XCP Computational Physics Workshop Final Report Draft

Scott Campbell

August 7, 2019

Abstract

The Thermal Radiative Transfer (TRT) equations describe the coupling between thermal radiation and material, which is an important physical process in many problems of interest to the astrophysics community. The Implicit Monte Carlo (IMC) method, originally developed over 40 years ago [1], is a standard solution methodology for the TRT equations. In this research, an improvement for IMC techniques for analyzing the interactions between a supernova and its circumstellar material is discussed, tested, and analyzed for a simplified geometry. This improvement uses a response function based variance reduction method to better estimate the observed time-dependent signal from a supernova system. The response function method improves the convergence, reliability, and accuracy of the IMC simulations. We were able to demonstrate said improvements by implementing this method in the open-source branson IMC code developed by Alex Long (*along@lanl.gov*) using modern object-oriented languages.

1 Introduction

Some ideas to add here:

- What is the supernova problem? Why is it difficult to obtain good tallies for supernova problems using IMC?
- Why is it important to improve efficiency for supernova simulations? (e.g. problem scale is large in space and time, so very computationally expensive... multiple physics involved that need to be modeled, etc.)
- (Very briefly) What are some of the basic variance reduction methods that people have used to try to improve efficiency for Monte Carlo? Mention weight windows / modified sampling – then mention that you will describe them in more detail later

RYAN: Here are two paragraphs for the first two points above (feel free to modify). Electromagnetic (EM) transients are the main observational probe of supernovae, providing insight into the explosion energy, dynamics, and compositions of the exploding stars. These transients are formed by the complex interaction of photons with matter expanding at high velocity, and with a circumstellar medium (CSM) that existed before the supernova. Modeling supernovae with CSM interactions in multiple dimensions and extracting numerical transients is a challenging computational problem, requiring high spatial resolution in areas, relative to the overall distance between the star and CSM [3, 5]. For the radiative transfer, in 1D deterministic [5] and Monte Carlo [2] have been applied to synthesize light curves and spectra.

However, to our knowledge, for 2D and 3D only simplified treatments of the radiation have been employed: parameterized radiative cooling [3, 4], the M1 moment closure approximation [6], or no treatment (only hydrodynamics) [7]. Nevertheless, a spherical (or multidimensional) geometry can significantly impact the properties of the EM transient (spectra, light curves) [6, 7], and the effect of higher-order radiative transfer on the observables may be non-negligible.

Monte Carlo lends itself well to multiple dimensions since the sourcing of particles in space can be adjusted to mitigate transport in low-energy regions.

In a simulation of a multidimensional supernova interacting with a CSM, obtaining well-sampled multi-frequency, multi-observer-angle spectra may generally be prohibitively expensive with Monte Carlo, assuming escaping particles are directly tallied as part of the transient. For instance, assuming a modest 100^3 cell 3D simulation, 10 observational views, and 100 observational wavelength bands, and further assuming 1% of particles from each cell are tallied as escaping flux, the potential number of particles required to obtain 1 tally in each point in the observational phase space would be $\sim 100^3 \times 10 \times 100 \times 100 = 10^{11}$ particles.

Given the expense of simulating this number of particles, variance reduction that takes into account statistics at a tally surface (i.e. the "telescope") is worth exploring. Variance reduction methods are implemented to improve simulation efficiency while producing equivalent (unbiased) results. Such methods include implicit capture, splitting, Russian Roulette, and weight windows, which are described in more detail in Section 2.3.

The remainder of this report outlines the background, motivation, and theory for the method, the algorithmic procedure, initial results, and an analysis of the cases and conditions where the method is useful.

2 Background and Theory

2.1 Thermal Radiation Transport

The scattering and absorption of a photon emitted from a material is described by the TRT equations:

$$\frac{1}{c}\frac{\partial I}{\partial t}(\vec{r},\vec{\Omega},\nu,t) + \vec{\Omega}\frac{\partial I}{\partial \vec{r}}(\vec{r},\vec{\Omega},\nu,t) + \sigma_a(\vec{r},\nu,T)I(\vec{r},\vec{\Omega},\nu,t) = 2\pi\sigma_a(\vec{r},\nu,T)B(\nu,T) + \frac{Q}{2}(\vec{r},\nu,t), \tag{1}$$

$$c_v(\vec{r},T)\frac{\partial T}{\partial t}(\vec{r},t) = \int_0^\infty \int_{-1}^1 \sigma_a(\vec{r},\nu',T)[I(\vec{r},\vec{\Omega}',\nu',t) - 2\pi B(\nu',T)]d\vec{\Omega}'d\nu' \tag{2}$$

where I is the specific intensity, T is the material temperature (keV), c is the speed of light, B is the Planck function, Q is the inhomogeneous source, c_v is the material-specific heat, and σ_a is the absorption opacity. Each of the terms in Eq. 1 corresponds to a loss or gain of photons from some phase space of the radiation field. The first term describes how the time behavior of the specific intensity depends on different gains and losses. The second is the streaming

term, which describes how photons are lost by spatial streaming out of the phase space. The third describes the loss due to absorption into the material. On the right-hand side, the first is a gain term describing the radiation source from material temperature, and the second term then describes an arbitrary source of radiation. Equations 1 and 2 are non-linearly coupled with material temperature.

2.2 Implicit Monte Carlo

Monte Carlo methods are used to model time-dependent, nonlinear, radiative transfer problems in complex three-dimensional configurations. This stochastic numerical method uses random sampling to determine where and how a particle moves through a material. Some Monte Carlo methods minimize the effects of discretization errors through a continuous treatment of energy, space, and/or angle leaving the primary errors to be rooted in stochastic uncertainties. However, Monte Carlo methods typically come at the expense of long run times (with a standard convergence rate $\propto \frac{1}{\sqrt{N}}$, where N is the number of histories simulated) and heavy use of machine resources.

