the fluxes. The second method is a combination of Cooper's weight window target values and knowing that the relative error in a region is proportional to the square root of the number of particles. Van Wijk et al. applied their methods to a PWR facility and observed a FOM increase by a factor of  $>200$  when compared to analog Monte Carlo. However, as with other Monte Carlo-generated VR parameters, for deep-penetration problems this approach relies on adequate sampling of all phase-space, which could be computationally prohibitive.

The Method of Automatic Generation of Importances by Calculation (MAGIC) method was proposed in parallel by Davis and Turner [43]. As with Van Wijk's method, the MAGIC method uses an analog forward Monte Carlo –potentially with several iterations–calculation to generate weight windows. The initial Monte Carlo runs used to generate the importance map took less time to converge by using multigroup (rather than continuous energy) cross section data as well as energy cutoffs. MAGIC converged on a finalized importance map by iteratively running several lower-fidelity Monte Carlo calculations.

Davis and Hendricks compared three variants of MAGIC to FW-CADIS in ITER fusion energy systems. These three variants used different weight window adjustments for importances: weight windows in cells based on existing weight information, weight windows in mesh cells based on flux information, and weight windows in cells based on population density. It was concluded that the most effective method for variance reduction of those proposed in the paper was MAGIC's weight window in mesh cells based on flux information. In this case, FW-CADIS' FOM was 65% that of MAGIC's. This compared similarly to Van

Wijk’s method, where the flux-based results continued improving the FOM as the computational time increased. The authors did not make it clear how many iterations were required, on average, to generate the finalized weight window map or if the time to iteratively generate the importance map was included in the FOM. While FW-CADIS’ FOM was lower than MAGIC’s, FW-CADIS had the highest fraction of cell voxels with very low relative errors.

Peplow et al. [45] compared the performance of Cooper’s method, Van Wijk’s method, Becker’s method, and FW-CADIS across a number of shielding calculations. For a simple shielding problem, FW-CADIS had the shortest runtime, which included the forward and adjoint deterministic runtimes, and had a FOM 80x higher than the analog calculation, and more than 3x higher than the next best hybrid method. Van Wijk’s method was the only method other than FW-CADIS to pass all statistical convergence checks for the problem, but its reported FOM was lower than either Becker’s method or FW-CADIS. In a second deep penetration shielding problem, FW-CADIS was the only method that passed all statistical convergence checks. FW-CADIS also had the highest reported FOM for this problem. The timing for all of the methods were comparable. Peplow et al. also ran two “challenge” problems. As with the first two problems, FW-CADIS outperformed the other methods and passed all statistical checks. Becker’s method was consistently comparable to FW-CADIS in reported FOMs, but only passed all of the statistical checks in a single challenge problem. Becker’s method also performed relatively better than the other methods in deep-penetration challenge problems.

The ubiquity and continued development of global variance reduction methods illustrates the need and desire for them in the nuclear engineering community. Some of the methods discussed in this section—including Becker’s global weight windows, Cooper’s weight windows, Van Wijk’s method, and FW-CADIS—have been applied to large application problems and compared to other methods. All of the methods reduce the time to a “good” solution—thus improving the final FOM—when compared to analog Monte Carlo. When compared against one another, FW-CADIS consistently outperforms the other methods.

## 2.5 Automated Angle-Informed Variance Reduction Methods

In a number of problems, the angular dependence of the flux is significant enough that biasing in space and energy exclusively is not sufficient. As a result, a subset of hybrid methods were developed to incorporate some degree of the flux anisotropy in variance reduction parameters. Without explicitly calculating the angular flux, which is memory- and storage-intensive, methods attempted to approximate the angular flux using other information more readily accessible to them. These approaches are broadly categorized as methods that bias using population control methods (such as weight windows), and methods that bias with modified sampling methods (such as the exponential transform). Initial approaches to angular biasing focused on approximating the angular flux,  $\psi$ , as a separable function of the scalar flux

and an angle-dependent multiplier. These approximations of the flux were then used to generate biasing parameters dependent on angle for highly angular-dependent problems. In this section, methods that generate variance reduction parameters dependent on angle or that include angular information are described.

## 2.5.1 Angular Biasing with Population Control Methods

### 2.5.1.1 AVATAR

The AVATAR [17, 46] (Automatic Variance and Time of Analysis Reduction) method generates three-dimensional, space-, energy- and angle-dependent weight windows for Monte Carlo. The implementation of AVATAR by the authors uses a relatively course-mesh and few-angle deterministic calculation in THREEDANT, approximating the angular flux as a function of the scalar flux, and then subsequently passes those flux values through a postprocessing code, Justine, to generate weight windows for MCNP [8]. The AVATAR approach to determining the angular flux uses an approximation of the angular flux based on the maximum entropy distribution, which is briefly summarized in the next few paragraphs.

#### Information Theory

First, for a given set of discrete values  $x_i$ ,  $i = 1, 2, \dots, n$  that are passed to a function,  $f(x)$ , the expectation value of that function is given by

$$\langle f(x) \rangle = \sum_{i=1}^n p_i f(x_i). \quad (2.53)$$

For the probability distribution  $p_i = p(x_i)$ ,  $i = 1, 2, \dots, n$ , the entropy of  $p$  is defined as

$$H(p) = -K \sum_i p_i \ln p_i, \quad (2.54)$$

where  $K$  is a positive constant. A proof that this is indeed the associated maximal entropy associated with all  $p_i$  is given in [47]. For a continuous probability density function  $p(x)$  over the interval  $I$ , the entropy of the continuous function is

$$H(p) = -K \int_I p(x) \ln p(x) dx. \quad (2.55)$$

To maximize either of these distributions, while also maintaining that  $\sum p_i = 1$ , one can use Lagrangian multipliers  $\lambda$  and  $\mu$

$$p_i = e^{-\lambda - \mu f(x_i)}. \quad (2.56)$$

This set of equations can be solved using

$$\langle f(x) \rangle = -\frac{\partial}{\partial \mu} \ln Z(\mu), \quad (2.57a)$$

and

$$\lambda = \ln [Z(\mu)], \quad (2.57b)$$

where

$$Z(\mu) = \sum_i e^{-\mu f(x_i)}. \quad (2.57c)$$

Jaynes [47, 48] showed that the maximum entropy probability distribution function corresponding to the previous equations is given by

$$p_i = \exp \left[ - \left( \lambda_0 + \lambda_1 f_1(x_1) + \cdots + \lambda_m f_m(x_i) \right) \right], \quad (2.58)$$

and the entropy of this distribution is given by

$$S_{max} = \lambda_0 + \lambda_1 \langle f_1(x) \rangle + \cdots + \lambda_m \langle f_m(x) \rangle. \quad (2.59)$$

In this case, the constant  $K$  from Eq. (2.54) has been set to 1.

