

Computing Temperatures in Optically Thick Protoplanetary Disks

Lawrence F. Capuder, Jr.¹
Ohio State University Columbus, OH 43210

Dr. Neal Turner²
Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA 91109

We worked with a Monte Carlo radiative transfer code to simulate the transfer of energy through protoplanetary disks, where planet formation occurs. The code tracks photons from the star into the disk, through scattering, absorption and re-emission, until they escape to infinity. High optical depths in the disk interior dominate the computation time because it takes the photon packet many interactions to get out of the region. High optical depths also receive few photons and therefore do not have well-estimated temperatures. We applied a modified random walk (MRW) approximation for treating high optical depths and to speed up the Monte Carlo calculations. The MRW is implemented by calculating the average number of interactions the photon packet will undergo in diffusing within a single cell of the spatial grid and then updating the packet position, packet frequencies, and local radiation absorption rate appropriately. The MRW approximation was then tested for accuracy and speed compared to the original code. We determined that MRW provides accurate answers to Monte Carlo Radiative transfer simulations. The speed gained from using MRW is shown to be proportional to the disk mass.

I. Introduction

Interstellar gas clouds are held up against gravity by internal kinetic energy, a pressure gradient, and magnetic forces. A star forms when a large gas cloud reaches the critical mass given its radius, called the Jeans mass, and collapses. That is, a star forms when the gravitational potential energy becomes strong enough to overcome the thermal and magnetic energy of the gas cloud. The gas cloud has angular momentum, which is a conserved quantity, and therefore a spinning disk of dust and gas forms around the young star. Standard theory suggests that planet formation occurs in these disks.

We worked with a Monte Carlo radiative transfer code to simulate the transfer of energy through these protoplanetary disks. The Monte Carlo radiative transfer code used is similar to that used by Bjorkman and Wood (2001). The code tracks photons from the star into the disk, through scattering, absorption and re-emission, until they escape to infinity. Tracking these photons in this manner allows for the calculation of both the temperature distribution within the disk and its spectral energy distribution (SED), which requires keeping track of where and at what frequency radiation leaves the disk. The program uses a technique where the photons are followed through a random optical depth to an interaction location, where they are either scattered or absorbed with a probability given by the albedo and emissivity of the dust. If the photon is scattered, it goes off at a randomized angle. If the photon is absorbed, it immediately gets re-emitted at a random angle with a new frequency set by the local envelope temperature. As the packet travels, it has a certain probability of interacting with the dust and giving it energy. The Monte Carlo code incorporates this using an energy prescription described by Lucy (1999). This process is repeated until the photon escapes the disk, traveling out to infinity.

High optical depths in the disk interior dominate the computation time because it takes the photon packet many interactions to get out of the region. High optical depths also receive few photons and therefore do not have well-estimated temperatures. We have applied a modified random walk approximation (MRW) for treating high optical depths, which is described by Min et. al. (2010) with corrections made by Robitaille (2010), to speed up the Monte Carlo calculations. The MRW is implemented by calculating the average number of interactions the photon packet will undergo in diffusing within a single cell of the spatial grid and then updating the packet position, packet frequencies, and local radiation absorption rate appropriately.

¹ USRP Student

² USRP Mentor

II. Modified Random Walk

A. Description

In regions of the disk where the length scale over which density, ρ , and temperature, T , change is much greater than the mean free path of the photon in the region, the energy transport is by radiative diffusion, which can be described by the radiative diffusion equation (Wehrse et al. 2000).

$$\nabla \cdot (D \nabla E) = \frac{1}{c} \frac{\partial E}{\partial t}, \quad (1)$$

where E is the local energy density, c is the speed of light, and D is the diffusion coefficient which in an approximation is given by

$$D = \frac{1}{3\rho} \frac{\int_0^\infty \frac{B_\nu(T)}{\chi_\nu} d\nu}{\int_0^\infty B_\nu(T) d\nu} = \frac{1}{3\rho \bar{\chi}}, \quad (2)$$

where B_ν is the Planck function, χ_ν is the frequency dependent mass extinction coefficient, and $\bar{\chi}$ is a mean opacity (Robitaille 2010).

