RADIATIVE TRANSFER IN GALAXIES

RADIATIVE TRANSFER IN GALAXIES

By
RORY WOODS, B.Sc., M.Sc.

A Thesis
Submitted to the School of Graduate Studies
in Partial Fulfillment of the Requirements
for the Degree of

Doctor of Philosophy

McMaster University

DOCTOR OF PHILOSOPHY (2015) McMaster University
(Physics and Astronomy) Hamilton, Ontario

TITLE: Radiative Transfer in Galaxy Formation

AUTHOR: Rory Woods, B.Sc. (Mount Allison University), M.Sc. (McMaster University)

SUPERVISOR: Professor James Wadsley

NUMBER OF PAGES: 1

Abstract

In this thesis, we present a novel algorithm for computing the radiation field in astrophysical simulations.

Dedicated to...

Acknowledgements

Thank you to all that helped.

"Some sort of quote?"

Albert Einstein (1879-1955)

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

It doesn't take much to convince a physicist of the importance of photons - astrophysical objects speak in photons. As astronomers, we receive all of our information in the universe through photons. In order to understand the objects we observe, we must understand photons; how are they created? How are they removed? What processes can alter a photon? To answer these questions, let's explore some of the many objects and photon energies present in astrophysics.

1.1 Astrophysics and Radiation

- FUV - 6.2-10.16 ev -
- Lyα - 10.16-10.25 -
- EUV - 12.4 - 124 ev -
- Soft X-rays - 124 ev - 5 kev -
- hard X-rays - 5-10 kev

- IR - 1.24mev - 1.7 ev -
- Cosmic Rays - ...

1.2 Overview

Chapter 2 will go over the background of radiative transfer and the currently available codes that solve the radiative transfer problem. It will also motivate the need for a new code in a particular niche. Chapter 3 will introduce the new radiative transfer method that we have developed. Chapter 4 will demonstrate the strengths and weaknesses of the new algorithm through a variety of numerical and phyical tests. Chapter 5 will show the results of using the algorithm on an isolated galaxy in the FUV band, and will also focus on future projects that the algorithm will be used for. Finally, chapter 6 contains the conclusions of this thesis.

Chapter 2

Radiative Transfer

This chapter will contain an overview of current radiative transfer methods and where we stand.

2.1 The Radiative Transfer Problem

When considering the transfer of photons, we must consider the scale we are dealing with. At the individual photon level, propagation is described by classical electrodynamics. Once we get to larger scales, however, it is more useful to treat radiation in "packets" or as an energy flux.

Consider an infinitesimal patch of area, dA, normal to a direction $\hat{n}$. We consider an infinitesimal solid angle, $d\Omega$, and consider all rays passing through the area and within the solid angle (see figure 2.1).

The energy through this area patch, within the solid angle, in time dt, and within the frequency range $d\nu$ is

$$dE = I_\nu dA dt d\Omega d\nu, \tag{2.1}$$

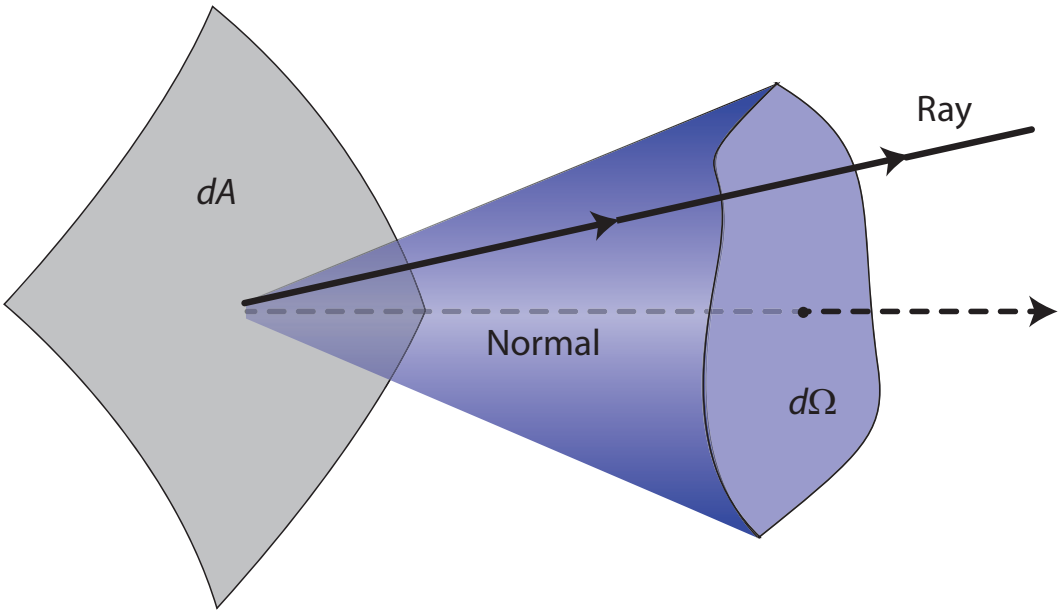


Figure 2.1: The geometry for all rays at a point p through area dA within solid angle $d\Omega$. Figure adapted from Rybicki and Lightman [1986], figure 1.2.

where I_ν is *specific intensity* (specific because it is within a frequency range; dropping the frequency dependence makes this intensity). Specific intensity has units of energy per unit area per unit time per unit solid angle per unit frequency. It is useful to consider radiation in terms of intensity because it enables a macroscopic description of radiation that includes microscopic effects like scattering and absorption.

We can recover familiar values such as flux (or pressure or density) by taking moments of the intensity,

$$F_\nu = \int I_\nu \cos\theta d\Omega, \tag{2.2}$$

where F_ν is the specific flux (flux at a particular wavelength).

Let us now consider the passage of these rays through some matter. If we consider a ray, then energy may be added or removed from this ray due to

absorption (removing photons), emission from the matter (adding photons), or scattering (scattering into or out of the ray). We first consider emission.

We define the specific (monochromatic) emission emission coefficient, j, as the energy emitted per unit time, per unit solid angle, per unit volume, and per unit frequency,

$$dE = j_\nu dV dt d\Omega d\nu. \tag{2.3}$$

If we trace along a ray with cross section dA some distance ds, it will cover a volume of $dV = dAds$. Since equation 2.3 and equation 2.1 only differ by a factor of distance (dA compared to dV), we can find the change intensity along the beam due to emission as

$$dI = j_\nu ds. \tag{2.4}$$

Equation 2.4 describes the amount of intensity added to a ray along some path ds due to spontaneous emission. If emission were the only process to worry about, finding intensity would be a simple matter of integrating the equation. [However, we must consider other physical processes for a more complete description.]

We next consider absorption. Consider again a ray traveling along a path ds. The amount of intensity lost due to absorption can be defined as

$$dI = -\alpha I ds, \tag{2.5}$$

where α is called the absorption coefficient and has units of distance^{-1}. It can be shown [Rybicki and Lightman, 1986] that α is a function of more

commonly known variables,

$$\alpha = -n\sigma I ds = -\rho\kappa I ds \tag{2.6}$$

where n is the number density of particles, σ is the cross section (in units of distance squared) of each absorbing particle, ρ is the mass density, and κ is the opacity (in units of distance squared per unit mass). Notice that the only difference between $n\sigma$ and $\rho\kappa$ is the average mass of the absorbing particles. Note that our algorithm has chosen to use ρ and κ.

Finally, we consider scattering. Scattering is a process that both subtracts and adds to the intensity. We can define a specific emission coefficient for scattering by equating the power per unit volume per frequency emitted to the power received,

$$j_{s,\nu} = \sigma_\nu J_\nu, \tag{2.7}$$

where σ_ν is the specific scattering coefficient, and J_ν is the specific mean intensity, defined as

$$J_\nu = \frac{1}{4\pi}\int I_\nu d\Omega. \tag{2.8}$$

Before combining all of the processes affecting radiative transfer, it is useful to introduce a variable called the specific *Source Function*,

$$S_{a,\nu} \equiv \frac{j_\nu}{\alpha_\nu}, \tag{2.9}$$

$$S_{s,\nu} \equiv \frac{j_\nu}{\alpha_\nu}. \tag{2.10}$$

The source function is the ratio of emission to absorption and describes the intensity that an object will tend to. In the case of pure absorption, emission is 0 and so the source function is 0, since the intensity would tend to 0. In the case of pure emission, the source function is infinite and intensity tends to infinity since nothing is removing photons.

We now have the base equations to put together a description of radiative transfer that includes the processes of spontaneous emission, absorption, and scattering. Combining equations 2.4, 2.5, 2.7, 2.8, 2.9, and 2.10, we can write

$$\begin{aligned}\frac{dI_\nu}{ds} &= (-\alpha_\nu I_\nu + j_\nu) - (\sigma_\nu I_\nu + j_{s,\nu}) \\ &= -\alpha_\nu(I_\nu - S_{a,\nu}) - \sigma_\nu(I_\nu - S_{s,\nu}) \\ &= -(\alpha_\nu + \sigma_\nu)(I_\nu - S_\nu),\end{aligned} \tag{2.11}$$

where the combined source function S_ν is defined as

$$S_\nu \equiv \frac{\alpha_\nu S_{a,\nu} + \sigma_\nu S_{s,\nu}}{\alpha_\nu + \sigma_\nu}. \tag{2.12}$$

According to equation 2.7, the source function for scattering is actually mean intensity (equation 2.8), meaning that the above equation is actually an integro-differential equation - it is a function of I_ν, $\frac{dI_\nu}{ds}$, and $\int I_\nu d\Omega$. Thus, any equation involving scattering is significantly more difficult to solve. Numerical solutions to integro-differential equations are usually specialized and complex [cite someone].

For this reason, scattering is often omitted from radiative transfer solver

due to very large added computational cost. In this thesis, our solutions do not explicitly account for scattering, though it is possible scattering-like behavior (due to properties of the RT algorithm and the properties of SPH) may result in some solutions [remove this line I think].

If scattering is omitted, equation 2.11 is simplified to a nicer form. We combine equations 2.4, 2.5, and 2.9 to obtain

$$\frac{dI_\nu}{ds} = -\alpha_\nu I_\nu + j_\nu. \tag{2.13}$$

It is now useful to introduce optical depth τ_ν,

$$\tau(s) = \int_{s_0}^{s} \alpha_\nu(s')ds' = \int_{s_0}^{s} \rho(s')\kappa_\nu(s')ds'. \tag{2.14}$$

Optical depth is a unitless value that describes the mean free path of a photon between interactions. The distance needed in the integral to give $\tau_\nu = 1$ should correspond to one mean free path given the absorption coefficient α_ν. It is useful to rewrite equation 2.13 in terms of τ_ν and S_ν by simply dividing by α_ν

$$\frac{dI_\nu}{d\tau_\nu} = -I_\nu + S_\nu. \tag{2.15}$$

Equation 2.15 is the transfer equation for radiation as it is most commonly seen. A solution can be obtained by using an integrating factor of e^{τ_ν}, which gives the formal solution to the transfer equation

$$I_\nu(\tau_\nu) = I_\tau(0)e^{-\tau_\nu} + \int_0^{\tau_\nu} e^{-(\tau_\nu - \tau'_\nu)} S_\nu(\tau'_\nu)d\tau'_\nu. \tag{2.16}$$

Solving the above equation at a point in a simulation would give you a

radiation field that accounted for emission and absorption at all other points in the simulation. This would then be repeated at all points for which a radiation field should be known.

It should now be clear why radiative transfer is a difficult problem; analytically, it involves integrals over source functions that are not necessarily known at all points in space, with both density and opacity varying as a function of position as well. Numerically, we are trying to solve a function of seven variables - three position, two angular, time, and frequency - $I = I(x, y, z, \theta, \phi, t, \nu)$.

[Add further simplifications to the equation here? Just talk about the absorption solution? Mention that integrating over all emission typically amounts to summing over all sources.]

2.2 Current Methods

The equations presented in section 2.1 are very difficult to solve if approximations are not made. Seven dimensions means that even if each dimension only has a resolution of 100 elements, we must keep track of 10^14 elements, or roughly one petabyte of data if each element is 10 bytes. In many cases, 100 elements is not nearly fine enough to resolve important features in a dimension, especially in frequency where many sharp features are present. The problem is already numerically impractical from a memory perspective.

As well, the transfer equation is an integro-differential equation, meaning that common numerical solvers are not useful. Solvers for this type of equation are generally complex and specific purpose, so the actual numerical method side is also difficult.

In order to overcome the above, different approximations to the equation are adopted. Different approximations give rise to different advantages and disadvantages in accuracy and speed and typically apply best to particular regimes.

Current popular strategies include monte-carlo, ray tracing, grid-based solvers, and moment methods. The following sections will give a brief description of each method as well as common properties of the methods.

2.2.1 Monte-Carlo Solvers

Monte-Carlo (MC) methods are perhaps the most obvious way to solve the radiative transfer problem. The most basic solution follows a photon from emission, through any scattering, absorption, and re-emission, until it leaves the simulation. At any point during the path, random numbers are used to determine whether the photon will be scattered, what direction it will be scattered, whether it will be absorbed and re-emitted, and what wavelength the re-emitted photon(s) will be.

In practice, following individual photons is not practical. Instead, following "photon packets" is more useful. Packets are typically defined as a group of photons [cite codes that do this] or as having a specified energy [codes that do this](in which case, the number of photons can be determined by using $E = h\nu$). The latter choice has the benefit that when re-emission occurs after an absorption at a lower wavelength, there are not now more photons to keep track of. The ray retains the same energy and more photons are implied since ν changes [Ercolano et al., 2003, Abbott and Lucy, 1985].