The maximum entropy approach to calculating a probability distribution function is an attractive option given limited information about that distribution. This method's power lies in that it deduces a function given limited information, but does not place too great of an importance on missing or unwarranted information. Furthermore, a distribution ascertained from this methodology will encompass all distributions with smaller entropies that satisfy the same constraints. Thus, the method provides the most widely applicable probability distribution function for the system that has been defined.

Moskalev showed that by using the maximum entropy approach, a distribution function could be reconstructed from a (truncated) Legendre expansion [49]. This is particularly applicable to radiation transport because scattering terms are often truncated Legendre expansions. In his application, Moskalev derived a generalized form of reconstructing a probability distribution from a truncated expansion, where the true function represented by a Legendre polynomial series,

$$f(L, \mu) = \sum_{l=0}^L -\frac{2l+1}{2} f_l P_l(\mu), \quad (2.60)$$

could be associated with an adjusted function (obtained from maximizing the entropy of the known values),

$$\tilde{f}(L, \mu) = \exp \left( \sum_{l=0}^L \lambda_l P_l(\mu) \right), \quad (2.61)$$

such that

$$(f, P_l) = (\tilde{f}, P_l); \quad l = 0, \dots, L. \quad (2.62)$$

Here,  $\lambda_l$  are the Lagrange multipliers,  $\tilde{f}$  and  $f$  are  $\in \phi$ , and are assumed to be a function of  $\mu$  such that  $f(\mu) \geq 0, \mu \in [-1, 1]$ . These generalized equations were then applied to group-to-group scattering probability distribution functions, as well as reconstructing a  $L = 3$  function. The reconstruction showed agreement except near the extrema of  $\mu$ .

Walters and Wareing [51, 50] showed that the angle-dependent source definition for a discrete ordinates transport problem could be calculated using Moskalev's approach [49]. In their method, they used this approach to reconstruct the source distribution of particles in each cell from the source moments. For standard methods, the source in a cell expanded in Legendre moments is

$$S_m(x) = S_{m,j} \left[ P_0(x) + \frac{S_{m,j}^x}{S_{m,j}} P_1(x) \right], \quad (2.63)$$

where  $S_{m,j}$  is the average source in cell  $j$ , direction  $m$ ,  $S_{m,j}^x$  is the  $P_1(x)$  moment of the source, and the  $P_0$  and  $P_1$  are the associated Legendre polynomials. Using a normalized source distribution  $s_m(x)$  where

$$S_m(x) = s_m(x) S_{m,j},$$

and the normalized distribution is

$$s_m(x) = [s_0 + s_1 P_1(x)]. \quad (2.64)$$

In this equation,  $s_0$  and  $s_1$  are the zeroth and first Legendre moments of the source, respectively. The source distribution derived from the maximum entropy distribution is

$$\tilde{s}(x) = \frac{\lambda_{1,k}}{\sinh(\lambda_{1,j})} e^{\lambda_{1,j} P_1(x)}. \quad (2.65)$$

$\tilde{s}$  has normalized Legendre moments  $s_0$  and  $s_1$  that match  $s_m(x)$ . Because  $\tilde{s}$  satisfies the information that can be obtained about  $s_m$ , it can be used to reconstruct  $S_m(x)$ :

$$S_m(x) = \tilde{s}_m(x) S_{m,j}. \quad (2.66)$$

$\lambda_{1,j}$  can be found with

$$s_1 = 3 \left[ \coth(\lambda_{1,j}) - \frac{1}{\lambda_{1,j}} \right]. \quad (2.67)$$

It should be noted that the same methodology that Walters and Wareing use to reconstruct the source distribution from the source moments can be used to reconstruct the angular flux in cells based on moments of the angular flux (i.e. the scalar flux and current) [51].

In their paper, Walters and Wareing [50] suggested that in place of solving Eq. (2.67) for  $\lambda_{1,j}$ , that a rational polynomial be used in its place to reduce computational time. The suggested polynomial for  $0 \leq \lambda_{1,j} \leq 5$  is:

$$\lambda_{1,j} = \frac{2.99821(\frac{s_{1,j}}{3}) - 2.2669248(\frac{s_{1,j}}{3})^2}{1 - 0.769332(\frac{s_{1,j}}{3}) - 0.519928(\frac{s_{1,j}}{3})^2 + 0.2691594(\frac{s_{1,j}}{3})^3}, \quad (2.68)$$

and for  $\lambda \geq 5$ :

$$\lambda_{1,j} = \frac{1}{1 - \bar{\mu}}. \quad (2.69)$$

A full derivation of Eq. (2.67) and how it satisfies the maximum entropy requirements can be found in Appendix A of Ref. [50].

In their application, Walters and Wareing found that this method was accurate over a fairly coarse mesh for the problems analyzed, and the computed fluxes remained positive over the solution space. When compared to other methods, this approach performed much better on coarse meshes. However, the analysis was limited to 1D problems. As mesh size grew finer, the method performed similarly to other methods. Near vacuum boundary conditions,  $\lambda_{1,j} \rightarrow \infty$  at the cell boundary, causing issues in calculating the flux in these cells.