The solution to Eq. 1 for an infinite homogenous medium is

$$\psi(r, t) = \frac{1}{(4\pi D ct)^{3/2}} \exp\left(-\frac{r^2}{4D ct}\right),$$

where $\Psi(r, t)$ is the fraction of the energy that has diffused to position r at time t (Min et. al. 2009).

B. Implementation

A criterion for deciding whether the MRW procedure should be started was suggested by Min et al. (2009). It consists of determining whether the distance to the nearest cell wall is greater than a few times the Rosseland mean free path:

$$d_{\min} > \frac{\gamma}{\rho \bar{\chi}}, \quad (4)$$

The γ parameter determines the balance between accuracy and speed. If gamma is set very low, the MRW approximation is worse because the considered sphere might be optically thin at long wavelengths, which means that the photon would escape the sphere directly rather than by diffusion. This leads to overestimated temperatures. Setting γ higher means decreasing performance gain because the photon packets would need to undergo more interactions before escaping the disk.

After the criterion is met in a region, a sphere of radius R_0 is set up around the position of the packet, R_0 being the distance from the packet to the cell wall. Inside this sphere, the diffusion approximation can be solved exactly (Min 2009). The following algorithm can then be applied to implement Eq. 3 (Robitaille 2010). First, a random number is sampled from $\zeta \in [0, 1]$ to solve the following equation for y :

$$\zeta = 2 \sum_{n=1}^{\infty} (-1)^{n+1} y^{n^2} \quad (5)$$

As y tends to 1, the sum needs to be computed to higher and higher values for numerical accuracy. The simplest way to do this is to accurately calculate y over a range of y values and then interpolate for y for a given ζ in the Monte-Carlo code. Once y is computed, the total distance traveled by the packet in its path out of the sphere can be computed, which is given by

$$ct = -\ln y \left(\frac{R_0}{\pi}\right)^2 \frac{1}{D} \quad (6)$$

where D is calculated using opacities at the temperature of the cell at the very end of the Monte-Carlo simulation. This can be estimated by dividing the energy of the cell by the fraction of the total packets that have been run in the simulation when calculating the temperature. The energy absorbed by the sphere for this distance is then

$$E = E_{\gamma} ct \rho \bar{\kappa}_p \tag{7}$$

where E_{γ} is the energy of the packet and $\bar{\kappa}_p$ is the Planck mean absorption coefficient, for which the temperature of the cell at the very end of the simulation is used.

The packet is then reemitted immediately at the frequency given by the Bjorkman and Wood (2001) temperature correction method, which is done by sampling the frequency from $dB_{\nu}(T)/dT$. The temperature used for $dB_{\nu}(T)/dT$ should be the current temperature of the cell, after the energy from the MRW step is added, in order to preserve radiative equilibrium and relax to the correct temperature. This is done by solving for the temperature given the energy of the cell without dividing by the fraction of the packets that have been run in the simulation. This same temperature is assigned as the temperature of the cell.

III. Results

A. Correctness of MRW

i) Spherical Symmetry

Correctness of the MRW routine was first shown for a simple spherically symmetric, isotropic single cell where the packet was emitted from the center. The results are shown in Table 1. The energy predictions are all normalized by the Eq. 7 energy prediction. The standard error in each case can be estimated by 1 over the square root of the number of packets. As the table shows, all results are within 2 standard errors and are therefore acceptable results.