The Implicit Monte Carlo (IMC) method is a specific variant of Monte Carlo originally developed by Fleck and Cummings in 1971 to solve the TRT equations [1]. IMC uses 'effective scattering' to model particle absorption/re-emission in a material for its current time step. This is represented by the Fleck factor, f, described as follows:

$$f = \frac{1}{1 + \frac{4acT^3\sigma\Delta t}{c_v}} \tag{3}$$

where a is the radiation constant, c is the speed of light, T is the material temperature, σ is the material opacity, Δt is the time step, and c_v is the material's specific heat.

The IMC method applies two note-worthy approximations: semi-implicit discretization of time, and the linearization of the TRT equations. By linearizing the TRT equations, IMC is known to lead to inaccurate or non-physical results. Additionally, the time discretization is not fully implicit as implied, but rather semi-implicit as it is typically too expensive to converge otherwise.

In the *branson* IMC code used for this research, the spatial domain is discretized into rectangular regions referred to as cells, each with a constant material temperature, radiation temperature, and density per timestep. Particles are tracked as they move through the cells and can be absorbed into the material, scattered, or stream out of the spatial domain. A particle is transported until it reaches the end of the time step or is fully absorbed by the material. Information about the system (e.g. fluence) is accumulated during the timestep and stored in a 'tally'.

2.3 Variance Reduction for IMC

Since the IMC method is stochastic, there will always be some statistical error in the result. Inherent issues with IMC methods include slow convergence and large computational requirements citeAW16. Furthermore, in cases where a material is optically thick and the distance from source to observation point (i.e. a tally surface) is large enough, the analog IMC method will fully absorb most particles into the material before they reach the tally, leaving the tally poorly sampled.

Due to these issues, variance reduction methods for IMC methods are necessary to provide equivalent answers while using fewer resources and converging faster. In the IMC code used, three standard variance reduction techniques were incorporated in addition to the response function method. Currently, there are a large variety of variance reduction methods suited to different purposes. Common examples include implicit capture coupled with history termination, splitting and Russian Roulette, exponential transformation, forced collisions, source biasing, correlated sampling, optical reciprocity, and weight windows [9].

Implicit capture is a standard method implemented in the *branson* code. Implicit capture, or absorption suppression, adjusts a particle's weight at every 'scattering' event according to the following relationship:

$$w_{n,\ i+1} = w_{n,\ i}(1 - \frac{\sigma_a}{\sigma_a + \sigma_s}) \tag{4}$$

where σ_a and σ_s are the absorption and scattering cross-sections respectively. Using the Fleck factor, the above becomes

$$E_{particle,\ i+1} = E_{particle,\ i} e^{-\sigma_a \cdot f \cdot d_{event}}, \tag{5}$$

where d_{event} is the distance to the next position the particle travels to. Implicit capture allows particles to stay active longer, increasing the likelihood that a particle reaches the tally surface.

Weight windows are another common variance reduction technique, in which each cell in the problem mesh is given a weight window center bounded by higher and lower values [9]. Once a particle enters a new cell, if a particle's weight is not within the weight window, one of two processes will occur. If the particle's wright is above the window, it will be split into additional particles. Particles whose weight is below the window will undergo a technique such as Russian Roulette [10] to remove the particle from the simulation.

All of the variance reduction methods listed above do not ensure that a tally surface will always be well sampled. In cases where particles have short mean free paths (i.e. traveling through a material with a high opacity), particles are still not guaranteed to pass through the tally surface. To ensure a tally is well-sampled, ideally, all particles would contribute to the tally at least once before being fully absorbed into the material. To this end, a response function, in theory, will meet this requirement.

2.4 Next Event Estimators

Next event surface crossing estimators (NXTEVT) are a standard method for analyzing Monte Carlo methods. NXTEVT contributes to a tally surface the expected value of a particle that will cross through the tally surface. This method is an effective tool to minimize variance in situations where particles have limited histories, and large mean free paths in the problem material [14, 11, 12].

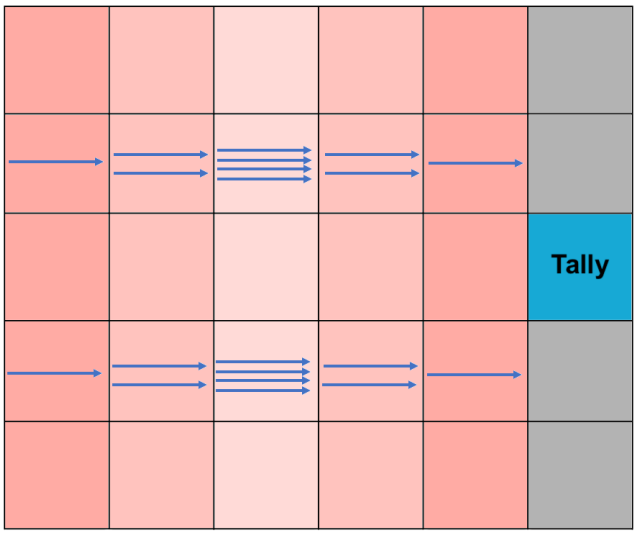

Figure 1: A visual demonstrating why splitting is ineffective in optically thin regions: the split particles will all have generally the same direction and will stream with few collisions.

NXTEVT estimators are implemented instead of splitting and Russian Roulette (e.g. weight windows) to get a well-sampled tally in small geometric regions. Figure 1 demonstrates why splitting is ineffective in problems of low opacity. This method is preferred in cases where a limited number of particles reach an area of interest, i.e. tally. For example, this can result from optically thin regions where few collisions occur, mitigating scattering which would statistically direct more particles towards the tally region. In this case, particles will be transported further, and consequently, will deposit more energy and be rouletted before they reach they can reach the tally. Additionally, splitting and weight windows do not change a particle's direction, furthering the effects described earlier [15].

This method points particles towards the region of interest to increase the sampling of important angles. The tally surface is 'scored' via the following equation: assuming that a particle is at a position $\vec{r}$, direction $\vec{\Omega}$, and weight w_0 intersecting a tally with a surface area S_{tally} at a position $\vec{r'}$ from the particle, the scored flux, $\phi(\mu)$, is given by

$$\phi(\mu) = w_0 \frac{e^{-\int_{\vec{r}}^{\vec{r'}} \Sigma_t(s)ds}}{S \cdot \mu} \tag{6}$$

where μ is the angle between $\vec{\Omega}$ and the surface normal at $\vec{r'}$, and Σ_t is the total cross-section in the material.

For materials with high opacities and long distances from source to tally, the NXTEVT method produces unphysical answers with increased variance relative to standard methods [16]. We are principally concerned in situations where such conditions exist, i.e. applications in astrophysical events, and therefore chose to investigate new methods.

2.5 Response Function Theory

A response function is thought to be an effective variance reduction method since it contributes an adjusted energy value to the tally at every scatter event, rather than adding a contribution only once the particle passes through the tally surface. Because of this feature, the tally surface should be well sampled using the response function compared to standard variance reduction methods.

A response function attempts to calculate the probability that a particle 'survives' to the tally surface following its current trajectory. The weight of the particle-based on this probability and current energy should the particle reach the tally surface is calculated by

The response function is generated by tracing a number of particles through the problem domain starting uniformly on the tally surface, directed at a cosine-distribution angle. Since each cell may have a unique σ_a, every possible path from the source to the tally surface will result in a potentially unique contribution to the tally. The response function attempts to model this by averaging the absorption opacity based on the accumulation of all weighted σ_a values from the previous cells into an effective opacity, σ_r. At every scattering event, the contribution of the particle to the tally is calculated by

$$E_{contribution} = E_{particle} e^{-(\sigma_r + \frac{1}{c\Delta t}) d_{tally}} \tag{7}$$

where $E_{particle}$ is the current energy of the particle, Δt is the current time step, and d_{tally} is the distance of the particle to the tally surface along its path.

2.6 Supernova and CSM Parameters

We attempt to model a snapshot of a supernova interacting with a circumstellar medium (CSM). The CSM has high enough mass to theoretically produce a superluminous supernova (e.g. SN 2006gy). The time for the snapshot is chosen to be after the supernova ejecta hits the CSM and produces a shock. This shock ionizes the CSM, which increases the opacity and consequently the optical depth. Moreover, the increase in density from the shock, coincident with the ionized region, also contributes to an increase in the optical depth of the CSM.

The approximate properties of each layer near the time of peak luminosity ($\sim$ 70 days) for model D2 of Moriya, Blinnikov, et al (2013) (hereafter M13) are as follows in Table 1.

Supernova Ejecta	Ejecta-CSM Shock	CSM (pre-shock)
$R_{\rm s,min} = 0$ cm	$R_{\rm s,min} = 10^{15.829}$ cm	$R_{\rm s,min} = 10^{15.831}$ cm
$R_{\rm s,max} = 10^{15.829}$ cm	$R_{\rm s,max} = 10^{15.831}$ cm	$R_{\rm s,max} = 10^{16}$ cm
$\rho_{\rm ej} = 10^{-14}$ g/cm^3	$\rho_{\rm s} = 10^{-12}$ g/cm^3	$\rho_{\rm CSM} = 10^{-14}$ g/cm^3
$T_{\rm ej} = 10^4$ K	$T_{\rm s} = 10^6$ K	$T_{\rm s} = 10^4$ K
$c_v = 10^6$ erg/K/g	$c_v = 10^6$ erg/K/g	$c_v = 10^6$ erg/K/g
$\kappa_{\rm ej} = 0.3$ cm^2/g	$\kappa_{\rm s} = 0.3$ cm^2/g	$\kappa_{\rm CSM} = 10^{-4}$ cm^2/g

Table 1

For spherical symmetry, these values give masses of about 6, 8.5, and 14 for the interior ejecta, shocked ejecta-CSM, and CSM, respectively, which are on the order of the model values presented by M13. The optical depth through the internal ejecta is about 20, and is about 10 through the shocked ejecta-CSM layer. The preshock CSM is evidently hot enough to be ionized, hence it can contribute another 10 mean-free-paths of optical depth. The spatial width of the shock layer is only 0.46 % of the radius of the shock, despite providing 1/3 of the total optical depth.

2.7 Rescaling Parameters

We may scale adjust dimensions and scale the parameters to simplify setup of the input. This can also be of use when attempting to test different types of supernovae conditions, but preserving physical properties or numerical resolution.

An adjustment of the overall domain size (radius),

$$\frac{R}{R_0} \approx 10^{-16} \quad , \tag{8}$$

where values subscripted with 0 are unscaled, gives dimensions of O(1 cm). Relevant properties to preserve are the optical depth, light crossing time, and ratio of total time to the absorption-emission timescale. To preserve the light crossing time, we simply scale the total simulation time,

$$t = \frac{R}{R_0} t_0 \quad , \tag{9}$$

Similarly, assuming opacity is constant or piecewise-constant, the optical depth is preserved when

$$\kappa = \kappa_0 \frac{R_0}{R} \quad , \tag{10}$$

where density has canceled from the left and right side. To preserve the ratio of the absorption-emission time scale to the total time, t,

$$t_{ae} = \frac{c_v}{4\kappa a c T^4} = \frac{R}{R_0} t_{ae,0} = \frac{R}{R_0} \frac{c_{v,0}}{4\kappa_0 a c T^4} \quad , \tag{11}$$

which implies

$$c_v = c_{v,0} \quad . \tag{12}$$

Density and temperature have been left unchanged. The important aspect of this problem is the geometric structure and optical depth. To lower the spatial resolution requirements, the ejecta-CSM shock layer may be spread from 0.4% to $\sim$10% of the problem length while preserving optical depth. To do so, the density of the layer can be lowered to compensate for the increased size of the layer.

3 Method and Technical Approach

The implementation of the response function variance reduction method largely follows a standard Monte Carlo approach with the exception of running an inverse transport to generate the response function before running the forward transport problem.