### AVATAR Implementation

AVATAR uses a deterministically-obtained solution of the adjoint scalar flux and adjoint currents to reconstruct the angular flux distribution. The angular flux distribution is then used to generate weight windows. AVATAR built off of the methodology described by Walters and Wareing [51, 50], but instead of reconstructing the source distribution inside the cell, the maximum entropy method was used to reconstruct the angular fluxes. Thus the angular flux,  $\psi$ , was reconstructed with the scalar flux,  $\phi$ , and the current,  $J$ .

AVATAR avoided generating explicit angular fluxes with THREEDANT by assuming that the adjoint angular flux is symmetric about the average adjoint current vector,  $J^\dagger$ :

$$\psi^\dagger(\hat{\Omega}) = \psi^\dagger(\hat{\Omega} \cdot n), \quad (2.70a)$$

where

$$n = \frac{J^\dagger}{\|J^\dagger\|}. \quad (2.70b)$$

Note that  $n$ ,  $J$ ,  $\psi$ , and  $\phi$  all have implied dependence on  $(\vec{r}, E)$ . The angular flux could then be reconstructed assuming that the angular flux is a product of the scalar flux and some angle-dependent function

$$\psi^\dagger(\hat{\Omega} \cdot n) = \phi^\dagger f(\hat{\Omega} \cdot n). \quad (2.70c)$$

Note that Eq. (2.70c) takes the form of Eq. (2.66). Thus  $f$  is derived from the maximum entropy distribution:

$$f(\hat{\Omega} \cdot n) = \frac{\lambda e^{(\hat{\Omega} \cdot n)\lambda}}{2 \sinh \lambda}, \quad (2.70d)$$

and  $\lambda$  is a function of the average cosine  $\bar{\mu}$

$$\begin{aligned} \lambda &= \frac{2.99821\bar{\mu} - 2.2669248\bar{\mu}^2}{1 - 0.769332\bar{\mu} - 0.519928\bar{\mu}^2 + 0.2691594\bar{\mu}^3} \\ &= \frac{1}{1 - \bar{\mu}} \end{aligned} \quad (2.70e)$$

for  $0 \leq \bar{\mu} < 0.8001$  and  $0.8001 \leq \bar{\mu} < 1.0$ , respectively. Also,  $\mu$  is given by

$$\bar{\mu}(\vec{r}, E) = \frac{\|J^\dagger(\vec{r}, E)\|}{\phi^\dagger(\vec{r}, E)}. \quad (2.70f)$$

Equations (2.70e) and (2.70f) are exact in both isotropic and streaming conditions [17].

Using the calculation of the angular flux described in Eqs. (2.70a) through (2.70f), angle-dependent weight windows can be constructed. AVATAR's space- energy- and angle-dependent weight window is given by

$$\bar{w}(\vec{r}, E, \hat{\Omega}) = \frac{k}{\phi^\dagger(\vec{r}, E) f(\hat{\Omega} \cdot n)}, \quad (2.71)$$

where  $k$  is a constant that can be adjusted to match the source distribution. In the case of AVATAR,  $k$  was used as a normalization factor to ensure that source particles are born with weights within the weight window. AVATAR exclusively generated weight windows, and did not attempt to consistently bias the source distribution. Physically, the assumption behind AVATAR is that the adjoint angular flux is locally one-dimensional, so azimuthal symmetry is assumed.

## AVATAR Results

The authors of AVATAR showed that AVATAR's angularly-dependent weight windows improved the FOM (from 5x to 7x) for a multiple-tally well-logging problem compared to the MCNP weight window generator. AVATAR was also compared to other methods in subsequent papers [16]. In an update of the MCNP weight window generator, AVATAR was compared to variants of the weight window generator and had a FOM of 79 while variants of the weight window generator had FOMs ranging from 105 to 119 [16]. However, the MCNP weight window generator required multiple iterations of Monte Carlo transport to converge on weight window values while AVATAR did not. Total runtimes for iteratively converging on weight window values were in the 200 to 300 minute range, while AVATAR took roughly 5 minutes to converge on weight window values for the problem. Whether these calculations were performed in serial or parallel were not discussed.

The MCNP weight window generator has also been adapted to use weight window values seeded by a solution from AVATAR [16]. This method had FOMs comparable to the default MCNP weight window generator, but only required 1 iteration to converge rather than 3. This reduced the total transport runtime from roughly 260 minutes to 140 minutes, but still required user experience and input to adequately set up and prepare the deterministic input for AVATAR.

The method used by AVATAR to produce angle-dependent weight windows successfully incorporated angular information into variance reduction parameters for Monte Carlo with very little additional computational burden. However, because AVATAR was not fully automated, the user had to have knowledge on the use of the  $S_N$  deterministic solver in addition

to the Monte Carlo methods they were trying to optimize. As a result, the user needed to adequately prepare the deterministic inputs, correctly specify the adjoint source for the deterministic solve, and then pass these values to postprocessing software [52, 16]. The FOMs reported with AVATAR did not incorporate the additional time required for user setup and preparation of inputs. Though this is not a customary time inclusion, the burden of time for this process may be significant. Though more computationally efficient than the weight window generator, this aspect of AVATAR may be too substantive of an obstacle for new-user approachability. Further, it leaves more room for user-induced error.

The AVATAR method [17, 46] used an approximation of the angular flux—without explicitly calculating it—to generate angle-dependent weight windows. It operated with the approximation that the angular flux was separable and symmetric about the average current vector. The angular flux was then calculated using a product of a deterministically-obtained scalar flux and an exponential function, derived from the maximum entropy distribution, that was a function of the scalar flux and the current. Space-, energy-, and angle-dependent weight windows for the Monte Carlo problem were then generated from the inverse of the angular flux. AVATAR improved the FOM for sample problems from 2 to 5 times, but did not apply to problems where the flux was not azimuthally symmetric.