Number of Packets	Standard Error	Eq 7 Energy Prediction	MRW On Energy Imparted	MRW off Energy Imparted
1,000	0.032	1	1.004	0.994
10,000	0.010	1	0.995	1.015

Table 1- Results of simple single-celled runs. All predictions are normalized to the Eq. 7 energy prediction.

ii) Axial Symmetry

The MRW routine was tested for correctness as compared to a run done without the subroutine as a control. All of these runs were made using a minimum mass solar nebula (MMSN) disk structure shown in Figure 1, which contains the minimum amount of mass needed to build the planets of our solar system. This model has a surface density of 1700 g/cm^2 at AU. This density then falls off as radius to the $3/2$. The packets are also bounced when they get within 20 g/cm^2 from the interior, where an isothermal approximation would normally be implemented but are not in these runs, in order to speed up computation times (Chiang & Goldreich 1997). We found that as the number of packets used in the runs with MRW turned on increased, the deviation from the control decreased. These results using $\gamma=1$ are shown in Figures 2-5. Red points indicate cells where at least one MRW step has occurred. It is reasonable that the cells where the MRW has occurred should be more deviant than those where it has not because MRW only occurs in optically thick regions that receive the fewest packets and therefore have the least well estimated temperatures. The root mean squares of the deviation in cells where at least one MRW occurred are 0.0711, 0.0470, 0.0400, and 0.0343 for cases using 1×10^6 packets, 5×10^6 packets, 1×10^7 packets, and 5×10^7 packets respectively. This implies that the results of the MRW runs grow more accurate as the number of packets used increases, and any error remaining for large numbers of packets is no bigger than a few percent.

The same sort of analysis was done for $\gamma=3$. $\gamma=3$ was chosen because it was the highest value of γ with which a significant number of cells still underwent at least one MRW step. This value would be higher if the packets were not bounced when they got into a certain optical depth from the interior. The interior is the most optically thick region of the disk, which would cause the MRW condition to be satisfied in more cells. The root mean squares of the deviation in cells where at least one MRW occurred are 0.0425 and 0.0349 for cases using 5×10^6 packets and 1×10^7 packets respectively. This is a very small increase in accuracy compared to the cases

where $\gamma=1$, especially considering γ was increased to its highest reasonable value given a 20 g/cm^2 bounce layer.

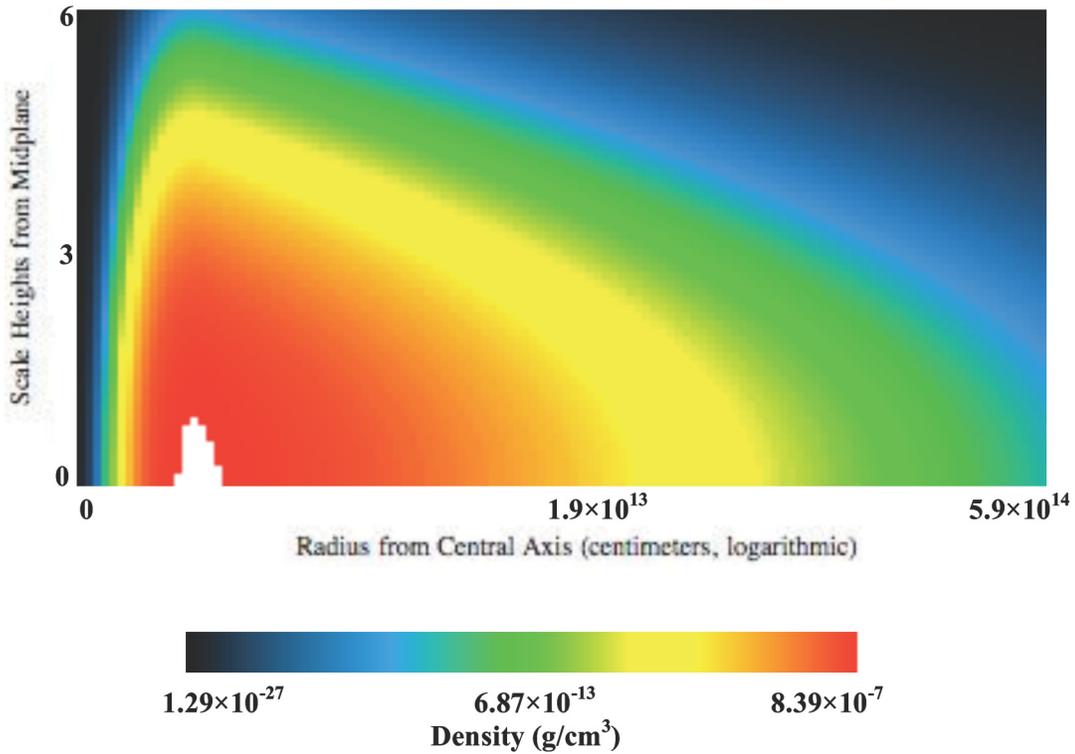


Figure 1-Density profile of the grid used for minimum mass solar nebula (MMSN) model. Color scheme is on a logarithmic scale.