3.1 Problem Initialization

At the start of the problem, several parameters are defined based on the properties of the materials in the problem domain. The absorption opacity, σ_a, is calculated using the Fleck factor:

$$\sigma_a = f\sigma. \tag{13}$$

The source emission, S_e for each cell is defined as

$$S_e = c \cdot a \cdot \Delta t \cdot \sigma_a \cdot T^4, \tag{14}$$

and total source emission, $S_{e,\,total}$, is the sum of S_e for each cell;

$$S_{e,\,total} = \sum_{i=1}^{n_{cell}} S_{e,\,n}. \tag{15}$$

The normalized weight, w_{ideal}, of each particle in the simulation is the quotient of $S_{e,\,total}$ and $N_{particles}$. The number of particles to be emitted in each cell is then the quotient of S_e and w_{ideal}. A check is run to ensure that each cell emits at least one particle. The time of emission for each particle is determined from a uniform distribution over the time step:

$$t_{0,\,particle} = \xi \Delta t \tag{16}$$

where $\xi \in [0, 1]$ is a uniformly distributed random number. The temperature of each cell is calculated by

$$T_{cell} = \rho c_v \Delta t \Delta E \tag{17}$$

where ρ is the material density and ΔE is the difference between S_e and the absorbed energy in each cell.

3.2 Inverse Transport

At the beginning of each time step, the response function is generated for the mesh. Our approach discretizes the function over the domain space, using each cell as a region to calculate the response opacity, σ_r. To calculate σ_r for each cell, a set number of particles, $N_{response}$, are traced over the mesh accumulating information.

The first step is to initalized each particle. Each particle's starting position is initialized uniformly on the tally surface using the following equations:

$$\phi = 2\pi\xi, \tag{18}$$

$$\mu = 1 - 2\zeta, \tag{19}$$

$$\theta = \arccos\mu \tag{20}$$

where $\xi,\ \zeta$ are uniformly random numbers, $\xi,\ \zeta \in [0, 1]$. The position vector, $\vec{r}$, is then given by

$$\vec{r} = \begin{bmatrix} r_0 \\ r_1 \\ r_2 \end{bmatrix} = \begin{bmatrix} x_{tally} + R_{tally}\sqrt{1-\mu^2}\cos\phi \\ y_{tally} + R_{tally}\sqrt{1-\mu^2}\sin\phi \\ z_{tally} + R_{tally}\mu \end{bmatrix} \tag{21}$$

where the tally is centered at $(x_{tally}, y_{tally}, z_{tally})$ with a radius of R_{tally}.

Additionally, the direction of the particle is initialized towards the source via a cosine-distributed angle. First, the inward unit normal from the point to the source **N** is given by:

$$\mathbf{N} = \begin{bmatrix} \hat{x} \\ \hat{y} \\ \hat{z} \end{bmatrix} = \frac{1}{n_r}\begin{bmatrix} x_{tally} - r_0 \\ y_{tally} - r_1 \\ z_{tally} - r_2 \end{bmatrix} \tag{22}$$

where $\frac{1}{n_r}$ is the unit vector fraction, $[(x_{tally} - r_0)^2 + (y_{tally} - r_1)^2 + (z_{tally} - r_2)^2]^{\frac{1}{2}}$. Then an angle factor, f_{angle}, is calculated by:

$$f_{angle} = \sqrt{|1 - \hat{z}^2|}. \tag{23}$$

Lastly, the angle $\vec{\Omega}$ is:

$$\vec{\Omega} = \begin{bmatrix} \hat{\Omega}_x \\ \hat{\Omega}_y \\ \hat{\Omega}_z \end{bmatrix} = \frac{1}{n_\Omega}\begin{bmatrix} \hat{x}\cos\theta + \hat{z}\hat{x}\sin\theta\cos\phi f_{angle} - \hat{y}\sin\theta\sin\phi f_{angle} \\ \hat{y}\cos\theta + \hat{z}\hat{y}\sin\theta\cos\phi f_{angle} + \hat{x}\sin\theta\sin\phi f_{angle} \\ \hat{z}\cos\theta - f_{angle}\sin\theta\cos\phi \end{bmatrix}. \tag{24}$$

where ϕ is from Eq. 18, $\cos\theta$ is from Eq. 20, and $\frac{1}{n_\Omega}$ is the vector unit fraction, similar to above.

With the particle initialized, it is then traced through the mesh. Scattering and absorbtion events are excluded. Instead, the following information is recorded as the particle passes a distance d_{cell} through each cell:

1. The total distance the particle has traveled through each cell, $d_{total,\ particle} = \sum d_{cell}$.
2. The sum of the distance the particle travels through each cell multiplied by the cells absorption opacity, $\sigma d_{total,\ particle} = \sum d_{cell}\sigma_{a,\ cell}$.
3. The total distance, $d_{total,\ cell} = \sum d_{n,\ cell}$ for every particle n that passes through each cell.
4. The total distance multiplied by the absorption opacity, $\sigma d_{total,\ cell} = \frac{\sigma d_{total,\ particle}}{d_{total,\ particle}} d_{cell}$ for every particle that passes through each cell.

where $\sigma_{a,\ cell}$ is the absorption opacity of the cell the particle is in. The average response value for each cell can then calculated using the following equation:

$$\sigma_r = \frac{\sigma d_{total,\ cell}}{d_{total,\ cell}}. \tag{25}$$

Figure 2 is a visual representation of this process.

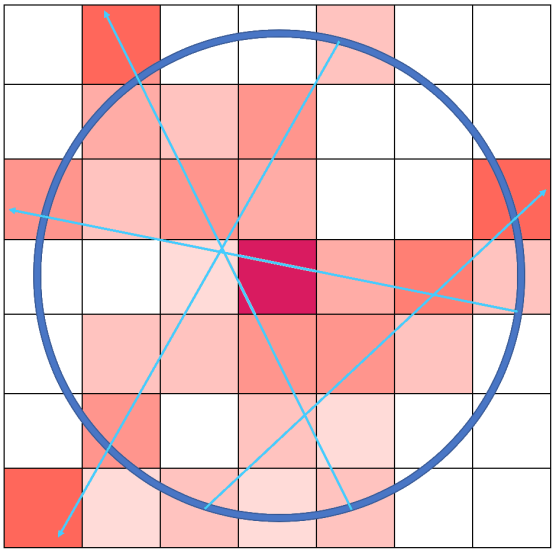

Figure 2: A simplified visualization of the response function method. Particles are traced from the surface of the tally (blue ring) 'towards' the source (purple square) until they exit the mesh. As the particle moves, an adjusted opacity, σ_r, is calculated based on how far the particle has traveled from its origin on the tally surface, as well as the $\sigma_{a,\,cell}$ values of the cells that it has passed through. A darker shade of red corresponds to a higher σ_r value – a less likely chance that a particle will 'make it' to the tally. These values are then used to calculate an effective contribution to the tally based on the cell the particle being transported is in.

3.3 Forward Transport

The forward transport problem follows closely to a typical IMC scheme. Each particle is sourced in a cell and is trasported over the duration of the timestep. Upon creation, a contribution is added to the tally using Eq. 7.

For each particle, while it remains in the problem domain and the timestep, a distance to the next scattering event is caluclated:

$$d_{scatter} = -\log \frac{\xi}{(1-f)(\sigma_a + \sigma_s)} \tag{26}$$

where $\xi \in [0, 1]$ is a uniformly distributed random number. Additionally, the distance to the cell boundary, $d_{boundary}$, distance to the tally surface, d_{tally}, and the distance to reach the end of the timestep, d_{census} are calculated. It should be noted that we are interested only in escaping flux, so d_{tally} is the distance to the surface of the tally where the particle would exit through. If the particle will not pass through the tally based on its current direction, d_{tally} is set to be $\approx \infty$ to machine limits, such that the contribution from Eq. 7 approaches 0. The distance to the end of the timestep, d_{census}, is also determined. The distance to the next 'event' is then calculated by:

$$d_{event} = min(d_{scatter}, d_{boundary}, d_{census}) \tag{27}$$

as the smallest distance is the most likely event to occur first. The particle is then moved a distance d_{event} and its weight reduced accd. to Eq. 5.

At this point, if the particles weight is below a set fraction, then it is determined to have an inconsequental effect on the simulation and is no longer tracked. If the event was determined to be a scatter ($d_{event} \equiv d_{scatter}$), then a new direction is sampled from a cosine-distribution and a contribution is added to the tally using Eq. 7. Because a limited number of particles are traced in the inverse problem, it is possible that some cells will not have a σ_r value. If this is the case, then more particles are traced trough the inverse problem. This is repeated as long as the cell has no contributions.

If instead the event was passing across a cell boundary ($d_{event} \equiv d_{boundary}$), then the particle's current cell information is updated to the new cell. Lastly, if the event is the particle reaching the end of the timestep ($d_{event} = d_{census}$), then the particle is put into a list of all particles that 'survived' the timestep to resume transport in the next timestep.

To maintain an unbiased simulation, at the end of the timestep after all particles have been transported, the total initial energy of all the census particles are redistributed randomly to a smaller number of representative particles. This allows for additional particles to be trasported within each timestep. Additionally, the temperature T of each cell is updated accd. to Eq. 17. The timestep information is then updated and the above scheme is repeated for the entirety of the simulation.

4 Test Problem

To check the validity of the response function method, a point source problem was devised where the analytic flux was known. This problem consisted of a $0.1 \times 0.1 \times 0.1$ cm^3 cube at an initial T_e and T_r temperature of $1\,\text{deg}$ C, encased in a $1.0 \times 1.0 \times 1.0$ cm^3 cube at an initial T_e and T_r temperature of $0.01\,\text{deg}$ C. The entire problem was then surrounded by a vacuum to mitigate all escaping flux from returning to the problem domain. The heat capacity of the heated 'source' and material was set to be high such that the re-emittance of photons would be small compared to the source. Similarly, the opacity of the heated cells was set to be high compared to the rest of the material. Not sure if these statements are correct or true - please edit/remove if needed! For the generation of the response function, $10,000$ particles were used to trace through the mesh and generate the σ_r values for each cell. A spherical tally surface was centered at the origin of the heated cell with a radius of 1.0 cm.

The validity of our method was tested using measures of average flux and variance as well as the figure of merit (FoM) of the method, both as a function of the number of particles run through the simulation.

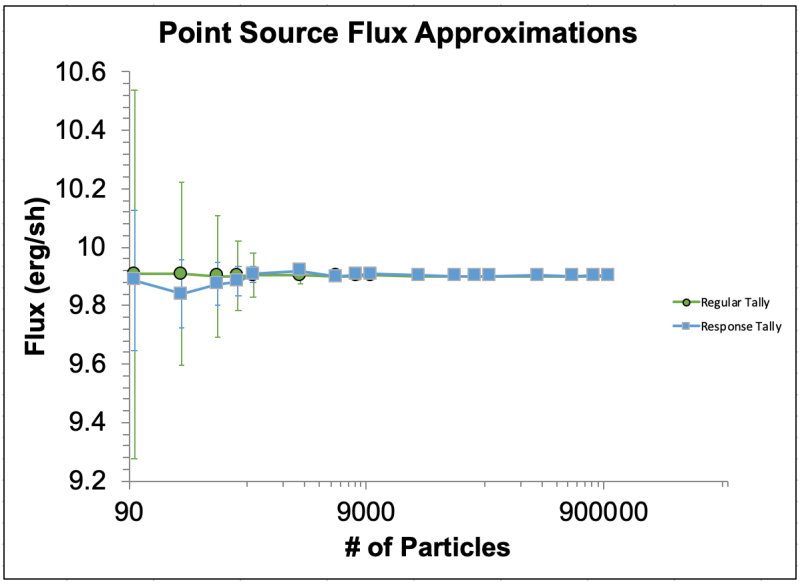

Figure 3: A graph showing the average values of the flux for a point source problem as well as the variance as a function of the number of particles used in the simulation. This shows that our method has far less variance for fewer simulated particles, which implies that it is a more reliable method under these conditions. Datapoint 2 for the response function results appears to be an anomaly – likely resulting from statistics or a minor bug in our code.

Results of the average flux and variance testing are shown in Fig. 3. The analytic flux for a point source problem is determined using the following equation:

$$Flux = \sigma_a a c T^4 \Delta t V_{src} \tag{28}$$

where a is the radiation constant, c is the speed of light, T is the initial temperature, Δt is the time step duration, and V_{src} is the source region volume. Average flux and variance were determined from all of the time steps in the situation averaged at the end of the simulation. Add a sentence about why this is valid The general trend of this plot demonstrates that the response function produced values largely centered around the analytic flux with significantly lower variance. It should be noted, however, that the second data-point is anomalous: likely resulting from statistics or a minor bug in the code.

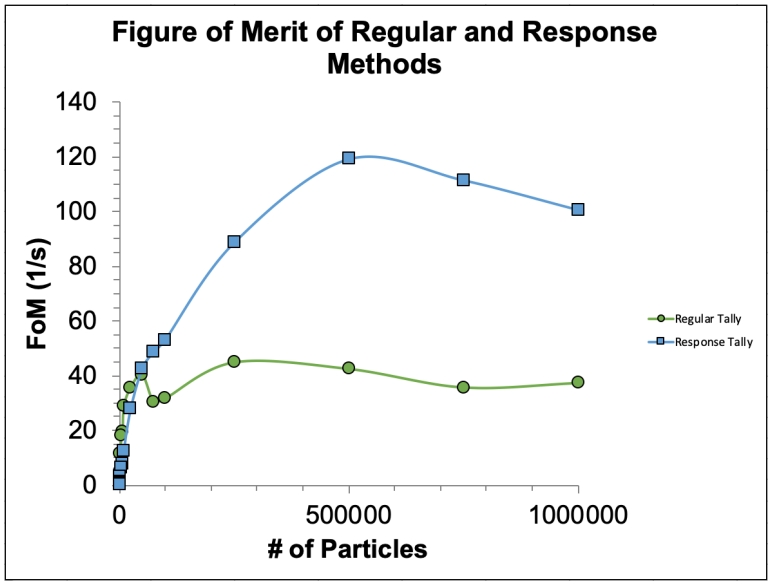

Figure 4: A graph showing the figure of merits (FoM) of the regular tally method as well as the response function. A higher FoM corresponds to a method that provides less variance in a more efficient period of time. This figure shows that after approx. 50k particles, our method results in a greatly improved FoM compared to a regular tally. This provides a strong basis for the usefulness of our method.

The FoM for the response function is compared to the standard tally method in Fig. 4. The FoM for the different methods is determined by the following equation outlined in Lewis and Miller 1993 [10]:

$$FoM = \frac{1}{\sigma^2(x) \cdot t_{run}} \tag{29}$$

where $\sigma^2(x)$ is the variance of the method and t_{run} is the time to run the simulation. A higher FoM for a method indicates a method that produces an answer more efficiently between variance and run time. As in Fig. 3, this plot also illustrates an improvement of the response function over the standard tally: the FoM of the response function is equivalent to or better than that of the standard tally. At around $\approx 50k$ particles run through the simulation, the FoM of the response function method consistently exceeds that of the standard tally method by more than a factor of 2.

The results of these two measures provide a strong basis for the validity and usefulness of the response function tally for our simplified point-source problem. Additionally, the integration of the response function theory into the computational method is validated further by Fig. 5. Our response function method has a far greater number of contributions ($\approx 40\%$ greater for the given input variables), matching the intended design for the problem types at hand. Considering the impact of these results gives us a strong foundation to test more complex problems.

5 Results & Supernova Applications

This section presents the results of the response function method applied to a simplified supernova problem. The geometry of the supernova is represented as cubic shells for the ease of the generation of an input file. This problem is further referred to as the 'cubanova' problem. This geometry can be seen on the left-hand side of Fig. 6. The innermost shell (yellow) is the supernova eject, the next out is the Ejecta-CSM Shock (red), followed by the CSM (pre-shock) (green) from Table 1 surrounded by void (blue). The simulation results were compared against standard tally methods as a point of reference. For all the simulations, a spherical tally surface was centered in the problem domain, $(0, 0, 0)$, with a radius of 2.

Figure 6 shows a comparison between the true σ_a of the problem at time t_0 (left-hand side) as well as the generated response values, σ_r, (right-hand side) for each cell. The response value for each cell represents the average opacity between the cell and every possible tally location.

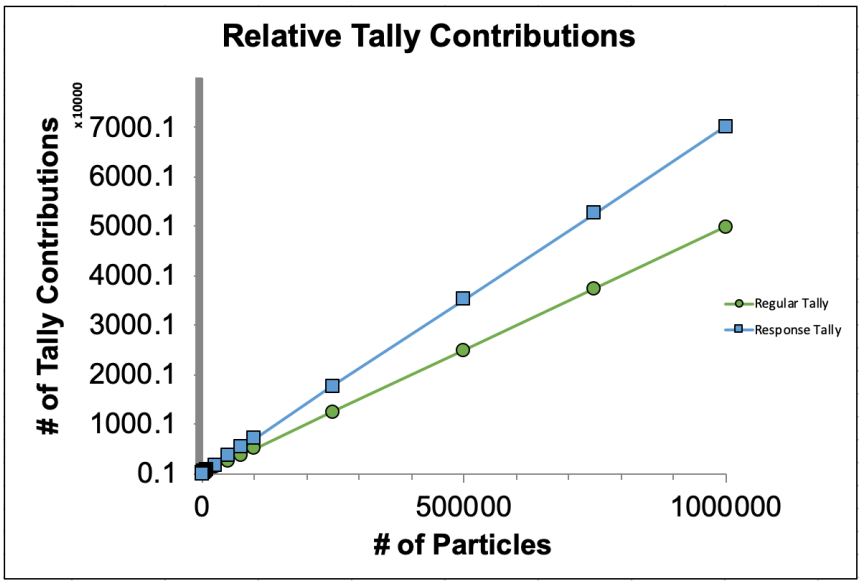

Figure 5: A graph showing the number of particles (or times in the case of the response function) that contribute to the tally. In general, a method will be expected to have a smaller variance with a greater number of particles that contribute to the tally.

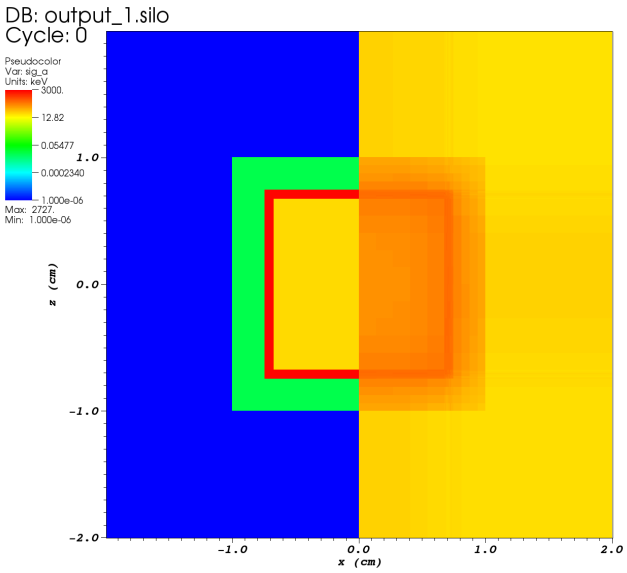

Figure 6: A plot of the geometrically simplified supernova problem showing the true σ_a value of the problem (left-hand side) as well as the σ_r values generated by the response function method (right-hand side).

From a visual standpoint, the response values match expectations. The voids outside the tally region have the smallest σ_r as the only way an outgoing particle could contribute to the tally would be to scatter backward and pass through to the other side of the tally. Similarly, the void regions inside the tally have the next highest σ_r as it can either pass through the tally in the direction it is traveling in, or pass through a different point after scattering. Similar logic follows for the supernova and CSM regions. The smearing artifacts along the axis in the response values show a limitation of the direction-independent σ_r calculation; we believe that disregarding the particle's directionality while it is traced through the mesh results in some cells receiving a smaller σ_r value.

Since the flux of the cubanova problem is time-dependent, i.e. no analytic flux, the average and variance of each

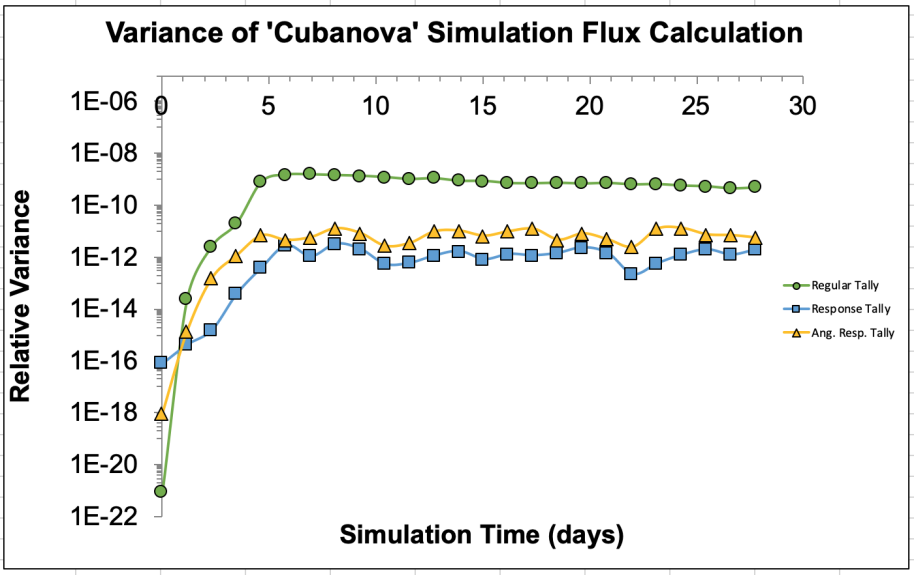

Figure 7: A graph demonstrating the variance of the regular tally, and response function tally for the simplified supernova simulation from Fig. 6 as a function of the simulation time.

method was calculated for each time step from a number of independent simulations each with a unique random seed. Figure 7 shows the results from twenty independent simulations for the standard tally and response tally methods. The general trends indicate that the response function method has a significantly lower variance at any given time step where the flux is statistically meaningful (see Fig. 8). It should also be noted that the variance of the response method is relatively stable compared to that of the regular tally, implying that the response method will be stable and predictable throughout the simulation relative to standard methods.

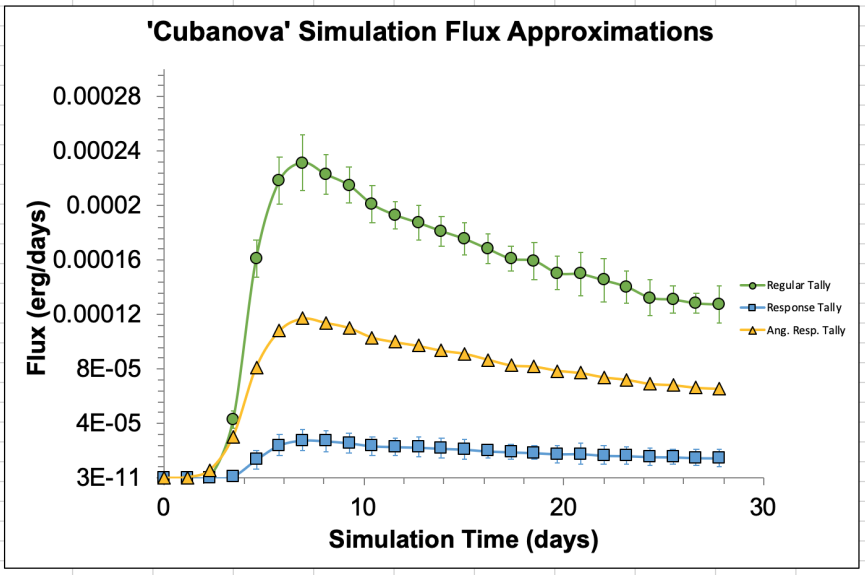

Figure 8: The relative flux calculations made by the regular tally method and the response function tally method. As in Fig. 3, the response method has a smaller variance, however, it also drastically underestimates the flux. This likely stems from the use of a spherical tally surface instead of a directional tally surface for this particular problem.

Figure 8 illustrates the average flux and standard deviation from the simulations run in Fig. 7. It is important to note that the response function produces an average flux significantly lower than the standard tally method which is

thought to closely approximate the true flux. We believe that this discrepancy is not in the fault of the method, but rather an artifact of the use of a spherical tally on which the response function is generated. Under this, we would expect that a planar tally surface will produce a flux that accurately represents the true value as it has its directionality built-in.

ADD REF. TO SECTION SHOWING RESULTS WITH PLANE TALLY HERE???

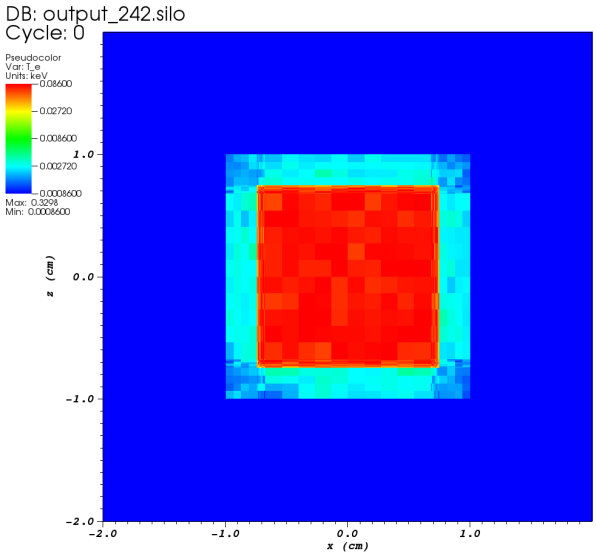

Figure 9: A plot of the electron temperature at the end of the simulations shown in Figs. 6, 7, 8. The overheating likely results from the small corner cells.

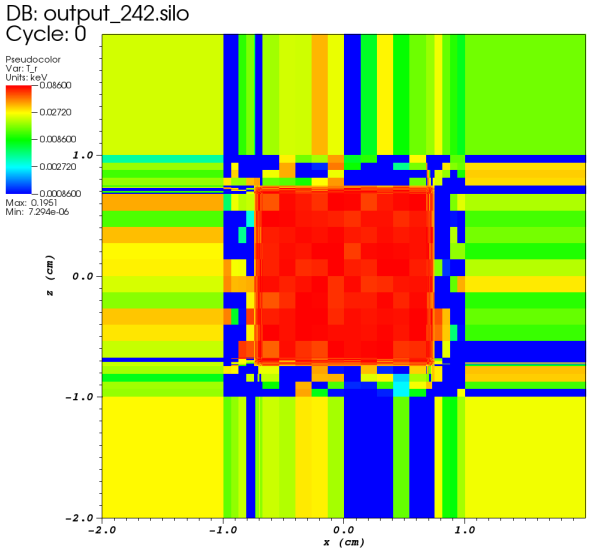

Figure 10: A plot of the radiation temperature from the plot in Fig. 9. The noise in the radiation field results from a limited number of photons, the optical thickness of the material, and the coarse mesh granularity of the void region.

6 Conclusions

References

[1] J.A. Fleck, Jr. and J.D. Cummings, Jr., "An implicit Monte Carlo scheme for calculating time and frequency dependent nonlinear radiation transport," *J. Comp. Phys.* 8, pp. 313–342, (1971).

[2] D. Kasen and S.E. Woosley, "Type II supernovae: model light curves and standard candle relationships," *ApJ* 703, pp. 2205–2216, (2009).

[3] A.J. van Marle, N. Smith, S.P Owocki, B. van Veelen, "Numerical models of collisions between core-collapse supernovae and circumstellar shells," *MNRAS* 407, pp. 2305–2327, (2010).

[4] A.J. van Marle, R. Keppens, "Multi-dimensional models of circumstellar shells around evolved massive stars," *A & A* 547, A3 (2012).

[5] T.J. Moriya, S.I. Blinnikov, N. Tominaga et al., "Light-curve modelling of superluminous supernova 2006gy: collision between supernova ejecta and a dense circumstellar medium," *MNRAS* 428, pp. 1020–1035, (2013).

[6] A. Vlasis, L. Dessart, and E. Audit, "Two-dimensional radiation hydrodynamics simulations of superluminous interacting supernovae of Type IIn," *MNRAS* 458, pp. 1253–1266, (2016).

[7] A.T. McDowell, P.C. Duffel, and D. Kasen, "Interaction of a Supernova with a Circumstellar Disk," *ApJ* 856, (2018).

[8] A.B. Wollaber, "Four Decades of Implicit Monte Carlo," *J. Comp. Phys.*, vol. 45, no. 1-2, pp. 1–72, (2016).

[9] J.T. Landman, "Variance reduction strategies for implicit Monte Carlo simulations," PhD thesis, Texas A&M University, *Texas A&M University*, (2016).

[10] E.E. Lewis and W.F. Lewis, Jr., *Computational Methods of Neutron Transport*, (American Nuclear Society, La Grange Park, IL), pp. 309–339.

[11] L.B. Lucy, "Improved Monte Carlo techniques for the spectral synthesis of supernovae," *A&A* 345, pp. 211-216, (1999).

[12] L.B. Lucy, "Monte Carlo techniques for time-dependent radiative transfer in 3-D supernovae," *A&A*, pp. 1–5, (2005).

[13] J.E. Sweezy, "A Monte Carlo volumetric-ray-casting estimator for global fluence tallies on GPUs," *J. Comp. Phys.* 372, pp. 426-445, (2018).

[14] W.T. Dailey, "Ray next-event estimator transport of primary and secondary gamma rays," PhD thesis, Air Force Institute of Technology, *Air Force Institute of Technology*, 2011.

[15] T.E. Booth, R. Arthur Forster, and R.L. Martz, "MCNP Variance Reduction Developments in the 21st Century," *Nuclear Technology* 180:3, pp. 355–371, (2012).

[16] M. Kawai and Y Hayashida, "Development of General Next Event Surface Crossing Estimators for Monte Carlo Method," *J. Nuc. Sci. & Tech.* 23, pp. 833–839, (1986).