### 2.5.1.2 Simple Angular CADIS

Simple Angular CADIS [52] is built on the theory of CADIS and FW-CADIS, but incorporates angular information in the methods. Simple Angular CADIS does so without explicitly using angular flux solutions from the deterministic solution. Instead, the method reconstructs the angular flux in the same manner employed by AVATAR, and additionally consistently biases the source distribution with the weight windows using the same methodology as CADIS and FW-CADIS. Recall that the original implementation of AVATAR did not have consistent source biasing. In their work, Peplow et al. implemented simple angular CADIS in MAVRIC, a hybrid methods software package distributed with the SCALE codebase [27]. The Simple Angular CADIS method was implemented with two different approaches to variance reduction: directionally-dependent weight windows with directionally-dependent source biasing and directionally-dependent weight windows without directional source biasing.

#### Theory

The Simple Angular CADIS approach, like AVATAR, uses a reconstruction of the angular flux derived from the maximum entropy distribution (Section 2.5.1.1). In Simple Angular CADIS, the authors approximate the adjoint angular flux such that

$$\psi^\dagger(\vec{r}, E, \hat{\Omega}) \cong \phi^\dagger(\vec{r}, E) \frac{1}{2\pi} f(\hat{\Omega} \cdot \hat{n}), \quad (2.72)$$

where  $f(\hat{\Omega} \cdot \hat{n})$  is given by the same Eqs. (2.70d), (2.70e), (2.70f) as AVATAR. Note that this differs from AVATAR's reconstruction of the angular flux, Eq. (2.70a), by a factor of  $1/2\pi$ .

As it was only dependent on  $\mu$ , AVATAR's original approach assumed azimuthal symmetry, but did not incorporate any factor of integration into the angular flux reconstruction. By including the azimuthal integration factor of  $1/2\pi$ , this version of  $\psi^\dagger$  satisfies

$$\phi^\dagger(\vec{r}, E) = \int \phi^\dagger \frac{1}{2\pi} f(\hat{\Omega} \cdot \hat{n}) d\hat{\Omega}.$$

The corresponding angle-dependent weight windows are then given by:

$$\bar{w}(\vec{r}, E, \hat{\Omega}) = \frac{2\pi k}{\phi^\dagger(\vec{r}, E) f(\hat{\Omega} \cdot \hat{n})}. \quad (2.73)$$

For the variant method with directionally-dependent weight windows and without directional source biasing, the biasing parameters are given by Eqs. (2.74). The biased source distribution,  $\hat{q}(\vec{r}, E, \hat{\Omega})$ , is given by a combination of the standard CADIS biased source,  $\phi^\dagger(\vec{r}, E)$  and the original directional source distribution,  $q(\hat{\Omega} \cdot \hat{d})$  such that

$$\begin{aligned} \hat{q}(\vec{r}, E, \hat{\Omega}) &= \frac{1}{R} q(\vec{r}, E) \phi^\dagger(\vec{r}, E) \frac{1}{2\pi} q(\hat{\Omega} \cdot \hat{d}) \\ &= \hat{q}(\vec{r}, E) \frac{1}{2\pi} q(\hat{\Omega} \cdot \hat{d}). \end{aligned} \quad (2.74a)$$

The direction  $\hat{d}$  is sampled using the original directional source distribution  $q(\hat{\Omega} \cdot \hat{d})$ . The birth weight matches standard CADIS with

$$\begin{aligned} w_0(\vec{r}, E, \hat{\Omega}) &= \frac{q(\vec{r}, E, \hat{\Omega})}{\hat{q}(\vec{r}, E, \hat{\Omega})} \\ &= \frac{R}{\phi^\dagger(\vec{r}, E)}, \end{aligned} \quad (2.74b)$$

and the weight window target value is given by

$$\begin{aligned} \bar{w}(\vec{r}, E, \hat{\Omega}) &= \frac{R}{\phi^\dagger(\vec{r}, E)} \frac{f(\hat{\Omega}_0 \cdot n(\vec{r}_0, E_0))}{f(\hat{\Omega} \cdot n)} \\ &= \bar{w}(\vec{r}, E) \frac{f(\hat{\Omega}_0 \cdot n(\vec{r}_0, E_0))}{f(\hat{\Omega} \cdot n)}. \end{aligned} \quad (2.74c)$$

Note that the biased source distribution,  $\hat{q}(\vec{r}, E, \hat{\Omega})$ , is a function of the biased source distribution from standard space- energy-CADIS and of the original directional source distribution. This is why this method has directional weight windows, but not directional source biasing. For the second method, with directionally-dependent weight windows and with directional source biasing, the biasing parameters are given by the equations summarized in Eqs. (2.75). The biased source distribution is given by a combination of the space-energy biased source

distribution, the original directional source distribution, and a directionally-dependent biased source distribution,  $f(\hat{\Omega} \cdot \hat{n}_0)$ , such that

$$\begin{aligned}\hat{q}(\vec{r}, E, \hat{\Omega}) &= \frac{1}{Rc} q(\vec{r}, E, \hat{\Omega}) \phi^\dagger(\vec{r}, E, \hat{\Omega}) \\ &= \left[ \frac{1}{R} q(\vec{r}, E) \phi^\dagger(\vec{r}, E) \right] \left[ \frac{1}{c} \frac{1}{2\pi} q(\hat{\Omega} \cdot \hat{d}) \frac{1}{2\pi} f(\hat{\Omega} \cdot n_0) \right] \\ &= \hat{q}(\vec{r}, E) \left[ \frac{1}{c} \frac{1}{2\pi} q(\hat{\Omega} \cdot \hat{d}) \frac{1}{2\pi} f(\hat{\Omega} \cdot n_0) \right].\end{aligned}\tag{2.75a}$$

The constant  $c$  is given by

$$c = \int \frac{1}{2\pi} q(\hat{\Omega} \cdot \hat{d}) \frac{1}{2\pi} f(\hat{\Omega} \cdot n_0) d\hat{\Omega}.\tag{2.75b}$$