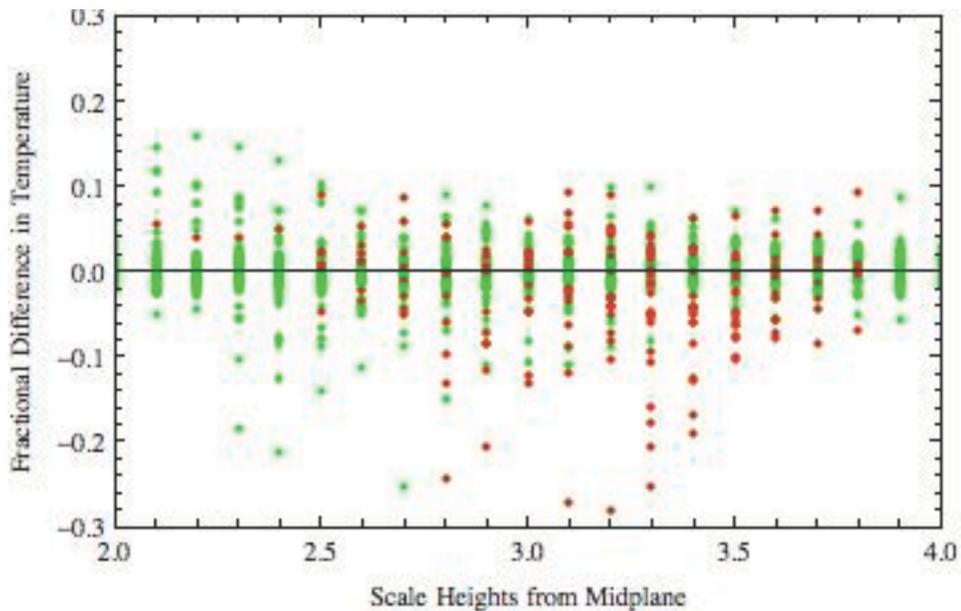


Figure 2-Scale heights from the midplane versus fractional deviation from the control model for a MRW run using 1×10^6 packets each. Red points indicate cells where at least one MRW step has occurred.