In order to determine when a photon packet will interact, most codes

use one of two methods. The first strategy is to use a probability distribution function (PDF) for optical depth. The PDF takes the form

$$P(l) = \frac{\int_0^{\tau(l)} e^{\tau} d\tau}{\int_0^{\infty} e^{\tau} d\tau} = 1 - e^{-\tau}, \tag{2.17}$$

where P(l) is the probability of an interaction happening at a distance l and $\tau(l)$ is the optical depth corresponding to the interaction. By inverting equation 2.17, one can use a random number for P(l) to determine the interaction optical depth (or distance). The photon packet is then assumed to interact at that position, and the process is repeated [Harries and Howarth, 1997].

Another strategy is to simply trace the photon packet from resolution element to resoultion element (cells, particles), and at each point, to use a random number to determine if the photon packet should interact at that cell. This has the advantage that the code does not need to calculate and normalize optical depths [Lucy, 1999, Ercolano et al., 2003].

The above process is repeated until a photon packet leaves the simulation volume, and for many photon packets. Once a large number of photon packets have been sent out, physical quantities must be estimated from observed MC quantities. In this case, a common physical quantity to determine is mean intensity J and the MC quantity is the photon packet. In order to relate the quantities, an *estimator* is needed. An obvious choice (though not necessarily the most optimal, Ercolano et al. [2003]) is to simply use the definition of intensity (equation 2.1),

$$\Delta E = I_{\nu}(r, \theta) \Delta A |\cos\theta| \Delta\nu \Delta\omega \Delta t, \tag{2.18}$$

Where ΔA is the reference surface, θ is the angle between the photon packet vector and surface normal vector, $\Delta\omega$ is the solid angle, $\Delta\nu$ is the frequency range, and Δt is the time interval. By combining with equation 2.8, one can obtain a mean intensity from a sum of photon packets [Ercolano et al., 2003],

$$J_\nu(r) = \frac{1}{4\pi}\frac{\Delta E}{\Delta t}\sum_i^{N_k}\frac{1}{\cos\theta}\frac{1}{\Delta A}\frac{1}{\Delta\nu}. \tag{2.19}$$

Once a mean intensity is found at a location, a solution for ionization can be iterated to by integrating out the ionization and heating terms. Note that it must be iterated since a chance in ionization and temperature may imply a change in local absorption properties.

The MC process is very accurate. It can deal with arbitrary spatial distributions, arbitrary scattering functions, polarization, and provides a natural way for "observing" a simulated object.

However, MC is *very* computationally costly. Large numbers of photon packets must be sent out and individually tracked in order to get a good estimate of the true mean intensity. While photons can be added to a packet along its path, new photon packets must be created for all sources [is this strictly true?] in order to ensure energy conservation (a star should radiate $L\Delta t$ energy in time Δt). This means as more sources are added, the computational cost rises quickly.

There is also an indirect cost associated with optical depth. In the case of very optically thick systems, interactions occur far more often between photon packets and the medium, meaning more computation is needed per photon packet. In the case of very optically thin systems, interactions are

very rare and very large numbers of photons must be cast to get accurate statistics on heating, ionization, and scattering. From a numerical perspective, MC provides poor error control. The typical solution to better accuracy is to simply increase the number of photons [check if other strategies?). However, this converges very slowly and often still does not guarantee good statistics on rare events such as low probability re-emission or scattering.

For the above reasons, MC radiative transfer is most commonly used as a post-processing technique for creating images of astrophysical objects. For more details on specific codes, please see Dullemond [2012], Cantalupo and Porciani [2011], Altay et al. [2008], Ercolano et al. [2003], Nenkova et al. [1999], Lucy [1999], Harries and Howarth [1997], among many others.

2.2.2 Ray Tracing

At the most basic level, Ray Tracing is a fairly natural way to go about solving the equations of radiative transfer and is probably the most popular method in astrophysics. Rays are cast from sources and the energy or photons contained in the ray are diminished as it passes through absorbing material [Razoumov and Scott, 1999, Abel et al., 1999b] [add more RT codes here]. This may sound familiar to section 2.2.1 because many Monte Carlo codes often use a ray tracing approach to follow their photon packets.

Most ray tracing codes tend to start by simplifying the radiative transport equation (2.13) by assuming the the emissivity of intervening material is 0. This leads to a much simplified version of the equation

$$\frac{dI_\nu}{d\tau_\nu} = -I_\nu, \tag{2.20}$$

which has the simple solution

$$I_\nu = I_\nu(0)e^{-\tau_\nu}. \tag{2.21}$$

It remains only for the ray tracer to calculate the optical depth between a source and another point. The simplest strategy to do this is to send rays out that intersect every cell in the simulation, and simply remove photons from the ray as they pass through each cell according to equations 2.14 and 2.21. This process is visualized in figure 2.2.

In principle, this process would solve the radiative transfer problem provided enough rays were cast. However, many practical problems arise that require special care. When casting rays, closer cells are intersected by far more rays than far away cells, as can be seen in figure 2.2, comparing cells a and b. This means redundant work is performed near the source in order to get sufficient resolution of intensity at large distances. Many codes have created adaptive techniques to reduce the number of rays that are needed. For example, Abel and Wandelt [2002] make use of the HEALPIX algorithm [Gorski et al., 1999] to determine rays that create an equal area per ray on a sphere. As the ray moves out and the ratio of the surface area of a cell to the solid angle of a ray decreases, the HEALPIX algorithm is recursively called on a single ray to subdivide it into four smaller rays to better sample further cells. This reduces the number of rays that need to be cast (see figure 2 in Abel and Wandelt [2002]).

A more difficult problem that arises is the computational cost associate with more sources. For most ray tracing codes, every additional source requires the whole tracing procedure to be performed again. While there are a few

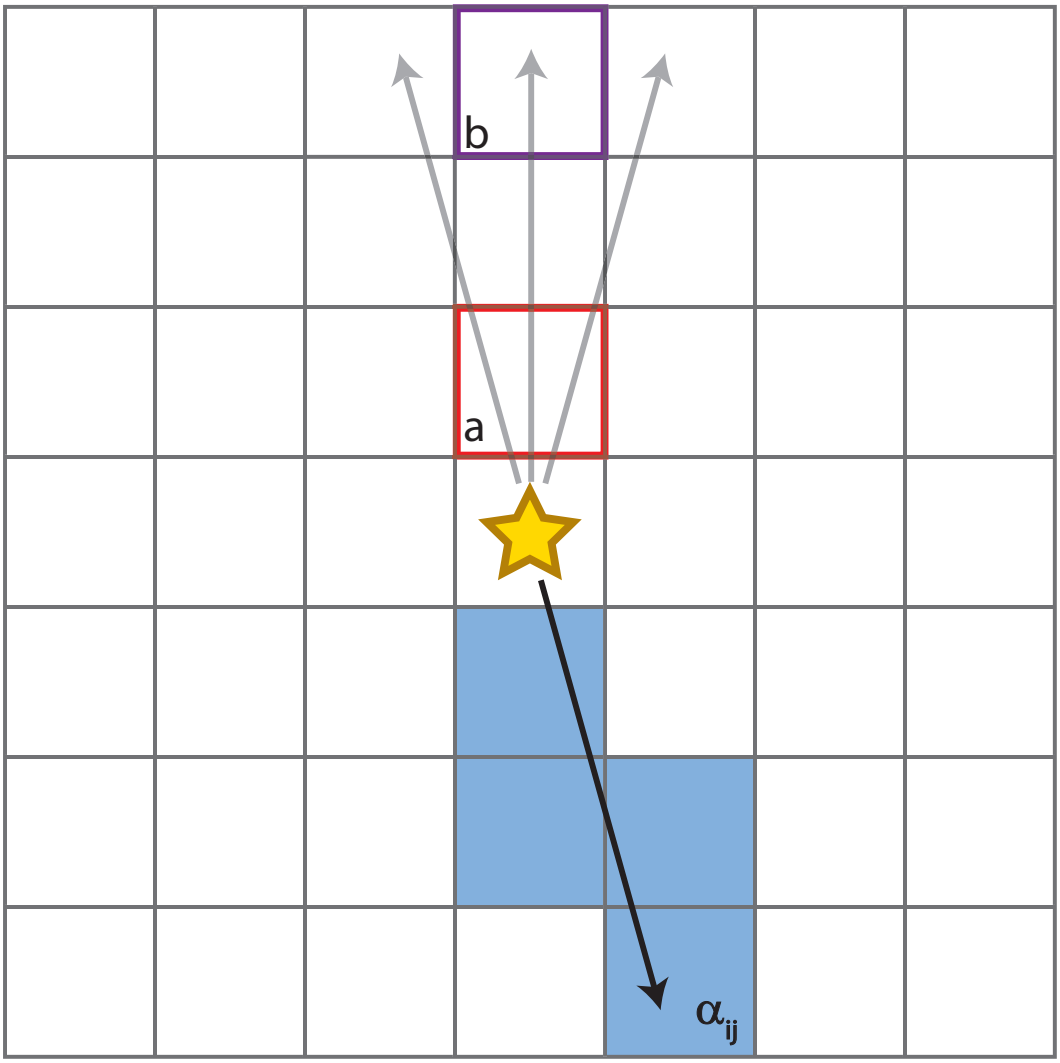


Figure 2.2: Rays being traced through a grid in a simulation. Each ray has an associated energy or number of photons that are removed from the ray as it traverses through each cell according to the properties of each cell. The blue highlighted cells show an example for one particular ray, where intensity is removed according to the absorption coefficient, $\alpha_{i,j}$, of each cell it passes through. Notice that closer cells, such as cell a, have more rays intersecting them than further cells, such as cell b, meaning redundant work is performed in the close cells.

codes that have attempted to remove this poor scaling [fill in authors], it is more often seen as a limitation of the method and so the method is not often applied to problems with large numbers of sources [I don't know if I like this sentence].

While the above description was based off of the initial assumption that scattering was zero, ray tracing codes do exist that attempt to model scattering in some way. Some authors have chosen to break the field into a direct component (due to ionizing sources) and a diffuse component (due to recombination in gas). The diffuse component can then be tracked and solved separately from the direct component. For example, Razoumov and Scott [1999] chooses to use an operator split explicit-implicit scheme to advect the radiation variable along the separate rays. Other authors, e.g. Abel et al. [1999b], use a similar approach advecting the diffuse radiation, but choose not to keep track of rays at this point.

The scenarios presented up until now have all focused on sending rays out from sources. URCHIN [Altay and Theuns, 2013] is a ray tracing code that has adopted the opposite strategy of sending rays out from sinks. While this may seem counter-intuitive at first, there are many computational advantages to doing this in particular physical scenarios. Altay and Theuns [2013] designed the algorithm to efficiently model the post-reionization Universe, where radiation is coming from all directions. In this case, tracing rays outward from sinks is guaranteed to find sources of radiation and alleviates any sampling issues associated with choosing sources to start tracing from.

Note that many of the algorithms mentioned above rely on tracing radiation outward along structured data. If the data is unstructured, it is far more difficult to properly estimate what portion of ray segments are affected by

absorbing media. As well, it becomes very difficult to ensure that sufficiently many rays are propagated to each resolution element. Unfortunately, this is exactly the scenario that is present in SPH simulations, where resolution elements are completely irregular.

?] have created a short-characteristics ray tracing scheme specifically designed to deal with the unstructured nature of SPH simulations called TRAPHIC. In this ray tracer, radiation is "traced" out in cones to neighboring SPH particles. However, since particles do no necessary exist in all directions for a given particle, virtual particles can be introduced to help propagate radiation through voids without particles. Note that by keeping a fixed solid angle for each particle, a natural adaptivity arises since the same solid angle on a particle further from a source will cover a smaller solid angle with respect to the source. TRAPHIC also introduces a method of merging sources that are close in angle for a receiving gas particles. Merging sources means that the algorithm can handle very large numbers of sources, making it one of the most currently powerful algorithms for cosmological simulations. [transition...]

The ray tracing method affords many advantages - it can handle arbitrary geometries and gives good error control, meaning very accurate results can be obtained. However, the method is typically limited to simulations that contain small numbers of sources due to poor scaling. As well, the method tends to have difficulty with very high optical depths and it usually becomes more practical to use a moment method (section 2.2.3) [why is this? is this because of time steps? Says more complicated solvers needed? Mention time steps...].

2.2.3 Moment Methods

[Add citations to OTVET, description of., davis et al? short vs long characteristic.]

Moment methods represent a large chunk of astrophysical radiative transfer codes currently available. Very broadly, these methods take moments of the radiative transfer equations and make simplifications to make the equations easier to solve by common techniques.

Specifically, we can start by taking angular moments of the radiative transfer equation (equation 2.15?). If this is done in a frame comoving with the radiating fluid and local thermodynamic equilibrium is assumed, we get (to first order in v/c) [Mihalas and Mihalas, 1984]

$$\frac{D\rho}{Dt} + \rho\nabla\cdot\mathbf{v} = 0, \tag{2.22}$$

$$\rho\frac{D\mathbf{v}}{Dt} = -\nabla P + \frac{1}{c}\chi_F\mathbf{F}, \tag{2.23}$$

$$\rho\frac{D}{Dt}\left(\frac{E}{\rho}\right) = -\nabla\cdot\mathbf{F} - \nabla\mathbf{v} : \mathbf{P} + 4\pi\kappa_p B - c\kappa_E E, \tag{2.24}$$

$$\rho\frac{D}{Dt}\left(\frac{e}{\rho}\right) = -P\nabla\cdot\mathbf{v} - 4\pi\kappa_P B + c\kappa_E E, \tag{2.25}$$

$$\frac{\rho}{c^2}\frac{D}{Dt}\left(\frac{\mathbf{F}}{\rho}\right) = -\nabla\cdot\mathbf{P} - \frac{1}{c}\chi_F\mathbf{F}. \tag{2.26}$$

In the above equations, D/Dt is the convective derivative, defined as $D/Dt \equiv \partial/\partial t + \mathbf{v}\cdot\nabla$. The quantities ρ, e, $\mathbf{v}$, and p are the mass density, energy density, velocity, and scalar isotropic pressure, respectively. E, $\mathbf{F}$, and $\mathbf{P}$ are the frequency-integrated radiation energy density, momentum density or flux, and pressure tensor, respectively. E, $\mathbf{F}$, and $\mathbf{P}$ are the zeroth, first,

and second order angular moments of intensity (equation 2.1),

$$A^n(\mathbf{x},t) = \oint I \cos^n\theta d\Omega. \tag{2.27}$$

χ_F is the flux mean total opacity, κ_P is the planck mean absorption opacity, and κ_E is the energy mean absorption opacity. The first opacity represents an effective opacity with contributions from absorption and scattering across all wavelengths. [The second and third are absorption opacities....?]. Finally, c is the speed of light. [What is B? Planck? What is : in math equation?]