The birth weights are also a function of direction, where

$$\begin{aligned}w_0(\vec{r}, E, \hat{\Omega}) &= \frac{q(\vec{r}, E, \hat{\Omega})}{\hat{q}(\vec{r}, E, \hat{\Omega})} \\ &= \frac{R}{\phi^+(\vec{r}, E)} \frac{2\pi c}{f(\hat{\Omega} \cdot n_0)},\end{aligned}\tag{2.75c}$$

as are the target weights

$$\begin{aligned}\bar{w}(\vec{r}, E, \hat{\Omega}) &= \frac{R}{\phi^\dagger(\vec{r}, E)} \frac{2\pi c}{f(\hat{\Omega} \cdot n_0)} \\ &= \bar{w}(\vec{r}, E) \frac{2\pi c}{f(\hat{\Omega} \cdot n)}.\end{aligned}\tag{2.75d}$$

Details about how the aforementioned equations were practically implemented are detailed in Ref. [52]. The motivated reader may explore this reference for details on the calculation of  $\lambda$ ,  $\bar{\mu}$ ,  $\|J^\dagger(\vec{r}, E)\|$ , and  $f(\hat{\Omega} \cdot \hat{n}_0)$

## Results

To test these two modifications of CADIS, the authors ran a number of test problems and compared them against standard implementations of CADIS and analog Monte Carlo runs. For a spherical boat test problem, Simple Angular CADIS without directional biasing improved the FOM by a factor of 2 to 3. Note that because the source is monodirectional, directional source biasing was not compared. Simple Angular CADIS with- and without-directional source biasing improved the FOM for active interrogation sample problems and for simple duct streaming problems. The methods did not improve the FOMs for sample problems using a neutron porosity tool or a gamma-ray litho-density tool.

The range in performance for angle-dependent problems was explained by the authors as a failure of the angular flux approximation to capture the true distribution of the angular flux. Because Simple Angular CADIS uses the same approximation in calculating the angular flux (Eq. (2.72)) as AVATAR, it is limited in the types of anisotropy that it can capture. As a result, the biasing parameters for a problem are unlikely to adequately reflect the flux distribution in problems where the flux is not captured effectively by the  $P_1$  expansion.

The authors also noted that because the weight window is dependent on space/energy/angle, the source birth weights only matched the weight window target values at a specific point in the weight window region. If the weight window covered a substantial region of phase-space, this could result in particle birth weights that do not adequately correspond to the importance of the region that they are born into, resulting in increased runtime and a more computationally-intensive calculation.

### 2.5.1.3 Cooper's Weight Windows

Cooper and Larsen, in addition to generating global isotropic weight windows from a deterministic forward solution (as described in Section 2.4.1), also developed angle-dependent weight windows [36]. Here, the forward angular flux is calculated in a similar manner as the AVATAR method, where the angular flux is a product of the scalar flux and an angle-dependent function. In this case, the adjustment factor also includes a factor of  $4\pi$ ,

$$\psi(\vec{r}, \hat{\Omega}) \approx A(\vec{r})e^{\vec{B}(\vec{r}) \cdot \hat{\Omega}}, \quad (2.76a)$$

where  $A(\vec{r})$  and  $\vec{B}(\vec{r})$  are given by:

$$A(\vec{r}) = \frac{\phi(\vec{r})}{4\pi} \frac{B(\vec{r})}{\sinh B(\vec{r})} \quad (2.76b)$$

$$\vec{B}(\vec{r}) = B(\vec{r}) \frac{\vec{\lambda}(\vec{r})}{|\vec{\lambda}(\vec{r})|} \quad (2.76c)$$

and

$$\lambda(\vec{r}) = \coth B(\vec{r}) - \frac{1}{B(\vec{r})}. \quad (2.76d)$$

If both  $A(\vec{r})$  and  $\vec{B}(\vec{r})$  are inserted into the equation for  $\psi(\vec{r}, \hat{\Omega})$ , Eq. (2.76a), the formulation will be very similar to AVATAR's reconstruction of the angular flux. However, Cooper's method differs from AVATAR in the calculation of  $\lambda(\vec{r})$ . Cooper noted that  $\lambda(\vec{r})$  could be estimated with either the scalar fluxes and currents from a fairly low-cost quasidiffusion calculation,

$$\begin{aligned} \lambda_i(\vec{r}) &= \frac{J_i(\vec{r})}{\phi(\vec{r})} \\ &= \frac{1}{\Sigma_{tr}(\vec{r})\phi(\vec{r})} \frac{\partial}{\partial r_j} E_{ij}(\vec{r})\phi(\vec{r}), \end{aligned} \quad (2.76e)$$

or with the scalar fluxes and currents directly from the Monte Carlo solution (recall that  $E_{ij}(\vec{r})$  is the Eddington factor described in Section 2.4.1). Cooper noted that because Monte Carlo robustly calculates these values, it is the more optimal choice. After obtaining these values from the deterministic calculation, Cooper's angle-dependent weight window could be calculated with

$$ww_{i,j}(\hat{\Omega}) = \frac{\psi_{i,j}(\vec{r}, \hat{\Omega})}{\max \phi_{i,j'} / 4\pi}. \quad (2.76f)$$

As mentioned in Section 2.4.1, Cooper's method was limited in that it used an iterative quasidiffusion / Monte Carlo solution to generate the biasing parameters for the problem. This method was not automated; and the ideal frequency between iterations was never explored. However, Cooper showed in two-dimensional example problems that the angularly-dependent weight windows significantly improved the figure of merit as compared to analog Monte Carlo. The distributions of the FOM and the resulting tally were also much smoother with the approach described in their work. Further, the angular weight windows performed slightly better than the isotropic weight windows in evenly distributing the particles, even in problems where the anisotropy was not significant. However, like AVATAR, this method is limited in the types of anisotropy it can quantify due to the approximations it uses to reconstruct the angular flux. In generating the estimates for  $\vec{\lambda}$ , the authors found that using a quasidiffusion estimate was more efficient than using Monte Carlo estimates, likely because the estimates of the factors could be periodically updated as the solution iteratively converged.