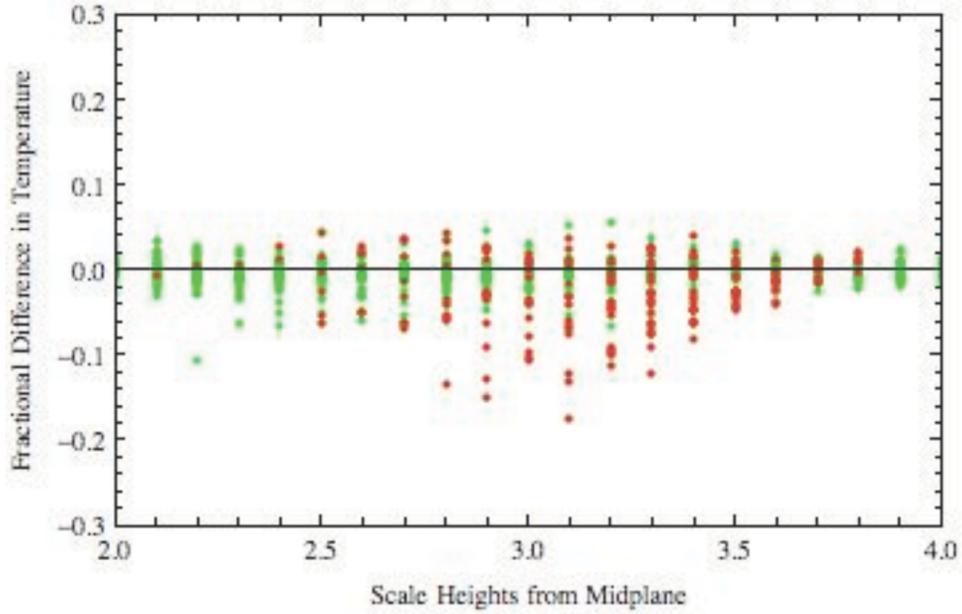


Figure 3-Scale heights from the midplane versus fractional deviation from the control model for a MRW run using 5×10^6 packets each. Red points indicate cells where at least one MRW step has occurred.

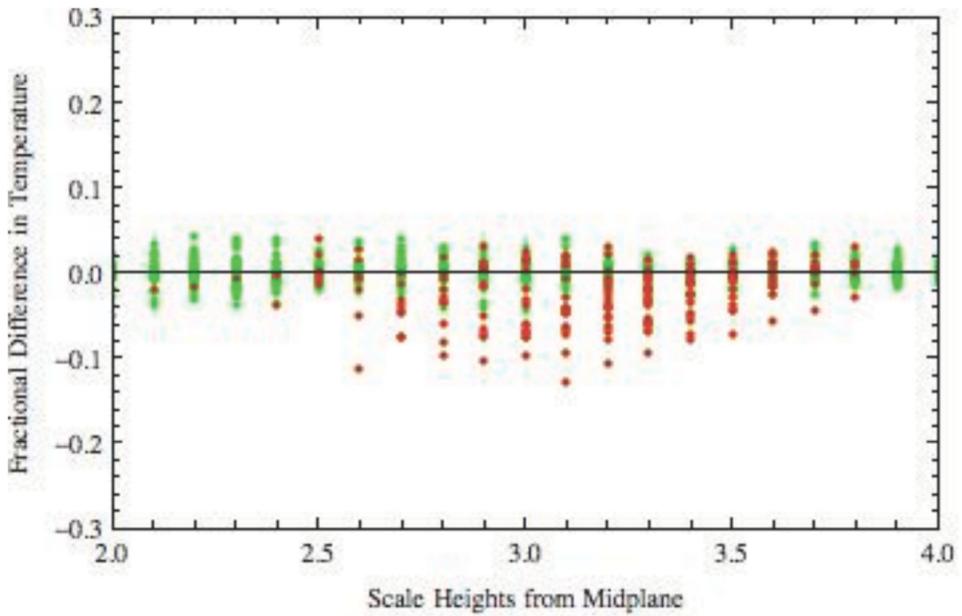


Figure 4-Scale heights from the midplane versus fractional deviation from the control model for a MRW run using 1×10^7 packets each. Red points indicate cells where at least one MRW step has occurred.

