LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

LLNL-JRNL-432539

# A Benchmark Comparison of Monte Carlo Particle Transport Algorithms for Binary Stochastic Mixtures

P. Brantley

May 18, 2010

Journal of Quantitative Spectroscopy and Radiative Transfer

**Disclaimer**

This document was prepared as an account of work sponsored by an agency of the United States government. Neither the United States government nor Lawrence Livermore National Security, LLC, nor any of their employees makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States government or Lawrence Livermore National Security, LLC. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States government or Lawrence Livermore National Security, LLC, and shall not be used for advertising or product endorsement purposes.

# A Benchmark Comparison of Monte Carlo Particle Transport Algorithms for Binary Stochastic Mixtures

Patrick S. Brantley

*Lawrence Livermore National Laboratory*
*P.O. Box 808*
*Livermore, California 94551*
*brantley1@llnl.gov*

**Abstract**

We numerically investigate the accuracy of two Monte Carlo algorithms originally proposed by Zimmerman [1] and Zimmerman and Adams [2] for particle transport through binary stochastic mixtures. We assess the accuracy of these algorithms using a standard suite of planar geometry incident angular flux benchmark problems and a new suite of interior source benchmark problems. In addition to comparisons of the ensemble-averaged leakage values, we compare the ensemble-averaged material scalar flux distributions. Both Monte Carlo transport algorithms robustly produce physically-realistic scalar flux distributions for the benchmark transport problems examined. The base Monte Carlo algorithm reproduces the standard Levermore-Pomraning model [3, 4] results. The improved Monte Carlo algorithm generally produces significantly more accurate leakage values and also significantly more accurate material scalar flux distributions. We also present deterministic atomic mix solutions of the benchmark problems for comparison with the benchmark and the Monte Carlo solutions. Both Monte Carlo algorithms are generally significantly more accurate than the atomic mix approximation for the benchmark suites examined.

*Keywords:*
Monte Carlo particle transport, binary stochastic mixture, stochastic media

## 1. Introduction

Particle transport through binary stochastic mixtures has received significant research attention in the last two decades [3, 4]. Much of the research has focused on the development and analysis of approximate deterministic models for the solution of such particle transport problems. The most common approach to solving particle transport problems involving binary stochastic media is to use the atomic mix approximation [4] in which the transport problem is solved using ensemble-averaged material properties. The atomic mix approximation is appealing because of its simplicity and computational efficiency but may not be accurate enough depending on the details of the stochastic material properties. The most ubiquitous approximate deterministic model developed specifically for solving binary stochastic media transport problems is often referred to as the Levermore-Pomraning model or the Standard Model [3, 4].

The accuracy of the Levermore-Pomraning model has previously been examined by Adams, Larsen, and Pomraning [5] using a suite of benchmark problems involving a non-stochastic isotropic angular flux incident on one boundary of a one-dimensional planar geometry binary stochastic medium. The benchmark suite is characterized by nine different sets of material cross sections, mean material slab widths, and material scattering ratios as well as three different total slab widths. The distribution of material slab widths is assumed to be described by Markovian statistics [4] that are spatially homogeneous. The fiducial quantities for comparison are the ensemble-averaged reflection from and transmission through the slab. We refer to this standard suite of incident angular flux benchmark test problems as benchmark Suite I. These benchmark comparisons demonstrated that the Levermore-Pomraning model produces qualitatively correct and semi-quantitatively correct results for the reflection and transmission values. Zuchuat, Sanchez, Zmijarevic, and Malvagi [6] reproduced the Markovian statistics benchmark results published in Ref. [5] and extended these benchmark solutions to additional non-Markovian material statistics. In addition to the reflection and transmission values, Zuchuat et al. compared the ensemble-averaged total and material scalar flux distributions. These results

demonstrated that although the reflection and transmission values computed by the Levermore-Pomraning model were reasonably accurate, the scalar flux distributions could exhibit significantly larger pointwise errors.

A comparatively limited amount of research has been performed into the development of Monte Carlo algorithms for the solution of these binary stochastic mixture transport problems. Zimmerman [1] and Zimmerman and Adams [2] first proposed a Monte Carlo algorithm that is equivalent to the Levermore-Pomraning equations (Algorithm A) and another Monte Carlo algorithm that should possess increased accuracy (Algorithm B) as a result of improved local material realization modeling. Murata, Mori, and Nakagawa [7] examined the use of similar algorithms for modeling spherical fuels in high-temperature gas-cooled reactors. Donovan and Danon [8] applied the Levermore-Pomraning algorithm proposed by Zimmerman (Algorithm A) to the specific case of a two-dimensional binary stochastic mixture composed of circular disks of one material randomly distributed in a background matrix material. Donovan and Danon generally refer to these types of Monte Carlo algorithms as "chord length sampling" (CLS) algorithms. Donovan and Danon also examined a "limited chord length sampling" (LCLS) algorithm for their problem in which the particle transport through the disks is modeled explicitly, and the transport through the background material is treated using CLS (Algorithm A). Donovan and Danon did not examine for their specific problem the approach of using Algorithm B for both materials in the problem. Donovan and Danon extended the LCLS algorithm to the simulation of particle transport through a binary stochastic mixture of spheres of constant radius mixed in a three-dimensional matrix material [9] as well as to the simulation of a single spherical pebble fuel cell [10]. A recent paper by Ji and Martin [11] examined Monte Carlo chord length sampling algorithms for the neutronics analysis of Very High Temperature Gas-Cooled Reactors. These researchers have generally concluded that incorporating the stochastic nature of the background medium into the Monte Carlo algorithm via chord length sampling improves the accuracy of the modeling. To complete and extend the comparisons of Ref. [2], we focus in this paper on the algorithms proposed by Zimmerman [1] and Zimmerman and Adams [2] applied uniformly over a one-dimensional spatial domain. The accuracy of the detailed global scalar flux distributions produced by these chord length sampling algorithms has not been previously investigated using benchmark comparisons.

Zimmerman and Adams [2] numerically demonstrated that the base Monte Carlo algorithm (Algorithm A) solves the Levermore-Pomraning equations and that the improved algorithm (Algorithm B) is more accurate by comparing the results of these algorithms to the standard suite (Suite I) of planar geometry incident angular flux binary stochastic mixture benchmark transport solutions [5]. In this paper, we extend the incident angular flux benchmark comparisons of these Monte Carlo algorithms to include the ensemble-averaged material scalar flux distributions produced. Evaluating the accuracy of the material scalar flux distributions is important, because these distributions determine the reaction rates in the materials of the system. Also, as demonstrated in Ref. [6], one approximate deterministic model that gave accurate reflection and transmission probabilities produced unphysical scalar flux distributions for some problems with optically thick and purely scattering materials. These unphysical scalar flux distributions exhibited a local maximum in the interior of the spatial domain for problems with no interior sources such as in benchmark Suite I. Our numerical results demonstrate that for the Suite I benchmark transport problems, Algorithms A and B robustly produce physically-realistic scalar flux distributions. This result for Algorithm A is largely expected (although not previously numerically demonstrated for these benchmark problems), because that algorithm solves the Levermore-Pomraning equations; this result for Algorithm B has not previously been explicitly demonstrated for this benchmark suite. The scalar flux distributions produced by Algorithm B are generally more accurate than those produced by Algorithm A.

In this paper, we also investigate the accuracy of these Monte Carlo transport algorithms using a new related suite of one-dimensional planar geometry interior source benchmark problems that we refer to as Suite II. This suite of benchmark problems was recently used [12] to investigate the accuracy of the Levermore-Pomraning model for interior source benchmark problems. (Subsequent to the work reported in Ref. [12], we became aware of unpublished interior source benchmark investigations of the accuracy of the Levermore-Pomraning approximation by Vasques [13].) Because the interior source scalar flux distributions are of an inherently different character than the distributions obtained for the incident angular flux benchmark problems, the present benchmark comparison significantly extends the domain of problems for which the accuracy of these Monte Carlo models has been investigated. The material specifications for benchmark Suite II are the same as for Suite I.

The numerical solutions to both benchmark suites were obtained using the Monte Carlo procedures described by Adams et al. [5] and Zuchuat et al. [6] in which independent material realizations are sampled from the Markovian statistics and the transport problem is solved for each realization using the discrete ordinates transport method [14]

and a high-order angular quadrature set. $5 \times 10^5$ independent realizations were simulated and the results averaged to obtain ensemble-averaged values for the leakage from the slab. We also tabulate the ensemble-averaged material scalar flux distributions using the procedure described in Ref. [6]. For comparison with the results obtained using the Monte Carlo algorithms, we also present deterministic atomic mix solutions for the benchmark suites. This paper is an expanded version of recent conference reports [15, 16].

The remainder of this paper is organized as follows. In Section 2, we describe the benchmark transport problem that we use to assess the accuracy of the Monte Carlo algorithms. In Section 3, we outline the Monte Carlo algorithms proposed by Zimmerman [1] and Zimmerman and Adams [2] for the solution of particle transport through stochastic media. We then present the numerical comparisons of the Monte Carlo algorithms and the atomic mix approximation in Section 4. We give general conclusions and suggestions for future work in Section 5.

## 2. Benchmark Transport Problem Suites

In this section, we first describe the binary stochastic mixture benchmark transport problems we use to assess the accuracy of the Monte Carlo algorithms. We then describe the procedure used to generate the benchmark solutions.

### *2.1. Benchmark Transport Problem Description*

We consider the following two time-independent monoenergetic neutron transport problems [5, 12] with isotropic scattering in a one-dimensional planar geometry spatial domain defined on $0 \leq x \leq L$:

$$\mu \frac{\partial}{\partial x} \psi(x,\mu) + \sigma_t(x)\psi(x,\mu) = \frac{1}{2}\sigma_s(x) \int_{-1}^{1} \psi(x,\mu')\, d\mu' + \frac{1}{2} q(x) \ ,$$

$$0 \leq x \leq L \ , \ -1 \leq \mu \leq 1 \ , \tag{1}$$

$$q(x) = \begin{cases} 0 \ , & \text{Suite I} \ , \\ \frac{1}{L} \ , & \text{Suite II} \ , \end{cases} \quad 0 \leq x \leq L \ , \tag{2}$$

$$\psi(0,\mu) = \begin{cases} 2 \ , & \text{Suite I} \ , \\ 0 \ , & \text{Suite II} \ , \end{cases} \quad \mu > 0 \ , \tag{3}$$

$$\psi(L,\mu) = 0 \ , \ \mu < 0 \ . \tag{4}$$

Eqs. (1)–(4) are written in standard neutronics notation [14]: $\psi(x,\mu)$ is the angular flux of particles at position $x$ traveling with direction cosine $\mu$ with respect to the $x$ axis, $\sigma_t(x)$ and $\sigma_s(x)$ are the macroscopic total and scattering cross sections, respectively, and $q(x)$ is an interior source of particles. Because the cross sections are random variables, the angular flux is also a random variable. Suite I is the standard suite of incident angular flux benchmark problems originally examined in Ref. [5]. Suite II is the interior source benchmark problem recently examined in Ref. [12]. The interior source defined by Eq. (2) is non-stochastic and spatially uniform. For Suite I, the source is zero throughout the spatial domain. For Suite II, the interior source is constant in both materials and results in one neutron sourced into the medium per unit time. A stochastic interior source (varying with material) could be considered; the present non-stochastic source specification represents an initial effort toward considering interior source problems and simplifies somewhat the interpretation of the results. The boundary conditions given by Eqs. (3) and (4) are non-stochastic. For Suite I, these boundary conditions represent an isotropic incident angular flux with a unity incoming partial current at $x = 0$ and a vacuum boundary at $x = L$. For Suite II, the boundary conditions describe vacuum boundaries at both $x = 0$ and $x = L$.

The binary stochastic spatial medium is assumed to be composed of alternating slabs of two materials, labeled with the indices 0 and 1, with the mean material slab width for material $i$ denoted as $\Lambda_i$. The total and scattering cross sections for each material are uniform and are denoted as $\sigma_t^i$ and $\sigma_s^i$, $i = 0, 1$, respectively. The distribution of material slab widths in the planar medium is assumed to be described by spatially homogeneous Markovian statistics [4], in which case a slab width for material $i$, $\lambda_i$, can be sampled from an exponential distribution given by

$$f_i(\lambda_i) = \frac{1}{\Lambda_i} \exp\left(-\frac{\lambda_i}{\Lambda_i}\right) \ , \tag{5}$$

where again $\Lambda_i$ is the mean material slab width for material $i$. Given the mean material slab widths, the probability of finding material $i$ at any given point in the spatial domain, $p_i$, is given by

$$p_i = \frac{\Lambda_i}{\Lambda_0 + \Lambda_1} \quad . \tag{6}$$

This material probability corresponds to the volume fraction of the material in the problem.

The fiducial comparison quantities of interest are the ensemble-averaged exiting partial currents at $x = 0$ and $x = L$, defined respectively as

$$\langle J_0 \rangle = \int_{-1}^{0} |\mu| \langle \psi(0,\mu) \rangle \, d\mu \quad , \tag{7}$$

$$\langle J_L \rangle = \int_{0}^{1} \mu \langle \psi(L,\mu) \rangle \, d\mu \quad . \tag{8}$$

For the Suite I benchmark with a unity incoming partial current, these partial exiting currents represent, respectively, the probability of reflection from the slab and the probability of transmission through the slab. For the Suite II benchmark, the leakage at the left and right boundaries of the slab are identical in the limit of an infinite number of realizations, i.e. $\langle J_0 \rangle = \langle J_L \rangle$. (For the finite number of realizations used in the generation of the benchmark results, the leakage values at the two boundaries agreed to typically three digits.) We choose to simply compare the leakage from the slab at $x = 0$, $\langle J_0 \rangle$. Another more subtle motivation for choosing to compare to the leakage value at $x = 0$ is described in Section 2.2 below. In addition, we focus in our comparisons on the ensemble-averaged material scalar flux distributions, $\langle \phi_i(x) \rangle$, $i = 0, 1$, as these distributions determine reaction rates in the materials of the system.

The material parameters for the benchmark transport problems are given in Table 1 using the notation of Ref. [6] (rather than the original notation of Ref. [5]), where the scattering ratio for material $i$ is defined as $c_i = \sigma_s^i / \sigma_t^i$. These material parameters are common between both the Suite I and Suite II benchmark problems. The different case numbers (i.e. 1, 2, and 3) represent permutations of materials with mean material slab widths of optical depth 0.1, 1.0, and 10.0. The different case letters (i.e. a, b, and c) represent varying amounts of scattering for each material. For each set of material parameters (cases 1, 2, and 3), three sets of scattering ratio combinations (cases a, b, and c) and three slab widths ($L = 0.1$, 1.0, and 10.0) are considered for each benchmark suite. For all cases, the ensemble-averaged total cross section, defined as $\langle \sigma_t \rangle = p_0 \sigma_t^0 + p_1 \sigma_t^1$, is unity. The ensemble-averaged scattering cross section is defined as $\langle \sigma_s \rangle = p_0 \sigma_s^0 + p_1 \sigma_s^1 = p_0 c_0 \sigma_t^0 + p_1 c_1 \sigma_t^1$. The ensemble-averaged cross section values for the different cases are given in Table 2. These ensemble-averaged material properties are used to generate the atomic mix approximation solutions for the benchmark suites.

Table 1: **Material parameters for benchmark transport problems**

| Case | $\sigma_t^0$ | $\Lambda_0$ | $\sigma_t^1$ | $\Lambda_1$ | Case | $c_0$ | $c_1$ | $L$ |
|---|---|---|---|---|---|---|---|---|
| 1 | 10/99 | 99/100 | 100/11 | 11/100 | a | 0.0 | 1.0 | 0.1 |
| 2 | 10/99 | 99/10 | 100/11 | 11/10 | b | 1.0 | 0.0 | 1.0 |
| 3 | 2/101 | 101/20 | 200/101 | 101/20 | c | 0.9 | 0.9 | 10.0 |

Table 2: **Ensemble-averaged material parameters**

| Case | $\langle \sigma_t \rangle$ | $\langle \sigma_s \rangle$ |
|---|---|---|
| 1a, 2a | 1 | 10/11 |
| 1b, 2b | 1 | 1/11 |
| 3a | 1 | 100/101 |
| 3b | 1 | 1/101 |
| 1c, 2c, 3c | 1 | 9/10 |

### 2.2. Generation of Benchmark Solutions

We generated the benchmark solutions for both suites, including scalar flux distributions, using the methodologies described in Refs. [5] and [6]. We now briefly describe this benchmark procedure. One instance of the material realization is generated by first sampling the material located at $x = 0$ using the probabilities defined in Eq. (6). Given this sampled material, a material slab width is sampled from the Markovian exponential distribution, Eq. (5), using the mean material slab width, $\Lambda_0$ or $\Lambda_1$, corresponding to the sampled material. The material slab width for the next (different) material is then sampled in the same manner. This process is repeated until the sum of the sampled material slab widths exceeds (or equals) the total slab width $L$. The last sampled material slab width typically requires truncation to the total slab width. We note that this truncation implies that the final material specification at the $x = L$ boundary does not rigorously follow Markovian statistics. This truncation and resulting approximation provides additional motivation for the comparison in Suite II to the leakage value at $x = 0$, $\langle J_0 \rangle$, instead of the leakage value at $x = L$. Given this single material realization, the transport problem described by Eqs. (1)–(4) is then solved for that realization using a discrete ordinates transport code. This procedure is repeated a large number $M$ of times and the results averaged to obtain ensemble-averaged values. The ensemble-averaged leakage values at $x = 0$ and $x = L$ are computed as [5]

$$\langle J_0 \rangle = \frac{1}{M} \sum_{m=1}^{M} \int_{-1}^{0} |\mu| \psi_m (0, \mu) \, d\mu \quad , \tag{9}$$

$$\langle J_L \rangle = \frac{1}{M} \sum_{m=1}^{M} \int_{0}^{1} \mu \psi_m (L, \mu) \, d\mu \quad , \tag{10}$$

where $\psi_m (x, \mu)$ is the angular flux computed for realization $m$, and the angular integral is performed using the same quadrature as in the discrete ordinates transport calculation. The ensemble-averaged material $i$ scalar flux distribution is computed as [6]

$$\langle \phi_i (x) \rangle = \frac{1}{M_i} \sum_{m_i=1}^{M_i} \phi_{m_i} (x) \quad , \tag{11}$$

where $\phi_{m_i} (x)$ is the material $i$ scalar flux distribution for realization $m$ at spatial location $x$. Here $M_i \leq M$ is the number of realizations with material $i$ present at location $x$, and the sum is computed only for those realizations.

The discrete ordinates transport code used to generate the benchmark solutions utilizes the linear discontinuous spatial discretization with the mesh spacing in each material chosen such that $\frac{\sigma_t^i \Delta x}{|\mu|_{min}} \leq \frac{1}{10}$ for accuracy, where $|\mu|_{min}$ is the minimum direction cosine in the quadrature set [14]. $5 \times 10^5$ independent statistical material realizations were sampled from Markovian statistics and simulated for each case. The material scalar flux distributions were tallied at the edges of 100 uniformly-spaced spatial zones. We enforced a minimum of 100 spatial zones for each independent material realization if the zone size restriction described above would have resulted in a smaller number of zones.

Because we are using the benchmark results to assess the accuracy of Monte Carlo algorithms that use no angular discretization, we used high-order discrete ordinates quadrature sets in the generation of the benchmark solutions. The Suite I incident angular flux benchmark solutions were obtained using a standard Gauss-Legendre quadrature set with $N = 64$. For the Suite II interior source benchmark problems, we found that angular convergence for the optically thin slabs (i.e. $L = 0.1$ and $L = 1.0$) could only be achieved using very high quadrature orders. This convergence difficulty is not entirely unexpected, as optically thin problems are known to require high order quadrature sets to converge because of the effect of the vacuum boundaries [14]. We examined in detail the case 1a benchmark problem for $L = 0.1$. This particular case has one highly probable material, material zero with $p_0 = 0.9$, with a small total cross section, $\sigma_t^0 = 10/99$, resulting in many realizations being composed of optically thin slabs of width $\sigma_t^0 \Delta x = 1/99$ mean free paths. The solution of this problem was not converged in angle using a standard $S_{96}$ Gauss-Legendre quadrature set. This same problem was converged in angle when using a $S_{96}$ double-$P_N$ quadrature set [14]. Based on this result, we used a double-$P_N$ quadrature set with $N = 96$ for the Suite II problems with total slab widths of $L = 0.1$ and 1.0 and a standard Gauss-Legendre quadrature set with $N = 64$ for the problems with a total slab width of $L = 10.0$. These high-order quadrature sets coupled with the spatial mesh size criterion described above resulted in the use of very fine spatial meshes.

We computed the transport solution for each material realization using an unmodified planar geometry discrete ordinates transport code written in Fortran. We performed the calculation of the ensemble-averaged results using a Python language driver script (less than 400 lines of actual code) that 1) samples each material realization, 2) generates and writes to disk an input file for the discrete ordinates transport code corresponding to the sampled material realization, 3) runs the transport code, 4) parses standard output files from the transport code to obtain the computed leakage values and scalar flux distributions for that material realization, 5) computes the required ensemble-averaged sums over all material realizations, and 6) writes the ensemble-averaged results to a summary file. As the generation of these benchmark solutions is a process to be performed only a limited number of times, this approach has the significant advantage of requiring no modification or specialization (and associated testing) of the discrete ordinates transport code for the solution of these benchmark problems. In addition, any transport code could potentially be used to solve the transport problems by generating the appropriate input file for the code and parsing its output. A disadvantage to this approach is that the linkage of the Python processing script to the transport code via writing input files and parsing output files possesses an inherent inefficiency. To ameliorate this inefficiency, we utilized the pyMPI [17] Python extension to parallelize using the MPI message passing interface [18] the sampling of the independent material realizations and the solution of the transport problems for each of these realizations. This parallelization is very efficient, as each MPI process can independently sample a subset of the total number of material realizations and solve the corresponding transport problems with no parallel communication with other processes required until all of its computational work is completed. Care must be exercised to ensure that independent random number streams are utilized on each MPI process. The benchmark solutions described in this paper were obtained using typically 512 processors of a Linux cluster with sixteen AMD Opteron 2.3 GHz processors per compute node. Each benchmark case required from less than one hour to several hours of simulation time. The simulation time per material realization varies significantly based on the total slab width, the material properties, and the quadrature order but was typically on the order of a fraction of a second to a few seconds.

We regenerated the benchmark solutions to the standard suite of benchmark problems (Suite I) originally defined in Ref. [5] using the same number of material realizations ($10^5$) and the same angular quadrature order ($N = 16$) as in the original reference. We compared our benchmark solutions against the probabilities of reflection and transmission published in Refs. [5] and [6], finding agreement to typically two to three digits, and against the scalar flux distribution data remaining available [19] from Ref. [6]. These comparisons establish confidence that our benchmark procedure is consistent with previously published benchmark results.

Finally, we note that an alternative approach to generating the benchmark solutions is to use a Monte Carlo algorithm for the solution of the transport problem associated with each material realization. This approach was followed by Donovan and Danon [8] in the generation of their two-dimensional benchmark results and by Ji and Martin [11] in the generation of their three-dimensional benchmark results.

## 3. Monte Carlo Algorithms

In the previous section, we described the binary stochastic mixture benchmark transport problems that we use to assess the accuracy of the Monte Carlo particle transport algorithms. In this section, we describe in more detail the Monte Carlo Algorithms A and B proposed by Zimmerman [1] and Zimmerman and Adams [2] for solving the benchmark transport problem described above.

For both Algorithms A and B, a particle history begins with sampling the source particle characteristics appropriately for either Suite I or II. For Suite I, the spatial position $x$ is set to zero, a direction of flight cosine $\mu$ is sampled from a cosine distribution modeling the isotropic incident angular flux, and a material identifier for the particle is sampled according to the probabilities defined in Eq. (6). For Suite II, the spatial position $x$ is sampled uniformly within the spatial domain, an isotropic direction cosine modeling the interior source is sampled, and a material identifier for the particle is sampled according to the probabilities defined in Eq. (6).

A new event, the distance to material interface, $d_i$, is introduced for these Monte Carlo algorithms for transport through stochastic mixtures. A distance to material interface is sampled by sampling a material slab width from the exponential distribution given by Eq. (5) and dividing by the magnitude of the particle direction cosine, $|\mu|$, to account for the direction of particle motion, i.e. $d_i = -\Lambda_i \ln(\xi)/|\mu|$, where $\xi$ is a random number.

Next, distances to the required standard Monte Carlo events are either sampled or computed. The distance to collision, $d_c$, is sampled using the macroscopic total cross section corresponding to the material in which the particle

exists. Because we are interested in comparing the material scalar flux distributions, we impose a uniform spatial mesh on the spatial domain in which to tally this information. As a result, the distance to zone boundary, $d_b$, is computed using the current position and direction of flight of the particle and the boundaries of the spatial zone in which the particle exists. We compute the Monte Carlo scalar flux tallies using a track length estimator [14]. The tally volume for a material scalar flux (i.e. the volume by which the summed track length is divided) is the zone volume times the volume fraction of the material in the zone, where the volume fraction of material $i$ is equal to the material probability $p_i$ defined in Eq. (6). For both Monte Carlo algorithms, the distance the particle travels in the zone is tallied whenever a particle is moved.

In the next sections, we describe in more detail the particle history flow for Algorithms A and B.

### *3.1. Algorithm A: The Levermore-Pomraning Solution*

For each particle history:

1. Sample the distance to material interface, $d_i$.
2. Compute the distance to zone boundary, $d_b$, and sample the distance to collision, $d_c$.
3. Compute the minimum of $d_b$, $d_c$, and $d_i$ to determine the sampled event.
4. If $d_b$ is the minimum distance, move the particle to the zone boundary. If the particle is escaping the spatial domain, update the appropriate leakage tally, terminate the history, and track the next particle. Otherwise, return to step 1.
5. If $d_c$ is the minimum distance, move the particle the appropriate distance, and sample the collision type using the macroscopic total and scattering cross sections for the material in which the particle exists. If the sampled collision is absorption, terminate the history and track the next particle. If the sampled collision is scattering, perform the scattering collision by sampling the outgoing characteristics of the scattered particle; the particle maintains its current material identifier. Return to step 1.
6. If $d_i$ is the minimum distance, move the particle the appropriate distance and switch the material identifier. Return to step 1.

Sahni [20] has argued that a "generic equation" approach to particle transport through a binary stochastic mixture such as employed in a Monte Carlo transport algorithm is equivalent to the Levermore-Pomraning model if each particle track is uncorrelated from its previous track. Because each particle track is uncorrelated with its previous tracks in Algorithm A, we expect that Algorithm A will agree with the Levermore-Pomraning model. The Levermore-Pomraning model is exact for a purely absorbing binary stochastic medium in which the materials are distributed according to Markovian statistics; this model becomes approximate when scattering is introduced [4].

Note that following a collision, a new distance to material interface is sampled in Algorithm A. As a result, the particle encounters a different material realization following a collision, which is unphysical. Therefore, we expect Algorithm A to be less accurate in highly scattering materials with optically thick mean material slab widths. As noted by Zimmerman and Adams [2], this algorithm is exact in a purely absorbing medium in which the materials are distributed according to Markovian statistics. Because we have imposed a spatial mesh on the problem, a new distance to material interface is also sampled following a zone boundary crossing. Since Algorithm A models a Markovian (i.e. a no-memory) transport process involving uncorrelated particle flights, sampling a new distance to material interface following a zone boundary crossing does not introduce additional error into the algorithm. Numerical experiments using Algorithm A have confirmed that consistent leakage values and scalar flux distributions are obtained when using significantly different numbers of zones.

### *3.2. Algorithm B: A More Accurate Solution*

For each particle history:

1. Sample the distance to material interface values in the forward and backward directions of particle motion, $d_i^+$ and $d_i^-$, respectively. (When sampling the initial source particle for benchmark Suite I, set the distance to material interface in the backward direction, $d_i^-$, to zero.)
2. Compute the distance to zone boundary, $d_b$, and sample the distance to collision, $d_c$.

3. Compute the minimum of $d_b$, $d_c$, and $d_i^+$ to determine the sampled event.
4. If $d_b$ is the minimum distance, initially treat as in Algorithm A, step 4. Instead of resampling the distance to material interface (returning to step 1) as in Algorithm A, adjust the distance to material interface values in the forward and backward directions to account for the distance the particle was moved. Return to step 2.
5. If $d_c$ is the minimum distance, initially treat as in Algorithm A, step 5. Instead of resampling the distance to material interface (returning to step 1) as in Algorithm A, adjust the distance to material interface values in the forward and backward directions to account for the distance the particle was moved. If the sampled collision is scattering, also adjust the distance to material interface values in the forward and backward directions to account for the change in direction of flight of the particle after the scatter. Switch the forward and backward distance to material interface values if the particle is backscattered (i.e. the value of the direction cosine $\mu$ changes sign). Return to step 2.
6. If $d_i^+$ is the minimum distance, move the particle the appropriate distance, switch the material identifier, sample a new $d_i^+$, and set $d_i^-$ to zero. Return to step 2.

In Algorithm B, a particle can move, undergo collisions, and cross zone boundaries within one material and encounter the same local material realization, which is physically more realistic than Algorithm A. As a result, we expect Algorithm B to be more accurate than Algorithm A. Algorithm B remains inexact if a particle reenters the same material at the same location within one history, as the sampled material realization will be different upon reentry. Zimmerman and Adams [2] also examined an Algorithm C that maintained additional realization information regarding the materials surrounding the current material in which the particle exists. This algorithm demonstrated improved accuracy over Algorithm B but may not be feasible in multiple dimensions. We do not discuss Algorithm C in further detail in this paper.

## 4. Numerical Comparisons to Benchmark Problems

In this section, we evaluate the accuracy of the Monte Carlo algorithms described in Section 3 using the benchmark problem suites described in Section 2. In addition to comparisons of the leakage values, we compare the detailed scalar flux distributions produced by these Monte Carlo algorithms with the benchmark scalar flux distributions. The scalar flux distributions were tallied in the Monte Carlo simulations using 100 uniform spatial zones. Each Monte Carlo simulation was performed using $10^9$ particle histories, resulting in relative standard deviations for the leakage values and pointwise relative standard deviations for the material scalar flux distributions of typically significantly less than 0.1%. The Monte Carlo solutions described in this paper were obtained using one processor of a Linux cluster with sixteen AMD Opteron 2.3 GHz processors per compute node (the same computational platform used to generate the benchmark solutions). Each Monte Carlo calculation required from less than one hour to a few hours of simulation time using a *single* processor, with a maximum run time of approximately seven hours. For all the problems considered, the Algorithm B solutions required less computing time than the Algorithm A solutions. We have not investigated this performance difference in detail, but we conjecture that one reason Algorithm A is less efficient than Algorithm B is the additional number of calls to the exponential function required to sample the distance to material interface for each particle track.

For comparison with the results obtained using the Monte Carlo algorithms, we also present deterministic atomic mix solutions for the benchmark problems. These atomic mix solutions use the ensemble-averaged material properties shown in Table 2. The atomic mix solutions for several of the cases are identical, because the ensemble-averaged total and scattering cross sections are identical for those cases. The atomic mix approximation produces only an ensemble-averaged total scalar flux distribution and, as a result, predicts the same scalar flux distribution for both materials in the system. In each of the cases examined, the deterministic atomic mix solutions were computed using the same quadrature set used to compute the benchmark solutions. The same maximum spatial zone size and minimum number of spatial zones restraints used for the generation of the benchmark results were also applied during the generation of the atomic mix solutions. The resulting atomic mix scalar flux distributions were averaged onto the same spatial mesh used to tally the benchmark scalar flux distributions. The atomic mix solutions were obtained using one processor of a Linux cluster with sixteen AMD Opteron 2.3 GHz processors per compute node (the same computational platform used to generate the benchmark and Monte Carlo solutions). Each atomic mix calculation typically required on the order of a fraction of a second to a few seconds using a *single* processor.

We compare the accuracy of the ensemble-averaged leakage values computed using the atomic mix (AM) approximation or the Monte Carlo (MC) algorithms to the benchmark values using relative errors computed as

$$E_{\langle X \rangle} = \frac{\langle X \rangle_{AM/MC} - \langle X \rangle_{benchmark}}{\langle X \rangle_{benchmark}} \quad , \tag{12}$$

where $\langle X \rangle$ represents either the ensemble-averaged leakage at $x = 0$, $\langle J_0 \rangle$, or at $x = L$, $\langle J_L \rangle$. We compare the accuracy of the scalar flux distributions using a root-mean-squared (RMS) relative error computed as

$$E_{\langle \phi_i \rangle} = \sqrt{\frac{1}{K} \sum_{k=1}^{K} \left( \frac{\left\langle \phi_{i,k}^{AM/MC} \right\rangle - \left\langle \phi_{i,k}^{benchmark} \right\rangle}{\left\langle \phi_{i,k}^{benchmark} \right\rangle} \right)^2} \quad , \tag{13}$$

where $\langle \phi_i(x) \rangle$, $i = 0, 1$, is the material $i$ scalar flux distribution, and the summation is over the $K = 101$ cell-edge values for the deterministic atomic mix solutions and the $K = 100$ spatial tally zones for the Monte Carlo solutions. The Monte Carlo scalar flux tallies were computed using track length estimators in the spatial zones. The benchmark and atomic mix scalar flux results were computed using a discrete ordinates code with a linear discontinuous spatial discretization [14]. We compare the Monte Carlo zonal scalar flux tally with the benchmark discrete ordinates cell-average value computed as the algebraic average of the cell-edge values (consistent with the linear discontinuous discretization). The RMS relative errors of the atomic mix scalar flux distributions were directly computed using the atomic mix cell-edge scalar flux values and the benchmark discrete ordinates cell-edge scalar flux values.

### *4.1. Suite I Comparisons*

The computed Suite I benchmark, atomic mix, and Monte Carlo leakage values and corresponding relative errors for cases 1 through 3 are shown in Tables 3–5. The RMS relative error results for the material scalar flux distributions for cases 1 through 3 are shown in Tables 6–8. The material scalar flux distributions computed using the benchmark procedure, the atomic mix approximation, and the Monte Carlo algorithms for all cases with $L = 10$ are plotted (on log scales) in Figs. 1–3.

The Monte Carlo Algorithm A leakage value results agree in all cases, to typically two to three digits, with previously-published Levermore-Pomraning model results [5, 6]. The Monte Carlo Algorithm B leakage value results agree, as far as can be discerned, with the subset of data published as relative errors in graphical form in Ref. [2].

The atomic mix approximation results are the least accurate overall for the benchmark Suite I problems, in error for some cases by a few orders of magnitude. Pomraning [4] has argued that the atomic mix approximation always underestimates the transmission through a source-free random mixture as a result of neglecting cross correlation terms. The $\langle J_L \rangle$ transmission leakage results in Tables 3–5 and the material scalar flux distributions in Figs. 1–3 are consistent with this assertion. The atomic mix approximation does not, in general, possess enough accuracy to adequately treat the incident angular flux problems in benchmark Suite I.

Two general trends regarding Monte Carlo Algorithms A and B can be observed from the Suite I numerical results. First, the accuracy of Algorithms A and B, as measured by both the reflection and transmission values and the scalar flux distributions, generally improves as the slab width decreases. Second, both Algorithms A and B generally underpredict the reflection from the slab and overpredict the transmission through the slab. These trends are consistent with observations by previous researchers [2, 5, 6] from similar numerical results. Zimmerman has suggested [21] that Algorithms A and B may overpredict the transmission because an optically thick material may be resampled as optically thin if a particle makes a repeated attempt to penetrate the material. Maintaining additional material realization information could improve the accuracy of these algorithms (e.g. Algorithm C in Ref. [2]).

The leakage values computed by both Monte Carlo algorithms for the $L = 0.1$ slab are accurate to within a couple of percent. The Monte Carlo algorithms become less accurate as the slab width increases. For the $L = 1.0$ slab, the maximum relative error for Algorithm A is -16% for case 1b and for Algorithm B is -5% for case 1c. For the $L = 10$ slab width, the typical relative errors are significantly larger. The largest relative error for Algorithm A is 78% for case 1a, while the largest relative error for Algorithm B is 48% for case 1a. Overall, we find that Algorithm B generally produces significantly more accurate leakage values than Algorithm A, although both algorithms exhibit some large errors for the leakage values in benchmark Suite I.

Focusing on the scalar flux distribution RMS relative error results in Tables 6–8, the accuracy of both Algorithms A and B generally improves as the slab width decreases. The scalar flux distributions computed by both algorithms for the $L = 0.1$ slab are accurate to within 1% RMS relative error. The RMS relative errors for the $L = 1.0$ slab are typically a few percent; the maximum Algorithm A error is 26% for case 2c and the maximum Algorithm B error is 4% for case 1c. The RMS relative errors for the $L = 10.0$ slab are generally somewhat larger; the maximum Algorithm A error is 64% for case 2c and the maximum Algorithm B error is 24% for case 1a. In general, when Algorithm A is reasonably accurate (i.e. within a couple of percent of the benchmark solution), Algorithm B is typically somewhat more accurate. When Algorithm A produces very inaccurate scalar flux distributions (i.e. larger than approximately 25% relative differences compared to the benchmark solution), Algorithm B is usually significantly more accurate. The overall Algorithm B RMS relative errors in the scalar flux distributions for the three cases examined are about a factor of one and one-half to six times smaller than the Algorithm A errors. Examining the material scalar flux distributions in Figs. 1–3, the Algorithm B scalar flux distributions are clearly in overall better agreement with the benchmark distributions than the Algorithm A distributions. The statistical fluctuations evident in some of the benchmark flux distributions derive from a small material probability resulting in a relatively small number of realizations contributing to the distribution. These statistical fluctuations were also observed in previous benchmark comparisons [6].

As described in Section 3.1, we expect Algorithm A to be least accurate in scattering materials with optically thick mean material slab widths. One particular example of this phenomenon is represented by case 2c for the $L = 10$ slab. For this case, materials zero and one have mean material slab widths of one and ten, respectively, and both materials have a scattering ratio of 0.9. The RMS relative scalar flux error values, given in Table 7, are 0.102 for material zero and 0.639 for material one. The error in the more optically-thick material one is significantly larger than in the less optically-thick material zero. Algorithm B is significantly more accurate than Algorithm A for this case, having RMS relative error values of 0.024 and 0.101 for materials zero and one, respectively.

Finally, we note that both Algorithms A and B produce physically-realistic material scalar flux distributions for the benchmark transport problems considered. This result is largely expected for Algorithm A, because it has previously been shown to produce the Levermore-Pomraning reflection and transmission probability results [2]; the demonstration of this fact for Algorithm B is new. The ability of an algorithm to accurately compute scalar flux distributions is important, as these distributions determine the reaction rates in the materials of the system.

Table 3: **Suite I leakage value comparisons for case 1**

| | | | | | | | Relative Error $E_{\langle J_0\rangle,\langle J_L\rangle}$ | | |
|---|---|---|---|---|---|---|---|---|---|
| $L$ | Case | Quantity | Benchmark | Atomic Mix | Algorithm A | Algorithm B | Atomic Mix | Algorithm A | Algorithm B |
| 0.1 | a | $\langle J_0\rangle$ | 0.04874 | 0.07544 | 0.04768 | 0.04884 | 0.548 | -0.022 | 0.002 |
| | | $\langle J_L\rangle$ | 0.93369 | 0.90690 | 0.93463 | 0.93348 | -0.029 | 0.001 | -0.000 |
| | b | $\langle J_0\rangle$ | 0.00862 | 0.00654 | 0.00849 | 0.00857 | -0.241 | -0.015 | -0.006 |
| | | $\langle J_L\rangle$ | 0.90118 | 0.83900 | 0.90080 | 0.90080 | -0.069 | -0.000 | -0.000 |
| | c | $\langle J_0\rangle$ | 0.04812 | 0.07455 | 0.04702 | 0.04813 | 0.549 | -0.023 | 0.000 |
| | | $\langle J_L\rangle$ | 0.93394 | 0.90602 | 0.93468 | 0.93368 | -0.030 | 0.001 | -0.000 |
| 1.0 | a | $\langle J_0\rangle$ | 0.25069 | 0.36040 | 0.21888 | 0.23928 | 0.438 | -0.127 | -0.046 |
| | | $\langle J_L\rangle$ | 0.59607 | 0.48096 | 0.62741 | 0.60746 | -0.193 | 0.053 | 0.019 |
| | b | $\langle J_0\rangle$ | 0.05418 | 0.01879 | 0.04552 | 0.05260 | -0.653 | -0.160 | -0.029 |
| | | $\langle J_L\rangle$ | 0.48537 | 0.23065 | 0.48446 | 0.48520 | -0.525 | -0.002 | -0.000 |
| | c | $\langle J_0\rangle$ | 0.25419 | 0.35288 | 0.21677 | 0.24058 | 0.388 | -0.147 | -0.054 |
| | | $\langle J_L\rangle$ | 0.60114 | 0.47477 | 0.62755 | 0.61091 | -0.210 | 0.044 | 0.016 |
| 10.0 | a | $\langle J_0\rangle$ | 0.43634 | 0.49564 | 0.37815 | 0.40086 | 0.136 | -0.133 | -0.081 |
| | | $\langle J_L\rangle$ | 0.01486 | 0.00474 | 0.02642 | 0.02202 | -0.681 | 0.778 | 0.482 |
| | b | $\langle J_0\rangle$ | 0.08549 | 0.01963 | 0.05864 | 0.07755 | -0.770 | -0.314 | -0.093 |
| | | $\langle J_L\rangle$ | 0.00166 | 0.00001 | 0.00154 | 0.00164 | -0.994 | -0.072 | -0.012 |
| | c | $\langle J_0\rangle$ | 0.47746 | 0.47819 | 0.36952 | 0.40690 | 0.002 | -0.226 | -0.148 |
| | | $\langle J_L\rangle$ | 0.01609 | 0.00386 | 0.02377 | 0.02091 | -0.760 | 0.477 | 0.300 |
| | | | | | | RMS of $E_{\langle J_0\rangle}$ | 0.476 | 0.161 | 0.069 |
| | | | | | | RMS of $E_{\langle J_L\rangle}$ | 0.516 | 0.306 | 0.189 |

Table 4: **Suite I leakage value comparisons for case 2**

| | | | | | | | Relative Error $E_{\langle J_0\rangle,\langle J_L\rangle}$ | | |
|---|---|---|---|---|---|---|---|---|---|
| $L$ | Case | Quantity | Benchmark | Atomic Mix | Algorithm A | Algorithm B | Atomic Mix | Algorithm A | Algorithm B |
| 0.1 | a | $\langle J_0\rangle$ | 0.04304 | 0.07544 | 0.04302 | 0.04321 | 0.753 | -0.000 | 0.004 |
| | | $\langle J_L\rangle$ | 0.93940 | 0.90690 | 0.93931 | 0.93912 | -0.035 | -0.000 | -0.000 |
| | b | $\langle J_0\rangle$ | 0.00885 | 0.00654 | 0.00880 | 0.00881 | -0.261 | -0.006 | -0.005 |
| | | $\langle J_L\rangle$ | 0.91465 | 0.83900 | 0.91419 | 0.91419 | -0.083 | -0.001 | -0.001 |
| | c | $\langle J_0\rangle$ | 0.04246 | 0.07455 | 0.04236 | 0.04255 | 0.756 | -0.002 | 0.002 |
| | | $\langle J_L\rangle$ | 0.94005 | 0.90602 | 0.93985 | 0.93968 | -0.036 | -0.000 | -0.000 |
| 1.0 | a | $\langle J_0\rangle$ | 0.12120 | 0.36040 | 0.10681 | 0.12074 | 1.974 | -0.119 | -0.004 |
| | | $\langle J_L\rangle$ | 0.72656 | 0.48096 | 0.74079 | 0.72711 | -0.338 | 0.020 | 0.001 |
| | b | $\langle J_0\rangle$ | 0.07337 | 0.01879 | 0.07063 | 0.07318 | -0.744 | -0.037 | -0.003 |
| | | $\langle J_L\rangle$ | 0.75974 | 0.23065 | 0.75920 | 0.75945 | -0.696 | -0.001 | -0.000 |
| | c | $\langle J_0\rangle$ | 0.14248 | 0.35288 | 0.12426 | 0.14165 | 1.477 | -0.128 | -0.006 |
| | | $\langle J_L\rangle$ | 0.76855 | 0.47477 | 0.77434 | 0.76876 | -0.382 | 0.008 | 0.000 |
| 10.0 | a | $\langle J_0\rangle$ | 0.23723 | 0.49564 | 0.18049 | 0.22235 | 1.089 | -0.239 | -0.063 |
| | | $\langle J_L\rangle$ | 0.09843 | 0.00474 | 0.12840 | 0.10597 | -0.952 | 0.304 | 0.077 |
| | b | $\langle J_0\rangle$ | 0.28763 | 0.01963 | 0.21829 | 0.28542 | -0.932 | -0.241 | -0.008 |
| | | $\langle J_L\rangle$ | 0.19553 | 0.00001 | 0.17938 | 0.19528 | -1.000 | -0.083 | -0.001 |
| | c | $\langle J_0\rangle$ | 0.43319 | 0.47819 | 0.28962 | 0.40092 | 0.104 | -0.331 | -0.074 |
| | | $\langle J_L\rangle$ | 0.18690 | 0.00386 | 0.19498 | 0.19556 | -0.979 | 0.043 | 0.046 |
| | | | | | | RMS of $E_{\langle J_0\rangle}$ | 1.049 | 0.169 | 0.033 |
| | | | | | | RMS of $E_{\langle J_L\rangle}$ | 0.634 | 0.106 | 0.030 |

Table 5: **Suite I leakage value comparisons for case 3**

| | | | | | | | Relative Error $E_{\langle J_0\rangle,\langle J_L\rangle}$ | | |
|---|---|---|---|---|---|---|---|---|---|
| $L$ | Case | Quantity | Benchmark | Atomic Mix | Algorithm A | Algorithm B | Atomic Mix | Algorithm A | Algorithm B |
| 0.1 | a | $\langle J_0\rangle$ | 0.07494 | 0.08343 | 0.07491 | 0.07492 | 0.113 | -0.000 | -0.000 |
| | | $\langle J_L\rangle$ | 0.92329 | 0.91479 | 0.92312 | 0.92312 | -0.009 | -0.000 | -0.000 |
| | b | $\langle J_0\rangle$ | 0.00098 | 0.00070 | 0.00098 | 0.00098 | -0.286 | 0.000 | 0.000 |
| | | $\langle J_L\rangle$ | 0.85192 | 0.83326 | 0.85188 | 0.85186 | -0.022 | -0.000 | -0.000 |
| | c | $\langle J_0\rangle$ | 0.06618 | 0.07455 | 0.06608 | 0.06608 | 0.126 | -0.002 | -0.002 |
| | | $\langle J_L\rangle$ | 0.91468 | 0.90602 | 0.91458 | 0.91458 | -0.009 | -0.000 | -0.000 |
| 1.0 | a | $\langle J_0\rangle$ | 0.32096 | 0.43624 | 0.31515 | 0.32016 | 0.359 | -0.018 | -0.002 |
| | | $\langle J_L\rangle$ | 0.66029 | 0.54446 | 0.66595 | 0.66096 | -0.175 | 0.009 | 0.001 |
| | b | $\langle J_0\rangle$ | 0.00877 | 0.00196 | 0.00840 | 0.00874 | -0.777 | -0.042 | -0.003 |
| | | $\langle J_L\rangle$ | 0.48297 | 0.22054 | 0.48293 | 0.48295 | -0.543 | -0.000 | -0.000 |
| | c | $\langle J_0\rangle$ | 0.24286 | 0.35288 | 0.23727 | 0.24237 | 0.453 | -0.023 | -0.002 |
| | | $\langle J_L\rangle$ | 0.60498 | 0.47477 | 0.60911 | 0.60525 | -0.215 | 0.007 | 0.000 |
| 10.0 | a | $\langle J_0\rangle$ | 0.69109 | 0.78629 | 0.60759 | 0.65416 | 0.138 | -0.121 | -0.053 |
| | | $\langle J_L\rangle$ | 0.16350 | 0.06677 | 0.24038 | 0.19804 | -0.592 | 0.470 | 0.211 |
| | b | $\langle J_0\rangle$ | 0.03651 | 0.00204 | 0.02402 | 0.03598 | -0.944 | -0.342 | -0.015 |
| | | $\langle J_L\rangle$ | 0.07678 | 0.00001 | 0.07567 | 0.07649 | -1.000 | -0.014 | -0.004 |
| | c | $\langle J_0\rangle$ | 0.44516 | 0.47819 | 0.32612 | 0.39897 | 0.074 | -0.267 | -0.104 |
| | | $\langle J_L\rangle$ | 0.10457 | 0.00386 | 0.11967 | 0.11799 | -0.963 | 0.144 | 0.128 |
| | | | | | | RMS of $E_{\langle J_0\rangle}$ | 0.467 | 0.151 | 0.039 |
| | | | | | | RMS of $E_{\langle J_L\rangle}$ | 0.543 | 0.164 | 0.082 |

Table 6: **Suite I scalar flux comparisons for case 1**

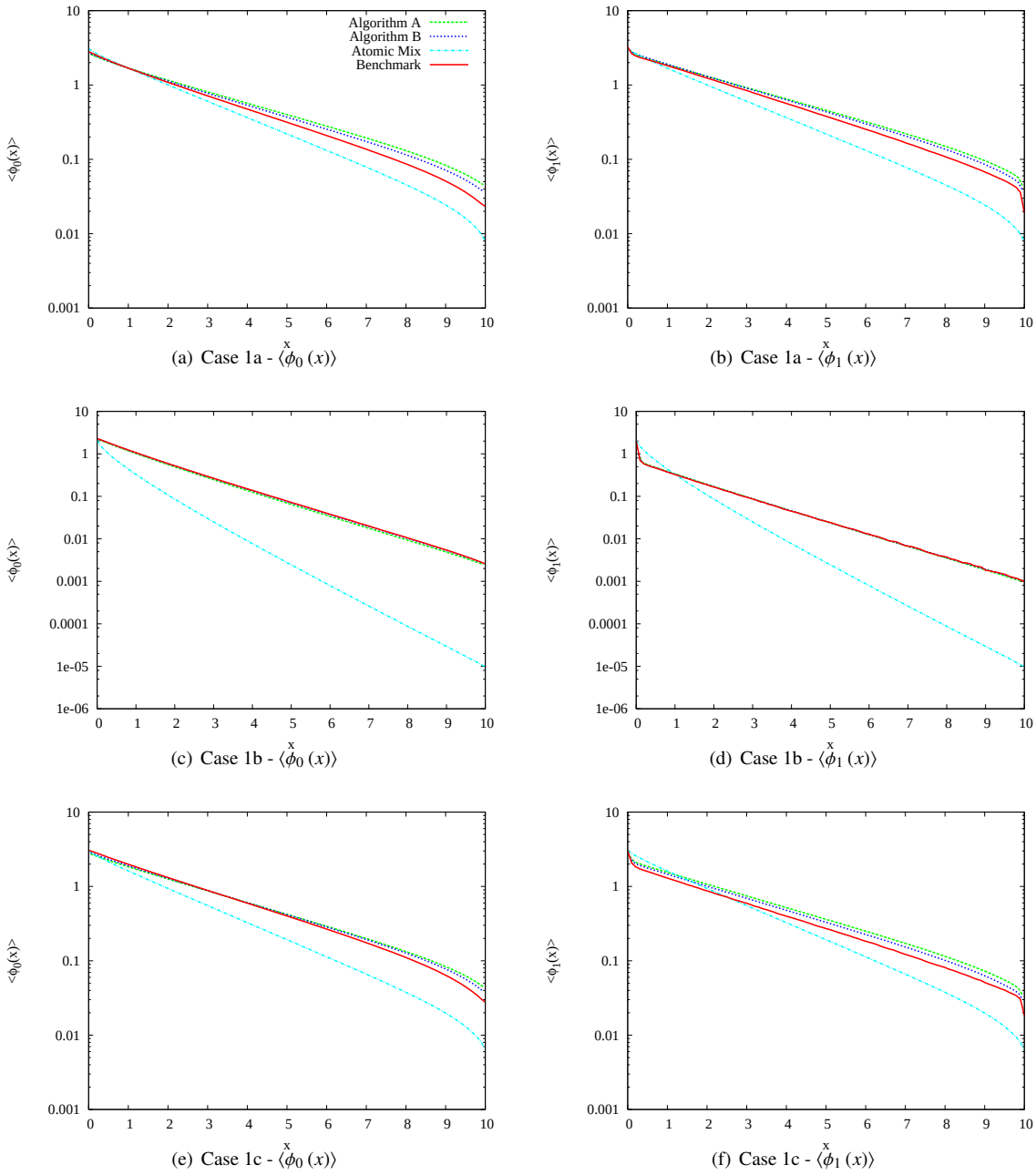
| | | | RMS Relative Error $E_{\langle\phi_i\rangle}$ | | |
|---|---|---|---|---|---|
| $L$ | Case | Quantity | Atomic Mix | Algorithm A | Algorithm B |
| 0.1 | a | $\langle\phi_0\rangle$ | 0.053 | 0.005 | 0.005 |
| | | $\langle\phi_1\rangle$ | 0.128 | 0.009 | 0.001 |
| | b | $\langle\phi_0\rangle$ | 0.144 | 0.001 | 0.001 |
| | | $\langle\phi_1\rangle$ | 0.867 | 0.003 | 0.003 |
| | c | $\langle\phi_0\rangle$ | 0.055 | 0.002 | 0.001 |
| | | $\langle\phi_1\rangle$ | 0.190 | 0.009 | 0.001 |
| 1.0 | a | $\langle\phi_0\rangle$ | 0.082 | 0.038 | 0.016 |
| | | $\langle\phi_1\rangle$ | 0.122 | 0.023 | 0.028 |
| | b | $\langle\phi_0\rangle$ | 0.498 | 0.010 | 0.002 |
| | | $\langle\phi_1\rangle$ | 0.672 | 0.021 | 0.006 |
| | c | $\langle\phi_0\rangle$ | 0.111 | 0.035 | 0.014 |
| | | $\langle\phi_1\rangle$ | 0.248 | 0.089 | 0.035 |
| 10.0 | a | $\langle\phi_0\rangle$ | 0.343 | 0.377 | 0.239 |
| | | $\langle\phi_1\rangle$ | 0.440 | 0.261 | 0.175 |
| | b | $\langle\phi_0\rangle$ | 0.909 | 0.093 | 0.025 |
| | | $\langle\phi_1\rangle$ | 0.828 | 0.042 | 0.030 |
| | c | $\langle\phi_0\rangle$ | 0.514 | 0.173 | 0.118 |
| | | $\langle\phi_1\rangle$ | 0.393 | 0.327 | 0.200 |
| | | RMS of $E_{\langle\phi_0\rangle}$ | 0.408 | 0.143 | 0.090 |
| | | RMS of $E_{\langle\phi_1\rangle}$ | 0.513 | 0.144 | 0.091 |