## 2.5.2 Angular Biasing Using the Exponential Transform

### 2.5.2.1 Early Work

As discussed in Section 2.1.2, the exponential transform is a modified sampling method that adjusts the distance-to-collision in Monte Carlo transport to encourage particle transport in preferential regions. This is done by modifying the sampled cross section. Recall from Eq. (2.21) that the exponential transform is dependent on a transform parameter  $p$  and the cosine angle  $\mu$ , such that  $\Sigma_t^* = \Sigma_t(1 - p\mu)$ . When used without angle biasing,

$$\psi_g^\dagger(r, \Omega) \approx e^{\Sigma_t \lambda \cdot r}, \quad (2.77)$$

the exponential transform can have undesirable weight fluctuations [2], especially as the number of collisions to reach a tally increases [53]. Eq. (2.77) shows that the importance function (the adjoint flux) can be approximated as an exponential function varying in space, dependent on the total cross section  $\Sigma_t$ , distance traveled  $r$ , and a parameter defining the amount and direction of biasing  $\lambda$ .

Dwivedi [54] showed that by adding an angle-dependent collision biasing scheme in addition to the exponential transform, the problematic weight fluctuations could be mitigated.

The collision biasing scheme introduced with the exponential transform takes the form

$$\psi_g^\dagger(r, \Omega) \approx \frac{\sigma_{s,0} e^{\Sigma_t \lambda \cdot r}}{4\pi \sigma_t (1 - \lambda \cdot \Omega)}. \quad (2.78)$$

Note that the ratio of cross sections outside of the exponential function  $\sigma_{s,0}/\sigma_t$ , where  $\sigma_{s,0}$  is the zeroth moment of the scattering cross section, the ratio of the cross sections is the survival probability in an interaction event, and the  $(1 - \lambda \cdot \Omega)$  term is consistent with the weight adjustment required for the exponential transform (Eq. (2.1.2)). This was applied to a monoenergetic problem with slab geometry and isotropic scattering, and the variance was reduced by a factor of more than 100 when compared with other exponential transform methods.

Gupta and Dwivedi's subsequent work [53] adjusted the factor described in the preceding paragraph by applying the exponential transform with angle biasing to deep-penetration problems with anisotropic scattering. The authors did not explicitly use the true distribution for anisotropic scattering, but rather chose to approximate the biased kernel to be a function of the isotropic angular distribution. The authors observed a reduction in the variance by a factor of 10, but they acknowledged that, while the combination of the biased kernel and exponential biasing reduced weight fluctuations, it also had the potential to introduce other weight fluctuations due to anisotropies in the flux.

Ueki and Larsen [55] generalized Dwivedi's importance transform and applied it to isotropic, linearly anisotropic, and quadratically anisotropic scattering. They observed that Dwivedi's method and the generalized Dwivedi method outperformed non-angle-dependent exponential biasing for all types of scattering, and that their generalized method outperformed Dwivedi's original method in higher order scattering. The work of Dwivedi, Gupta, Ueki and Larsen was applied and each compared with one-dimensional sample problems. Ueki and Larsen pointed out that their method could be extended to three-dimensional problems using Turner and Larsen's methodology (described in Section 2.5.2.2) [55].

In 1985, Henricks and Carter [56] described a method by which photon transport could be biased in angle with an exponential transform adjustment factor. In this study, the authors performed studies on three test problems with the exponential transform adjustment factor and with a synergistic angular bias and exponential transform adjustment. In all studies, the synergistic biasing outperformed the exponential transform adjustment alone. However, their method performed best in highly absorbing media. The authors noted that this performance was due to the fact that the biasing could be strong without undersampling scattering in the problem. They also pointed out that, while the weight window method was comparable in efficiency to the method described, their method avoided choosing importances and weight window values for biasing. Their method was limited to exclusively photon transport biasing, and not neutron transport. However, the authors were optimistic that the method could be extended to neutron transport with relative ease. Both, Niemal, and Vergnaud [57] also derived VR parameters for the exponential transform and for collision biasing based on the adjoint solution as a measure of importance.

### 2.5.2.2 LIFT

The LIFT [58, 59] method developed by Turner and Larsen, and like Dwivedi's exponential transform, is a modification of the zero variance solution (see Section 2.2.3). Consequently, the LIFT method uses a calculation of the adjoint flux as a measure for importance in the problem to distribute particles according to the contribution density in the problem. LIFT uses a deterministic calculation to generate biasing parameters for the exponential transform and weight window variance reduction techniques.

As with the form of the importance function derived by Dwivedi (Eq. (2.78)), the LIFT method generates an angle-dependent importance function by taking the product of a space-based exponential function and an angle-informed collision estimator. Additionally, LIFT uses a deterministic calculation of the adjoint scalar flux to inform the angular flux reconstruction. The adjoint angular flux is approximated as piecewise continuous in space and angle with Eqs. (2.79a) through (2.79d):

$$\psi_{g,n}^\dagger(r, \Omega) \approx \phi_{g,n}^\dagger V_n \left[ \beta_{g,n} \frac{\sigma_{s_0, g \rightarrow g, n} b_{g,n}(\Omega)}{\sigma_{t, g, n} - \rho_{g,n} \cdot \Omega} e^{\rho_{g,n} \cdot (r-r_n)} \right], \quad (2.79a)$$

where the physical system is comprised of  $N$  regions of volume  $V_n$ , and  $\psi_{g,n}^\dagger$  is the approximation of the angular flux for group  $g$  and region  $n$ . Further,  $\beta$ , the normalization factor, is given by:

$$\beta_{g,n} = \frac{1}{\int_{V_n} e^{\rho_{g,n} \cdot (r-r_n)} dr \int_{4\pi} \frac{\sigma_{s_0, g \rightarrow g, n} b_{g,n}(\Omega)}{\sigma_{t, g, n} - \rho_{g,n} \cdot \Omega} d\Omega}; \quad (2.79b)$$

$b_{g,n}$ , the linearly anisotropic factor, is

$$b_{g,n}(\Omega) = 1 + 3\mu_{g \rightarrow g, n} \frac{\sigma_{t, g, n} - \sigma_{s_0, g \rightarrow g, n}}{|\rho_{g,n}|^2} \rho_{g,n} \cdot \Omega; \quad (2.79c)$$

and the biasing parameter  $\rho_{g,n}$  is given by the product of the cross section and the biasing parameter  $\lambda$  seen previously in Eqs. (2.77) and (2.78),

$$\rho_{g,n} = \sigma_{t, g, n} \lambda_{g,n}. \quad (2.79d)$$

Turner showed that  $\rho_{g,n}$  can be obtained from the deterministic solution to the adjoint equation, rather than from the cross section and  $\lambda$ , which requires some assumptions on the distribution of particles. Instead, Turner showed that  $\rho$  can be found in terms of the deterministic scalar fluxes, where

$$\rho_{x, g, n} = \frac{1}{\Delta x_n} \ln \left( \frac{\phi_{g, i+1/2}^\dagger}{\phi_{g, i-1/2}^\dagger} \right) \quad (2.80a)$$

$$\rho_{y, g, n} = \frac{1}{\Delta y_n} \ln \left( \frac{\phi_{g, j+1/2}^\dagger}{\phi_{g, j-1/2}^\dagger} \right) \quad (2.80b)$$

and

$$\rho_{z,g,n} = \frac{1}{\Delta z_n} \ln \left( \frac{\phi_{g,k+1/2}^\dagger}{\phi_{g,k-1/2}^\dagger} \right) \quad (2.80c)$$

are all defined using cell-edge flux values in Cartesian coordinates.

Eq. (2.79a) is an adjustment of the exponential transform described by Dwivedi [54]. However, rather than relying upon an isotropic scattering law, like earlier implementations of the exponential transform, the LIFT method adjusts the transform to instead be linearly anisotropic in angle. The derivation of this equation for both linearly anisotropic scattering and isotropic scattering is available in [58]. To summarize: the parameters  $\beta_{g,n}$ ,  $b_{g,n}$ , and  $\rho_{g,n}$  are calculated from values obtained from the deterministic calculation and are used to calculate  $\psi_{g,n}^\dagger$ .

In addition to using the exponential transform to bias the particles in angle, the LIFT method also uses weight windows for particle weight adjustment. However, the computational cost of generating angle-dependent weight windows from the previous equations led the authors to choose space-energy exclusive weight windows. The weight window target values were calculated to be inversely proportional to the adjoint solution, as with other methods

$$ww_{center,g,n} = \frac{\phi_{g,src}^\dagger}{\phi_{g,n}^\dagger}. \quad (2.81)$$

The LIFT method [58, 59], like AVATAR, calculated the angular flux for a region by assuming the angular flux was a product of the scalar flux and an exponential function. The angular flux values were then used to generate values for the exponential transform variance reduction technique to bias the particles in space, energy, and angle. Like AVATAR, LIFT also generated weight window parameters. However, generating a full angle-dependent weight window map and running Monte Carlo transport with those weight windows was computationally limiting, and the authors chose to only generate space- and energy-dependent weight windows. Turner showed that LIFT outperformed AVATAR for several example problems, but both methods performed poorly in voids and low-density regions.

Turner compared a number of variants of LIFT [59] against AVATAR to determine the efficiency of LIFT. In his investigation, Turner compared LIFT and AVATAR using approximations for the adjoint solution with diffusion and  $S_N$  transport calculations, and with various methods to calculate weight window parameters, including using LIFT combined with AVATAR's weight window parameters. In most cases, LIFT outperformed AVATAR. In problems with voids and low-density regions, the efficiency of the LIFT method decreased, but so did AVATAR. This independently confirmed the findings of the previous study. However, an important note that Turner mentioned was that while increasing the accuracy of the deterministic solution may decrease the variance, it is not necessarily the best for the FOM. This is a valuable lesson for all automated variance reduction methods: an overly accurate solution for the adjoint problem may reduce the variance but come at such a high computational cost such that it decreases the FOM.

More recently, Keady and Larsen showed that LIFT could be improved upon further by using cell-averaged currents and fluxes rather than cell-edge values for angular biasing [60]. By using this modified variation of LIFT, material interfaces do not create strong flux discontinuities on cell edges, resulting in a solution that is both smoother and more realistic. Results were presented for a one-dimensional monoenergetic slab problem with material interfaces. The modified version of LIFT outperformed both the original LIFT method and Monte Carlo weight windows generated with forward deterministic weight windows.

## 2.6 Variance Reduction in Large Application Problems

Variance reduction methods exist for Monte Carlo methods to achieve a more accurate answer in a shorter amount of time. Automated variance reduction methods have been designed to aid users in generating variance reduction parameters where it might not be intuitive or obvious what variance reduction parameters are best for a problem. The most successful variance reduction methods construct or estimate an importance function for the desired response from a preliminary calculation. This importance function may be derived from the adjoint solution to the transport equation, or it may be derived from contribution theory.

The methods described in Sections 2.3 through 2.5 have been implemented and tested in a number of software packages. The problem spaces over which they have been applied is extensive, and show that a large subset of application problems can be successfully simulated with the assistance of existing variance reduction techniques. Local variance reduction methods can be used to reduce the variance in source-detector problems where the detector constitutes a small subset of the problem phase-space. Global variance reduction methods can be used to distribute response sampling equally throughout several tallies or a problem-wide tally. Angle-based variance reduction methods are used in problems where space- and energy- variance reduction methods alone are not sufficient. For large and complex problems, automated versions of each of these methods are required as the user expertise to obtain even remotely adequate parameters is significant. Here, the existing state of automated variance reduction methods and the applications on which they have been tested will be summarized.

Presently, numerous hybrid methods packages that use the methods described in the preceding sections are available. These packages are targeted towards deep-penetration radiation transport and shielding applications. The CADIS and FW-CADIS methods are distributed with MAVRIC [27, 40] and ADVANTG [61] from Oak Ridge National Laboratory (ORNL), which use the discrete ordinates code Denovo [62] to make VR parameters for the Monte Carlo codes Monaco [27] and MCNP[8], respectively. CADIS and FW-CADIS are also available in Tortilla [63], which uses the hybrid methods software using the deterministic code Attila [64]. Tortilla also includes a version of LIFT and LIFT-based weight windows. The Deterministic Adjoint Weight Window Generator (DAWWG) from Los Alamos National Laboratory (LANL) [65] uses the adjoint solution from a deterministic solve in PARTISN

[66] to generate biasing parameters for MCNP, and also includes AVATAR functionality. MCNP [8] is distributed with a weight window generator (WWG) that uses a preliminary Monte Carlo solution to estimate an importance function for the problem. Though this list is not exhaustive, it illustrates the present ubiquity and need for hybrid methods to analyze realistic problems. In the analysis of realistic problems, ensuring that a “good” answer is achieved is necessary for safety and security. In the next few paragraphs, how and how effectively each of these methods have been applied to application problems is summarized. The degree to which each is successful is also discussed.

CADIS and FW-CADIS have been used for a number of studies of spent fuel storage facilities. Radescleu et al. used FW-CADIS in MAVRIC to calculate spent fuel dose rates of a single dry cask with finely detailed geometry and spent fuel isotopic compositions [67]. Chen et al. used MAVRIC [27] to analyze dose rates on spent fuel storage containers [68]. The fueled region of the storage container was homogenized into an effective fuel region. They found that in a coarse energy group calculation (27G19N) MAVRIC underestimated neutron dose rates at high energies. However, MAVRIC’s ability to generate importances in three dimensions allowed it to have better problem-wide results, while the compared to methods (SAS4) struggled generating satisfactory results in the axial direction. This was demonstrated to a greater extent in an analysis of an independent spent nuclear fuel storage installation (ISFSI) [69] by Sheu et al. The FOM achieved by MAVRIC appeared inferior to those obtained with SAS4 or TORT/MCNP in a single cask. However, when applied to a storage bed of 30 casks MAVRIC was able to generate VR parameters at all, which were unfeasible for the other two methods. These studies demonstrated that CADIS and FW-CADIS are desirable methods for which to obtain global and three-dimensional variance reduction parameters for realistic problems.

ADVANTG [61], developed at ORNL [70, 35, 71] is a hybrid methods package for automated variance reduction of the Monte Carlo transport package, MCNP [11]. ADVANTG uses the deterministic transport code Denovo [62] to perform the forward and adjoint calculations for CADIS and FW-CADIS. At its inception, ADVANTG was used to analyze various threat-detection nonproliferation problems [61]. FOM improvements on the order of  $10^2$  to  $10^4$  when compared with analog Monte Carlo have been observed. However, Mosher et al. noted that the methods struggled with problems exhibiting strongly anisotropic behavior. In particular, they noted that low-density materials and strongly directional sources posed issues. This indicated that while CADIS and FW-CADIS are very useful methods, they have limitations in highly angle-dependent applications.

The deterministic adjoint weight window generator (DAWWG), utilizes the discrete ordinates code PARTISN [65] to generate space- energy- and angle-dependent weight windows. It is an internal feature of MCNP. The angle-dependent weight windows are calculated with the same methodology as AVATAR [65, 17]. Sweezy and colleagues compared DAWWG to the standard MCNP WWG on an oil well logging problem, a shielding problem, and a dogleg neutron void problem. The deterministic weight window generator obtained similar relative errors as the standard WWG for the first two problems, but in a fraction of the time. However, for the dogleg void problem, which exhibited strong angular dependence in the neutron

flux, the authors noted that DAWWG was not as effective as the standard MCNP WWG. This was attributed to ray effects from the  $S_N$  transport influencing the weight windows obtained by DAWWG, which is not an issue for the standard WWG.

A variety of automated variance reduction methods, including CADIS and LIFT have been implemented into the Attila / Tortilla deterministic and hybrid transport code packages [63]. These methods were used on several nonproliferation test problems. For the most part, LIFT and LIFT combined with weight windows outperformed CADIS' weight windows and source biasing, indicating that the addition of angular information was of benefit for these more realistic nonproliferation application problems.

Peplow et. al. formulated an adjustment to CADIS in the ORNL code suite [52] to incorporate angular information into the VR parameters (see Section 2.5.1.2. Two different methods to generate weight windows and source biasing parameters were investigated: CADIS with directional source biasing, and CADIS without directional source biasing. For the method without directional source biasing, the biased source distribution matched that of the original CADIS, but the weight window values were directionally-dependent. The method with directional source biasing used the transform function to obtain directionally-dependent weight windows and directional source biasing. Peplow and his colleagues found that these methods generally increased the FOM by a factor of 1-5 as compared to traditional CADIS, but in some cases decreased the FOM. This was attributed to the  $P_1$  approximation used to calculate the angular flux, which limited the physical applicability of the method, just as with AVATAR.

CADIS and FW-CADIS have shown to be the existing “gold standard” of local and global variance reduction methods for large application problems, a selection of which were described in the preceding paragraphs. These problems include active interrogation of cargo containers [61], spent fuel storage casks [68, 67] and beds [69], and other nonproliferation and shielding applications [63]. For additional applications, one may refer to [35]. In some of these application problems, the parameters generated by CADIS or FW-CADIS were sufficient for the problem application. However, for other problems that had strong angular dependence or geometric complexity, the parameters were insufficient [68, 63, 52]. This can be remedied with additional angular information in the variance reduction parameters, such as LIFT [63], but the benefits of consistent source biasing are lost in this case. Alternatively, the angular flux can be reconstructed in a manner similar to AVATAR [65, 52] to generate angle-dependent weight windows, but this approximates the angular flux to be linearly anisotropic in angle (from the  $P_1$  reconstruction), and is also dependent on the deterministic flux not having ray effects [65].

Although numerous methods have been proposed and implemented to obtain adequate angle-informed variance reduction parameters for application problems, they have limited applicability, and determining in which problems they will be useful is not always straightforward. No single method has been successful for problems with all types of anisotropy, and no existing angle-informed method captures the anisotropy in the flux without significant approximation. For large-scale, highly anisotropic, deep-penetration radiation transport problems, there exists a need for improvements in hybrid methods.