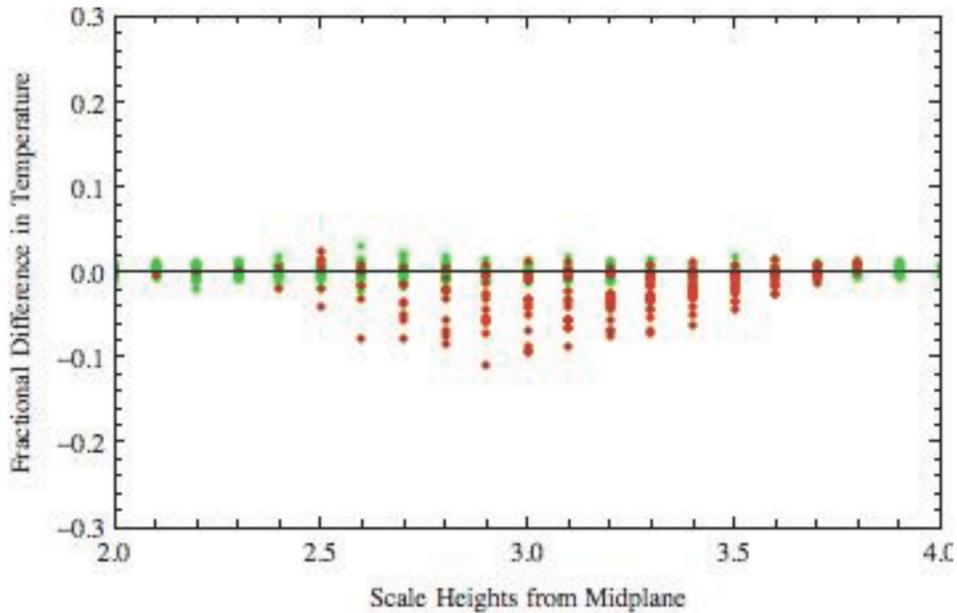


Figure 5-Scale heights from the midplane versus fractional deviation from the control model for a MRW run using 5×10^7 packets each. Red points indicate cells where at least one MRW step has occurred.

B. Speed of MRW

The MRW routine was tested for speed as compared to a run done without the subroutine as a control. Tests were done for disks that have 1×10^{-4} , 0.1, and 1 times the MMSN mass without bouncing packets that get in regions that are too optically thick. The results are shown in Table 2. The similarity for all of the 1×10^{-4} *MMSN can be explained by the fact that the disk was too optically thin for MRW to occur. The MMSN simulation for MMSN without MRW was just over halfway finished at the time listed in Table 2. Table 2 shows that as the disk mass increases, so too does the speed benefit from using MRW. This is the same result received by Min et. al. (2009). We achieved an order of magnitude speed-up or more in the calculations with disk mass exceeding 10% of the MMSN.

Tests were also done for disks that have 1, 10, and 100 times the MMSN mass that bounced packs in regions that are less than 20 g/cm^2 away from the midplane of the disk. The $\gamma=3$ run was only done for the 1*MMSN mass case. As discussed in Section A.ii, $\gamma=3$ was chosen because it was the highest value of γ with which a significant number of cells still underwent at least one MRW step. Table 3 again shows that as the disk mass increases, so too does the speed benefit from using MRW. However, the speed up is far less than in the runs shown in Table 3 where no bounce threshold was used. A negligible amount of speed was lost from increasing γ from 1 to 3, however this is mainly because $\gamma=1$ had a very small speed up from the no MRW simulation time.

Mass	Time without MRW	Time with MRW $\gamma=10$	Time with MRW $\gamma=1$
1×10^{-4} *MMSN	1.00	1.02	1.01
1×10^{-1} *MMSN	440	50.6	29.2
MMSN	1000+	110	37.9

Table 2- Speed of a full simulation for different masses of the disk. The results are normalized to the 1×10^{-4} *MMSN mass run time without MRW, which took 490 seconds on one core of a 2.8 GHz Quad-core Intel Xeon processor. These simulations used no bounce threshold.

Mass	Time without MRW	Time with MRW $\gamma=3$	Time with MRW $\gamma=1$
MMSN	1.000	0.956	0.855
10*MMSN	1.145	-	0.957
100*MMSN	1.319	-	1.021

Table 3- Speed of a full simulation for different masses of the disk. The results are normalized to the MMSN mass run time without MRW, which took 21515 seconds on one core of a 2.8 GHz Quad-core Intel Xeon processor. These simulations used a bounce threshold of 20 g/cm².

IV. Acknowledgements

This research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, and was sponsored by the Undergraduate Student Research Program and the National Aeronautics and Space Administration.

V. References

- Bjorkman, J.E. & Wood, K. 2001, ApJ, 554, 615
 Chiang, E.I. & Goldreich, P. 1997 ApJ 490, 368
 Lucy, L. B. 1999, A&A, 344, 282
 Min, M., Dullemond, C.P., Dominik, C., de Koter, A., & Hoenier, J.W. 2009, A&A, 497, 155
 Robitaille, T. P. 2010, A&A, 520, A70
 Wehrse, R., Baschek, B., & von Waldenfels, W. 2000. A&A, 359, 780