One popular moment method is Flux Limited Diffusion (FLD) [Alme and Wilson, 1974, Levermore and Pomraning, 1981, Pomraning, 1983, Melia and Zylstra, 1991, Anile and Romano, 1992]. In FLD, the assumption is made that intensity is a slowly varying function of space and time. This is certainly true in the limit of very high or very low optical depth. It is the intermediate region ($\tau \approx 1$) where this assumption may not be true. This assumption allows the radiative flux to be written in the form of Fick's Law of diffusion [Levermore and Pomraning, 1981],

$$\mathbf{F} = -D\nabla E, \tag{2.28}$$

where D is the diffusion coefficient, given by

$$D = \frac{c\lambda}{\chi}. \tag{2.29}$$

λ is a dimensionless function of energy called the flux limiter.

In order to solve equations 2.22-2.26, the system must be closed by

relating the moments of radiation. An obvious first choice is the Eddington Approximation, which assumes the field is isotropic and implies that

$$\mathbf{P} = \frac{1}{3}E. \tag{2.30}$$

Using this approximation, equation 2.26 becomes

$$\mathbf{F} = -\frac{c}{3\chi}\nabla E, \tag{2.31}$$

which is correct in the optically thick limit, but gives an infinite speed of light in optically thin regions. Thus, the flux limiter in equation 2.29 functions to allow the Eddington Approximation to be made by creating physical behavior in the optically thin regime by limiting the radiation propagation speed [reword].

By combining equations 2.28 and a relationship between moments of intensity, equations 2.22-2.25 can be solved numerically [Turner and Stone, 2001].[Should I go into the actual solving process? I don't think so...]

Other common moment methods follow a similar path, but use different assumptions for the closure relation. Another popular choice is called the "M1 closure relation" [Levermore, 1984, Skinner and Ostriker, 2013][add more codes here]. This assumes that intensity is rotationally invariant about the direction of radiative flux, rather than fully isotropic. This assumption allows better results in particular scenarios (e.g. shadowing behind dense objects [Skinner and Ostriker, 2013]) and a more efficient numerical solution [González et al., 2007, Aubert and Teyssier, 2008]. However, its applications are limited as it cannot deal with complex radiation fields from distributed sources since it uses entirely local information to get the closure relation.

The majority of moment methods tend to have a difficult time dealing with complex source distributions because they adopt a closure relation that only accounts for local characteristics of the radiation field. One moment code (among others) that has managed to get around this is OTVET [Gnedin and Abel, 2001]. OTVET explicitly constructs the radiation pressure tensor from the source distribution in the simulation, removing the need to assume a relationship between moments. This enables a combination of contributions to the intensity from all sources.

Overall, moment methods provide a useful and efficient way to solve problems that consist of high optical depth, where long characteristics are not important, or low optical depth with simple source distributions, so the intensity field is simple. Note that the method also tends to be very diffusive in many cases, and so special care must be taken to avoid this behavior if it is not desired.

2.2.4 Other Methods

Sections 2.2.1-2.2.3 cover some of the most common radiative transfer methods currently used in astrophysics. However, it is worth mentioning a few other methods that aren't as easily grouped into the above categories.

In order to overcome some of the shortcomings of moment methods and ray tracing methods, some authors have created hybrid codes that use both methods in different regimes in the simulation. Kuiper et al. [2010] has created one such scheme. The basic idea is to attempt to use each method in the regime where it's most advantageous to save on computation and improve accuracy. In the case of Kuiper et al. [2010], the algorithm is designed to approach the

problem of massive star formation. It uses a first order ray tracer to transfer stellar photons at higher frequencies to the gas. A secondary moment method (FLD) then diffuses the photons through high density gas. This is meant to efficiently model transfer of high frequency photons from a massive star and reprocessing of those photons to lower frequency emission.

The method avoids the difficulties that ray tracing can have in high optical depth regions and benefits from the speed and accuracy of FLD in appropriate physical regions. While the hybrid code of Kuiper et al. [2010] was specialized to do simulations of a single massive star, the code has since been extended to work on an arbitrary number of sources [Klassen et al., 2014].

Another strategy is to make use of the Fast Fourier Transform (FFT) techniques. Cen [2002] makes use of the property that if the sum of a quantity over a volume can be written in the standard convolution form, than it can be solved using an FFT. By re-writing equations **??** and **??** in this form, one can solve the equations in $N \log(N)$ time for each direction. Using trees to discretize the angles, you end up with roughly $\log(N)$ angles to solve for, and so the solution scales as $N \log^2(N)$. [More here.... I don't get this method.]

Finally, Clark et al. [2012] have created an algorithm most similar to ray tracers, but whose purpose is to calculate column depths for exterior sources [check if this is right]. The basic idea is to map the simulation volume onto a sphere that has been divided into equal area segments by the HEALPIX algorithm [**?**]. By performing this action during the tree walk, columns can be calculated to any point within the simulation at a cost of $N \log(N)$. However, since this algorithm only calculates column depth as a function of θ, ϕ for each particle, it is limited to cases in which sources are located outside of all absorbing material.

2.3 Summary of Methods

Section 2.2 gave an overview of some of the most common methods currently used in astrophysics, so we are now well posed to assess the field of computational radiative transfer.

In order to solve the equations of radiative transfer, codes must make certain approximations. In all but MC codes, authors typically start by dropping scattering. Moment methods typically make a further assumption about the relationship between the moments of radiation, with the most common being the Eddington Approximation (eqation 2.30), or the assumption of radiation isotropy.

The above assumptions are still often not sufficient to make the problem computationally viable. Ray tracing has needed to make algorithm-specific improvements to become more computationally efficient. Making rays adaptive as they get further from the source reduces redundant work near the source, and in some scenarios, tracing the rays backwards from the sinks can provide another efficiency boost. Some codes, both ray tracing and moment methods, have also made the improvement of merging sources, which can reduce the number of sources to do calculations for from N to $\log(N)$.

Very roughly, the current code base occupies particular spaces in the computational radiative transfer volume. Monte carlo codes are typically the most accurate, but also the most computationally expensive. For this reason, they are usually limited to post processing and image creation in simulations. Ray tracers are the most popular, and offer very good accuracy, but typically scale quite poorly with the number of sources, and so are usually limited to simulations containing small numbers of sources. Moment codes are a step

up in computational speed, but are usually only appropriate in simulations that have high optical depth where the assumption of radiation isotropy is appropriate. Otherwise, unwanted diffusion in the radiation field can give large errors. Despite the simplifying assumptions, many moment codes are still the dominant computational cost in simulations compared to other common physics solvers such as gravity and hydrodynamics.

Currently, there is a gap in the market for solvers that can deal with large numbers of sources over a range of optical depths without any diffusive assumptions. OTVET [Gnedin and Abel, 2001] is close to this regime, and perhaps the most widely recognized tool for cosmological simulations at the time, but is still a diffusive code [?]. TRAPHIC is currently the only code that satisfies the above criteria, and has recently started to take impressive steps forward on the computational cosmology front. TRAPHIC is SPH-specific, and so the algorithm is limited in this way. However, it is probably the most appropriate tool at the time for cosmological radiative transfer.

The goal of this thesis is to present a new algorithm for solving radiative transfer that is capable of dealing with large numbers of sources, non-diffusive, at a similar computational cost to gravity. In order to achieve these goals, we do not stress achieving exact solutions, but aim for correct qualitative behavior and correct equilibrium solutions.

2.4 TO-DO (REMOVE THIS)

- RAMSES-RT [Rosdahl and Teyssier, 2015, Rosdahl et al., 2013]
- Mention error convergence

Chapter 3

The Numerical Method

The main goal of the algorithm presented in this chapter is to solve the radiative transfer equation. We prioritize the ability to deal with a large number of sources and computational speed over high accuracy, though we still insist equilibrium behavior be correct.

In order to accomplish this, we first start by dropping scattering to simplify the equations of RT. We also note that by summing radiation over all sources, we need not explicitly include emission in the equation. This means equation 2.11 simplifies down to equation 2.5, which has the solution

$$I_\nu(\mathbf{r}) = I_{\nu,0} e^{-\tau}, \tag{3.1}$$

where $I_{\nu,0}$ is the intensity from a particular source.

In the absence of absorbing material, the optical depth is 0 and equation 3.1 implies that we need only sum over all sources. This problem is almost identical to gravity, and so we choose to use the same tree-based technique as gravity to solve it.

The tree-based gravity solver of Barnes and Hut [1986] has become

commonplace in astrophysical simulations [Wadsley et al., 2004][cite codes that use it]. They scale well with the number of resolution elements ($N \log(N)$) and are easily tuned to the desired accuracy. It is also easily made parallel. For these reasons, we choose to implement a tree to perform radiative transfer.

Please note that following algorithm has been implemented in SPH. As such, we choose to refer to resolution elements as particles, as is common in SPH codes. However, the algorithm is not specific to SPH, and "particle" can be replaced with whatever represents the resolution element in another code.

3.1 Tree Data Structures

In order to understand the radiative transfer algorithm that we are presenting, it is important to understand tree data structures.

In computer science, a tree is any data structure that stores data in a hierarchical way. Typically, the data is stored in a "node" or "cell." Each node then has 1 or more "child nodes" that branch off from it, and each of those child nodes in turn has child nodes, and so on. In order to store simulation data in a tree, we simply need to partition the data in a hierarchical way. The easiest way to do this is to partition the simulation volume itself.

The entire simulation volume is stored in the "root node", or the top of the tree. We then partition the root node into a smaller unit. A common choice is an octree, which splits a cube into eight sub-cubes, or a binary tree, which splits a volume into two smaller volumes. In the former case, every node has eight children, while in the latter case, each node has two children. Note that all children with a common parent are referred to as siblings. While the algorithm we have developed is independent of tree-type, we will introduce it

in the context of a binary tree, since this is what GASOLINE uses.

Starting with the root node, the volume is split into two smaller volumes. In memory, particles are partitioned about the split value, which is an $\mathcal{O}(N)$ operation. The tree-build continues recursively, splitting each child node into smaller volumes. In a binary tree, it is common to choose your split value as either the midpoint of the longest axis in the volume, or the midpoint that gives an equal number of particles on either side. Both of these options are available to our RT algorithm. The splitting continues until a condition is met. Typically, this condition is that a volume contains less than a specified number of particles in it. A cell that meets this criteria will be at the bottom of the tree, and is called a "leaf" or "bucket."

During this process, a node can request average properties from the nodes beneath it, such as total luminosity, center of luminosity, and so on. Once the tree has been partitioned down to leaves, the leaves can calculate the requested properties and pass them to their parents. Parents then calculate their average properties from their children and return to their parents, all the way up to the root of the tree. A graphical representation of a tree is shown in figure 3.1. Figure **??** shows the tree in real space, while figure **??** shows how it might look in memory.

Note that average properties of interest for radiation are total luminosity, center of luminosity, average density, average opacity, and the variance in opacity. The reasons for these properties will be discussed in section 3.3.

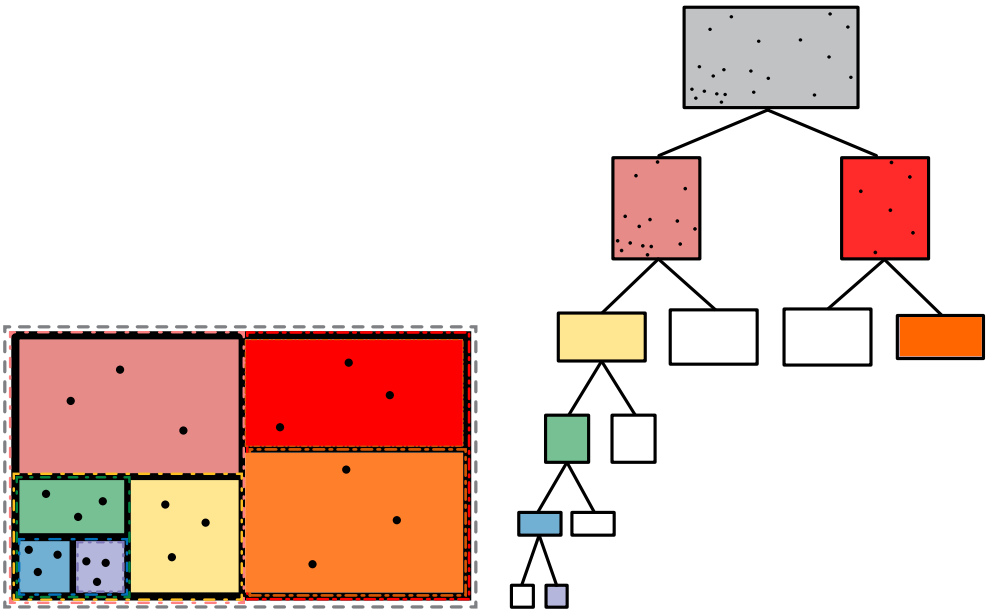


Figure 3.1: This is an example of a binary tree. The volume is represented by a tree node, and each volume is then split into two subvolumes, which are represented by two "child" nodes of the original node. This splitting can continue indefinitely on either side, making the tree an effective way at splitting volumes.

3.2 Exchanging Radiation

Once the tree has been built, calculating the radiation (gravity) at any particular point can be accomplished by traversing the tree structure, a process called a "tree walk." First, a "post-order" tree walk is performed in which the children of a node are always checked before its sibling. The walk continues until it arrives at a leaf node, at which point the radiation (gravity) arriving at that leaf is calculated. This leaf node will be called the receiving leaf.

A second tree walk occurs during the radiation (gravity) calculation. We must check what cells are acceptable to interact with based on a particular criteria. Gravity calculations use what is called an opening angle criteria. The idea is that for any cell, if the cell takes up a sufficiently small solid angle on

the sky, then the entire contents of the cell can be approximated as a single object located at the center of mass of the cell. In order to determine this, the simplest criteria to check is whether

$$\frac{b_{\max}}{r} < \theta_{\text{crit}}, \tag{3.2}$$

where θ_{crit} is a user set parameter, $b_{\max}$ is the largest extent of the cell, and r is the radius from the receiving cell to the cell in question. If a cell does not satisfy this criteria, it must then examine each child of the cell. If it does satisfy the criteria, then it can interact with that cell and move on to checking the next one. If a cell fails this criteria, but is a leaf node and cannot go down any further, then all particles within the lead node are interacted with individually. Note that in practice, it is more efficient to rewrite equation 3.2 in terms of radius,

$$r_{\text{crit}} = \frac{b_{\max}}{\theta_{\text{crit}}}. \tag{3.3}$$

This process is illustrated graphically in figure 3.2. In this figure, Cell A is the receiving cell, and cells B, C, and D are cells to interact with. In this case, cell B fails the criteria, but cannot be opened any further and so the particles inside of B are interacted with individually. Cell C fails the criteria as well, but since it is not a leaf, each of its children are checked. Cell D passes the criteria, so the interaction is done with the center of luminosity of the cell. The interaction is depicted in figure 3.3.

Once radiation has been calculated for the receiving leaf, we move on to the next leaf, which is accomplished by moving to the sibling if the current leaf is the left child of the parent node, or to the sibling of the parent node if

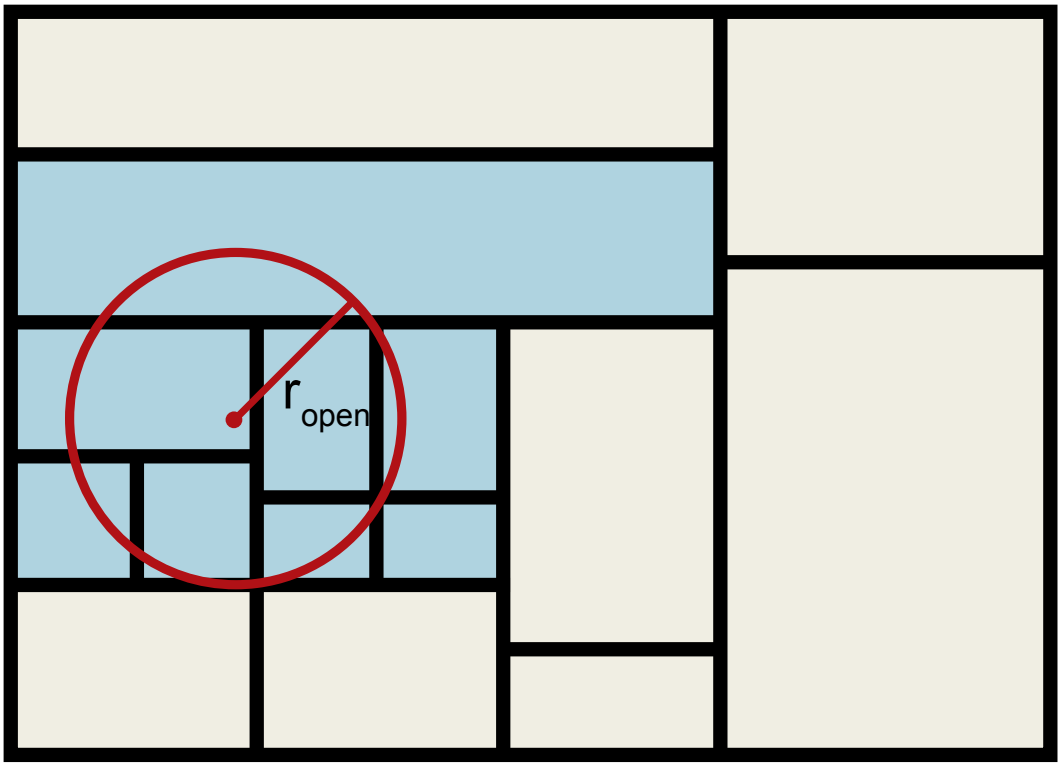


Figure 3.2: Cell A in this image is the receiving cell, while cells B, C, and D are cells that A will receive flux from. Cell B is close enough so that it should be opened, but is a leaf and so it requires a direct n^2 summation. Cell C is close enough and is not a leaf, so it will have its two children checked for the same criteria (the left child will be too close, the right child will be acceptable to interact with). Cell D is not a leaf, but is sufficiently far away that leaf A can interact with the full cell.

we are the right child. An example radiation exchange is shown in figure 3.3.

The above algorithm will run in $N \log N$ time, as with gravity. However, unlike gravity, not all objects emit radiation. Thus, technically the more specific scaling is $N_{\rm sink} \log N_{\rm source}$. The slow growth rate of computation time with the number of sources makes the algorithm a very strong candidate for cosmological applications in which there are often similar numbers of star particles to gas particles. As was mentioned in chapter 2, some codes have already made use of this basic idea [Gnedin and Abel, 2001, **?**, Kannan et al.,

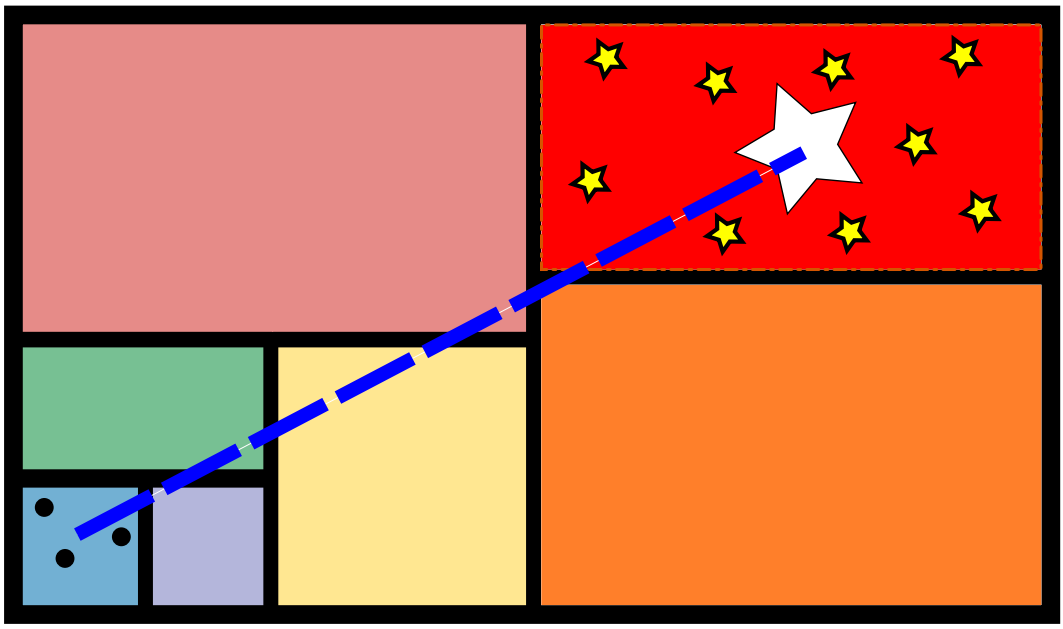


Figure 3.3: In this image, cell A is receiving radiation from cell B. Cell B is sufficiently far away that we can find the center of luminosities of all the sources inside of it, and calculate flux based on that single value rather than summing each one individually.

2014].

3.3 Absorption

The algorithm presented in sections **??** and 3.2 assumes that no change to the radiation happens in between the sending and receiving cells. In gravity, this is acceptable because forces are not "absorbed" in any way. However, radiation tends to be absorbed and scattered by intervening material and thus the intensity of the radiation at a point is not only due to the sending source, but to all material in between the source and the sink. As was mentioned in the introduction of this chapter, we choose to omit scattering and focus only on absorption. The goal, then, is to find the optical depth between two

interacting cells without adding significant computational cost. In order to accomplish this, we have developed the algorithm to make use of the tree during the optical depth calculation as well.

The crucial point to the algorithm lies in the fact that for any two interacting cells, there exists a common parent node above them. Thus, all intervening space between the cells must lie within the subtree in which the common parent is the root [should add figure to show this]. If we traverse up the depth of the tree (hereafter referred to as a tree climb) from each interacting node to the common parent node, we will have performed roughly $\log(N)$ extra operations per interaction on average. If we do no other work than this, then our scaling for radiative transfer changes to $N_{\rm sink} \log N_{\rm source} \log N$. While the extra factor of $\log N$ is certainly worth noting, it does not tend to increase scaling by a significant amount. Our goal then becomes to perform (∞) amount of work during this additional tree climb.

As was mentioned in section 3.1, the tree records average properties as it is built, including average opacity and density. Referring to the definition of optical depth (equation 2.14), we see that we can get an estimate of the optical depth through a cell by using the average opacity, the average density, and finding the segment of the ray inside the cell,

$$\tau_i = \bar{\rho}\bar{\kappa}ds, \tag{3.4}$$

where τ_i is the optical depth in cell i, $\bar{\rho}$ is the average density in the cell, $\bar{\kappa}$ is the average opacity, and ds is the length of the ray segment contained in the cell.

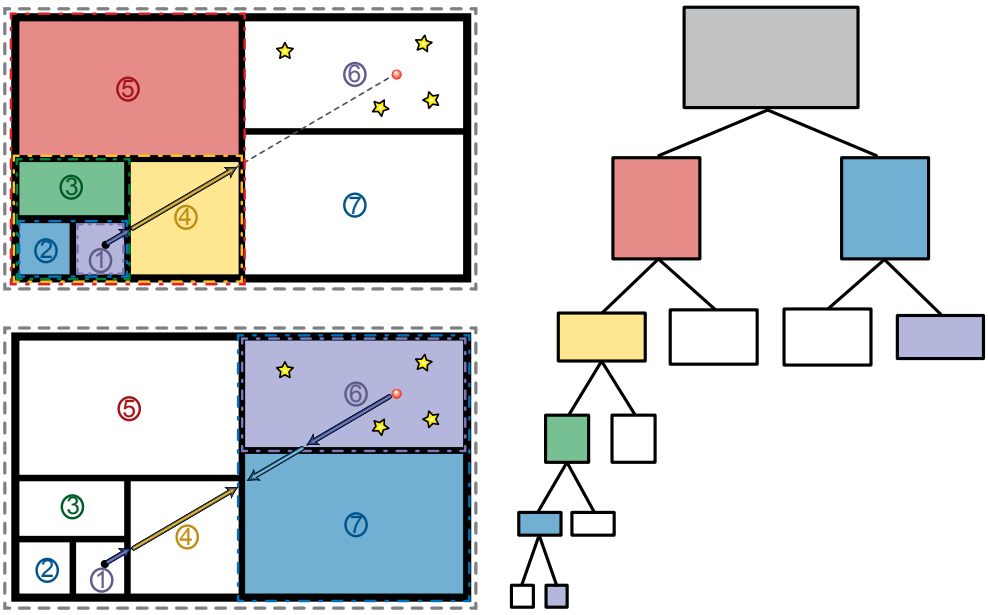
At each higher cell during the tree climb, we obtain a larger represen-

tative volume from that cell. The new volume contains the previous volume as well as a new contribution from the previous cell's sibling. This sibling's volume may or may not lie on the vector connecting the two interacting cells. This can be determined by calculating the distance to the edge of the current volume along the vector from the centers of the original interacting cells, an operation that takes (∞) time.

At each new parent cell, if the calculated line segment is longer than the accumulated distance so far, than the difference is the amount of the ray contained in the sibling cell. By recording this new line segment, the average density of the cell, and the average opacity of the cell, we have everything needed to calculate the optical depth of the line segment. By summing the optical depth of each line segment, we will have obtained the full optical depth between the interacting cells in order $logN$ time, giving a full scaling of $\mathcal{O}(N_{sink} \log(N_{source}) \log(N))$. The algorithm is depicted graphically in figure 3.4

3.4 Refinement

While section 3.3 introduces a very fast algorithm for calculating a radiation field, it relies heavily on the geometry of the underlying tree. In volumes with very smooth density and opacity, the above algorithm performs very well. However, in cases with sharp density or opacity gradients, the gradient is discretized into widths of order the cell size at the current tree depth. This can become problematic, causing the tree structure to be imposed into the calculated radiation field. In order to solve this, we introduce a refinement process to the algorithm that allows a descent back down the tree during the



(a) A representation of the algorithm in real space, with the tree overlaid.

(b) The colors correspond to the volumes on the left.

Figure 3.4: The absorption algorithm.

tree climb in order to obtain a more detailed description of the medium.

- The refinement is a fairly straightforward addition to the algorithm. At the point where the average properties of the cell would normally be considered, we simply check if the current cell passes a refinement criteria.

- If the cell passes the criteria to refine, rather than recording the average properties, we recursively check the children of section of the tree we did *not* ascend from.

- Once we arrive at a cell that fails the criteria to refine (or at a leaf and can no longer refine), we record the line segment within the cell and the average properties as normal, and return up the recursive call. See figure 3.5 for a visual representation.

- The specific refinement criteria has deliberately been left vague until this point. In principle, one can refine on any cell property desired.

- For the purposes of this paper, we have decided to use an opacity refinement criteria. Within any cell, if a constant times the standard deviation of the average opacity is larger than the average opacity, the cell is refined. We find this produces a reasonable amount of refinement in code tests.

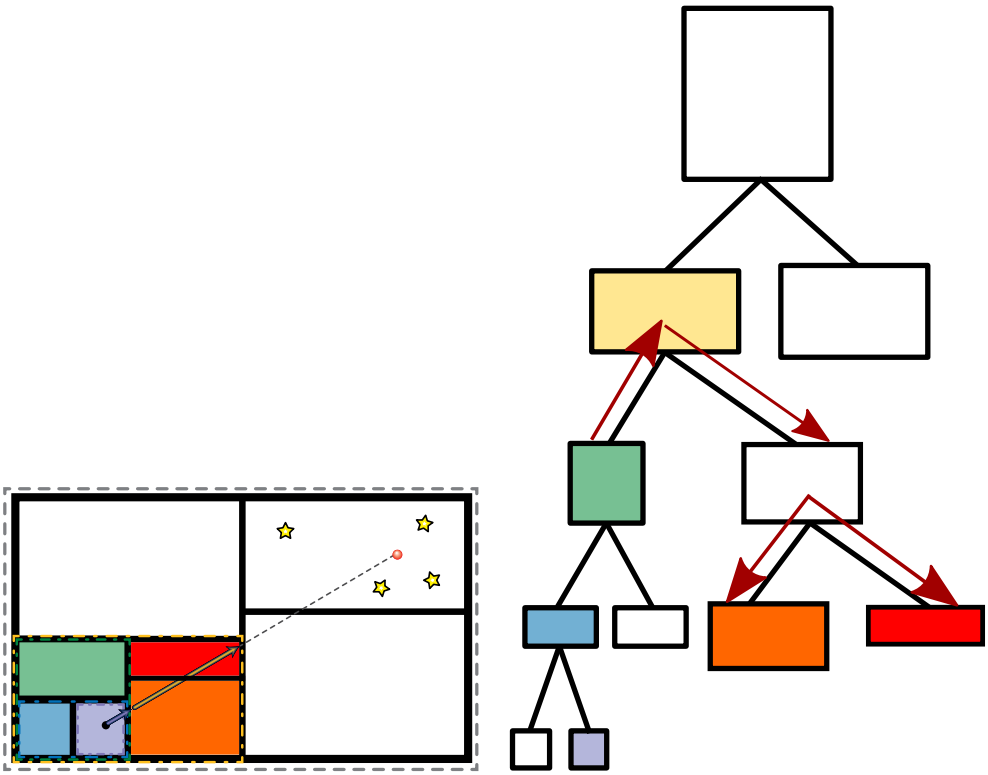
- Note that this is not necessarily the ideal criteria for physical simulations. It would be wise not only to look at the variation in opacity, but also the absolute value. In cases where the optical depth is very high, most of the radiation will be absorbed anyway, and the algorithm can be terminated since this particular vector yields a negligible flux of photons to the receiving cell.

Extension of the refinement to ray-tracing

- If very high accuracy is required, the refinement routine is flexible enough that sub-leaf refinement is possible. While this has not currently been tested since it leaves the regime of low computational expense, it could easily be implemented.

- If a leaf was reached during refinement and still passed the criteria to be refined on, the individual particles inside the cell could be considered.

- A ray tracing scheme through the cell similar to SPHRay [Altay et al., 2008] could be performed. The machinery to do this ray trace is already established for use within the receiving and sending cells (see section 3.5 and figure 3.6).



(a) The tree splits the line segment into the red and orange sections.

(b) The colors correspond to the volumes on the left.

Figure 3.5: When the line segment is too rough in some physical sense, refinement can be triggered. Visually, the algorithm descends back down the tree the opposite direction it came from until the criteria to refine is no longer satisfied or until a leaf is reached.

3.5 Resolving the Receiving Cells

During testing, we ran into issues with ionization fronts "stalling" in certain cells. If a sharp ionization front is passing through a receiving bucket, then the effects of averaging can cause issues if the optical depth of the bucket is of order unity or higher. (below section needs re-wording and more specifics).

- Consider an ionization front that has passed halfway through a leaf node (half of the particles are ionizaed, half are not).

- The average opacity will be $\kappa/2$, where κ is the opacity of the unionized particles.
- The ionized particles will use an opacity that is much too large, therefore reducing the flux that particles at the "rear" of the leaf see.
- This means that particles at the rear of the leaf are harder to ionize than at the front, and the propagation speed of the front is drastically reduced.
- In order to combat this, more detailed tracing is required *only in the receiving leaf.*
- This is easily accomplished by implementing a scheme similar to SPHRay [Altay et al., 2008]. (introduce simpler method that we use where we order particles along vector and linearly add optical depth?). See figure 3.6.

Introducing the ray tracing machinery for the above purpose also creates the ability to ray trace within leaves during the refine mentioned in section 3.4. In principle, this means the code can easily be forced into a full ray trace if this behavior is desired.

3.5.1 High Optical Depth Particles

High tau particles are problematic. This is how we deal with them...

3.6 Periodicity

Move to future work?

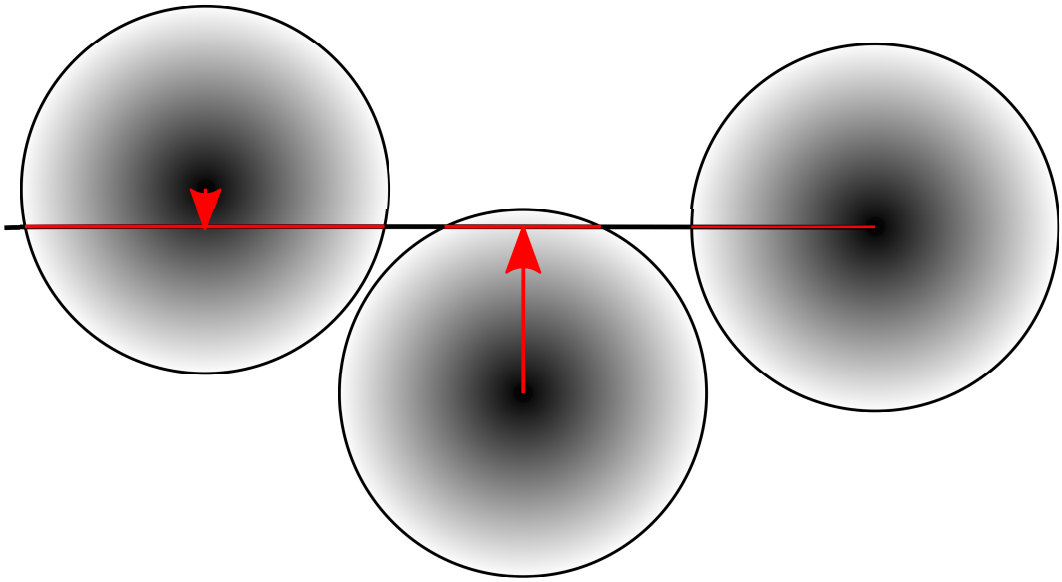


Figure 3.6: Two ray tracing schemes. The first is the scheme of Altay et al. [2008] in which the photons are diminished by optical depth along each particle's smoothing length that they pass through. The second scheme is much simpler and yields very similar results. It relies on the fact that the ray actually represents a very large cone of photons, and that most particles in the cell will probably contribute to the absorption of photons at the receiving particle. It essentially simplifies to setting b = 0 for every particle in the receiving leaf.

3.7 Cosmological Background Radiation

In order to do cosmological simulations properly, we must account for the radiation coming from the rest of the universe outside of the simulation volume. Most current codes apply a constant UV field to the entire box, essentially the lowest order approximation possible.

- Few codes have moved past this for cosmology, though some have.
- OTVET [Petkova and Springel, 2009] uses their radiation scheme with periodic boundaries.
- Other more detailed scheme apply a constant radiation field at the boundaries coming into the volume (add citations).

- While our scheme is perfectly capable of doing a periodic option, we have opted to set up a number of "background sources." The sources are distributed on the surface of a sphere at the very edge of the simulation (or larger if required) and the number of sources can be varied to match the required angular resolution of the background.

- Finding the flux at the center of a sphere of sources is a problem akin to Newton's Shell Theorem. However, because the flux doesn't cancel like force, the solution does not work out the same. See equation 3.5.

- Since the solution is logarithmic in r, the we can assume the flux is roughly constant at similar radii. Since most cosmological zoom simulations only consider gas at a fairly small radius, this is acceptable.

$$F = K\left[\log R + r - ln(R - r)\right] \tag{3.5}$$

3.8 Cosmological Effects

Move this section to "future plans" section?

3.8.1 Cosmological Redshift of Radiation

3.8.2 Accounting for a Finite Speed of Light

Move this section to "'future plans" section?

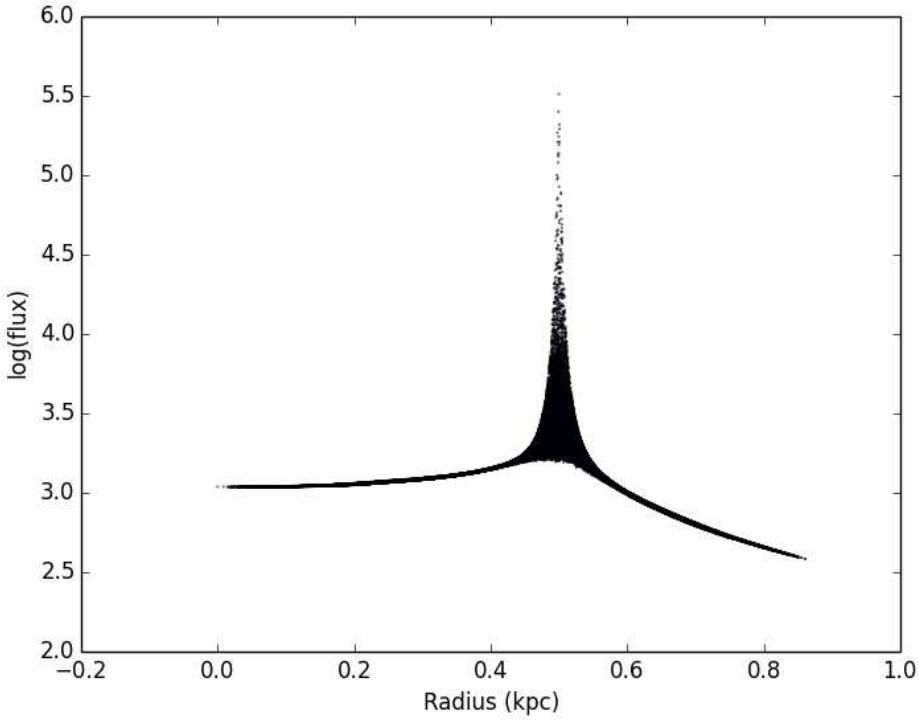


Figure 3.7: The distribution of flux particles receive due to cosmological background particles when distributed in a sphere at the edge of the box. Note that value of the flux at the center can be easily scaled by simply scaling the luminosity of all sources on the sphere.

3.9 Summary of Algorithm

We have presented a flexible and computationally inexpensive algorithm for calculating the radiation field within a simulation. The algorithm affords many benefits (note: need to introduce many of stated benefits below in previous sections):

- It is flexible enough to allow a wide range of accuracy depending on the application. Speed starts at $N \log N \log N$ and approaches that of ray tracing (check this...) when the algorithm is tuned to that level of refinement.

- Because radiation is transferred instantaneously, the speed of light does not become a limiting time step. If ionization dynamics are important, then the propagation of the ionization front becomes the limiting time step. If only end behavior is required, then the there is very little the algorithm does to limit the time step.

- There is no scan dependence. Because flux is accumulated at each receiving bucket without explicitly depositing the photons into the intervening material, ionization/heating/cooling is performed completely separate to radiation. This means that the solution will not change based on the order in which the sources are visited.

- The algorithm is independent of wavelength or even number of wavelengths. The algorithm need only perform the tree walk and tree climb a single time in order to obtain the line segments in each cell. Performing different wavebands simply equates to recording multiple average opacities. This enables multi-band radiative transfer at little additional cost.

However, it is important to keep in mind the limitations and assumptions of this algorithm.

- Photons are not explicitly conserved. In order to save computational time, we can not keep track of the photons deposited in intervening material during an exchange. We obtain an optical depth and simply assume that the photons lost in the process have been deposited in the intervening material. When the intervening material is the receiving bucket at a later point in the algorithm, it should receive roughly the correct number of photons due to a matching initial segment (wording...).

- Light is transferred instantaneously, meaning that photon fronts could travel faster than allowed, and that sinks could receive photons from a source too far away to have sent photons there yet.

- Very large opacities in single particles can be problematic for both cooling (calculating the emitted flux from a particle is complex if the particle itself is optically thick) and ionization propagation. Particle self-absorption can impart the same "stall" in a single particle that was mentioned in section 3.5 for leaves.

- Extra computation time can be required in the heating and cooling code due to intense local radiation fields. However, if the goal is to obtain a radiation field, this is already a built in cost to any algorithm. We simply mention it to suggest that increased computation time is due not only to the radiation algorithm, but the increased computation time for the cooling integrations (remove this point?).

Chapter 4

Code Tests

In this chapter, I present a variety of tests to demonstrate the strengths and limitations of the above algorithm. Many test cases have been drawn from previous RT papers including Iliev et al. [2006], Gendelev and Krumholz [2012], Skinner and Ostriker [2013]. This chapter also include tests of accuracy and scaling of the algorithm.

4.1 Glass

A glass of particles is a simple way to demonstrate the most basic functionality of the algorithm. Each particle effectively acts to sample the radiation field at a particular point and has an easily calculated exact solution to compare to. A glass is also the relaxed state for an SPH simulation.

In the optically thin case, we simply want to ensure that we obtain a $1/r^2$ dropoff with flux. There should be no errors present in this test case because in the case of a single source, no averaging is needed and both exact luminosity and position of the source is used for every sink.

$$F = \frac{L}{4\pi r^2} \tag{4.1}$$

In the case of absorption, the glass of particles has a roughly homogeneous density and thus the flux is still easily calculated for comparison.

- The equation for flux in this case is still fairly simple. If we refer to equation **??**, we can see the theoretical flux is simply equation 4.1 multiplied by the exponential of optical depth. See equation 4.2.
- This assumes a homogeneous density field. The glass does not have an exactly homogeneous field, but has little variance from the average.
- Figure 4.1 shows the error distribution of the particles.
- Note that in the case that the density field is exact, the errors reduce down to machine precision. This emphasizes the importance of accurately modeling the density distribution.

$$F = \frac{L}{4\pi r^2} \exp -\tau \tag{4.2}$$

4.2 Multi-Source Glass

We now show the effect of including many sources. The code is now performing at its most "stressed;" having a large number of randomly distributed sources means the code will run at its slowest and will include a large amount of averaging (not quite right).

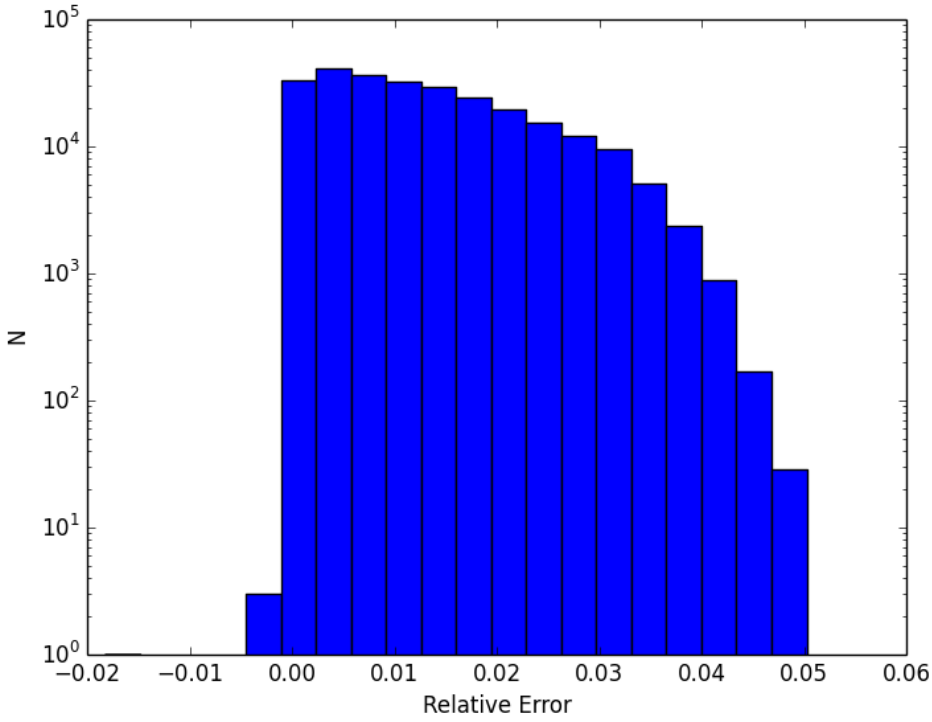


Figure 4.1: The distribution of flux errors among particles.

- We first present the optically thin case. In this case, the glass has had half of its gas particles replaced with sources so that there are an equal number of each.
- The error distribution present in this case demonstrates the errors associated with the tree method. [make mean error vs theta plot?]
-

4.3 Effects of Averaging the Source

We now look closely at what effects averaging sources can have on results. To demonstrate this, we have created a box of gas with two stars placed offset

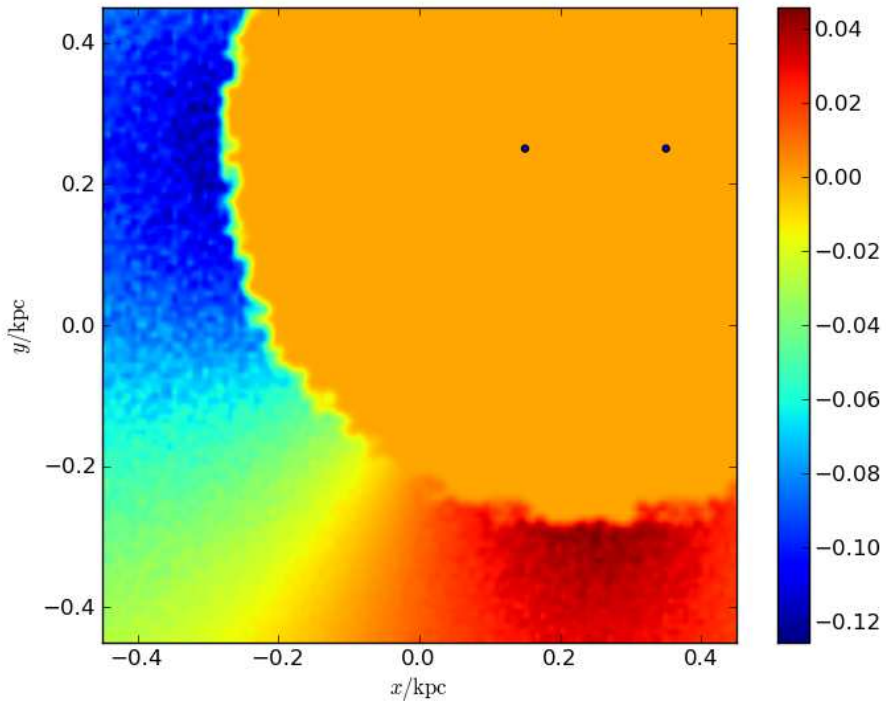


Figure 4.2: The error associated with averaging star positions.

from each other.

4.4 The Strömgren Sphere

The strömgren sphere is a theoretical ionized sphere of gas. It was first discussed by Bengt Strömgren in 1938 [Strömgren, 1939]. We start with a cloud of homogeneous neutral Hydrogen gas and an ionizing source, commonly representing an O or B-type star, at the center. As the photons from the source ionize the hydrogen, the optical depth of the gas decreases, and so the ionizing photons move further out creating an ionization front. As the front moves out, the photon density as a function of radius falls off simply due to $1/r^2$ geometry and eventually a point is reached where the ionization rate falls to

the recombination rate. At this point, the front stops in equilibrium.

4.4.1 The Isothermal Case

In the simplest case, the ionizing source is assumed to emit photons at exactly 13.7 ev, meaning that the hydrogen gas is ionized but not heated. Cooling is also disabled, meaning that the gas is isothermal. If we assume that the ionization front propagates until the ionization rate drops low enough (due to geometric diminishment) to equal the recombination rate of the ambient medium, then we can solve for the equilibrium ionization radius by setting the two rates equal.

The ionization rate per unit volume can be written as the (fill in here)

$$R_S = \left(\frac{3}{4\pi} \frac{\dot{N}_\gamma}{\alpha n_H^2} \right) \tag{4.3}$$

$$R(t) = R_S[1 - \exp\left(t/t_{\rm recomb}\right)]^{1/3} \tag{4.4}$$

The above derivation assumes a "sharp" ionization front, meaning the transition from ionized to neutral is across an infinitesimal region. In practice, the transition region is small compared to the size of the ionized region, but there is structure interior to the Strömgren radius that is not accounted for by simply solving for the equilibrium radius. In order to solve for a non-sharp ionization front, we must integrate the above ionization equations [Osterbrock and Ferland, 2006]. (ADD HERE). In the following tests, we include both the sharp and non-sharp ionization front solutions for comparison to our results.

We follow the initial conditions of Iliev et al. [2006]; the medium is

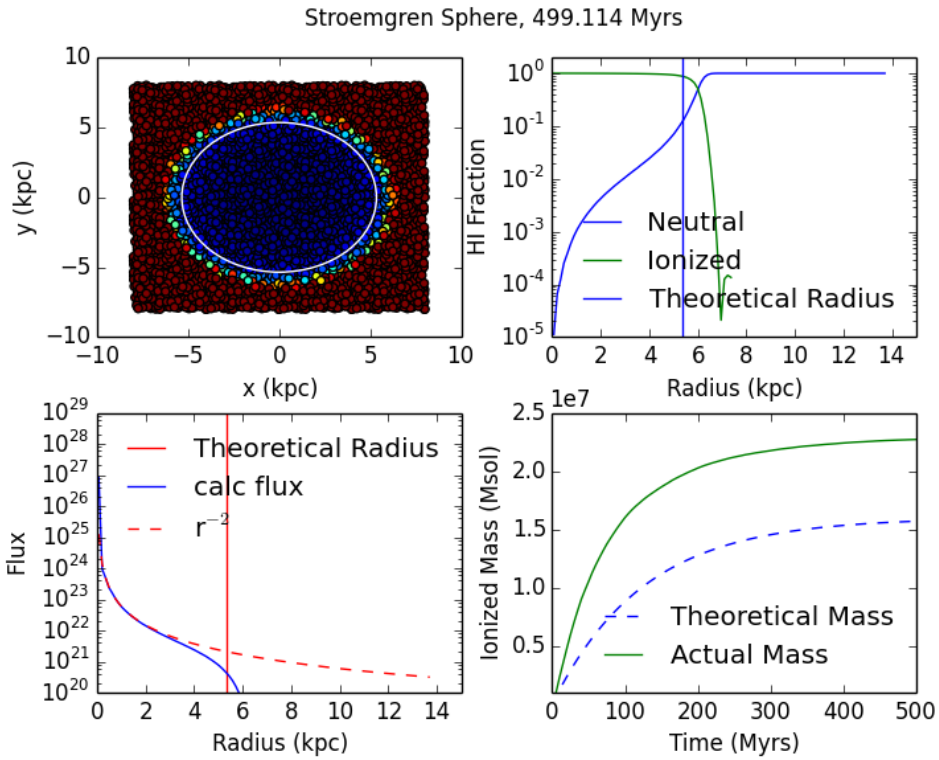


Figure 4.3: A slice of particles showing the ionization state.

initially neutral with a temperature 1e5 K and a density of 1e-3 cm^{-3}. An ionizing source is turned on at t = 0 that emits $\dot{N} = 5e48$ photons s^{-1} at 13.6 ev. We use a cross section $\sigma = 6.3e-18cm^2$ and a recombination rate of $\alpha = 2.59e-13cm^{-3}s^{-1}$, typical of 1e5 K gas. These values give a Strömgren radius of 5.38 kpc and a recombination time of 125 Myr.

4.4.2 The Thermal Case

The above formulation assumed the gas was isothermal and that all incident photons had the same energy. In reality, photons range across many wavelengths (commonly in a Planck spectrum) with differing cross sections for each wavelength. As well, absorption typically causes heating, which effects,

among many properties, recombination rate.

In order to do a more realistic test,the incident photons are assumed to be from a black body with temperature 1e5 K. The cross section is changed to an integrated cross section, obtained by integrating the cross section as a function of wavelength over all wavelengths having energies between 13.6 ev and 29.65 ev. The gas has an initial temperature, This scenario does not have an analytic solution to compare to, and so we instead compare to the results of Iliev et al. [2006] and Petkova and Springel [2009]. (NOTE - seems to be inconsistency. 10^5 K source corresponds to 43 ev = 29.65 + 13.6, but other areas seem to indicate total energy of 29.65, not heating energy = 29.65. Figure this out).

This test includes heating due to absorption and cooling due to recombination Λ_r, collisional ionization Λ_{ci}, line cooling Λ_l, and Bremsstrahlung radiation Λ_B. The rate are taken from Cen [1992] in order to match Petkova and Springel [2009]. The following are those rates in ergs cm^{-3} s^{-1}:

$$\Lambda_r = 8.7 \times 10^{-27} \sqrt{T} \left(\frac{T}{10^3 K}\right)^{-0.2} \Bigg/ \left[1 + \left(\frac{T}{10^6 K}\right)^{0.7}\right], \tag{4.5}$$

$$\Lambda_{ci} = 1.27 \times 10^{21} \sqrt{T} \left(1 + \sqrt{\frac{T}{10^5 K}}\right) e^{157809.1/T} n_e n_{HII}, \tag{4.6}$$

$$\Lambda_l = 7.5 \times 10^{-19} \left(1 + \sqrt{\frac{T}{10^5 K}}\right)^{-1} e^{-118348/T} n_e n_{HI}, \tag{4.7}$$

$$\Lambda_B = 1.42 \times 10^{-27} g_{ff} \sqrt{T} n_e, \tag{4.8}$$

where $g_{ff} = 1.3$ is the gaunt factor.

Figure 4.4 shows a radially averaged profile of temperature. We see a

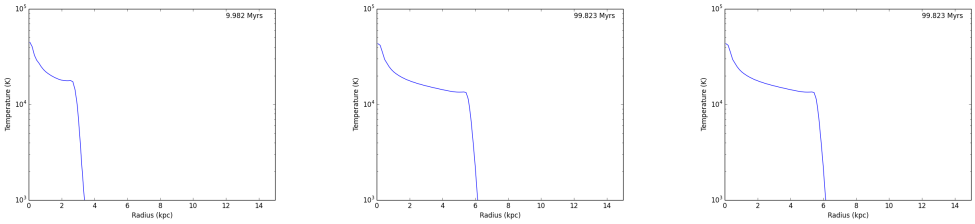


Figure 4.4: Temperature vs radius for the thermal Strömgren sphere.

peak temperature of roughly $3 \times 10^4 K$ with a radial dropoff stimilar to results from Iliev et al. [2006]. This demonstrates the code's ability to couple radiation to the thermodynamics of the gas (ADD MORE).

4.5 The Gas Wall

In order to test the algorithm's ability to handle a sharp density jump, we again perform the isothermal stromgren front (section 4.4.1), but with a large density jump. We keep all of the same initial parameters, but change the density to the left of x = 0 to $\rho/2$ and the density to the right of x = 0 to $3/2\rho$.

4.5.1 Only Radiation

In the case that hydrodynamics is off, the solution is two strömgren hemispheres centered at x = 0.

4.5.2 With Hydrodynamics - the Champagne Flow

In order to test the coupling of radiation to the hydrodynamics, we perform a similar test in which the code now uses its hydrodynamics solvers. We

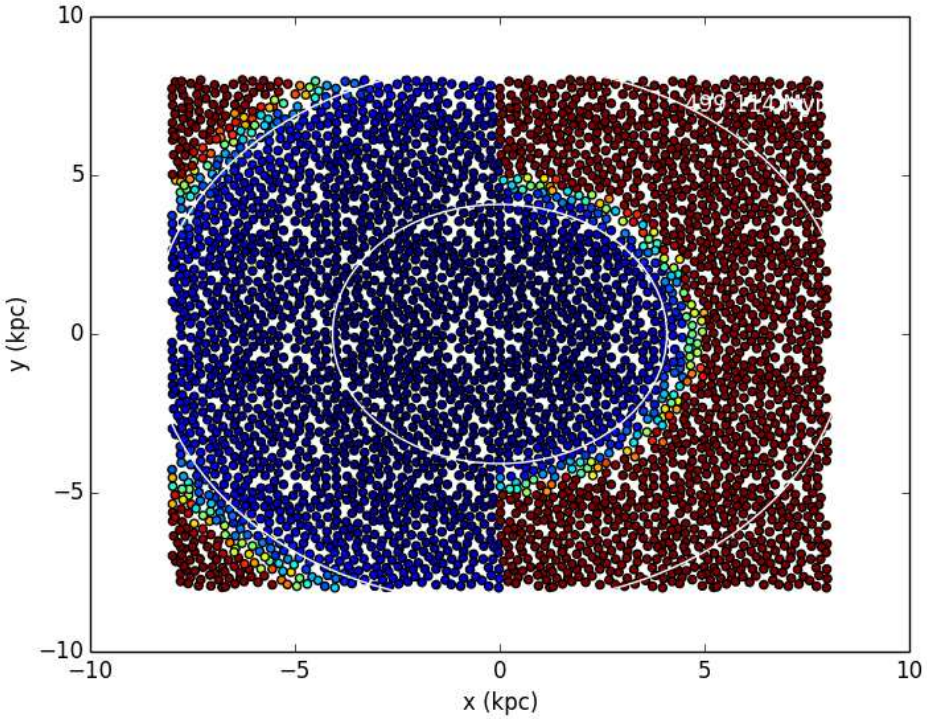
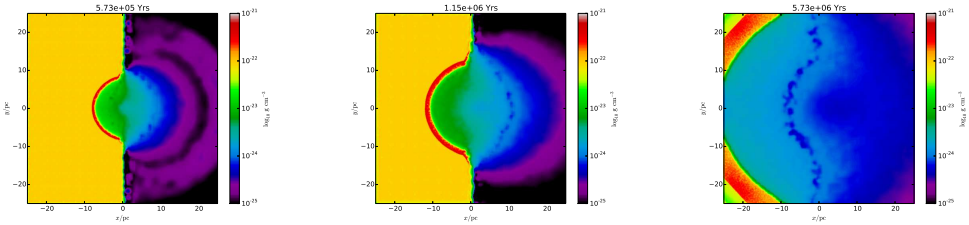


Figure 4.5: Two slabs of gas at different densities.

follow the setup of Gendelev and Krumholz [2012]; A 50 pc cube is initialized with a density of 0.055 atoms cm^{-3} to the right of x = 0, and 63 atoms cm^3 to the left. The temperatures of the left and right halves are 55 K and 6.3e3 K, respectively. This density/temperature combination gives pressure equilibrium at the boundary. An ionizing source is turned on at the origin that emits 5.3e47 photons/s, similar to a type BO.5 star. The combination of density and luminosity gives a stromgren sphere radius of 1.5 pc in the dense regio ns and a recombination time of 2.48e4 Years. The simulation is run for more than 4e6 years, meaning that the gas has time to heat and expand.



(a) 0.5 Myr (b) 1 Myr (c) 5 Myrs

Figure 4.6: The Blister HII region from Gendelev and Krumholz [2012].

4.6 Shadowing

We present the shadowing test from Hayes and Norman [2003], González et al. [2007], Skinner and Ostriker [2013]. A thin 2-D tube of gas with dimensions 0.12 cm tall x 1 cm long is irradiated from the left by a distance ((x,y) = (-100,0) cm) source with a characteristic temperature of 1740 K. The gas has an ambient density of $\rho_0 = 0.001 g cm^{-3}$ and an ambient temperature of 290 K. An oblate spheroid with dimensions $(x_0, y_0) = (0.1, 0.06) cm$ is placed in the tube at $(x_c, y_c) = (0.5, 0) cm$. The spheroid has a central density of $\rho_1 = 1000\rho_0 = 1 g cm^{-3}$ and the same ambient temperature. Note that hydrodynamics is turned off, so a mismatch in pressure is not an issue. The cloud has a density structure described by

$$\rho_{cloud}(x, y) = \rho_0 + \frac{(\rho_1 - \rho_0}{)}(1 + e^{\Delta}), \tag{4.9}$$

where

$$\Delta = 10 \left\{ \left[\frac{x - x_c)}{x_0} \right]^2 + \left[\frac{y - y_c}{y_0} \right]^2 - 1 \right\}. \tag{4.10}$$

This structure gives the cloud a "fuzzy" surface in that the density smoothly transitions rather than sharply jumps. The opacity is set as a func-

tion of density and temperature,

$$\kappa(T_{gas}, \rho) = \kappa_0 \left(\frac{T_{gas}}{T_0} \right)^{-3.5} \left(\frac{\rho}{\rho_0} \right), \tag{4.11}$$

with $\kappa_0 = 100 cm^2 g^{-1}$. This gives an optical depth of 0.1 across the length of the box, excluding the dense cloud, and an optical depth of roughly 2000 across the diameter of the blob.

The tube is resolved by 280x80 SPH resolution elements, and the above density structure is obtained by changing the mass of the SPH particles (as opposed to increasing the number of particles). Note that this is not realistic for SPH, where higher densities are usually represented by a higher density of particles. However, we have elected to keep the same resolution as previous papers to aid in comparison.

At t = 0, the source is turned on and the simulation is evolved for 0.1 s, or 3×10^9 light crossing times. Figure 4.7 shows the simulation at t = 0.1 s.

Need to fill in results...

4.7 Ionization Front Trapping

Along the same lines of section 4.6, this section performs a shadowing test. However, this test includes ionizing radiation and is constructed in such a way that the ionization front should stall out inside of the dense sphere. The test is taken from Iliev et al. [2006].

A 6.6 kpc cube is initiated with mean background density $2 \times 10^{-4} cm^{-3}$ and background temperature 8000 K. The dense clump is located at (x,y,z) = (5,3.3,3.3) and is given a density of 200 times the background, or 0.04 cm^{-3},

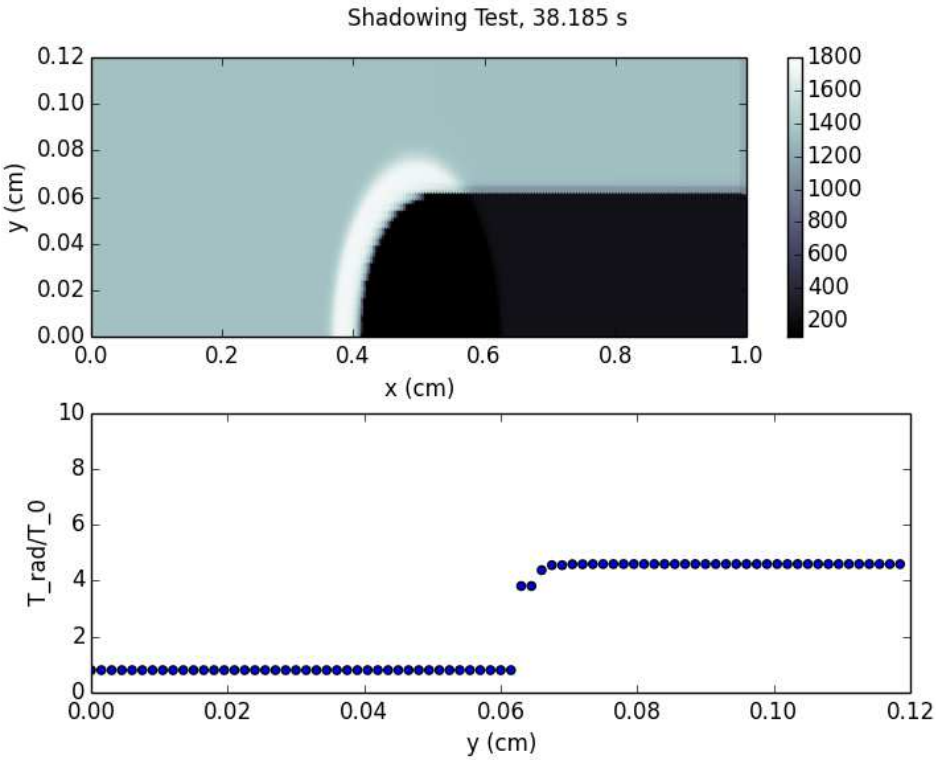


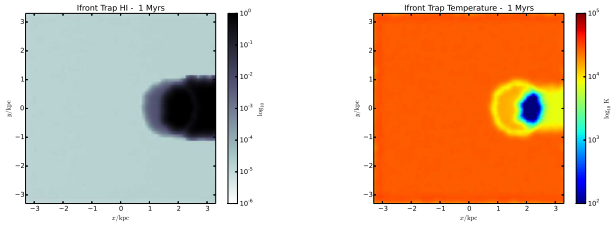
Figure 4.7: Demonstrating the codes ability to shadow.

and a temperature of 40 K.

The conditions for trapping an ionization front are presented in Shapiro et al. [2004]. The authors show that a clump can trap an ionization front if

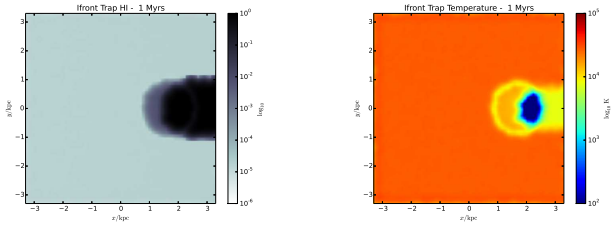
$$l_s = \frac{F}{\alpha n_H^2} \tag{4.12}$$

is less than the diameter of the clump. In the above simulation $\alpha = 2.59 \times 10^{-13}(T/10^4 K)^(- 3/4)$, which suggests the clump should trap the ionization front at 10^4 somewhere around halfway through.



(a) HI, 1 Myr (b) T, 1 Myr

Figure 4.8



(a) HI, 1 Myr (b) T, 15 Myrs

Figure 4.9

4.8 Timings and Scaling

Text to fix the formatting.

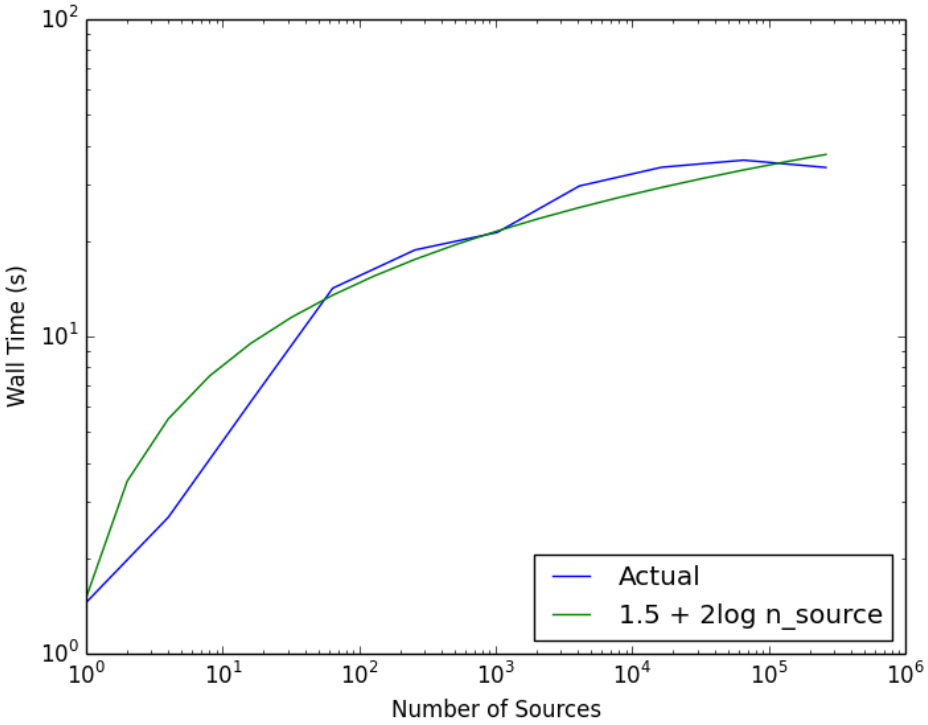


Figure 4.10: Wall time vs the number of sources.

Chapter 5

Applications to Galaxy Formation (Future Work?)

In this chapter, we discuss current and planned applications of the algorithm.

5.1 Galactic UV Fields

Currently, there is very little work in computational astrophysics that models the UV fields in and around galaxies. While many models have been created from the observational side [references], due to large computational cost, simulations have left this area largely unexplored or explored only at high redshift [references].

We have used [will be using] the above algorithm to re-run the MUGS2 comparison project [cite Ben].

- What effect does radiation have on the galaxy in MUGS2 (as a function of redshift, all the way to redshift 0)?
  - What effect does UV have on the ISM? (sam?)

  - What effect does UV have on satellite galaxies, gas properties, SFR - shut down mechanism?
  - What is the typical escape fraction of UV in galaxies (isolated disk from Sam, consistent with results from MUGS? Compare to Kannan et al. escape fractions)
  - Compare to observations of UV fields

Future work of this algorithm is quite broad; the flexibility allows application to a wide range of problems. The following are immediately planned projects, and following that is a short list of unplanned but possibly interesting projects.

- Look at H2 formation and destruction using Ly-Werner bands of radiation (Charlotte). How does cloud shielding depend on density? Compare to OWLS/EAGLE w/ TRAPHIC (Rahmati+ 13ab) (this is HI shielding and ntot H2 shielding... not great comparison?).
- Considering the properties of the ISM and molecular clouds (Samantha, Sijing). [Author] suggests 4 radiation bands are needed to sufficiently find ISM properties. Using these 4 bands, calculate effect of radiation on ISM. How do gas properties effect SFR and vice versa? [ask Sam for more info here]
- Potential to look at the effect of radiation processing. How does processing radiation (UV re-emitted as IR) effect the gas properties? How important of an effect is it in determining SFR? This project depends on a successful implementation of a self consistent gas source function. Can we do this in a stable way with respect to the cooling code?

- An obvious application is cosmic re-ionization, but this has already been done a bit since it does not require running to low redshift. That said, can we do it better/cheaper? Do our results agree?

Chapter 6

Conclusions and Future Work

Appendix A

Appendix A

Bibliography

D. C. Abbott and L. B. Lucy. Multiline transfer and the dynamics of stellar winds. *ApJ*, 288:679–693, January 1985. doi: 10.1086/162834.

T. Abel and B. D. Wandelt. Adaptive ray tracing for radiative transfer around point sources. *MNRAS*, 330:L53–L56, March 2002. doi: 10.1046/j.1365-8711.2002.05206.x.

T. Abel, M. L. Norman, and P. Madau. Photon-conserving Radiative Transfer around Point Sources in Multidimensional Numerical Cosmology. *ApJ*, 523: 66–71, September 1999a. doi: 10.1086/307739.

T. Abel, M. L. Norman, and P. Madau. Photon-conserving Radiative Transfer around Point Sources in Multidimensional Numerical Cosmology. *ApJ*, 523: 66–71, September 1999b. doi: 10.1086/307739.

M. L. Alme and J. R. Wilson. Numerical study of X-ray induced mass transfer in the HZ Herculis/Hercules X-1 binary system. *ApJ*, 194:147–164, November 1974. doi: 10.1086/153232.

G. Altay and T. Theuns. URCHIN: a reverse ray tracer for astrophysical applications. *MNRAS*, 434:748–764, September 2013. doi: 10.1093/mnras/stt1067.

G. Altay, R. A. C. Croft, and I. Pelupessy. SPHRAY: a smoothed particle hydrodynamics ray tracer for radiative transfer. *MNRAS*, 386:1931–1946, June 2008. doi: 10.1111/j.1365-2966.2008.13212.x.

A. M. Anile and V. Romano. Covariant flux-limited diffusion theories. *ApJ*, 386:325–329, February 1992. doi: 10.1086/171018.

D. Aubert and R. Teyssier. A radiative transfer scheme for cosmological reionization based on a local Eddington tensor. *MNRAS*, 387:295–307, June 2008. doi: 10.1111/j.1365-2966.2008.13223.x.

J. Barnes and P. Hut. A hierarchical O(N log N) force-calculation algorithm. *Nature*, 324:446–449, December 1986. doi: 10.1038/324446a0.

S. Cantalupo and C. Porciani. RADAMESH: cosmological radiative transfer for Adaptive Mesh Refinement simulations. *MNRAS*, 411:1678–1694, March 2011. doi: 10.1111/j.1365-2966.2010.17799.x.

R. Cen. A hydrodynamic approach to cosmology - Methodology. *ApJS*, 78: 341–364, February 1992. doi: 10.1086/191630.

R. Cen. A Fast, Accurate, and Robust Algorithm for Transferring Radiation in Three-dimensional Space. *ApJS*, 141:211–227, July 2002. doi: 10.1086/339805.

P. C. Clark, S. C. O. Glover, and R. S. Klessen. TreeCol: a novel approach to estimating column densities in astrophysical simulations. *MNRAS*, 420: 745–756, February 2012. doi: 10.1111/j.1365-2966.2011.20087.x.

C. P. Dullemond. RADMC-3D: A multi-purpose radiative transfer tool. Astrophysics Source Code Library, February 2012.

B. Ercolano, M. J. Barlow, P. J. Storey, and X.-W. Liu. MOCASSIN: a fully three-dimensional Monte Carlo photoionization code. *MNRAS*, 340:1136–1152, April 2003. doi: 10.1046/j.1365-8711.2003.06371.x.

B. Ercolano, M. J. Barlow, and P. J. Storey. The dusty MOCASSIN: fully self-consistent 3D photoionization and dust radiative transfer models. *MNRAS*, 362:1038–1046, September 2005. doi: 10.1111/j.1365-2966.2005.09381.x.

L. Gendelev and M. R. Krumholz. Evolution of Blister-type H II Regions in a Magnetized Medium. *ApJ*, 745:158, February 2012. doi: 10.1088/0004-637X/745/2/158.

N. Y. Gnedin and T. Abel. Multi-dimensional cosmological radiative transfer with a Variable Eddington Tensor formalism. *New Astron.*, 6:437–455, October 2001. doi: 10.1016/S1384-1076(01)00068-9.

M. González, E. Audit, and P. Huynh. HERACLES: a three-dimensional radiation hydrodynamics code. *A&A*, 464:429–435, March 2007. doi: 10.1051/0004-6361:20065486.

K. M. Gorski, B. D. Wandelt, F. K. Hansen, E. Hivon, and A. J. Banday. The HEALPix Primer. *ArXiv Astrophysics e-prints*, May 1999.

K. M. Górski, E. Hivon, A. J. Banday, B. D. Wandelt, F. K. Hansen, M. Reinecke, and M. Bartelmann. HEALPix: A Framework for High-Resolution Discretization and Fast Analysis of Data Distributed on the Sphere. *ApJ*, 622:759–771, April 2005. doi: 10.1086/427976.

T. J. Harries and I. D. Howarth. Raman scattering in symbiotic stars. II. Numerical models. *A&AS*, 121:15–44, January 1997. doi: 10.1051/aas:1997110.

J. C. Hayes and M. L. Norman. Beyond Flux-limited Diffusion: Parallel Algorithms for Multidimensional Radiation Hydrodynamics. *ApJS*, 147:197–220, July 2003. doi: 10.1086/374658.

I. T. Iliev, B. Ciardi, M. A. Alvarez, A. Maselli, A. Ferrara, N. Y. Gnedin, G. Mellema, T. Nakamoto, M. L. Norman, A. O. Razoumov, E.-J. Rijkhorst, J. Ritzerveld, P. R. Shapiro, H. Susa, M. Umemura, and D. J. Whalen. Cosmological radiative transfer codes comparison project - I. The static density field tests. *MNRAS*, 371:1057–1086, September 2006. doi: 10.1111/j.1365-2966.2006.10775.x.

I. T. Iliev, D. Whalen, G. Mellema, K. Ahn, S. Baek, N. Y. Gnedin, A. V. Kravtsov, M. Norman, M. Raicevic, D. R. Reynolds, D. Sato, P. R. Shapiro, B. Semelin, J. Smidt, H. Susa, T. Theuns, and M. Umemura. Cosmological radiative transfer comparison project - II. The radiation-hydrodynamic tests. *MNRAS*, 400:1283–1316, December 2009. doi: 10.1111/j.1365-2966.2009.15558.x.

M. Jeon, A. H. Pawlik, T. H. Greif, S. C. O. Glover, V. Bromm, M. Milosavljević, and R. S. Klessen. The First Galaxies: Assembly with Black Hole Feedback. *ApJ*, 754:34, July 2012. doi: 10.1088/0004-637X/754/1/34.

M. Jeon, A. H. Pawlik, V. Bromm, and M. Milosavljević. Radiative feedback from high-mass X-ray binaries on the formation of the first galaxies and early reionization. *MNRAS*, 440:3778–3796, June 2014a. doi: 10.1093/mnras/stu444.

M. Jeon, A. H. Pawlik, V. Bromm, and M. Milosavljević. Recovery from Population III supernova explosions and the onset of second-generation star formation. *MNRAS*, 444:3288–3300, November 2014b. doi: 10.1093/mnras/stu1980.

M. Jeon, V. Bromm, A. H. Pawlik, and M. Milosavljevic. The first galaxies: simulating their feedback-regulated assembly. *ArXiv e-prints*, January 2015.

P. Jonsson. SUNRISE: polychromatic dust radiative transfer in arbitrary geometries. *MNRAS*, 372:2–20, October 2006. doi: 10.1111/j.1365-2966.2006.10884.x.

R. Kannan, G. S. Stinson, A. V. Macciò, J. F. Hennawi, R. Woods, J. Wadsley, S. Shen, T. Robitaille, S. Cantalupo, T. R. Quinn, and C. Christensen. Galaxy formation with local photoionization feedback - I. Methods. *MNRAS*, 437:2882–2893, January 2014. doi: 10.1093/mnras/stt2098.

M. Klassen, R. Kuiper, R. E. Pudritz, T. Peters, R. Banerjee, and L. Buntemeyer. A General Hybrid Radiation Transport Scheme for Star Formation Simulations on an Adaptive Grid. *ApJ*, 797:4, December 2014. doi: 10.1088/0004-637X/797/1/4.

R. Kuiper, H. Klahr, C. Dullemond, W. Kley, and T. Henning. Fast and accurate frequency-dependent radiation transport for hydrodynamics simulations in massive star formation. *A&A*, 511:A81, February 2010. doi: 10.1051/0004-6361/200912355.

C. D. Levermore. Relating Eddington factors to flux limiters. *J. Quant. Spec. Radiat. Transf.*, 31:149–160, February 1984. doi: 10.1016/0022-4073(84)90112-2.

C. D. Levermore and G. C. Pomraning. A flux-limited diffusion theory. *ApJ*, 248:321–334, August 1981. doi: 10.1086/159157.

L. B. Lucy. Computing radiative equilibria with Monte Carlo techniques. *A&A*, 344:282–288, April 1999.

F. Melia and G. J. Zylstra. Flux-limited diffusion in a scattering medium. *ApJ*, 374:732–740, June 1991. doi: 10.1086/170157.

D. Mihalas and B. W. Mihalas. *Foundations of radiation hydrodynamics*. 1984.

M. Nenkova, Z. Ivezic, and M. Elitzur. DUSTY: A Publicly Available Code for Continuum Radiative Transfer in Astrophysical Environments. *LPI Contributions*, 969:20, 1999.

S. R. Och, L. B. Lucy, and M. R. Rosa. Diffuse radiation in models of photoionized nebulae. *A&A*, 336:301–308, August 1998.

D. E. Osterbrock and G. J. Ferland. *Astrophysics of gaseous nebulae and active galactic nuclei*. 2006.

A. H. Pawlik and J. Schaye. TRAPHIC - radiative transfer for smoothed particle hydrodynamics simulations. *MNRAS*, 389:651–677, September 2008. doi: 10.1111/j.1365-2966.2008.13601.x.

A. H. Pawlik and J. Schaye. Multifrequency, thermally coupled radiative transfer with TRAPHIC: method and tests. *MNRAS*, 412:1943–1964, April 2011. doi: 10.1111/j.1365-2966.2010.18032.x.

A. H. Pawlik, V. Bromm, and M. Milosavljević. Assembly of the first disk galaxies under radiative feedback from the first stars. *Mem. Soc. Astron. Italiana*, 85:565, 2014.

A. H. Pawlik, J. Schaye, and C. Dalla Vecchia. Spatially adaptive radiation-hydrodynamical simulations of galaxy formation during cosmological reionization. *ArXiv e-prints*, January 2015.

M. Petkova and V. Springel. An implementation of radiative transfer in the cosmological simulation code GADGET. *MNRAS*, 396:1383–1403, July 2009. doi: 10.1111/j.1365-2966.2009.14843.x.

C. Pinte, F. Ménard, G. Duchêne, and P. Bastien. Monte Carlo radiative transfer in protoplanetary disks. *A&A*, 459:797–804, December 2006. doi: 10.1051/0004-6361:20053275.

G. C. Pomraning. Flux-limited diffusion with relativistic corrections. *ApJ*, 266:841–847, March 1983. doi: 10.1086/160832.

A. Rahmati, A. H. Pawlik, M. Raicevic, and J. Schaye. On the evolution of the H I column density distribution in cosmological simulations. *MNRAS*, 430:2427–2445, April 2013a. doi: 10.1093/mnras/stt066.

A. Rahmati, J. Schaye, A. H. Pawlik, and M. Raicevic. The impact of local stellar radiation on the H I column density distribution. *MNRAS*, 431: 2261–2277, May 2013b. doi: 10.1093/mnras/stt324.

A. O. Razoumov and D. Scott. Three-dimensional numerical cosmological radiative transfer in an inhomogeneous medium. *MNRAS*, 309:287–298, October 1999. doi: 10.1046/j.1365-8711.1999.02775.x.

E.-J. Rijkhorst, T. Plewa, A. Dubey, and G. Mellema. Hybrid characteristics: 3D radiative transfer for parallel adaptive mesh refinement hydrodynamics. *A&A*, 452:907–920, June 2006. doi: 10.1051/0004-6361:20053401.

J. Rosdahl and R. