Differentiable Neutron Transport

XI DENG, MAOSEN TANG, MICHAEL CZEKANSKI, DAVID BINDEL, and STEVE MARSCHNER, Cornell University, USA

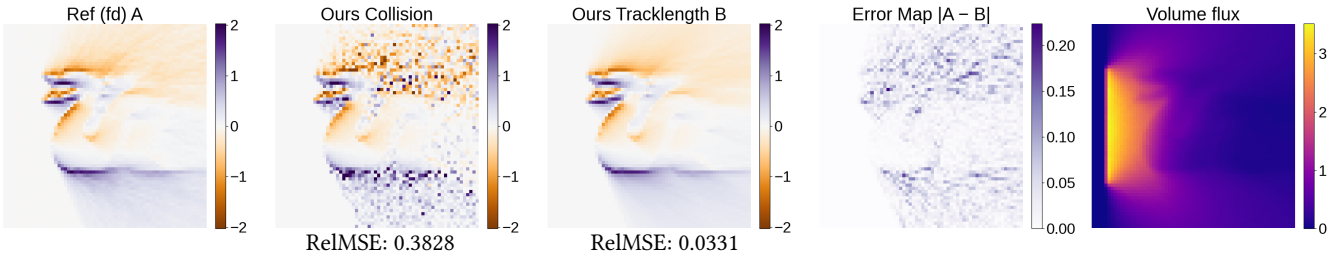


Fig. 1. Gradient of volume flux (attenuation) for a geometric parameter (vertical translation) in 3D space, computed by auto-differentiating our Monte Carlo neutron transport simulation, which is built using our new differentiable estimators. The scene consists of two materials separated by a boundary defined by the dragon shape. We show the relative mean square error over collision and tracklength estimators against the finite differences.

Neutron transport simulation plays a central role in nuclear engineering, yet its inverse problems—such as shielding design and sensitivity analysis—remain extremely challenging. The governing transport equation shares a common mathematical foundation with the volume rendering equation used for participating media in computer graphics. Recent advances in differentiable rendering have demonstrated that inverse problems—such as recovering scene properties from observations—can be efficiently solved through gradient-based optimization, where the gradients are estimated through Monte Carlo methods.

Inspired by these developments, we extend differentiable rendering techniques to neutron transport problems, enabling inverse design and sensitivity analysis in nuclear systems. To adapt these methods to the neutron transport problem, we generalize the automatic differentiation of Monte Carlo light transport simulation to multi-energy neutron transport; propose a new simplified way to handle moving geometric discontinuities; and propose new estimators for derivatives of volume integrals. We demonstrate the effectiveness of the proposed framework on several model inverse problems, including gradient-based optimization of multi-group neutron moderation, sensitivity analysis of material properties, and diagnostic sensor shape optimization. These results show that differentiable Monte Carlo neutron transport can support optimization and analysis tasks, opening new opportunities for inverse design in nuclear shielding and related applications.

CCS Concepts: • **Computing methodologies** → **Ray tracing**.

Additional Key Words and Phrases: Rendering, Differentiable Rendering, Neutron Transport Simulation

ACM Reference Format:
Xi Deng, Maosen Tang, Michael Czekanski, David Bindel, and Steve Marschner. 2026. Differentiable Neutron Transport. In *Proceedings of ACM SIGGRAPH 2026 (SIGGRAPH '26).* ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3811296

Authors' Contact Information: Xi Deng, xd93@cornell.edu; Maosen Tang, mt872@cornell.edu; Michael Czekanski, mc2589@cornell.edu; David Bindel, bindel@cornell.edu; Steve Marschner, srm@cs.cornell.edu, Cornell University, Ithaca, New York, USA.

SIGGRAPH '26, Los Angeles, CA, USA

ACM ISBN 978-1-4503-XXXX-X/2018/06
https://doi.org/10.1145/3811296

1 Introduction

Nuclear reactors, including both fission- and fusion-based systems, generate large fluxes of high-energy neutrons that propagate through surrounding materials. Capturing and moderating these neutrons is essential not only for efficient energy production and system diagnostics, but also for maintaining structural integrity and operational safety. To this end, it is crucial to understand how the spatial, angular, and energy distributions of measured neutrons depend on design parameters such as source characteristics, material properties, and the configuration of diagnostic devices. Selecting these underlying scene parameters to achieve desired neutron measurements constitutes what we refer to as the inverse neutron transport problem.

Although neutron transport [Davison 1958] arises in a very different physical context than computer graphics, it shares a common mathematical foundation with light transport in participating media. Both describe the propagation of particles through space governed by probabilities of different reactions (like emission, absorption, and scattering) along their paths. While forward simulation is well studied, the corresponding inverse problems—such as determining optimal material configurations or quantifying the impact of uncertain material properties—remain significantly challenging. In particular, understanding how variations in material composition and geometry affect energy distribution is critical for reactor design, safety assessment, and performance optimization. These challenges motivate the development of inverse methods capable of computing sensitivities with respect to both geometry and material parameters, which is the focus of this work.

Conventional approaches to the inverse problems include adjoint neutron transport [Hoogenboom 1977; Saracco et al. 2016], which computes an importance function that quantifies how neutrons at each phase-space point contribute to a specified response, as well as

finite-difference Monte Carlo methods, which require a large number of forward simulations to probe the response landscape. More recently, data-driven [Li et al. 2024; Sahadath et al. 2025; Zhu et al. 2026], gradient-based approaches have also emerged. However, adjoint methods become increasingly difficult to apply in scenes with complex geometries and multiple responses, while data-driven surrogate models often struggle to provide accurate gradients in high-dimensional parameter spaces and lack physical guarantees outside their training regimes. To date, there exists no auto-differentiable Monte Carlo neutron transport simulator that provides efficient and unbiased gradient estimates with respect to both complex geometry and material parameters.

This paper addresses that gap by building a theoretical framework and providing a gradient-based method for inverse neutron transport. Unlike previous work focused on simple parameters like the thickness of a shield [Wang et al. 2024b], our approach encompasses the full range of possible shapes, opening up more design dimensions for this problem. Our key idea is to generalize the concept of differentiable rendering [Li et al. 2018; Loubet et al. 2019; Zhang et al. 2019] to differentiable neutron transport.

We show that extending the mathematical framework of differentiable light transport to neutron transport simultaneously simplifies and complicates the problem. On the one hand, differentiation with respect to geometric parameters becomes simpler: neutrons predominantly transmit across material boundaries, and complications arising from reflection and refraction—central in light transport—do not arise. On the other hand, neutron transport is substantially more challenging because it predicts a spatial neutron flux field, whose dimensionality is higher than the two-dimensional image space typical in inverse light transport. The material parameters that it takes as input are also energy dependent. These higher-dimensional input and output spaces necessitate the development of more efficient estimators, which must also remain fully differentiable. In differentiable rendering, geometry is typically represented as the shape of visible objects and interfaces, such as meshes, and primarily affects visibility. In neutron transport, by contrast, geometry is more often represented as a partition of space into material regions, using constructive solid geometry(CSG). Making CSG directly differentiable, however, is nontrivial. Thus neutron transport is not merely an application, but a foundational extension of differentiable transport theory—one that opens a new domain in which the mathematical and algorithmic tools of computer graphics can naturally evolve.

To the best of our knowledge, this is the first Monte Carlo neutron transport framework that is fully differentiable with respect to both geometry and energy-dependent material parameters via automatic differentiation. Achieving that, our contributions include:

- A novel method for computing radiation gradients that avoids sampling of geometric boundaries.
- An algorithm for computing gradients via traditionally non-differentiable track-length estimators, ensuring smoother gradient estimation for volume flux.
- A formulation of the differentiable neutron transport simulation supporting energy-dependent scattering.
- An alternative engineering solution that preserves the convenience of CSG-based material geometry while maintaining differentiability with respect to mesh parameters, without involving a complex CSG solver.

We show three applications of gradients in solving inverse neutron transport problems, including shielding design, sensor design, and sensitivity analysis.

2 Related Work

Neutron Transport. Monte Carlo methods are widely used in neutron transport [Davison 1958], particularly in complex scenarios such as neutron slow-down, where fast neutrons are moderated to thermal energies through a sequence of collision events. Several Monte Carlo simulators are available for neutron transport, including OpenMC [Romano et al. 2015], MCNP [Kulesza et al. 2024], and Tripoli-4 [Petit et al. 2008]. While these tools are highly effective for forward simulation, they offer limited support for inverse problems and differentiable analysis. Our work focuses on providing a theoretical framework that enables automatic differentiation of Monte Carlo estimators with respect to both geometry and material parameters.

Classical estimators—including the collision estimator, the track-length estimator, and their variants [Spanier and Gelbard 2008]—have also been adopted in computer graphics in the form of photon beams [Habel et al. 2013; Křivánek et al. 2014], photon planes [Bitterli and Jarosz 2017], and photon surfaces [Deng et al. 2019], enabling efficient volumetric and time-of-flight rendering [Liu et al. 2022]. Unlike light transport, where an additional integration dimension is required to make photon primitives unbiased estimators, the track-length estimator in neutron transport is naturally unbiased for estimating volume flux. Our goal here, however, is to make these estimators differentiable with respect to geometry and material parameters.

Inverse Neutron Transport. A wide range of inverse problems arise in neutron transport. These include sensor design [Sjoden 2002; Wang et al. 2024b; Yi et al. 2012], where adjoint methods and Monte Carlo simulations are used to optimize sensor geometry for maximal response; shielding optimization [Chen and Yan 2023; Rockwell 1956; Yun et al. 2019], which typically assumes simplified geometries and relies on Monte Carlo analysis to evaluate candidate designs; and sensitivity analysis of nuclear data [de la Lluvia et al. 2026; Leray et al. 2017; Rochman et al. 2014]. Classical sensitivity approaches typically derive estimators for each class of parameters with theory in [de Lataillade et al. 2002], and [Roger et al. 2005], whereas our formulation expresses the transport estimator in a differentiable form and obtains gradients through automatic differentiation. As a result, derivatives with respect to multiple parameters can be accumulated simultaneously within a single Monte Carlo simulation using the same set of sampled paths.

Physically-based Differentiable Rendering. Starting from [Li et al. 2018], which estimates geometry gradients of pixel values by explicitly sampling discontinuities in the integrand of the path integral, the main focus of physically based differentiable rendering has been to handle discontinuities arising from geometry and visibility terms. In general, existing approaches fall into two categories. The first category [Zhang et al. 2021a, 2020, 2019] reformulates the path-space

integral into a material-space integral and separates the gradient into interior and boundary contributions by generalizing the Leibniz integral rule. The second category [Bangaru et al. 2020; Loubet et al. 2019; Vicini et al. 2021; Zeltner et al. 2021] reparameterizes the integral such that the integrand has well-defined geometry derivatives that smoothly interpolate boundary motion. To efficiently sample discontinuity boundaries, [Wang et al. 2024a; Zhang et al. 2023] collect samples near discontinuities during the forward rendering pass and project or reweight them during gradient computation. Our surface flux gradient computation falls into the reparameterization category; however, unlike previous methods, our problem setup allows us to move all gradient terms entirely into the integrand, eliminating all explicit handling of boundaries or edges.

Differentiable Volumetric Rendering. [Gkioulekas et al. 2013] derives analytic gradients with respect to participating-media parameters represented on volume grids. Subsequently, [Yu et al. 2023; Zhang et al. 2021b] enable geometry gradients in scenes containing participating media by formulating the problem in path space and discretizing volumes using tetrahedralization. [Nimier-David et al. 2022] introduces differential ratio tracking to compute unbiased gradients of volumetric properties defined on voxel grids. In the context of subsurface scattering, [Deng et al. 2022] estimates gradients with respect to both geometry and volumetric parameters by modeling the volume beneath refractive surfaces using a diffusion approximation. In contrast, our method jointly computes gradients with respect to both geometry and material parameters in scenes containing piecewise-homogeneous participating media separated by surfaces. The scattering and absorption properties are energy-dependent, and energy shifts during scattering are fully supported. Our approach does not require tetrahedralizing the volume or sampling the boundary, and it accommodates arbitrary geometry representations, requiring only ray intersection queries.

Applications. Differentiable light transport through volumes has been applied to a range of inverse problems, including reconstructing complex materials such as leaves [Deng et al. 2024], designing projection patterns for volumetric 3D printing [Nicolet et al. 2024], plasma visualization [Öztürk et al. 2025], and atmospheric sensing [He et al. 2026]. Our work shares this general gradient-based inverse optimization framework, but differs in the governing scattering law, the role of energy dependence, and the physical quantity predicted by the forward solver.

3 Overview

In nuclear reactor design, neutron transport is treated separately from the underlying physics of neutron generation and microscopic material interactions. We assume that the neutron source is known and specified as a prescribed function, and that neutron–material interactions are characterized statistically through known scattering distributions. Under these assumptions, the neutron transport problem reduces to predicting the statistical spatial, angular, and energy distributions of neutrons after emission and multiple scattering, given the source, the geometry and the properties of surrounding materials. Mathematically, this problem is closely analogous to light transport: given a source, scene geometry, and optical material properties, the distribution of photons is obtained by physically simulating the transport of light.

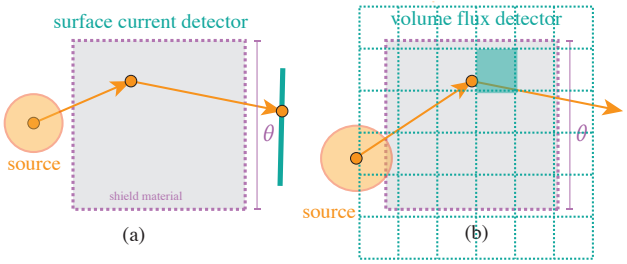


Fig. 2. Problem setup with an emission source, a shielding material, and two measurement types: (a) surface current on the detector and (b) volumetric flux.

Although neutron transport is mathematically analogous to light transport, there are important differences between the two problems. First, neutrons exist at multiple energy levels, and their scattering behavior is strongly energy dependent. Second, the detectors in neutron transport are typically not limited to an image plane, as is common in light transport simulations. Depending on the analysis objective, neutron transport may be solved to compute (a) the energy leakage at the surface of a shielding structure (surface integral), (b) the spatial distribution of neutrons (volume integral), or (c) the energy distribution of neutrons. Each of these quantities corresponds to a different type of measurement process, analogous to different types of "cameras" in light transport. Third, surface effects, such as reflection and refraction, are not important in neutron transport.

The inverse problem, in this context, consists of inferring scene parameters—such as source characteristics, material properties, or geometric configurations—to achieve desired values of these measurements. When gradients of these measured quantities with respect to the underlying parameters are available, gradient-based optimization methods, including stochastic gradient descent, can be employed to efficiently solve such inverse problems.

This work focuses on deriving differentiable estimators for neutron transport with emphasis on gradients of two commonly used measurements: the surface current ($\mathbf{J}$), defined as the net flow of neutrons across a surface, and the volumetric flux (Φ), defined as a volume integral of neutron density (see definitions in Table 1).

Fig. 2 illustrates a simple problem setup: a neutron source is placed in space, and a shielding material (in the purple shape) blocks the fast neutrons emitted from the source. Let $N(\mathbf{x}, \omega)$ be the neutron radiance representing the angular neutron density, marginalized over energy. Two measurements we are considering here are:

Current. ($\mathbf{J}$) Current, analogous to irradiance in rendering, is the net count of neutrons crossing an oriented surface (shown in Fig. 2(a), marked green), which is the integral of neutron radiance over the area A of the surface and over all directions of propagation Ω:

$$\mathbf{J} = \int_A \int_\Omega N(\mathbf{x}, \omega)(\mathbf{n}(\mathbf{x}) \cdot \omega) \mathrm{d}\mathbf{x} \mathrm{d}\omega. \tag{1}$$

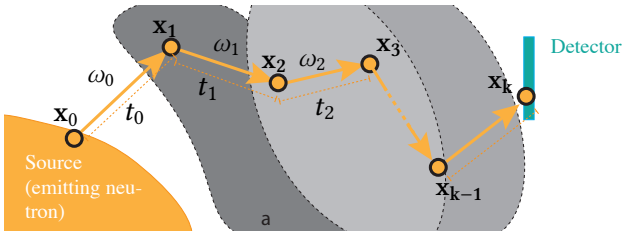


Fig. 3. Path Sample formulated in sequence of directions and distances

This measurement is important because it gives us information about neutron leakage at the surface of a shield or at the interface of a detector that counts all neutrons absorbed within a volume. Understanding how each parameter in the scene affects this quantity is crucial.

Flux. (Φ) Flux is average neutron density within a volume (shown in Fig. 2(b), marked green), integrated over all propagation directions:

$$\Phi = \frac{1}{\mathcal{V}} \int_{\mathcal{V}} \int_{\Omega} N(\mathbf{x}, \omega) \mathrm{d}\mathbf{x} \mathrm{d}\omega. \tag{2}$$

Flux is important because relevant reactions, such as absorption (converting a neutron's energy to heat) and Tritium breeding (fissioning Lithium to produce Tritium) happen at collisions, and flux is proportional to the collision density through the macroscopic total cross section, which gives the collision rate $\sigma_t \Phi$. Measuring the flux and its gradient for the scene parameters helps us understand how the outcomes of these reactions are affected by the scene parameters.

Both types of measurements can be expressed as integrals of the neutron density against a score function $W(\mathbf{x}, \omega)$ (of the detector) over all directions and the detector domain $\mathcal{V}_d$:

$$\mathbf{I} = \int_{\mathcal{V}_d} \int_{\Omega} N(\mathbf{x}, \omega) W(\mathbf{x}, \omega) \mathrm{d}\mathbf{x} \mathrm{d}\omega, \tag{3}$$

The distribution N arises from neutrons that start at the source, travel through space, and undergo scattering in the material. To model this process we consider paths like the one shown in Fig. 3. The neutrons travel in straight lines with constant velocity before collision events occur. The points $\mathbf{x}_i$ for $0 < i < k$ represent the locations of collision events. When a collision happens, the neutron either is absorbed or is scattered to travel at another velocity. With this model of neutron propagation, the neutron density $N(\mathbf{x}_k, \omega)$ can be computed by summing over all paths weighted by the emission $E(\mathbf{x}_0)$ at the start and the throughput of the path. The weighting function $f(\mathbf{x}_0 \dots \mathbf{x}_k, \theta)$ assigns a weight to each neutron path, accounting for attenuation between collisions, energy- and direction-dependent scattering probabilities, and geometric factors arising from the probability measure over paths. The measurement $\mathbf{I}_k$ due to paths of length k is an integral over this neutron path space:

$$\mathbf{I}_k = \int_{\Gamma} E(\mathbf{x}_0) f(\mathbf{x}_0 \dots \mathbf{x}_k, \theta) W(\mathbf{x}_k) \mathrm{d}\mathbf{x}_k \dots \mathrm{d}\mathbf{x}_0. \tag{4}$$

Symbol	Description
$N(\mathbf{x}, \omega)$	Neutron Radiance, angular neutron density, marginalized over energy.
$\mathbf{J}$	Current, the net neutron flow across a surface (unit: #neutron $cm^{-2}s^{-1}$).
Φ	Flux, the total distance neutrons propagate per unit volume per unit time (unit: #neutron $cm^{-2}s^{-1}$).
$\mathbf{I}$	General Measurement.
$\mathbf{x}$	Point position in Cartesian coordinate system.
t	Distance in Cartesian coordinate system.
ω	Solid Angle.
Ω	Solid angle space.
Γ	Conventional path space.
Ξ	Direction–distance path space.
$\bar{\mathbf{d}}$	Path in direction–distance space, where $\bar{\mathbf{d}} = \mathbf{x}_0, \omega_0, t_0, \dots, \omega_{k-2}, t_{k-2}, \mathbf{x}_k$.
$f(\bar{\mathbf{d}})$	Path contribution function in direction–distance space.
$f_\omega(\omega_i, \omega_{i+1})$	Scattering term between direction ω_i and ω_{i+1}.
$f_t(t)$	Attenuation along distance t.
$E(\mathbf{x})$	Emission function.
$W(\mathbf{x})$	Sensor score function.
$\mathrm{d}\mathcal{V}(\mathbf{x})$	Infinitesimal volume element.
r	Distance along the ray to the nearest surface.
θ	Scene parameters, including the geometry parameter (vertices position, transformation) and the material properties (like total cross section σ_t, scattering cross section σ_s)

Table 1. List of symbols and definitions.

In practice, E can depend on the direction towards $\mathbf{x}_1$ and W can depend on the direction from $\mathbf{x}_{k-1}$, but for simplicity of notation we assume these functions are isotropic in the following derivations.

Some particular challenges when differentiating this quantity are (a) that $f(\mathbf{x}_0 \dots \mathbf{x}_k, \theta)$ contains discontinuities whose positions depend on θ, and (b) that we need to compute estimates for large collections of score functions $W(\mathbf{x}_k)$ that arise from voxel grids.

As shown in Fig. 3, a neutron travels from $\mathbf{x}_i$ to $\mathbf{x}_{i+1}$ along direction ω_i by distance t_i. In our case, the weighting function is only piecewise continuous, and it has jumps not just at the boundary of the shielding geometry but also where different materials meet in the interior. We use the boundary to refer both to the boundary of the geometry and to the interfaces between materials (marked as purple lines). The integration domain is defined by path vertices being inside the boundaries of the materials. But it is equivalent to consider the vacuum outside the materials as another material, turning the outer boundary into another interior discontinuity, so these two types of boundaries can be handled in the same way.

Previous work like Yu et al. [2023] and Zhang et al. [2021b] computes geometric gradients by decomposing into an interior integral and a boundary integral. This approach requires additional boundary sampling and also does not directly cover the case where there

are material interface boundaries in the interior, like the multi-layer material configurations considered in this work. In contrast, we present a simpler method for estimating shape gradients that avoids explicit boundary integrals. By applying a suitable reparameterization, the geometry dependence can be handled entirely within the interior integral. To introduce this reparameterization, we first express the neutron transport path integral in a direction–distance formulation.

Any collision point $\mathbf{x}_i$ $(i < k)$ on the path can be expressed as the result of a sequence of offsets from the emission point $\mathbf{x}_0$ as follows:

$$\mathbf{x}_i(\theta) = \mathbf{x}_0 + \sum_{j=0}^{i-1} t_j(\theta)\omega_j(\theta). \tag{5}$$

Therefore, we write a neutron path as $(\mathbf{x}_0, t_0, \omega_0, \ldots, t_{k-2}, \omega_{k-2}, \mathbf{x}_k)$ with $t_i = \|\mathbf{x}_{i+1} - \mathbf{x}_i\|$ and $\omega_i = t_i^{-1}(\mathbf{x}_{i+1} - \mathbf{x}_i)$. We will show that this formulation is easier to work with in the problem of differentiating neutron transport.

With this parameterization, we can write a path $\bar{\mathbf{d}}_k$ of length k as a sequence of directions $\bar{\omega}_k = \omega_0 \cdots \omega_{k-2}$, a sequence of distances $\bar{t}_k = t_0 \cdots t_{k-2}$, and two points residing on the source $\mathbf{x}_0$ and sensor $\mathbf{x}_k$, respectively:

$$\bar{\mathbf{d}}_k = \{\mathbf{x}_0, \bar{\omega}, \bar{t}, \mathbf{x}_k\}. \tag{6}$$

Note that $\bar{\mathbf{d}}_1 = \mathbf{x}_0\mathbf{x}_1$ represents the neutron path with no collisions. This forms the modified path space Ξ_k with $k-1$ collision events as the Cartesian product of directional space $\Xi^\omega(\bar{\omega} \in \Xi^\omega)$, distance space Ξ^t $(\bar{t} \in \Xi^t)$, the space of the source A_d $(\mathbf{x}_0 \in A_d)$, and the space of the sensor S_d $(\mathbf{x}_k \in S_d)$: $\Xi_k = A_d \times S_d \times \Xi^\omega \times \Xi^t$.

The corresponding weighting function for path $\bar{\mathbf{d}}_k$ in this direction-distance formulation is :

$$f(\bar{\mathbf{d}}_k) = E(\mathbf{x}_0)W(\mathbf{x}_k)\prod_{i=1}^{k-1} f_\omega(\omega_i)\prod_{i=0}^{k-1} f_t(t_i), \tag{7}$$

where $f_t(t_i) = e^{-\int_0^{t_i}\sigma_t(t)\mathrm{d}t}$ and $f_\omega(\omega_i, \theta) = \sigma_s f_p(\mathbf{x}_i, \omega_i, \omega_{i-1}, \theta)$ is the phase function which can depend on θ. Note that $f_t(t_i) = e^{-t_i\sigma_t}$ in a homogeneous volume. We use $f_t(\bar{t}_k, \theta)$ and $f_\omega(\bar{\omega}_k, \theta)$ to abbreviate $\prod_{i=1}^{k-1} f_\omega(\omega_i, \theta)$ and $\prod_{i=0}^{k-1} f_t(t_i, \theta)$. Then we write the path space integral using the distance and direction formulation. Given a source with volume A_d and a sensor with volume S_d, the total measurement $\mathbf{I}_k$ of the neutron is:

$$\int_{A_d}\int_{S_d} E(\mathbf{x}_0)W(\mathbf{x}_k)\underbrace{\int_{\Xi^\omega}\int_{\Xi^t} f_t(\bar{t}_k, \theta)f_\omega(\bar{\omega}_k, \theta)\mathrm{d}\bar{t}_k\mathrm{d}\bar{\omega}_k}_{\mathbf{I}_k^m(\theta)}\mathrm{d}\mathbf{x}_0\mathrm{d}\mathbf{x}_k. \tag{8}$$

In this space, we will show that the discontinuities can be stabilized with a simple reparameterization of only the t variables. Furthermore, we will show in Sec. 5 that we can integrate out the variable t_{k-1} when the score function W has a convenient form, leading to a track-length estimator.

4 Reparameterization of the Distance Subspace

In this section, we reparameterize the distance subspace, consisting of the t variables, to handle discontinuities.

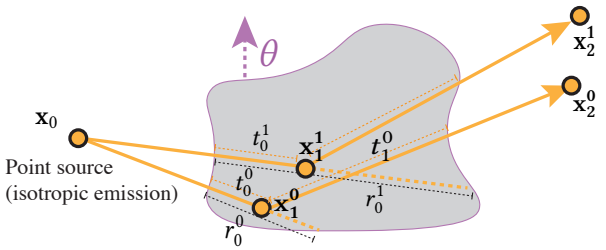


Fig. 4. Two path samples with path length of $k = 2$ in an arbitrary shape. Sample 1, with $\bar{t}_0 = t_0^0 t_1^0$, is the path that connects $\mathbf{x}_0, \mathbf{x}_1^0, \mathbf{x}_2^0$. Sample 2, with $\bar{t}_1 = t_0^1 t_1^1$, is the path that connects $\mathbf{x}_0, \mathbf{x}_1^1, \mathbf{x}_2^1$.

4.1 Reparameterization for Single Scattering

We first consider reparameterizing the distance subspace with a single collision event ($k = 2$). Given a surface enclosing a material with total cross-section σ_t and single scattering albedo α, we compute the flux at the detector surface due to neutrons that scatter once in the material. This quantity is equal to the path space integral Eq. (8) for a path length of $k = 2$, whose gradient with respect to θ can be written as

$$\frac{\partial}{\partial\theta}\mathbf{I}_2 = \int_{A_d}\int_{S_d} E(\mathbf{x}_0)W(\mathbf{x}_k)\frac{\partial}{\partial\theta}\mathbf{I}_2^m(\theta)\mathrm{d}\mathbf{x}_2\mathrm{d}\mathbf{x}_0 \tag{9}$$

Our task now is to estimate $\frac{\partial}{\partial\theta}\mathbf{I}_2^m$. Writing $\mathbf{I}_2^m(\theta)$ in Fig. 4 as:

$$\int_\Omega\int_0^{r_0(\omega_0,\theta)} f_t(t_0, t_1, \theta)f_\omega(\omega_0, \omega_1, \theta)\mathrm{d}t_0\mathrm{d}\omega_0, \tag{10}$$

the key idea is to apply the change of variable $t_0 = t_0' r_0(\omega_0, \theta)$ to remove the parameter dependency from the integral bounds. Then the gradient is:

$$\frac{\partial}{\partial\theta}\mathbf{I}_2^m(\theta) = \int_\Omega\int_0^1 \frac{\partial}{\partial\theta}\left[f_t(t_0' r_0, t_1, \theta)f_\omega(\omega_0, \omega_1, \theta)r_0(\theta)\right]\mathrm{d}t_0'\mathrm{d}\omega_0, \tag{11}$$

which is straightforward to estimate by Monte Carlo integration and automatic differentiation because its integrand is continuous and its bounds are constant.

This reparameterization is powerful because it allows us to compute an unbiased gradient estimate by simply applying automatic differentiation to the forward Monte Carlo simulation.

4.2 Generalizing the reparameterization to multiple scattering

Now we are ready to consider the case where there is more than one scattering event and generalize the expression to $\mathbf{I}_k$.

Consider how the reparameterization works for $\mathbf{I}_3$, when there are two scattering events in the material $\frac{\partial}{\partial\theta}\mathbf{I}_3^m(\theta)$:

$$\frac{\partial}{\partial\theta}\int_\Omega\int_0^{r_0(\theta)}\int_\Omega\int_0^{r_1(\theta,t_0)} f_t(t_0, t_1, t_2, \theta)f_\omega(\omega_0, \omega_1, \theta)\mathrm{d}t_1\mathrm{d}t_0\mathrm{d}\bar{\omega}_3. \tag{12}$$

Using the reparameterization introduced in the last section, where

$$t_0 = t_0' r_0(\omega_0, \theta), \qquad t_1 = t_1' r_1(\bar{\omega}_2, t_0' r_0, \theta), \tag{13}$$

we can move the gradient operator of $\frac{\partial}{\partial\theta}\mathbf{I}_3^m(\theta)$ into the integrand and get,

$$\int_{\Xi_2^\omega}\int_0^1\int_0^1 \frac{\partial}{\partial\theta}\Big[f_t(t_0'r_0,t_1'r_1,t_2,\theta)f_\omega(\omega_0,\omega_1,\theta)|\mathcal{J}_3(\theta)|\Big]\mathrm{d}t_1'\mathrm{d}t_0'\mathrm{d}\overline{\omega}_3, \tag{14}$$

where the Jacobian matrix $\mathcal{J}_3(\theta)$ between t_0, t_1 and t_0', t_1' is:

$$\begin{bmatrix}\mathrm{d}t_0\\ \mathrm{d}t_1\end{bmatrix} = \begin{bmatrix} r_0(\theta,\omega_0) & 0\\ \frac{\partial}{\partial t_0'}r_1(\omega_1,\theta,t_0'r_0) & r_1(\omega_1,\theta,t_0'r_0)\end{bmatrix}\begin{bmatrix}\mathrm{d}t_0'\\ \mathrm{d}t_1'.\end{bmatrix} \tag{15}$$

Notice that $\mathcal{J}_3(\theta)$ is a triangular matrix and thus its determinant equals the product of the diagonal elements:

$$|\mathcal{J}_3(\theta)| = r_0(\theta,\omega_0)r_1(\omega_1,\theta,t_0'r_0). \tag{16}$$

We know from the sampling process that the location of a point i on the path is independent of the later points j, $(j > i)$ on the path. Therefore, the Jacobian matrix is always a lower triangular matrix. In other words, for a path of length k ($k > 1$), we can reparameterize $t_0 \dots t_{k-2}$ as $t_0'r_0 \dots t_{k-2}'r_{k-2}$, where $t_0' \dots t_k'$ is a variable in the unit hypercube, and the determinant of the Jacobian between the two parameterizations is then:

$$|\mathcal{J}_k(\theta)| = \prod_{i=0}^{k-2} r_i(\theta,\omega_i), \tag{17}$$

thus to compute the gradient of $\mathbf{I}_k^m(\theta)$ for a geometry parameter θ, we have

$$\int_{\Xi_k^\omega}\int_{\Xi_k'^t}\frac{\partial}{\partial\theta}\Big[|\mathcal{J}_k(\theta)|f_t(t_0'r_0,\dots,t_{k-1},\theta)f_\omega(\overline{\omega}_k,\theta)\Big]\mathrm{d}\overline{t}_{k-1}'\mathrm{d}\overline{\omega}_k. \tag{18}$$

After the reparameterization, the t_i' are defined on the normalized space $[0,1]$. We use $\Xi_k'^t$ to denote the normalized distance space of path length k.

4.3 Reparameterization for a Piecewise Continuous Integrand

As shown in Fig. 3, a ray may travel through multiple materials, each with different cross-sections. This leads to a piecewise integral, separated by the position of the material interface, which depends on the shape parameter θ.

For single scattering, the piecewise integral is

$$\mathbf{I}_2^m = \int_\Omega\sum_{j=1}^M\int_0^{r_j(\omega_0,\theta)} f_t(t_0,t_1,\sigma_j,\theta)f_\omega(\omega_0,\omega_1,\sigma_j,\theta)\mathrm{d}t_0\mathrm{d}\omega_0, \tag{19}$$

where M is the number of materials a neutron passes through when traveling from $\mathbf{x}_0$ along direction ω_0, and $r_j(\omega_0,\theta)$ is the distance, for which the neutron is in material j with material properties σ_j. For each material, we do the piecewise reparameterization such that $t_0 = r_j(\omega_0,\theta)t_0^j$, and putting this back into the integral gives the following expression:

$$\mathbf{I}_2^m = \int_\Omega\sum_{j=1}^M\int_0^1 f_t(t_0^jr_j,t_1,\sigma_j,\theta)f_\omega(\omega_0,\omega_1,\sigma_j,\theta)r_j(\theta)\mathrm{d}t_0^j\mathrm{d}\omega_0. \tag{20}$$

Again, because the bounds are fixed and the integrand is continuous, the gradient can be computed by integrating the gradient of the integrand:

$$\frac{\partial}{\partial\theta}\mathbf{I}_2^m = \int_\Omega\sum_{j=1}^M\int_0^1\frac{\partial}{\partial\theta}f_t(t_0^jr_j(\theta),t_1,\sigma_j)f_\omega(\omega_0,\omega_1,\sigma_j,\theta)r_j(\theta)\mathrm{d}t_0^j\mathrm{d}\omega_0. \tag{21}$$

A proof of the proposed piecewise reparameterization in 1D is provided in the supplementary material.

5 Gradient for the Volume Flux

In addition to estimating surface current, we also aim to estimate volume flux, which is the integral of the neutron radiance over a volume. Starting from the general version of the measurement in Eq. (8), for a flux sensor in volume $\mathcal{V}_d$, the sensor response is $W(\mathbf{x}_k) = \frac{1}{\|\mathcal{V}_d\|}$, and thus

$$\Phi = \frac{1}{\|\mathcal{V}_d\|}\int_{A_d}\int_{\mathcal{V}_d}\int_{\Xi^\omega}\int_{\Xi^t} E(\mathbf{x}_0)f_t(\overline{t})f_\omega(\overline{\omega})\mathrm{d}\overline{t}_k\mathrm{d}\overline{\omega}_k\mathrm{d}\mathbf{x}_k\mathrm{d}\mathbf{x}_0. \tag{22}$$

If we replace the $\mathbf{x}_k$ with $\mathbf{x}_{k-1} + \omega_{k-1}t_{k-1}$, the flux can be written as:

$$\Phi = \frac{1}{\|\mathcal{V}_d\|}\int_{A_d}E(\mathbf{x}_0)\int_{\Xi^\omega}\int_{\Xi^t}f_t(\overline{t}) \tag{23}$$

$$\int_{\Omega(\mathbf{x}_{k-1},\mathcal{V}_d)}f_\omega(\overline{\omega})\underbrace{\int_{r(\mathbf{x}_{k-1},\mathcal{V}_d)}f_t(t_{k-1})\mathrm{d}t_{k-1}}_{\mathbf{I}_a}\mathrm{d}\omega_{k-1}\mathrm{d}\overline{t}_k\mathrm{d}\overline{\omega}_k\mathrm{d}\mathbf{x}_0, \tag{24}$$

where $r(\mathbf{x}_{k-1},\mathcal{V}_d)$ is the interval that the neutron could travel inside the volume $\mathcal{V}_d$, and $\Omega(\mathbf{x}_{k-1},\mathcal{V}_d)$ is the solid angle subtended by the volume from the point $\mathbf{x}_{k-1}$. The main idea of pre-integration, underlined in Eq. (24), is to compute a closed-form expression to replace the inner-most integral, which integrates over the last distance with the precomputed solution $\mathbf{I}_a$.

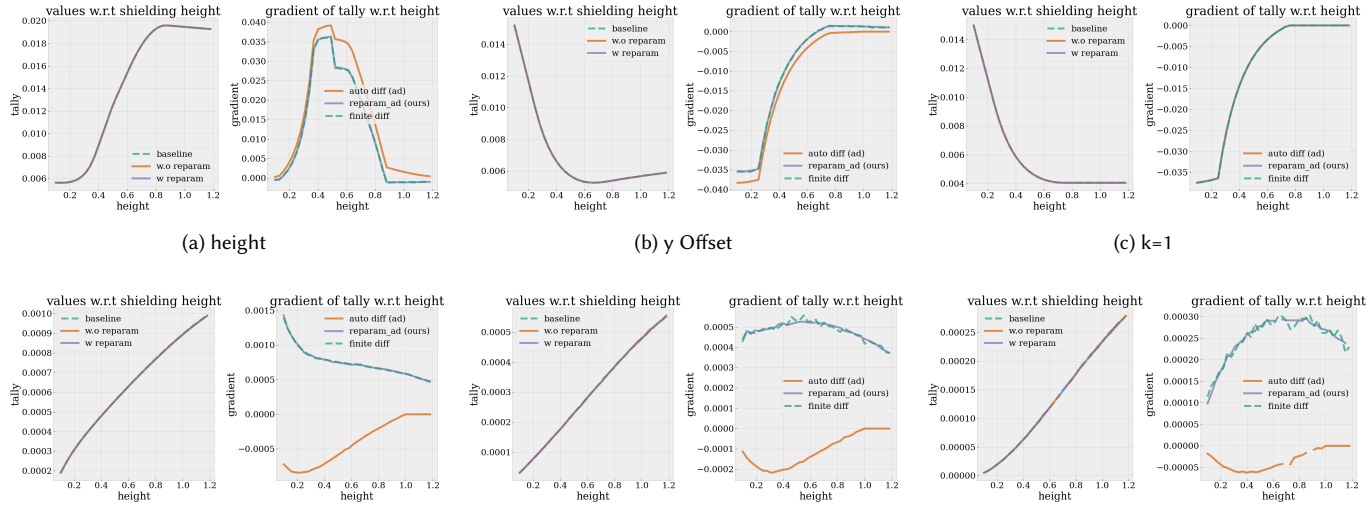
5.1 Track-length Estimator for Flux

For one volume sensor, this computation seems affordable. But what is really needed is a tensor grid of volume flux sensors, like the grids in Fig. 6. Computing the analytic solution of $\mathbf{I}_a$ for a large number of voxel sensors requires evaluating every voxel within an infinite distance along the travel direction of the neutron sample, and for each voxel, an attenuation at the entry point is needed. This, in combination, makes the estimates of volume flux expensive.

Instead, we want an estimator that allows one path sample to contribute to many sensors at once. This is related to the track-length estimator, which contributes the length that each path travels in a voxel to that voxel's integral. We briefly review the idea of the track-length estimator here, starting from a one-dimensional example.

Imagine there is a one-dimensional space, where the neutrons are propagating along the $+x$ direction. The space is discretized into bins of width w; and the flux of each bin is

$$\mathbf{I}_a = \int_r^{r+w} f_t(d)\mathrm{d}d \tag{25}$$



(a) height (b) y Offset (c) k=1

(d) k=2 (e) k=3 (f) k=4

Fig. 5. Gradient validation of tally (surface current on the sensor) value with respect to the upward y offset (b) and the height (a) of the shield. We visualize for the "y-offset" case, the value and the gradient of path space of length k=1, k=2, k=3, k=4 separately, showing that our method is capable of computing the gradient of multiple scattering events. In contrast, the naive autodiff could only compute the gradient with uncollided quantities.

where d is the distance between a point in the bin and the origin. Inside the integral in Eq. (25), the closed form expression for the attenuation at a distance d from the source is

$$f_t(d) = e^{-d\sigma_t} = \int_d^\infty \sigma_t e^{-s\sigma_t}\,\mathrm{d}s. \tag{26}$$

The interpretation of Eq. (26) is that the flux at a distance $t = d$ from the source equals the total number of neutrons that collide at a distance beyond $t = d$.

By definition, the expected value of a function of a random variable $\mathbf{X}$ is

$$[g(\mathbf{X})] = \int_R g(x)p(x)dx \tag{27}$$

where $p(x)$ is the probability density of x. Therefore, $f_t(d)$ can be formulated as the expected value of an indicator function over a random variable defined on $t \in [0, +\infty]$ where

$$f_t(d) = \int_0^\infty \mathbf{1}(t > d)p(t)\mathrm{d}t; \qquad \text{with} \quad p(t) = \sigma_t e^{-t\sigma_t}. \tag{28}$$

The Monte Carlo estimator for $f_t(d)$ is

$$[f_t(d)] = \frac{1}{N}\sum_{j=1}^{N} \frac{\mathbf{1}(t^j > d)p(t^j)}{p(t^j)}, \tag{29}$$

where the probability cancels out and what remains is a constant contribution for each sample j. Putting this back into Eq. (25), reordering the integration, and analytically integrating over d gives:

$$[\mathbf{I}_a] = \frac{1}{N}\sum_{j}^{N}\int_r^{r+w} \frac{\mathbf{1}(t^j > d)p(t^j)}{p(t^j)}\mathrm{d}d = \frac{1}{N}\sum_{j=1}^{N} L_w(\mathbf{x}, t^j), \tag{30}$$

where $L_w(\mathbf{x}, t_j)$ is the distance that the jth neutron sample traveled in the interval $[r, r + w]$. This is called the track-length estimator for the volume flux. This approach makes the estimate of the flux smoother by pre-integrating one distance dimension. As shown in Fig. 6, given the same number of neutrons, the track length estimator can achieve significantly smoother volume flux estimates.

5.2 Track-length Estimator for Flux Gradients

The key question here is whether we can estimate the gradient of the volume flux using the same set of neutron samples as the track-length estimator of the flux and get similar, smoother estimates of gradients. However, we observe that the main difficulty arises from the piecewise nature of the function $g(t, d)$, where for each neutron sample, the length $t_j(\theta)$ depends on the parameters. Direct automatic differentiation of the standard track-length estimator does not yield a valid track-length estimator for the gradient. This is expected: both geometry and material parameters influence the estimator through the lengths of neutron trajectories, and differentiating these lengths produces contributions only from volumes that contain the beam endpoints (where the collision happens), effectively resulting in a collision-type gradient estimator rather than a track-length one.

In this section, we derive an estimator for flux that is equal to the track-length estimator but when differentiated, yields a lower-variance estimator for the gradient of the flux. Suppose there is a material interface located $r(\theta)$ from emission, where $r(\theta) < r$, and the total cross section parameters of the two materials are σ_t^a when $0 < t < r(\theta)$ and σ_t^b where $t > r(\theta)$. Then the attenuation up to a distance $d > r(\theta)$ from the emitter is therefore $f_t(d, r(\theta)) = e^{-\sigma_t^a r(\theta) - \sigma_t^b (d - r(\theta))}$, and its gradient with respect to θ is:

$$\frac{\partial}{\partial\theta} f_t(d, r(\theta)) = -(\sigma_t^a - \sigma_t^b) f_t(d, r(\theta)) \frac{\partial}{\partial\theta} r(\theta), \tag{31}$$

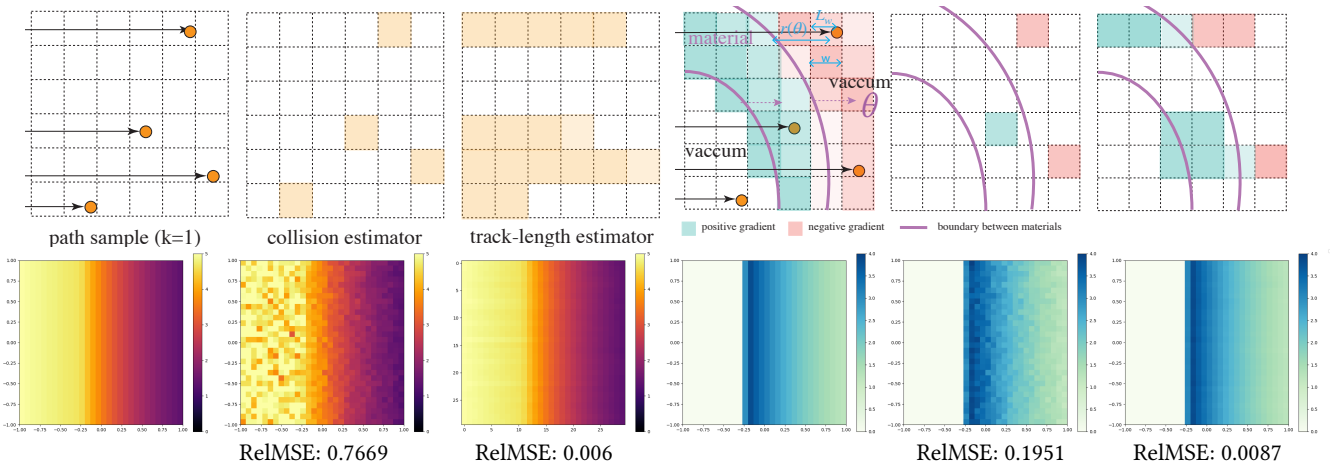


Fig. 6. Collision Estimator and Track-length estimator: The first row illustrates several physical quantities – the collision estimator, track-length estimator, and their gradients in a 2D scene. In the second row, we demonstrate the accuracy of our track-length estimator by comparing the visualization of the 2D neutron fluxes computed with different estimation methods. The fluxes are visualized in the left three plots, while their respective flux gradients are visualized in the right three plots. Out of the 6 plots, the 1st and the 4th plots visualize the analytical solutions of the flux and its gradient in this setup. The same two quantities, derived with our track-length estimator in the 3rd and 6th plots, demonstrate significantly lower variance compared to those derived from the baseline collision estimator in the 2nd and 5th plots. Both methods estimate the flux using the same number of samples. We compute the relative mean square error of the collision and tracklength estimators geometry gradients against the analytic results.

where the attenuation itself can be replaced by the track-length estimator, yielding

$$\frac{\partial}{\partial\theta} f_t(d, r(\theta)) \approx -(\sigma_t^a - \sigma_t^b)\frac{\partial}{\partial\theta} r(\theta)\frac{1}{N}\sum_{j=1}^{N} \mathbf{1}(t^j > d). \tag{32}$$

Integrating the flux gradient in Eq. (32) over the interval $[r, r+w]$ gives us:

$$[\frac{\partial}{\partial\theta}\mathbf{I}_a] = \frac{1}{N}\frac{\partial}{\partial\theta} r(\theta)\int_r^{r+w}(\sigma_t^b - \sigma_t^a)\sum_{j=1}^{N}\mathbf{1}(t^j > d)\mathrm{d}d \tag{33}$$

$$= \frac{1}{N}\frac{\partial}{\partial\theta} r(\theta)\sum_{j=1}^{N} L_w(\mathbf{x}(\theta), t^j)(\sigma_t^b - \sigma_t^a). \tag{34}$$

Now, we want to enforce $\frac{\partial}{\partial\theta}[\mathbf{I}_a] = [\frac{\partial}{\partial\theta}\mathbf{I}_a]$ by multiplying by a parameterized constant $c(r(\theta), \sigma_t^D)$ such that

$$\frac{\partial}{\partial\theta} c(r(\theta), \sigma_t^D) = \sigma_t^D \frac{\partial}{\partial\theta} r(\theta) \tag{35}$$

$$c(r(\theta), \sigma_t^D) = \frac{\mathbf{C}(r(\theta), \sigma_t^D)}{\widehat{\mathbf{C}}(r(\theta), \sigma_t^D)} = 1, \tag{36}$$

where $\sigma_t^D = \sigma_t^b - \sigma_t^a$ represents the difference of scattering properties between the materials on two sides of the interface, and the symbol $\widehat{\ }$ stands for the part of the computation whose computational graph is detached from the propagation of gradients, thus considered as a scalar, that is to say $\widehat{\mathbf{C}(d)} = \mathbf{C}(d)$ and $\frac{\partial}{\partial d}\widehat{\mathbf{C}(d)} = 0$. These two conditions give the following equation:

$$\frac{\partial}{\partial r(\theta)}\mathbf{C}(r(\theta), \sigma_t^D) = \sigma_t^D \mathbf{C}(r(\theta), \sigma_t^D) \tag{37}$$

Solving the ODE in Eq. (37) gives us

$$\mathbf{C}(r(\theta), \sigma_t) = e^{-\sigma_t^D r(\theta)}. \tag{38}$$

If we generalize this to M material interfaces along the path segment, the constant modifying the track-length estimator is

$$\mathbf{C}_M = e^{-\sum_{i=1}^{M}\sigma_t^{D_i} r_i(\theta)}, \tag{39}$$

where $\sigma_t^{D_i}$ is the difference between the two materials on the ith boundary along the direction of the path segment and $r_i(\theta)$ is the distance from the interface to the starting point of the current path segment. Summarizing the aforementioned modification to the track-length estimator, we have

$$[\mathbf{I}_a] = \frac{1}{N}\sum_{j=1}^{N} L_w(t^j)\frac{\mathbf{C}_M(t^j)}{(\widehat{\mathbf{C}_M(t^j)})}. \tag{40}$$

When the value of this estimator is computed, it is equivalent to (30), and when its derivative is computed through automatic differentiation, the result is a valid estimator for $\frac{\partial}{\partial\theta}\mathbf{I}_a$.

5.3 Track-length Estimator in Multiple Scattering

When extending the track-length estimator to multi-bounce paths, the most direct approach is to apply it to the final segment of the neutron trajectory. Specifically, we replace $\mathbf{I}_a$ in Eq. (24) with the modified track-length estimator from Eq. (40), and then reparameterize the distance subspace. This yields

$$\Phi = \frac{1}{\|\mathcal{V}_d\|}\int_{A_d} E(\mathbf{x}_0)\int_{\Xi^\omega}\int_{\Xi'_t} f_t(t'_0 r_0, \ldots, t'_{k-1} r_{k-1}, \theta) f_\omega(\overline{\omega}, \theta)$$

$$\mathcal{J}(\theta)\int_{\Omega(\mathbf{x}_{k-1}, \mathcal{V}_d)} [\mathbf{I}_a(\mathbf{x}_{k-1}, \omega_{k-1}, w, \theta)]\mathrm{d}\omega_{k-1}\mathrm{d}\overline{t}'_k\mathrm{d}\overline{\omega}'_k\mathrm{d}\mathbf{x}_0,$$

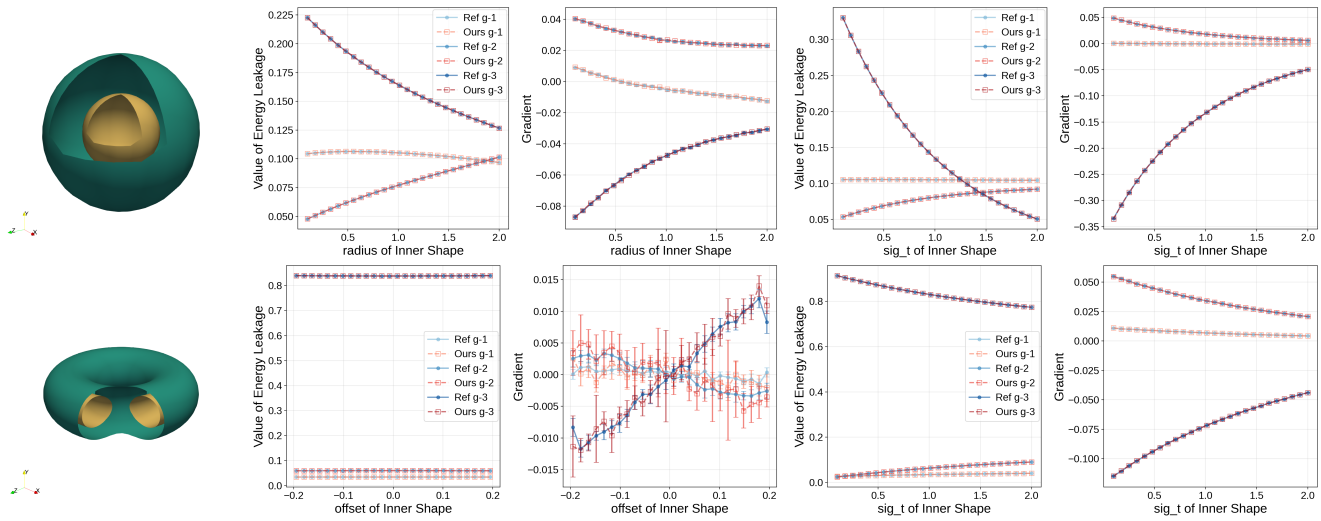


Fig. 7. We validate our neutron transport implementation against a conventional neutron transport simulator (OpenMC) by comparing both surface flux estimates and their gradients with respect to geometric and material parameters (specifically, the total cross section) at multiple energy groups. Gradients from OpenMC are approximated using finite differences and are shown with error bars, while our gradients are obtained via automatic differentiation of the Monte Carlo simulation. The leftmost column illustrates the material composition and scene geometry. The sphere scene has an isotropic point emission source in the center of the sphere, and the torus scene has an isotropic ring emission source in the center of two tori.

Here, the starting point of the final segment, $\mathbf{x}_{k-1}$, depends on all previous sampled distances and directions. Consequently, after reparameterization, the beam length term $L_w(\mathbf{x}_{k-1}(\theta), t_j)$ also depends on the geometry parameter θ. The angular domain $\Omega(\mathbf{x}_{k-1}, \mathcal{V}_d)$ likewise depends on θ through the location of $\mathbf{x}_{k-1}$. However, the overlap length between a beam and a voxel is continuous with respect to the beam orientation, except at degenerate configurations where the beam becomes parallel to voxel edges. Under the assumed scattering distribution, such parallel configurations form a measure-zero set. Therefore, the differentiation operator can be safely moved inside the integral.

6 Implementation

We implemented our differentiable neutron transport simulator in Python using the differentiable numerical backend Dr. Jit [Jakob et al. 2022a], alongside ray–geometry intersection routines adapted from the open-source renderer Mitsuba 3 [Jakob et al. 2022b]. Although Mitsuba 3 is originally designed for light transport and provides automatic differentiation support, it lacks several components essential for neutron transport. These components are energy-dependent scattering functions, energy-dependent material properties, efficient support for a large number of volumetric flux measurements, and a geometry representation capable of handling non-manifold structures in which material interfaces may exist within the interior of the geometry.

Geometry Representation. We represent the scene geometry using constructive solid geometry (CSG), in which regions of different materials are defined through Boolean combinations of primitive shapes. The leaves of the CSG tree may consist of either analytic primitives or triangle meshes. Rather than explicitly constructing the geometry from Boolean operations—which would require recomputation or remeshing and introduce non-differentiable topology changes—we evaluate the Boolean relationships implicitly during ray tracing by operating directly on the leaf geometries. This implicit evaluation is particularly advantageous for gradient-based optimization, as it allows us to optimize only a subset of the leaf geometries while supporting multiple parameterizations of a shape without altering the underlying scene topology.

Energy Dependence of Material Properties. Neutron transport depends on energy in two fundamental ways. First, material properties—such as the total cross section and albedo—depend on the neutron's speed, or equivalently, its energy. Second, the scattering process may transfer a neutron from one energy level to another according to the energy-dependent scattering kernel. We model this behavior using a discrete set of energy groups. Each neutron sample belongs to a single energy group at any given time, and material properties are queried conditionally based on that group. At a scattering event, the neutron may either remain in its current energy group or transition to a different group, per sampling of the scattering function.

Simulator. Our simulator works by tracing neutrons from the source and making contributions to sensors, computing the path weights using the differentiable approach from Section 4. We support two types of sensors: surface current sensors, which accumulate the total energy per energy group incident on a surface, and volume flux sensors, which estimate the neutron flux on a user-specified volumetric grid of resolution $N \times N \times N$. For surface current sensors, we use next-event estimation, connecting each sensor to every vertex along each neutron path. For volume flux sensors, neutron

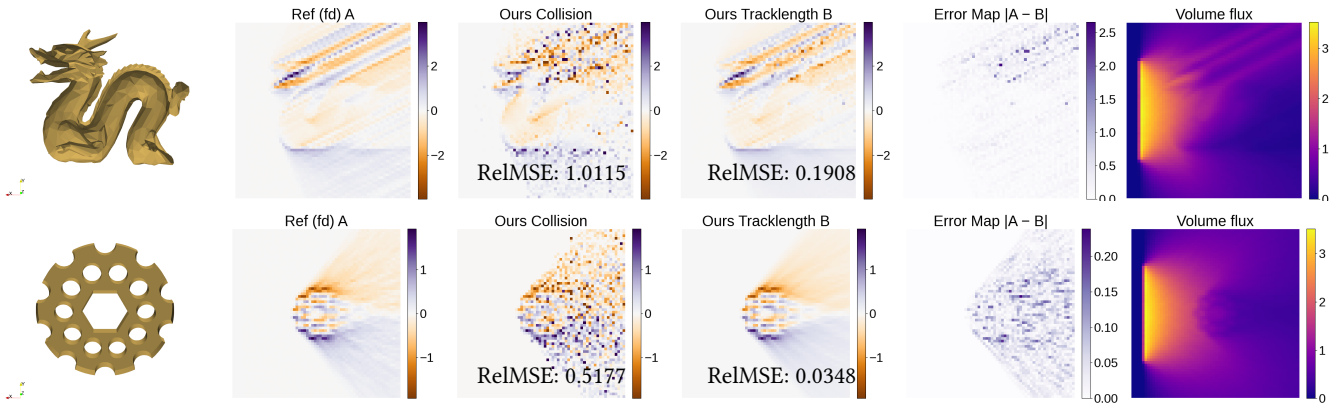


Fig. 8. Equal sample comparison on gradient estimation between the track-length estimator and the collision estimator on 3D scenes with two materials separated by the interface defined by the dragon shape (row 1), and the gear shape (row 2). The gradient is computed with respect to a geometry vertical offset to the geometry. We compute the relative mean square error of collision and tracklength estimator against the finite differences.

paths are stored as beam segments and then contributed to the corresponding grid cells using the differentiable track-length estimator from Section 5. With differentiable estimators in use throughout, the derivatives are propagated in reverse mode, so that they propagate from measurements to any desired parameters.

Optimization. We employ stochastic gradient descent with the Adam optimizer for all optimization tasks. For optimization over triangle meshes, we also use the large-step preconditioner proposed by Nicolet et al. [2021] to improve smoothness and robustness.

7 Experiments and Results

In this section, we first present experiments validating our gradient computations for both surface current measurements and volumetric flux measurements. We then demonstrate the effectiveness of these gradients in solving three representative problems of neutronics: (1) shielding optimization, (2) sensor design, and (3) sensitivity analysis.

7.1 Gradient Validation

We validate our simulation framework against a conventional Monte Carlo neutron transport simulator, OpenMC [Romano et al. 2015]. Specifically, the gradient estimates from our method are compared with finite differences of measurements computed by OpenMC. In Fig. 7, we consider two benchmark scenes composed of two randomly generated materials and three energy groups. We evaluate the gradients of the surface-escaping neutron energy with respect to both geometric parameters (the radius of the inner sphere and the offset of the inner torus) and material parameters (the total cross section in the highest energy group).

In Fig. 6, we validate the gradient of the volume neutron flux with respect to the position of a material interface using a two-dimensional scene for which an analytical solution is available. This comparison demonstrates the effectiveness of our gradient estimator relative to the collision estimator when using the same number of samples.

Finally, in Fig. 8, we extend this validation to three-dimensional scenes under different source configurations, evaluating the gradients of the volume neutron flux with respect to geometric parameters such as the offset of the geometry.

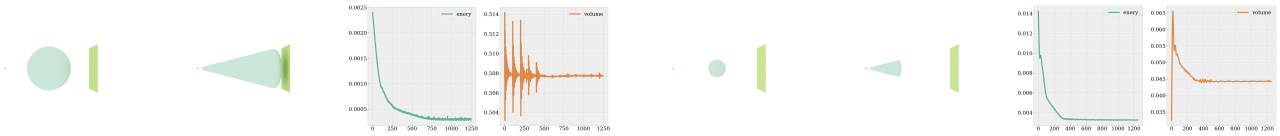
7.2 Shielding Optimization

In Fig. 9, we demonstrate an example of shielding shape optimization in which an isotropic neutron source emits fast neutrons, and the optimization objective is to minimize the neutron current collected at the surface of a rectangular sensor, subject to a volume constraint. In this setting, the optimal solution is expected to concentrate material within the pyramid-shaped region defined by the source and the rectangular sensor, while removing material outside this region. This is because the material placed elsewhere would primarily increase the number of neutrons scattered toward the sensor.

Consistent with this physical intuition is the optimal shield shape solved using our method. Under a strict material budget, the optimized shape progressively pushes material closer to the source, thereby maximizing the distance neutrons travel within the shielding material. This increased path length enhances attenuation and ultimately minimizes the radiation reaching the sensor.

7.3 Sensor Design

For diagnostic purposes, the neutron flux in a tokamak hall is measured using neutron detectors. One of the most prevalently used detectors, built with Boron-10, is primarily sensitive to neutrons at relatively low energies, whereas the neutron field in the hall is dominated by fast neutrons. Consequently, fast neutrons must be moderated before reaching the detector in order to produce a measurable signal. To this end, additional energy-shaping components are typically introduced to slow down neutrons and enhance detector response.



(a) iter=0 (b) iter=1200 (c) Energy and Volume (d) iter=0 (e) iter=1200 (f) Energy and Volume

Fig. 9. The scene consists of a point source on the left and an area sensor (green plane) on the right, with a shielding material initially placed between them (blue geometry). The optimization seeks a shield shape that minimizes the neutron surface current incident on the sensor under a volume constraint. In (a)–(c), the material is expected to be redistributed from less effective regions into the region that directly blocks the sensor from the source, and our optimization matches this expectation. In (d)–(f), the optimization not only predicts a similar blocking geometry but also shifts material toward the source, which increases neutron path lengths within the shield and thereby enhances attenuation of neutrons reaching the sensor. The geometry is represented as a triangle mesh, with the vertex positions treated as optimization variables.

Wang et al. [2024b] investigated the effect of the thickness of a cylindrical high-density polyethylene sleeve on the Boron-10 detector response. Specifically, they examined how well the sleeve moderates fast neutrons. In Fig. 11, we reproduce their experimental setup within a synthetic scene and extend it by exploring more general design spaces, enabling optimization over additional geometric degrees of freedom beyond sleeve thickness alone.

Following the basic scene setup of Wang et al. [2024b], we initialize the optimization with a cylindrical sensor and a cylindrical neutron moderator enveloping it. Then, we optimize the moderator thickness to maximize the sensor response. For validation, we perform a parameter sweep over the moderator thickness and simulate neutron transport to evaluate the energy collected by the sensor in the target energy group. The optimal thickness and corresponding response obtained from this sweep agree with the results produced by gradient-based optimization.

We then relax the optimization parameterization to model the moderator's surface as a mesh, rather than restricting it to a cylindrical shape. During this process, we constrain the moderator volume to remain equal to that of the optimal cylindrical moderator. Under this more flexible parameterization, the optimization discovers a new moderator shape and its placement relative to the cylindrical sensor. This new design found by our method achieves a higher response than the optimal cylindrical configuration.

7.4 Sensitivity Analysis

Real-world material properties are typically obtained from experimental measurements and are therefore subject to uncertainty. In neutronics, it is often the case that the full probability distributions of these uncertainties are unavailable; instead, only bounded intervals for quantities such as the mean and variance are known. Under such epistemic uncertainty, practitioners are interested in estimating upper and lower bounds on physical quantities of interest, as well as identifying which parameters the results are most sensitive to, in order to prioritize design optimization efforts.

Traditionally, this type of sensitivity and uncertainty analysis in neutronics is performed using brute-force approaches that rely on a large number of Monte Carlo simulations. Each simulation samples material properties within the prescribed uncertainty intervals and computes the corresponding response. Aggregating the results yields empirical bounds or cumulative distribution functions (CDFs), from which uncertainty ranges—such as the gap between upper and lower envelopes—can be estimated.

In Fig. 10, we demonstrate an example of propagating uncertainty using local gradient information. Specifically, we approximate the response uncertainty by treating the perturbed quantities as normally distributed and use this approximation to construct an estimated probability box (p-box). We analyze uncertainty arising from the total cross section (a) and from geometric parameters (d) for a single energy group. In addition, we study uncertainty in the total cross section across multiple (three) energy groups, both without covariance (b) and with covariance (c). In all cases, we compare our results against a Monte Carlo baseline, a standard approach widely used in contemporary neutronics sensitivity analysis [de la Lluvia et al. 2026].

8 Conclusion and Discussion

We have presented a theoretical framework for solving inverse problems in neutron transport using techniques from differentiable rendering. Our approach first applies a reparameterization technique for the Monte-Carlo algorithm to estimate surface current gradients with respect to shield geometry. We then introduce a modification to the track-length estimator that enables direct estimation of the geometry gradients of volume flux. This method is generalized to handle arbitrary shapes and multiple material layers, providing designers with greater flexibility. Such flexibility enables optimization strategies that go beyond uniform shield thicknesses, allowing the exploration of novel and previously unseen design solutions.

This work also raises several directions for further research:

Geometry Representation. In the current formulation, for shape optimization of in non-symmetrical devices like stellarators, we prevent self-intersection by enforcing positive displacements for inner surfaces. However, in complex, non-symmetric geometries, singularities may arise where displacements from the innermost surface converge to the same point. Addressing these cases will require more robust geometry representations to ensure stable and accurate optimization in the presence of such singularities.

Sensitivity Analysis. We demonstrate an example of probability box (p-box) estimation for uncertainty quantification using gradient-based methods. While this approach is computationally efficient, it may produce overly conservative bounds with long tails when the

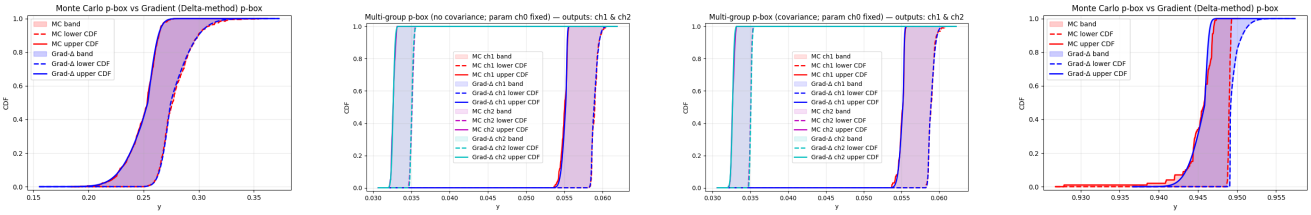


Fig. 10. We show that differentiable neutron transport enables efficient, gradient-based p-box construction for uncertainty estimation under epistemic parameter uncertainty. From left to right, the examples increase in simulation complexity, progressing from a single cross-section parameter, to a three-parameter multi-group transport setting without covariance, to the same multi-group setting with parameter covariance. In the multi-group examples, we fix parameter channel 0 and introduce uncertainty in channels 1 and 2, and compare the Monte Carlo p-box against the first-order gradient delta-method approximation. In the rightmost column, we further illustrate the approximation errors that arise when the gradient-based approach is applied to nonlinear and non-monotone parameters, such as geometric variables.

parameter-to-response mapping is strongly nonlinear. This behavior is evident in Fig. 10(d), where the geometric uncertainty shown in the second row of Fig. 7 overlaps regions in which the first-order gradient approaches zero. In such cases, linear gradient-based approximations fail to capture local curvature in the response surface. Addressing these limitations—by incorporating higher-order derivatives via local Taylor expansions or by adopting hybrid gradient sampling strategies—is not the primary focus of this work. Nevertheless, the availability of accurate local gradients enables a range of sensitivity analysis tasks, including identifying influential parameters, characterizing locally linear response regimes, and designing measurement configurations that are maximally informative under uncertainty. Exploring these directions constitutes an interesting avenue for future work.

Track-length Estimates with Multiple Bounces. Our current implementation applies the track-length estimator only to the final segment of each neutron path. When combined with the reparameterization in Sec. 4, this can lead to collision-like gradients for paths with $k > 1$ scattering events, see Fig. 12 for reference, and may introduce singularities when beam directions become nearly parallel to voxel boundaries. Replacing the box filter with continuous basis functions could smooth the voxel response and mitigate these degeneracies. More broadly, it would be interesting to investigate track-length estimators applied to transport segments other than the final one, which may distribute gradient information more effectively along multi-collision paths.

Beyond Neutrons. As noted in the introduction, fusion reactions produce multiple particle types, including neutrons, gamma rays, alpha particles, and beta particles. Gamma rays exhibit different interaction behavior at material interfaces, potentially posing new challenges for gradient estimation. Likewise, the leakage patterns of alpha and beta particles from the plasma core are closely tied to the plasma's state. Designing shielding to mitigate damage from these charged particles will require incorporating plasma simulations into the optimization process—a non-trivial but promising extension of this work. In addition, neutral beam injection systems—commonly used to heat and trigger fusion plasmas—are themselves modeled using Monte Carlo particle transport. Extending differentiable transport techniques to neutral beam simulations could enable gradient-based optimization of source design and beam shaping, providing further opportunities for integrated design optimization across fusion systems.

Acknowledgments

We thank Bruce Walter for valuable discussions and insights on reparameterization. We are grateful to Anima Anandkumar for her support in connecting us with the neutronics community. We thank Wenzel Jakob for guidance on using Mitsuba functions and tensors. We appreciate Michael Churchill and Enrique Miralles-Dolz for providing application scenarios related to sensitivity analysis. We thank Xinyan Wang and Qiyun Cheng for suggesting the sensor optimization example. We also thank Jiatian Sun for reviewing the paper during the writing pass. This work was supported in part by NSF Award IIS-2212084, the U.S. Department of Energy under Contract DE-SC0024548, and the Simons Foundation Collaboration on Hidden Symmetries and Fusion Energy, Grant No. 601958.

References

Sai Praveen Bangaru, Tzu-Mao Li, and Frédo Durand. 2020. Unbiased warped-area sampling for differentiable rendering. *ACM Trans. Graph.* 39, 6 (2020).

Benedikt Bitterli and Wojciech Jarosz. 2017. Beyond Points and Beams: Higher-Dimensional Photon Samples for Volumetric Light Transport. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 36, 4 (July 2017). doi:10/gfznbr

Y.Q. Chen and B.H. Yan. 2023. The technology of shielding design for nuclear reactor: A review. *Progress in Nuclear Energy* 161 (2023), 104741. doi:10.1016/j.pnucene.2023.104741

B. Davison. 1958. *Neutron Transport Theory*. Clarendon Press. https://books.google.com/books?id=ZhlRAAAAMAAJ

Juan A. Monleon de la Lluvia, Mariya Brovchenko, Dimitri Rochman, and Eric Dumonteil. 2026. Toward Efficient Nuclear Data Uncertainty Quantification in Radiation Shielding Calculations. *Nuclear Science and Engineering* 200, 2 (2026), 257–279. doi:10.1080/00295639.2025.2510048

A de Lataillade, S Blanco, Y Clergent, J.L Dufresne, M El Hafi, and R Fournier. 2002. Monte Carlo method and sensitivity estimations. *Journal of Quantitative Spectroscopy and Radiative Transfer* 75, 5 (2002), 529–538. doi:10.1016/S0022-4073(02)00027-4

Xi Deng, Shaojie Jiao, Benedikt Bitterli, and Wojciech Jarosz. 2019. Photon Surfaces for Robust, Unbiased Volumetric Density Estimation. *ACM Trans. Graph.* 38, 4, Article 46 (July 2019), 12 pages. doi:10.1145/3306346.3323041

Xi Deng, Fujun Luan, Bruce Walter, Kavita Bala, and Steve Marschner. 2022. Reconstructing Translucent Objects using Differentiable Rendering. In *Proceedings of SIGGRAPH 2022*.

Xi Deng, Lifan Wu, Bruce Walter, Ravi Ramamoorthi, Eugene d'Eon, Steve Marschner, and Andrea Weidlich. 2024. Reconstructing translucent thin objects from photos. In *SIGGRAPH Asia 2024 Conference Papers* (Tokyo, Japan) *(SA '24)*. Association for

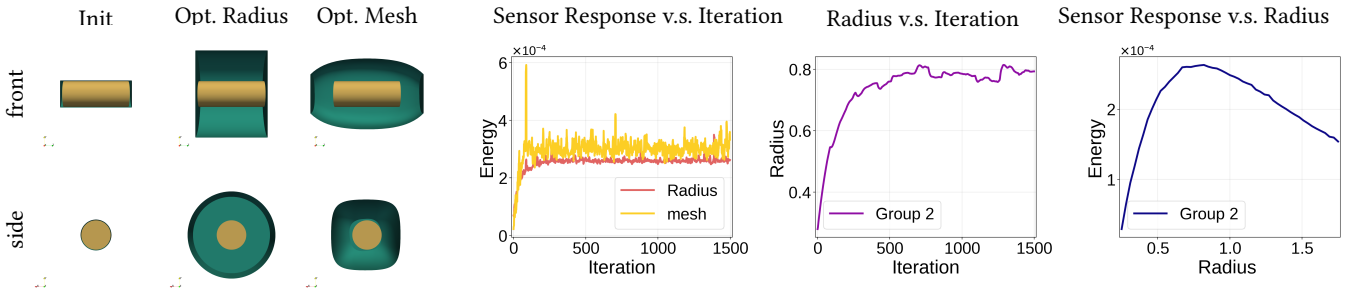


Fig. 11. Sensor Optimization. The sensor (yellow cylinder) is placed inside the reactor to capture high-energy neutrons. As shown in the side view (the second row), the neutron source is located to the right of the sensor. Because the sensor is most sensitive to neutrons in the lower energy group, a moderator (green envelope) is attached to the sensor to slow down fast neutrons through scattering events, thereby increasing the sensor response. In the classical design, the moderator is assumed to be cylindrical, and the optimal radius is determined by sweeping over a range of radii and simulating neutron transport for each configuration. We show that, by computing the gradient of the sensor response with respect to the radius, the optimal radius can instead be obtained efficiently using gradient descent. Furthermore, we fix the moderator volume to match that of the cylinder at the optimal radius and allow the shape to deform freely from the cylindrical initial condition using vertex-based geometry gradients. This gradient-based full-mesh optimization (the opt. mesh illustration) produces a new moderator shape with a higher sensor response (yellow curve). Notably, the optimized shape is no longer symmetric and shifts slightly away from the neutron source, as seen in the side view (row two of the opt. mesh illustration). This shift reduces neutron energy attenuation while increasing the number of neutrons scattered toward the sensor, resulting in improved response.

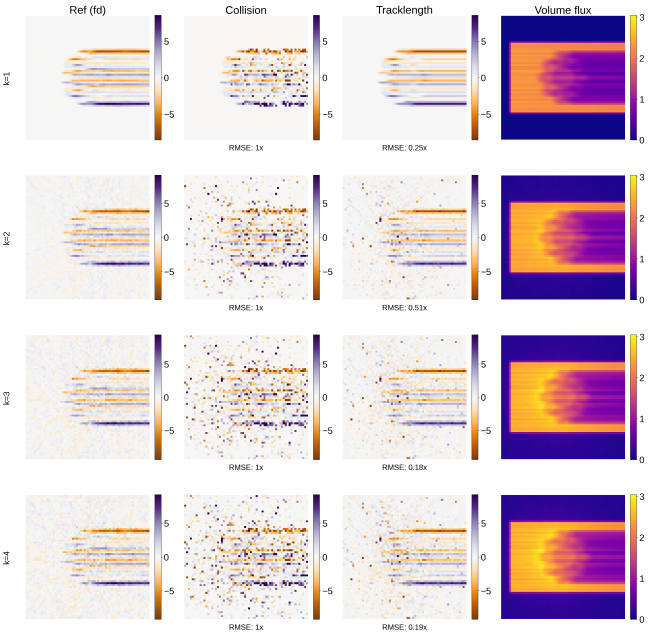


Fig. 12. We evaluate the combination of the track-length estimator and reparameterization on longer transport paths with $k = 1, 2, 3, 4$. We visualize a representative slice of the simulated volume flux field for the gear shape, comparing the collision estimator against our reparameterized track-length estimator applied to the final path segment.

Computing Machinery, New York, NY, USA, Article 124, 11 pages. doi:10.1145/3680528.3687572

Ioannis Gkioulekas, Shuang Zhao, Kavita Bala, Todd Zickler, and Anat Levin. 2013. Inverse volume rendering with material dictionaries. *ACM Trans. Graph.* 32, 6 (2013).

Ralf Habel, Per H Christensen, and Wojciech Jarosz. 2013. Photon beam diffusion: A hybrid monte carlo method for subsurface scattering. In *Computer Graphics Forum*, Vol. 32. Wiley Online Library.

Zili He, Sandrine Vinatier, Vincent Eymet, Vincent Forest, Bruno Bézard, Pascal Rannou, Sébastien Rodriguez, Emmanuel Marcq, Richard Fournier, Stéphane Blanco, Nada Mourtaday, Yaniss Nyffenegger-Péré, Sébastien Lebonnois, and Anni Määttänen. 2026. Simultaneous estimation of radiance and its sensitivities to radiative properties in a spherical-heterogeneous atmospheric radiative transfer model by Monte Carlo method: Application to Titan. *Journal of Quantitative Spectroscopy and Radiative Transfer* 350 (2026), 109722. doi:10.1016/j.jqsrt.2025.109722

Jan Eduard Hoogenboom. 1977. *Adjoint Monte Carlo methods in neutron transport calculations.* Ph. D. Dissertation. Delft University Press.

Wenzel Jakob, Sébastien Speierer, Nicolas Roussel, Merlin Nimier-David, Delio Vicini, Tizian Zeltner, Baptiste Nicolet, Miguel Crespo, Vincent Leroy, and Ziyi Zhang. 2022b. *Mitsuba 3 renderer.* https://mitsuba-renderer.org.

Wenzel Jakob, Sébastien Speierer, Nicolas Roussel, and Delio Vicini. 2022a. DrJit: A Just-In-Time Compiler for Differentiable Rendering. *Transactions on Graphics (Proceedings of SIGGRAPH)* 41, 4 (July 2022). doi:10.1145/3528223.3530099

Joel A. Kulesza, Terry R. Adams, Jerawan C. Armstrong, Simon R. Bolding, Forrest B. Brown, Jeffrey S. Bull, Timothy P. Burke, Alexander R. Clark, Robert Arthur Forster, III, Jesse F. Giron, Avery S. Grieve, Colin J. Josey, Roger L. Martz, Gregg W. McKinney, Eric J. Pearson, Michael E. Rising, Clell J. Solomon, Jr., Sriram Swaminarayan, Travis J. Trahan, Colin A. Weaver, Stephen C. Wilson, and Anthony J. Zukaitis. 2024. *MCNP® Code Version 6.3.1 Theory & User Manual.* Technical Report LA-UR-24-24602, Rev. 1. Los Alamos National Laboratory, Los Alamos, NM, USA. doi:10.2172/2372634

Jaroslav Křivánek, Iliyan Georgiev, Toshiya Hachisuka, Petr Vévoda, Martin Šik, Derek Nowrouzezahrai, and Wojciech Jarosz. 2014. Unifying Points, Beams, and Paths in Volumetric Light Transport Simulation. *ACM Trans. Graph.* 33, 4, Article 103 (July 2014), 13 pages. doi:10.1145/2601097.2601219

O. Leray, H. Ferroukhi, M. Hursin, A. Vasiliev, and D. Rochman. 2017. Methodology for core analyses with nuclear data uncertainty quantification and application to Swiss PWR operated cycles. *Annals of Nuclear Energy* 110 (2017), 547–559. doi:10.1016/j.anucene.2017.07.006

Tzu-Mao Li, Miika Aittala, Frédo Durand, and Jaakko Lehtinen. 2018. Differentiable monte carlo ray tracing through edge sampling. *ACM Trans. Graph.* 37, 6 (2018).

Xiaoqi Li, Youqi Zheng, Xianan Du, and Bowen Xiao. 2024. A new surrogate method for the neutron kinetics calculation of nuclear reactor core transients. *Nuclear Engineering and Technology* 56, 9 (2024), 3571–3584. doi:10.1016/j.net.2024.04.007

Yang Liu, Shaojie Jiao, and Wojciech Jarosz. 2022. Temporally sliced photon primitives for time-of-flight rendering. *Computer Graphics Forum (Proceedings of EGSR)* 41, 4 (July 2022). doi:10/jgzq

Guillaume Loubet, Nicolas Holzschuch, and Wenzel Jakob. 2019. Reparameterizing discontinuous integrands for differentiable rendering. *ACM Trans. Graph.* 38, 6 (2019).

Baptiste Nicolet, Alec Jacobson, and Wenzel Jakob. 2021. Large Steps in Inverse Rendering of Geometry. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 40, 6 (Dec. 2021). doi:10.1145/3478513.3480501

Baptiste Nicolet, Felix Wechsler, Jorge Madrid-Wolff, Christophe Moser, and Wenzel Jakob. 2024. Inverse Rendering for Tomographic Volumetric Additive Manufacturing. *ACM Trans. Graph.* 43, 6, Article 228 (Nov. 2024), 17 pages. doi:10.1145/3687924

Merlin Nimier-David, Thomas Müller, Alexander Keller, and Wenzel Jakob. 2022. Unbiased Inverse Volume Rendering with Differential Trackers. *ACM Trans. Graph.* 41, 4, Article 44 (July 2022), 20 pages. doi:10.1145/3528223.3530073

O. Petit, F.X. Hugot, Y.K. Lee, C. Jouanne, A. Mazzolo, and 91 Gif sur Yvette (France) CEA Saclay, Dept. Modelisation de Systemes et Structures. 2008. Tripoli-4 version 4 user guide. (2008).

D. Rochman, W. Zwermann, S. C. van der Marck, A. J. Koning, H. Sjöstrand, P. Helgesson, and B. Krzykacz-Hausmann. 2014. Efficient Use of Monte Carlo: Uncertainty Propagation. *Nuclear Science and Engineering* 177, 3 (2014), 337–349. doi:10.13182/NSE13-32

Theodore Rockwell, III. 1956. *Reactor Shielding Design Manual.* Technical Report. United States Atomic Energy Commision, Washington, DC (United States). doi:10.2172/4360248

Maxime Roger, Stephane Blanco, Mouna El Hafi, and Richard Fournier. 2005. Monte Carlo Estimates of Domain-Deformation Sensitivities. *Phys. Rev. Lett.* 95 (Oct 2005), 180601. Issue 18. doi:10.1103/PhysRevLett.95.180601

Paul K. Romano, Nicholas E. Horelik, Bryan R. Herman, Adam G. Nelson, Benoit Forget, and Kord Smith. 2015. OpenMC: A state-of-the-art Monte Carlo code for research and development. *Annals of Nuclear Energy* 82 (2015), 90–97. doi:10.1016/j.anucene.2014.07.048 Joint International Conference on Supercomputing in Nuclear Applications and Monte Carlo 2013, SNA + MC 2013. Pluri- and Trans-disciplinarity, Towards New Modeling and Numerical Simulation Paradigms.

Md Hossain Sahadath, Qiyun Cheng, Shaowu Pan, and Wei Ji. 2025. Deep Operator Network Based Surrogate Model for Neutron Transport Computation. 742–751. doi:10.13182/xyz-47261

Paolo Saracco, Sandra Dulla, and Piero Ravetto. 2016. The adjoint neutron transport equation and the statistical approach for its solution. *The European Physical Journal Plus* 131, 11 (2016), 412.

G.E Sjoden. 2002. Deterministic adjoint transport applications for He-3 neutron detector design. *Annals of Nuclear Energy* 29, 9 (2002), 1055–1071. doi:10.1016/S0306-4549(01)00096-2

Jerome Spanier and Ely M Gelbard. 2008. *Monte Carlo principles and neutron transport problems.* Courier Corporation.

Delio Vicini, Sébastien Speierer, and Wenzel Jakob. 2021. Path replay backpropagation: differentiating light paths using constant memory and linear time. *ACM Trans. Graph.* 40, 4 (2021).

X. Wang, R. Gocht, J. Ball, S. Mackie, E. Panontin, R. A. Tinguely, P. Raj, I. Holmes, A. A. Saltos, A. Johnson, and A. Grieve. 2024b. Neutronics simulations for the design of neutron flux monitors in SPARC. *Review of Scientific Instruments* 95, 8 (08 2024), 083560. doi:10.1063/5.0219508

Zichen Wang, Xi Deng, Ziyi Zhang, Wenzel Jakob, and Steve Marschner. 2024a. A Simple Approach to Differentiable Rendering of SDFs. In *SIGGRAPH Asia 2024 Conference Papers* (Tokyo, Japan) *(SA '24)*. Association for Computing Machinery, New York, NY, USA, Article 119, 11 pages. doi:10.1145/3680528.3687573

C. Yi, K. Manalo, M. Huang, M. Chin, C. Edgar, S. Applegate, and G. Sjoden. 2012. Optimization of a neutron detector design using adjoint transport simulation. American Nuclear Society, Inc., 555 N. Kensington Avenue, La Grange Park, Illinois 60526 (United States), American Nuclear Society - ANS; La Grange Park, IL (United States). https://www.osti.gov/biblio/22105838

Z. Yu, C. Zhang, O. Maury, C. Hery, Z. Dong, and S. Zhao. 2023. Efficient Path-Space Differentiable Volume Rendering With Respect To Shapes. *Computer Graphics Forum* 42, 4 (2023).

Sunghwan Yun, Cheol Woo Lee, Dong Won Lee, Sun-Ho Kim, Bongki Jung, Doo-Hee Chang, Hyung Gon Jin, and Chang Wook Shin. 2019. An optimization study for shielding design of D-D and D-T neutron generators. *Fusion Engineering and Design* 146 (2019), 531–534. doi:10.1016/j.fusengdes.2019.01.014 SI:SOFT-30.

Tizian Zeltner, Sébastien Speierer, Iliyan Georgiev, and Wenzel Jakob. 2021. Monte Carlo estimators for differential light transport. *ACM Trans. Graph.* 40, 4 (2021).

Cheng Zhang, Zhao Dong, Michael Doggett, and Shuang Zhao. 2021a. Antithetic sampling for Monte Carlo differentiable rendering. *ACM Trans. Graph.* 40, 4 (2021).

Cheng Zhang, Bailey Miller, Kan Yan, Ioannis Gkioulekas, and Shuang Zhao. 2020. Path-space differentiable rendering. *ACM Trans. Graph.* 39, 4 (2020).

Cheng Zhang, Lifan Wu, Changxi Zheng, Ioannis Gkioulekas, Ravi Ramamoorthi, and Shuang Zhao. 2019. A differential theory of radiative transfer. *ACM Trans. Graph.* 38, 6 (2019).

Cheng Zhang, Zihan Yu, and Shuang Zhao. 2021b. Path-Space Differentiable Rendering of Participating Media. *ACM Trans. Graph.* 40, 4 (2021).

Ziyi Zhang, Nicolas Roussel, and Wenzel Jakob. 2023. Projective Sampling for Differentiable Rendering of Geometry. *Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 42, 6 (Dec. 2023). doi:10.1145/3618385

Yekun Zhu, Min Tang, and Zheng Ma. 2026. DeepRTE: Pre-trained attention-based neural network for radiative transfer. *Computer Methods in Applied Mechanics and Engineering* 449 (2026), 118556. doi:10.1016/j.cma.2025.118556

E. Öztürk, R. Akers, S. Pamela, P. Peers, A. Ghosh, and The MAST Team. 2025. Inverse rendering of fusion plasmas: inferring plasma composition from imaging systems. *Nuclear Fusion* 65, 2 (jan 2025), 026020. doi:10.1088/1741-4326/ad9ab5

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009