Table 7: **Suite I scalar flux comparisons for case 2**

| | | | RMS Relative Error $E_{\langle\phi_i\rangle}$ | | |
|---|---|---|---|---|---|
| $L$ | Case | Quantity | Atomic Mix | Algorithm A | Algorithm B |
| 0.1 | a | $\langle\phi_0\rangle$ | 0.067 | 0.004 | 0.005 |
| | | $\langle\phi_1\rangle$ | 0.220 | 0.002 | 0.001 |
| | b | $\langle\phi_0\rangle$ | 0.162 | 0.001 | 0.001 |
| | | $\langle\phi_1\rangle$ | 1.581 | 0.001 | 0.001 |
| | c | $\langle\phi_0\rangle$ | 0.068 | 0.001 | 0.001 |
| | | $\langle\phi_1\rangle$ | 0.335 | 0.002 | 0.000 |
| 1.0 | a | $\langle\phi_0\rangle$ | 0.199 | 0.014 | 0.003 |
| | | $\langle\phi_1\rangle$ | 0.209 | 0.029 | 0.005 |
| | b | $\langle\phi_0\rangle$ | 0.611 | 0.003 | 0.002 |
| | | $\langle\phi_1\rangle$ | 5.573 | 0.030 | 0.015 |
| | c | $\langle\phi_0\rangle$ | 0.224 | 0.015 | 0.002 |
| | | $\langle\phi_1\rangle$ | 1.772 | 0.255 | 0.009 |
| 10.0 | a | $\langle\phi_0\rangle$ | 0.610 | 0.162 | 0.045 |
| | | $\langle\phi_1\rangle$ | 0.702 | 0.091 | 0.040 |
| | b | $\langle\phi_0\rangle$ | 0.959 | 0.092 | 0.001 |
| | | $\langle\phi_1\rangle$ | 1.981 | 0.189 | 0.034 |
| | c | $\langle\phi_0\rangle$ | 0.782 | 0.102 | 0.024 |
| | | $\langle\phi_1\rangle$ | 1.274 | 0.639 | 0.101 |
| | | RMS of $E_{\langle\phi_0\rangle}$ | 0.517 | 0.071 | 0.017 |
| | | RMS of $E_{\langle\phi_1\rangle}$ | 2.184 | 0.240 | 0.039 |

Table 8: **Suite I scalar flux comparisons for case 3**

| | | | RMS Relative Error $E_{\langle\phi_i\rangle}$ | | |
|---|---|---|---|---|---|
| $L$ | Case | Quantity | Atomic Mix | Algorithm A | Algorithm B |
| 0.1 | a | $\langle\phi_0\rangle$ | 0.076 | 0.002 | 0.002 |
| | | $\langle\phi_1\rangle$ | 0.043 | 0.001 | 0.001 |
| | b | $\langle\phi_0\rangle$ | 0.176 | 0.001 | 0.001 |
| | | $\langle\phi_1\rangle$ | 0.155 | 0.002 | 0.002 |
| | c | $\langle\phi_0\rangle$ | 0.078 | 0.001 | 0.001 |
| | | $\langle\phi_1\rangle$ | 0.050 | 0.001 | 0.001 |
| 1.0 | a | $\langle\phi_0\rangle$ | 0.224 | 0.013 | 0.004 |
| | | $\langle\phi_1\rangle$ | 0.079 | 0.005 | 0.001 |
| | b | $\langle\phi_0\rangle$ | 0.625 | 0.001 | 0.000 |
| | | $\langle\phi_1\rangle$ | 0.858 | 0.002 | 0.002 |
| | c | $\langle\phi_0\rangle$ | 0.257 | 0.013 | 0.001 |
| | | $\langle\phi_1\rangle$ | 0.229 | 0.009 | 0.001 |
| 10.0 | a | $\langle\phi_0\rangle$ | 0.185 | 0.292 | 0.149 |
| | | $\langle\phi_1\rangle$ | 0.300 | 0.060 | 0.023 |
| | b | $\langle\phi_0\rangle$ | 0.954 | 0.030 | 0.001 |
| | | $\langle\phi_1\rangle$ | 0.873 | 0.025 | 0.010 |
| | c | $\langle\phi_0\rangle$ | 0.783 | 0.151 | 0.077 |
| | | $\langle\phi_1\rangle$ | 0.556 | 0.410 | 0.113 |
| | | RMS of $E_{\langle\phi_0\rangle}$ | 0.484 | 0.110 | 0.056 |
| | | RMS of $E_{\langle\phi_1\rangle}$ | 0.470 | 0.138 | 0.039 |




(a) Case 1a - $\langle \phi_0(x) \rangle$

(b) Case 1a - $\langle \phi_1(x) \rangle$

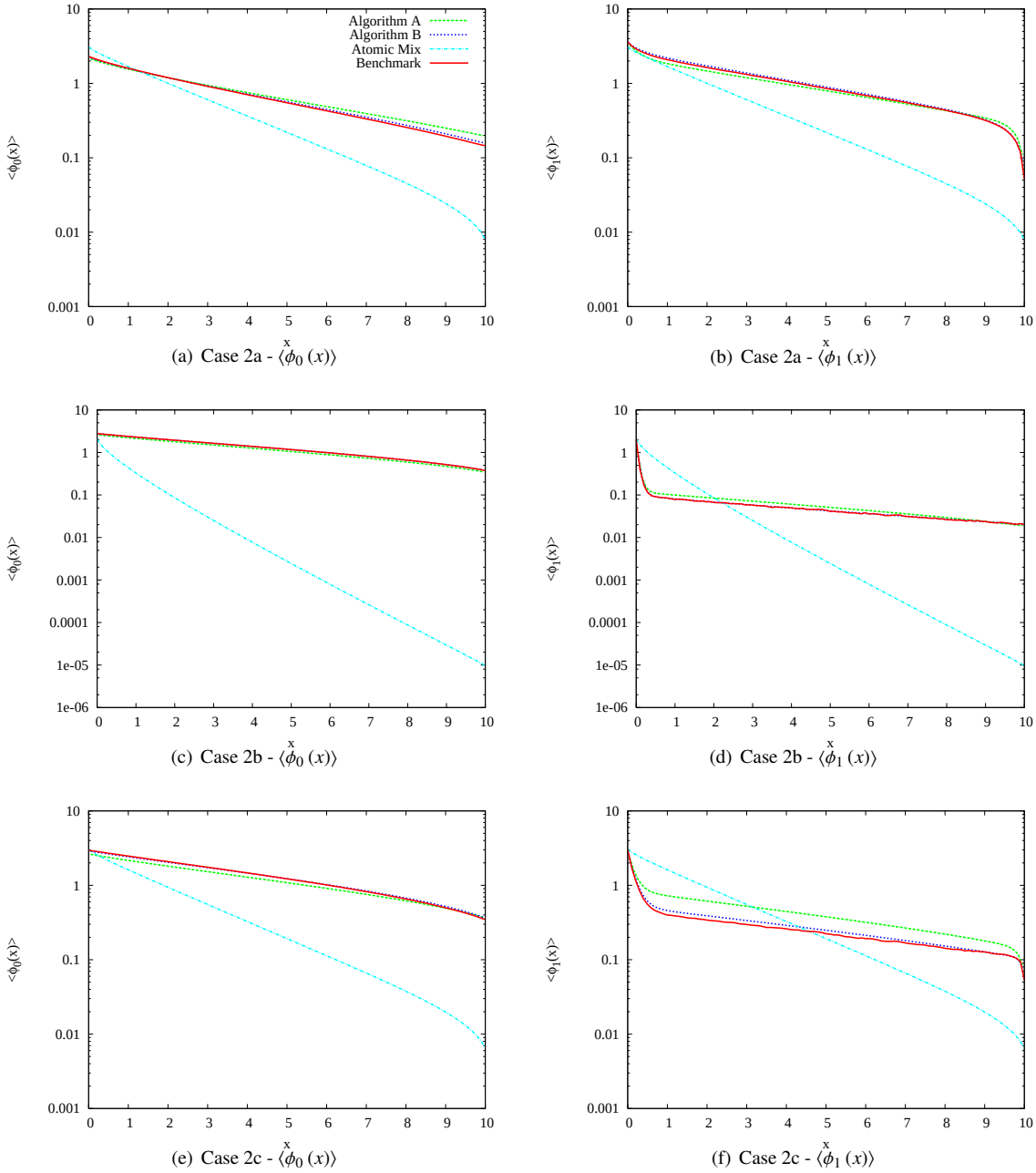
(c) Case 1b - $\langle \phi_0(x) \rangle$

(d) Case 1b - $\langle \phi_1(x) \rangle$

(e) Case 1c - $\langle \phi_0(x) \rangle$

(f) Case 1c - $\langle \phi_1(x) \rangle$

Figure 1: **Suite I scalar flux distributions for case 1 and L = 10**




(a) Case 2a - $\langle \phi_0(x) \rangle$

(b) Case 2a - $\langle \phi_1(x) \rangle$

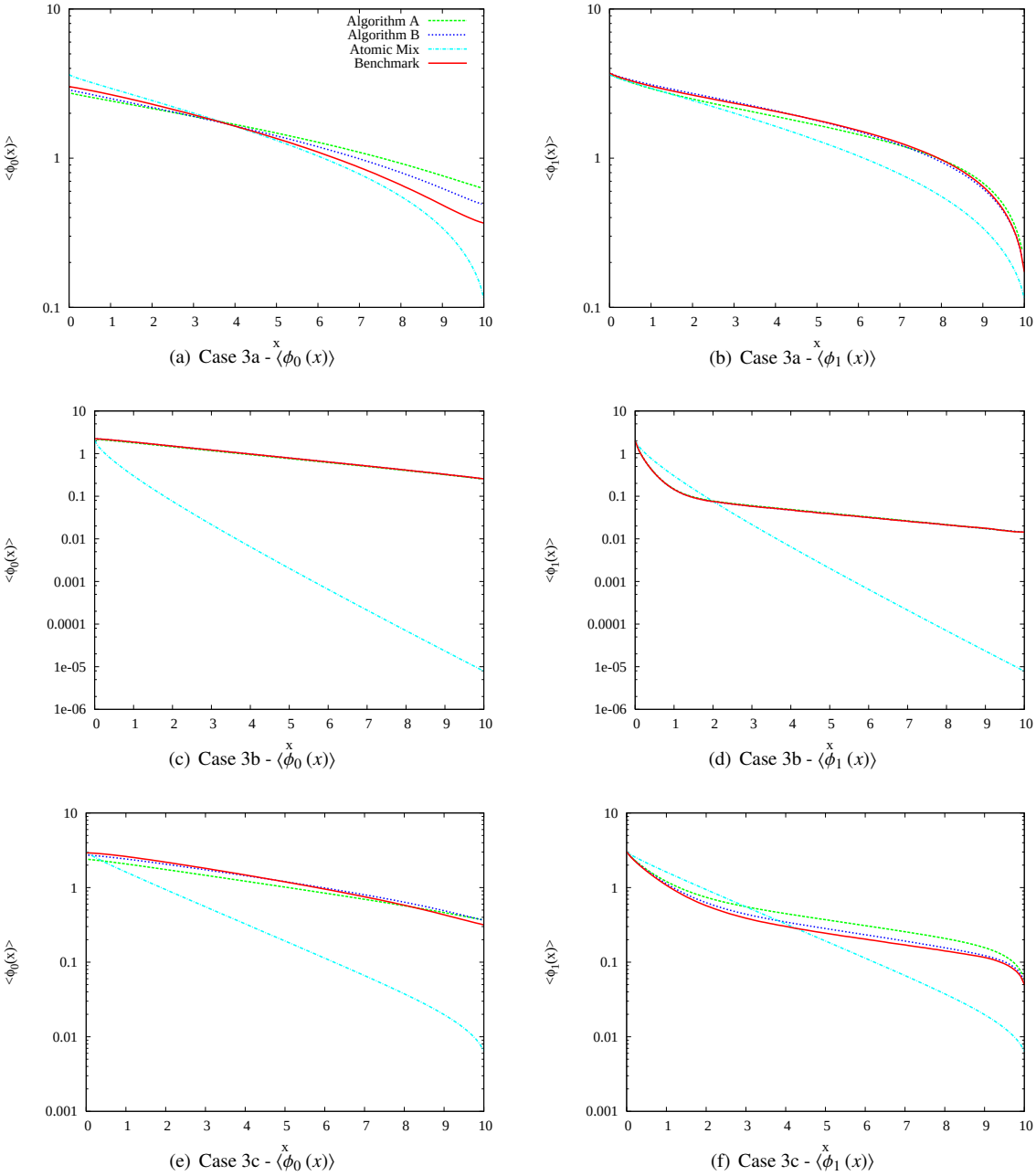
(c) Case 2b - $\langle \phi_0(x) \rangle$

(d) Case 2b - $\langle \phi_1(x) \rangle$

(e) Case 2c - $\langle \phi_0(x) \rangle$

(f) Case 2c - $\langle \phi_1(x) \rangle$

Figure 2: **Suite I scalar flux distributions for case 2 and L = 10**




(a) Case 3a - $\langle \phi_0(x) \rangle$

(b) Case 3a - $\langle \phi_1(x) \rangle$

(c) Case 3b - $\langle \phi_0(x) \rangle$

(d) Case 3b - $\langle \phi_1(x) \rangle$

(e) Case 3c - $\langle \phi_0(x) \rangle$

(f) Case 3c - $\langle \phi_1(x) \rangle$

Figure 3: **Suite I scalar flux distributions for case 3 and L = 10**

*4.2. Suite II Comparisons*

The computed Suite II benchmark, atomic mix, and Monte Carlo leakage values and corresponding relative errors for cases 1 through 3 are shown in Tables 9–11. The RMS relative error results for the material scalar flux distributions for cases 1 through 3 are shown in Tables 12–14. The material scalar flux distributions computed using the benchmark procedure, the atomic mix approximation, and the Monte Carlo algorithms for all cases with $L = 10$ are plotted in Figs. 4–6.

We first make some observations about the results of the benchmark simulations. We note that the leakage values for the optically-thin $L = 0.1$ slab imply that nearly all particles escape the spatial domain through the boundaries for nearly all cases, as expected. The only significant exception is case 3b in which material one is purely absorbing and has a mean material slab width of 10 mean free paths. As for the Suite I benchmark results, statistical fluctuations are evident in some of the benchmark flux distributions, most notably material one of case 2b shown in Fig. 5(d). These fluctuations have been observed in previous investigations [6] and attributed to a small material probability resulting in a relatively small number of realizations contributing to the flux distribution. The fluctuations are more pronounced in benchmark Suite II than in Suite I, although the numerical results for both suites were generated using the same number of material realizations. Finally, we note the mildly concave-up nature of the ensemble-averaged material zero scalar flux distribution near the boundaries in case 3a. This concavity is unique to case 3a and was not expected. We further investigated this characteristic of the scalar flux distribution by generating independent spherical harmonics $P_{15}$ transport solutions for the case 3a benchmark problem with $L = 10.0$. The ensemble-averaged material zero scalar flux distribution generated using the $P_{15}$ approximation also exhibits the concave-up nature near the boundaries. Given the consistency in the scalar flux behavior between the $S_{64}$ and the $P_{15}$ results and the fact that our benchmark Suite I case 3a scalar flux distributions agree with those obtained by Zuchuat et al. [6], we believe this characteristic of the benchmark scalar flux distribution to be correct.

The Monte Carlo Algorithm A results for benchmark Suite II agree in all cases, to typically three to four digits for the leakage values and to typically three digits for the material scalar flux RMS relative error, with previously-published Levermore-Pomraning model results for this interior source benchmark suite [12].

The atomic mix approximation results are the least accurate overall for the benchmark Suite II problems. Although the material scalar flux distributions computed using the atomic mix approximation are reasonably accurate in limited instances (e.g. material one of case 1a), in general the flux distributions exhibit significant errors.

The leakage values computed by both Monte Carlo algorithms for the optically-thin slabs ($L = 0.1$ and 1.0) are accurate to better than 1%. The Monte Carlo algorithms become less accurate as the slab width increases. For the $L = 10$ slab width, the typical relative error is a few percent. The largest relative error for Algorithm A is -9% for case 2c, while the largest relative error for Algorithm B is 5% for case 1a. Overall, we find that Algorithm B generally produces significantly more accurate leakage values than Algorithm A. These leakage results are generally consistent with observations from benchmark Suite I. However, we note that the errors in the leakage values for the interior source benchmark suite are generally somewhat smaller than those for the incident angular flux benchmark suite.

The scalar flux distributions computed using Monte Carlo Algorithms A and B are generally more accurate for the optically-thin slabs. The scalar flux distributions computed by both algorithms for the $L = 0.1$ slab are accurate to within about 1% RMS relative error. The RMS relative errors for the $L = 1.0$ slab are typically a few percent; the maximum Algorithm A error is 27% for case 2a and the maximum Algorithm B error is 5% for case 1a. The RMS relative errors for the $L = 10.0$ slab are generally somewhat larger; the maximum Algorithm A error is 62% for case 2a and the maximum Algorithm B error is 11% for case 3a. We observe that the largest error in the Algorithm A scalar flux distributions occurs in case 2a. The material zero scalar flux computed by Algorithm A for this case is reasonably accurate (RMS relative error of 2.5%), while the material one scalar flux distribution exhibits large pointwise errors (RMS relative error of 61.9%). For this case, materials zero and one have mean material slab widths of one and ten, respectively. Material one is purely scattering and optically thick, conditions under which we expect Algorithm A to be inaccurate. Material zero is purely absorbing, and hence Algorithm A should be generally accurate in this material. Algorithm B produces significantly more accurate scalar flux distributions for this case.

Overall, we find that Algorithm B generally produces significantly more accurate leakage values than Algorithm A and also significantly more accurate material scalar flux distributions. Both Monte Carlo transport algorithms robustly produce physically-realistic scalar flux distributions for this interior source benchmark problem. These conclusions are generally consistent with results from the incident angular flux benchmark suite.

Table 9: **Suite II leakage value comparisons for case 1**

| | | | | | | | Relative Error $E_{\langle J_0 \rangle, \langle J_L \rangle}$ | | |
|---|---|---|---|---|---|---|---|---|---|
| $L$ | Case | Quantity | Benchmark | Atomic Mix | Algorithm A | Algorithm B | Atomic Mix | Algorithm A | Algorithm B |
| 0.1 | a | $\langle J_0 \rangle$ | 0.48755 | 0.49130 | 0.48756 | 0.48755 | 0.008 | 0.000 | 0.000 |
| | b | $\langle J_0 \rangle$ | 0.46596 | 0.42483 | 0.46580 | 0.46583 | -0.088 | -0.000 | -0.000 |
| | c | $\langle J_0 \rangle$ | 0.49251 | 0.49045 | 0.49242 | 0.49247 | -0.004 | -0.000 | -0.000 |
| 1.0 | a | $\langle J_0 \rangle$ | 0.42497 | 0.43679 | 0.42604 | 0.42541 | 0.028 | 0.003 | 0.001 |
| | b | $\langle J_0 \rangle$ | 0.34733 | 0.20641 | 0.34412 | 0.34704 | -0.406 | -0.009 | -0.001 |
| | c | $\langle J_0 \rangle$ | 0.44668 | 0.43135 | 0.44497 | 0.44707 | -0.034 | -0.004 | 0.001 |
| 10.0 | a | $\langle J_0 \rangle$ | 0.15268 | 0.13745 | 0.16571 | 0.16101 | -0.100 | 0.085 | 0.055 |
| | b | $\langle J_0 \rangle$ | 0.07316 | 0.02697 | 0.06934 | 0.07225 | -0.631 | -0.052 | -0.012 |
| | c | $\langle J_0 \rangle$ | 0.17500 | 0.12954 | 0.17286 | 0.17439 | -0.260 | -0.012 | -0.003 |
| | | | | | | RMS of $E_{\langle J_0 \rangle}$ | 0.269 | 0.034 | 0.019 |

Table 10: **Suite II leakage value comparisons for case 2**

| | | | | | | | Relative Error $E_{\langle J_0 \rangle, \langle J_L \rangle}$ | | |
|---|---|---|---|---|---|---|---|---|---|
| $L$ | Case | Quantity | Benchmark | Atomic Mix | Algorithm A | Algorithm B | Atomic Mix | Algorithm A | Algorithm B |
| 0.1 | a | $\langle J_0 \rangle$ | 0.48746 | 0.49130 | 0.48744 | 0.48746 | 0.008 | -0.000 | 0.000 |
| | b | $\langle J_0 \rangle$ | 0.47026 | 0.42483 | 0.47014 | 0.47012 | -0.097 | -0.000 | -0.000 |
| | c | $\langle J_0 \rangle$ | 0.49245 | 0.49045 | 0.49241 | 0.49243 | -0.004 | -0.000 | -0.000 |
| 1.0 | a | $\langle J_0 \rangle$ | 0.42557 | 0.43679 | 0.42477 | 0.42554 | 0.026 | -0.002 | -0.000 |
| | b | $\langle J_0 \rangle$ | 0.43596 | 0.20641 | 0.43487 | 0.43577 | -0.527 | -0.003 | -0.000 |
| | c | $\langle J_0 \rangle$ | 0.45497 | 0.43135 | 0.45332 | 0.45482 | -0.052 | -0.004 | -0.000 |
| 10.0 | a | $\langle J_0 \rangle$ | 0.19077 | 0.13745 | 0.19535 | 0.19362 | -0.279 | 0.024 | 0.015 |
| | b | $\langle J_0 \rangle$ | 0.29265 | 0.02697 | 0.26787 | 0.29226 | -0.908 | -0.085 | -0.001 |
| | c | $\langle J_0 \rangle$ | 0.31315 | 0.12954 | 0.28362 | 0.31199 | -0.586 | -0.094 | -0.004 |
| | | | | | | RMS of $E_{\langle J_0 \rangle}$ | 0.413 | 0.043 | 0.005 |

Table 11: **Suite II leakage value comparisons for case 3**

| | | | | | | | Relative Error $E_{\langle J_0 \rangle, \langle J_L \rangle}$ | | |
|---|---|---|---|---|---|---|---|---|---|
| $L$ | Case | Quantity | Benchmark | Atomic Mix | Algorithm A | Algorithm B | Atomic Mix | Algorithm A | Algorithm B |
| 0.1 | a | $\langle J_0 \rangle$ | 0.49823 | 0.49904 | 0.49821 | 0.49822 | 0.002 | -0.000 | -0.000 |
| | b | $\langle J_0 \rangle$ | 0.43532 | 0.41922 | 0.43534 | 0.43534 | -0.037 | 0.000 | 0.000 |
| | c | $\langle J_0 \rangle$ | 0.49140 | 0.49045 | 0.49140 | 0.49142 | -0.002 | 0.000 | 0.000 |
| 1.0 | a | $\langle J_0 \rangle$ | 0.48803 | 0.49223 | 0.48803 | 0.48810 | 0.009 | 0.000 | 0.000 |
| | b | $\langle J_0 \rangle$ | 0.29940 | 0.19632 | 0.29928 | 0.29946 | -0.344 | -0.000 | 0.000 |
| | c | $\langle J_0 \rangle$ | 0.43569 | 0.43135 | 0.43574 | 0.43576 | -0.010 | 0.000 | 0.000 |
| 10.0 | a | $\langle J_0 \rangle$ | 0.41136 | 0.37153 | 0.40933 | 0.41550 | -0.097 | -0.005 | 0.010 |
| | b | $\langle J_0 \rangle$ | 0.12966 | 0.02520 | 0.12580 | 0.12944 | -0.806 | -0.030 | -0.002 |
| | c | $\langle J_0 \rangle$ | 0.22563 | 0.12954 | 0.21176 | 0.22570 | -0.426 | -0.061 | 0.000 |
| | | | | | | RMS of $E_{\langle J_0 \rangle}$ | 0.327 | 0.023 | 0.003 |

Table 12: **Suite II scalar flux comparisons for case 1**

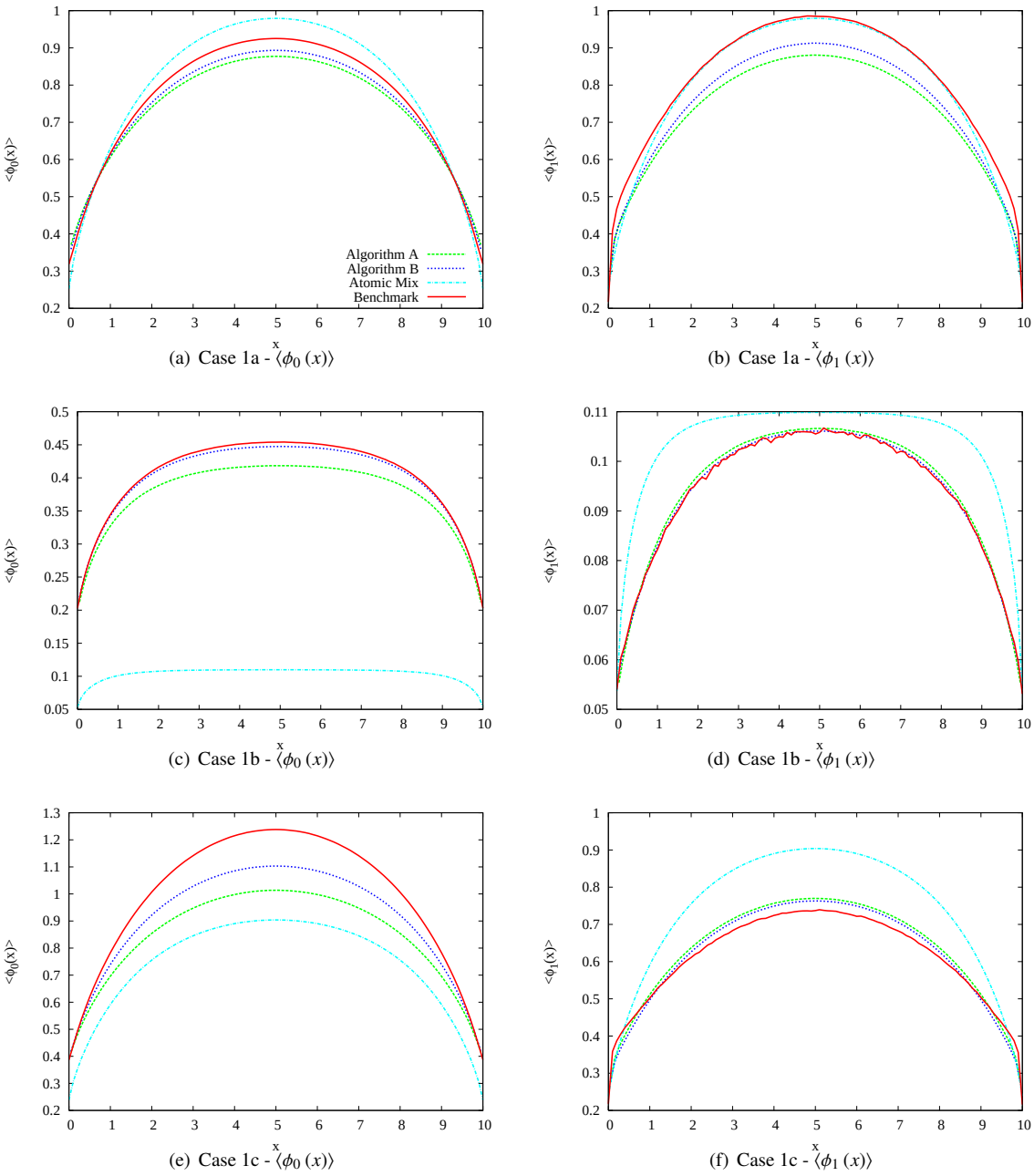
| | | | RMS Relative Error $E_{\langle\phi_i\rangle}$ | | |
|---|---|---|---|---|---|
| $L$ | Case | Quantity | Atomic Mix | Algorithm A | Algorithm B |
| 0.1 | a | $\langle\phi_0\rangle$ | 0.303 | 0.001 | 0.001 |
| | | $\langle\phi_1\rangle$ | 0.282 | 0.003 | 0.000 |
| | b | $\langle\phi_0\rangle$ | 0.407 | 0.001 | 0.001 |
| | | $\langle\phi_1\rangle$ | 1.203 | 0.002 | 0.002 |
| | c | $\langle\phi_0\rangle$ | 0.320 | 0.001 | 0.001 |
| | | $\langle\phi_1\rangle$ | 0.408 | 0.008 | 0.000 |
| 1.0 | a | $\langle\phi_0\rangle$ | 0.164 | 0.016 | 0.006 |
| | | $\langle\phi_1\rangle$ | 0.101 | 0.085 | 0.054 |
| | b | $\langle\phi_0\rangle$ | 0.593 | 0.010 | 0.001 |
| | | $\langle\phi_1\rangle$ | 0.922 | 0.024 | 0.004 |
| | c | $\langle\phi_0\rangle$ | 0.254 | 0.025 | 0.005 |
| | | $\langle\phi_1\rangle$ | 0.397 | 0.057 | 0.021 |
| 10.0 | a | $\langle\phi_0\rangle$ | 0.058 | 0.047 | 0.031 |
| | | $\langle\phi_1\rangle$ | 0.064 | 0.108 | 0.082 |
| | b | $\langle\phi_0\rangle$ | 0.740 | 0.066 | 0.011 |
| | | $\langle\phi_1\rangle$ | 0.137 | 0.012 | 0.004 |
| | c | $\langle\phi_0\rangle$ | 0.262 | 0.150 | 0.086 |
| | | $\langle\phi_1\rangle$ | 0.210 | 0.043 | 0.040 |
| | | RMS of $E_{\langle\phi_0\rangle}$ | 0.398 | 0.058 | 0.031 |
| | | RMS of $E_{\langle\phi_1\rangle}$ | 0.556 | 0.053 | 0.036 |

Table 13: **Suite II scalar flux comparisons for case 2**

| | | | RMS Relative Error $E_{\langle\phi_i\rangle}$ | | |
|---|---|---|---|---|---|
| $L$ | Case | Quantity | Atomic Mix | Algorithm A | Algorithm B |
| 0.1 | a | $\langle\phi_0\rangle$ | 0.308 | 0.001 | 0.001 |
| | | $\langle\phi_1\rangle$ | 0.245 | 0.001 | 0.001 |
| | b | $\langle\phi_0\rangle$ | 0.417 | 0.001 | 0.001 |
| | | $\langle\phi_1\rangle$ | 1.538 | 0.001 | 0.001 |
| | c | $\langle\phi_0\rangle$ | 0.326 | 0.001 | 0.001 |
| | | $\langle\phi_1\rangle$ | 0.409 | 0.001 | 0.000 |
| 1.0 | a | $\langle\phi_0\rangle$ | 0.168 | 0.010 | 0.000 |
| | | $\langle\phi_1\rangle$ | 0.442 | 0.273 | 0.005 |
| | b | $\langle\phi_0\rangle$ | 0.661 | 0.002 | 0.000 |
| | | $\langle\phi_1\rangle$ | 3.567 | 0.014 | 0.004 |
| | c | $\langle\phi_0\rangle$ | 0.286 | 0.006 | 0.000 |
| | | $\langle\phi_1\rangle$ | 0.796 | 0.042 | 0.002 |
| 10.0 | a | $\langle\phi_0\rangle$ | 0.209 | 0.025 | 0.013 |
| | | $\langle\phi_1\rangle$ | 0.621 | 0.620 | 0.076 |
| | b | $\langle\phi_0\rangle$ | 0.912 | 0.082 | 0.001 |
| | | $\langle\phi_1\rangle$ | 1.281 | 0.137 | 0.008 |
| | c | $\langle\phi_0\rangle$ | 0.442 | 0.145 | 0.019 |
| | | $\langle\phi_1\rangle$ | 1.673 | 0.314 | 0.039 |
| | | RMS of $E_{\langle\phi_0\rangle}$ | 0.470 | 0.056 | 0.008 |
| | | RMS of $E_{\langle\phi_1\rangle}$ | 1.526 | 0.253 | 0.029 |

Table 14: **Suite II scalar flux comparisons for case 3**

| | | | RMS Relative Error $E_{\langle\phi_i\rangle}$ | | |
|---|---|---|---|---|---|
| $L$ | Case | Quantity | Atomic Mix | Algorithm A | Algorithm B |
| 0.1 | a | $\langle\phi_0\rangle$ | 0.457 | 0.004 | 0.004 |
| | | $\langle\phi_1\rangle$ | 0.107 | 0.000 | 0.000 |
| | b | $\langle\phi_0\rangle$ | 0.547 | 0.004 | 0.004 |
| | | $\langle\phi_1\rangle$ | 0.251 | 0.000 | 0.000 |
| | c | $\langle\phi_0\rangle$ | 0.469 | 0.004 | 0.004 |
| | | $\langle\phi_1\rangle$ | 0.125 | 0.000 | 0.000 |
| 1.0 | a | $\langle\phi_0\rangle$ | 0.357 | 0.005 | 0.001 |
| | | $\langle\phi_1\rangle$ | 0.081 | 0.013 | 0.001 |
| | b | $\langle\phi_0\rangle$ | 0.750 | 0.001 | 0.000 |
| | | $\langle\phi_1\rangle$ | 0.516 | 0.001 | 0.001 |
| | c | $\langle\phi_0\rangle$ | 0.456 | 0.003 | 0.001 |
| | | $\langle\phi_1\rangle$ | 0.100 | 0.001 | 0.001 |
| 10.0 | a | $\langle\phi_0\rangle$ | 0.461 | 0.099 | 0.077 |
| | | $\langle\phi_1\rangle$ | 0.286 | 0.432 | 0.108 |
| | b | $\langle\phi_0\rangle$ | 0.913 | 0.023 | 0.010 |
| | | $\langle\phi_1\rangle$ | 0.279 | 0.012 | 0.004 |
| | c | $\langle\phi_0\rangle$ | 0.537 | 0.148 | 0.038 |
| | | $\langle\phi_1\rangle$ | 0.379 | 0.063 | 0.023 |
| | | RMS of $E_{\langle\phi_0\rangle}$ | 0.573 | 0.060 | 0.029 |
| | | RMS of $E_{\langle\phi_1\rangle}$ | 0.274 | 0.146 | 0.037 |




(a) Case 1a - $\langle \phi_0(x) \rangle$

(b) Case 1a - $\langle \phi_1(x) \rangle$

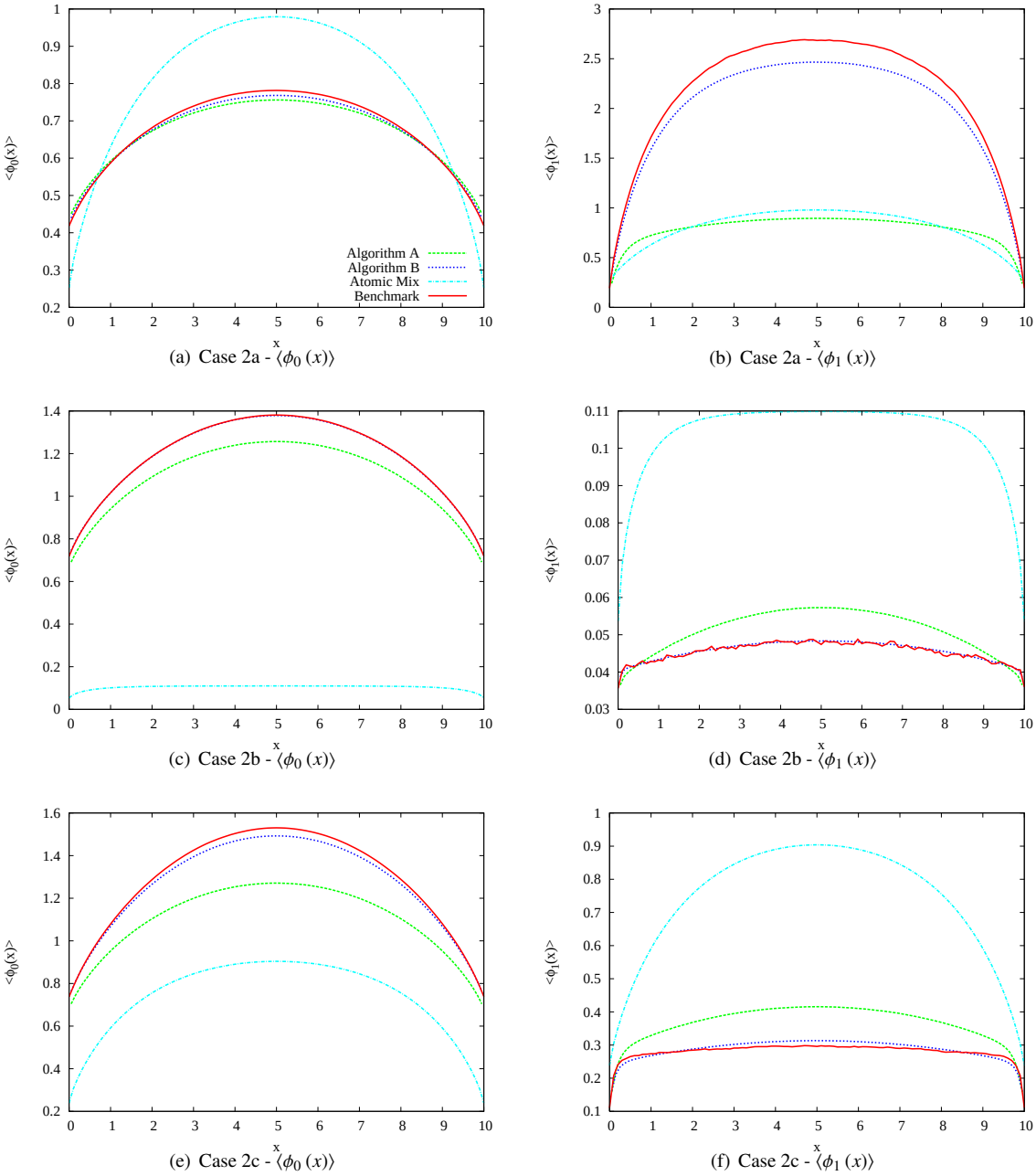
(c) Case 1b - $\langle \phi_0(x) \rangle$

(d) Case 1b - $\langle \phi_1(x) \rangle$

(e) Case 1c - $\langle \phi_0(x) \rangle$

(f) Case 1c - $\langle \phi_1(x) \rangle$

Figure 4: **Suite II scalar flux distributions for case 1 and L = 10**




(a) Case 2a - $\langle \phi_0(x) \rangle$

(b) Case 2a - $\langle \phi_1(x) \rangle$

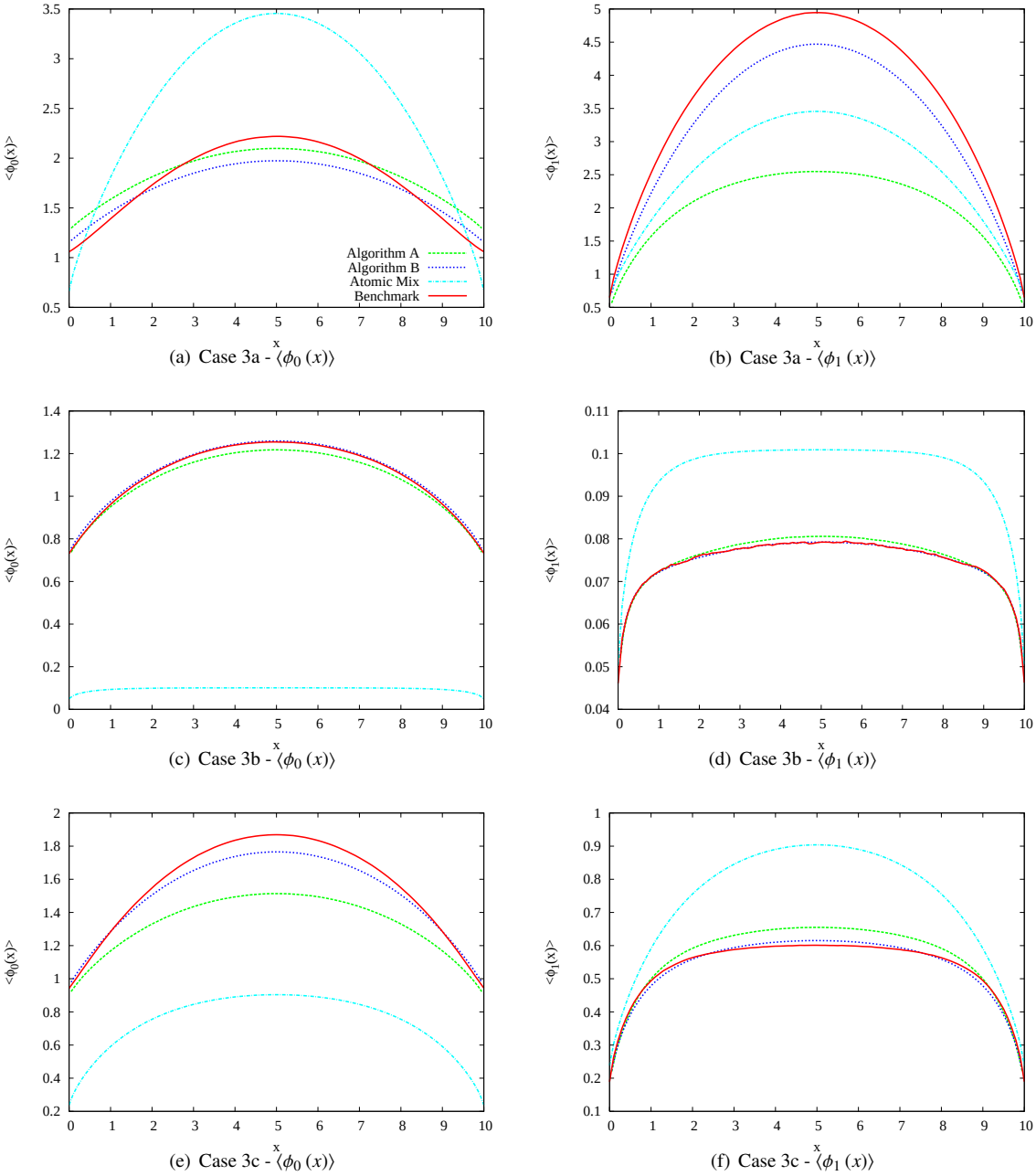
(c) Case 2b - $\langle \phi_0(x) \rangle$

(d) Case 2b - $\langle \phi_1(x) \rangle$

(e) Case 2c - $\langle \phi_0(x) \rangle$

(f) Case 2c - $\langle \phi_1(x) \rangle$

Figure 5: **Suite II scalar flux distributions for case 2 and L = 10**




(a) Case 3a - $\langle \phi_0(x) \rangle$

(b) Case 3a - $\langle \phi_1(x) \rangle$

(c) Case 3b - $\langle \phi_0(x) \rangle$

(d) Case 3b - $\langle \phi_1(x) \rangle$

(e) Case 3c - $\langle \phi_0(x) \rangle$

(f) Case 3c - $\langle \phi_1(x) \rangle$

Figure 6: **Suite II scalar flux distributions for case 3 and L = 10**

## 5. Conclusions

We have numerically investigated the accuracy of two Monte Carlo algorithms originally proposed by Zimmerman [1] and Zimmerman and Adams [2] for particle transport through a binary stochastic mixture. The accuracy of these algorithms was assessed by comparing to two suites of planar geometry binary stochastic mixture benchmark problems. Suite I is a standard suite of benchmark problems driven by an isotropic incident angular flux on one boundary [5]; Suite II is driven by an isotropic interior source [12]. In addition to comparisons of the ensemble-averaged leakage values, we compared the ensemble-averaged material scalar flux distributions. We also presented deterministic atomic mix solutions of the benchmark problems for comparison with the benchmark and the Monte Carlo solutions.

Although relatively simple and computationally efficient, the atomic mix approximation produces generally inaccurate results overall for the leakage values and the material scalar flux distributions for both benchmark Suites I and II. The atomic mix approximation does not appear to possess enough accuracy for use with binary stochastic mixture transport problems characterized by material parameters such as examined in this benchmark study.

Both Monte Carlo Algorithms A and B produce qualitatively and semi-quantitatively correct results for the leakage values and the material scalar flux distributions for both benchmark suites. Our numerical results demonstrate that Algorithm B generally produces significantly more accurate leakage values than Algorithm A and also significantly more accurate material scalar flux distributions. Both Monte Carlo transport algorithms robustly produce physically-realistic material scalar flux distributions for the transport problems examined. This attribute of these algorithms is important, as these scalar flux distributions determine reaction rates in the materials of the system. Both Monte Carlo algorithms are generally significantly more accurate than the atomic mix approximation for the benchmark suites examined.

The leakage values predicted by both Monte Carlo algorithms are significantly more accurate for the interior source benchmark suite than for the incident angular flux benchmark suite. However, the material scalar flux distributions (particularly for Algorithm A) can exhibit significantly larger errors than the leakage values in both benchmark suites. This discrepancy underscores the need to investigate the accuracy of both leakage values and scalar flux distributions in fully assessing algorithms for these types of problems.

The material parameters used in the benchmark suite examined in this work are the same as used in previous incident angular flux benchmark suite investigations [2, 5, 6, 15]. For some mean material slab width and total slab width values used in the benchmark suite, a small number of distinct material slabs are present in typical realizations (although some realizations have significantly larger numbers). Examining additional ranges of material parameters and larger slab widths may be beneficial in further assessing the relative accuracy of Monte Carlo Algorithms A and B.

In future work, we would like to investigate the accuracy of these Monte Carlo algorithms for additional benchmark problems. In particular, Davis, Palmer, and Larsen [22] compared the accuracy of the atomic mix and Levermore-Pomraning models for a benchmark suite composed of binary solid-void stochastic mixtures motivated by the modeling of pebble bed reactors. We would like to investigate the accuracy of Monte Carlo Algorithms A and B for this benchmark suite. In addition, research by Larsen, Vasques, and Vilhena [23] has demonstrated that the Levermore-Pomraning model (and hence Algorithm A) does not possess the correct asymptotic limit for diffusive problems that are optically thick with weak absorption and sources. Investigating the accuracy of Algorithm B for these types of problems would be interesting. Finally, higher-order closure models have been investigated for the Levermore-Pomraning deterministic model that produce increased accuracy over the standard closure [24]. Comparing the results obtained using the higher-order closure models and these Monte Carlo algorithms would be interesting.

## Acknowledgments

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344. The author would like to acknowledge helpful conversations with George Zimmerman and Brian Yang. The author is indebted to Olivier Zuchuat, who went to extraordinary measures to retrieve the available scalar flux distribution data previously published in graphical form in Ref. [6].

## References


[1] Zimmerman GB. "Recent Developments in Monte Carlo Techniques," Lawrence Livermore National Laboratory Report UCRL-JC-105616 (1990).

[2] Zimmerman GB, Adams ML. "Algorithms for Monte-Carlo Particle Transport in Binary Statistical Mixtures," *Trans. Am. Nucl. Soc.*, San Francisco, CA, November 10-14, 1991, Vol. 64, p. 287 (1991).

[3] Levermore CD, Pomraning GC, Sanzo DL, Wong J. "Linear transport theory in a random medium," *J. Math. Phys.*, **27**, pp. 2526-2536 (1986).

[4] Pomraning GC. *Linear Kinetic Theory and Particle Transport in Stochastic Mixtures*, World Scientific Publishing Co. Pte. Ltd., River Edge, New Jersey USA (1991).

[5] Adams ML, Larsen EW, Pomraning GC. "Benchmark Results for Particle Transport in a Binary Markov Statistical Medium," *J. Quant. Spectrosc. Radiat. Transfer*, **42**, pp. 253-266 (1989).

[6] Zuchuat O, Sanchez R, Zmijarevic I, Malvagi F. "Transport in Renewal Statistical Media: Benchmarking and Comparison With Models," *J. Quant. Spectrosc. Radiat. Transfer*, **51**, pp. 689-722 (1994).

[7] Murata I, Mori T, Nakagawa M. "Continuous Energy Monte Carlo Calculations of Randomly Distributed Spherical Fuels in High-Temperature Gas-Cooled Reactors Base on a Statistical Geometry Model," *Nucl. Sci. Engr.*, **123**, pp. 96-109 (1996).

[8] Donovan TJ, Danon Y. "Application of Monte Carlo Chord-Length Sampling Algorithms to Transport Through a Two-Dimensional Binary Stochastic Mixture," *Nucl. Sci. Engr.*, **143**, pp. 226-239 (2003).

[9] Donovan TJ, Danon Y. "Implementation of Chord Length Sampling for Transport Through a Binary Stochastic Mixture," *Proceedings of the Nuclear Mathematical and Computational Sciences: A Century in Review, A Century Anew*, Gatlinburg, TN, April 6-11, 2003, on CD-ROM, American Nuclear Society, La Grange Park, IL (2003).

[10] Donovan TJ, Danon Y. "HTGR Unit Fuel Pebble k-infinity Results Using Chord Length Sampling," *Trans. Am. Nucl. Soc.*, New Orleans, LA, November 16-20, 2003, Vol. 89, pp. 37-39 (2003).

[11] Ji W, Martin WR. "Monte Carlo Simulation of VHTR Particle Fuel with Chord Length Sampling," *Joint International Topical Meeting on Mathematics & Computation and Supercomputing in Nuclear Applications (M&C + SNA 2007)*, Monterey, CA, April 15-19, 2007, on CD-ROM, American Nuclear Society, La Grange Park, IL (2007).

[12] Brantley PS, Palmer TS. "Levermore-Pomraning Model Results for an Interior Source Binary Stochastic Medium Benchmark Problem," *Proceedings of the International Conference on Mathematics, Computational Methods & Reactor Physics (M&C 2009)*, Saratoga Springs, New York, May 3-7, 2009, on CD-ROM, American Nuclear Society, La Grange Park, IL (2009).

[13] Vasques R. "A Review of Particle Transport Theory in a Binary Stochastic Medium," Masters Dissertation, Universidade Federal Do Rio Grande Do Sul (2005).

[14] Lewis EE, Miller WF. *Computational Methods of Neutron Transport*, American Nuclear Society, La Grange Park, Illinois, USA (1993).

[15] Brantley PS. "A Comparison of Monte Carlo Particle Transport Algorithms for Binary Stochastic Mixtures," *Proceedings of the International Conference on Mathematics, Computational Methods & Reactor Physics (M&C 2009)*, Saratoga Springs, New York, May 3-7, 2009, on CD-ROM, American Nuclear Society, La Grange Park, IL (2009).

[16] Brantley PS. "A Comparison of Monte Carlo Particle Transport Algorithms for an Interior Source Binary Stochastic Medium Benchmark Suite," *Trans. Am. Nucl. Soc.*, Washington D.C., November 15-19, 2009, on CD-ROM (2009).

[17] Miller P. "pyMPI - An Introduction to Parallel Python Using MPI ," Lawrence Livermore National Laboratory report UCRL-WEB-151052 (2002). See also http://pympi.sourceforge.net (2009).

[18] Gropp W, Lusk E, Skjellum A. *Using MPI: Portable Parallel Programming with the Message-Passing Interface*, The MIT Press, Cambridge, Massachusetts, USA (1999).

[19] Zuchuat O. Personal communication (2008).

[20] Sahni DC. "Equivalence of Generic Equation Method and the Phenomenological Model for Linear Transport Problems in a Two-State Random Scattering Medium," *J. Math. Phys*, **30**, pp. 1554-1559 (1989).

[21] Zimmerman GB. Personal communication (2009).

[22] Davis IM, Palmer TS, Larsen EW. "A Comparison of Binary Stochastic Media Transport Models in "Solid-Void" Mixtures," *PHYSOR 2004 - The Physics of Fuel Cycles and Advanced Nuclear Systems: Global Developments*, Chicago, Illinois, April 25-29, 2004, on CD-ROM, American Nuclear Society, La Grange Park, IL (2005).

[23] Larsen EW, Vasques R, Vilhena MT, "Particle Transport in the 1-D Diffusive Atomic Mix Limit," *Proceedings of the International Topical Meeting (M&C 2005) on Mathematics and Computation, Supercomputing, Reactor Physics and Nuclear and Biological Applications*, Palais des Papes, Avignon, France, September 12-15, 2005, on CD-ROM, American Nuclear Society, La Grange Park, IL (2005).

[24] Su B, Pomraning GC. "Modification to a Previous Higher Order Model for Particle Transport in Binary Stochastic Media," *J. Quant. Spectrosc. Radiat. Transfer*, **54**, pp. 779-801 (1995).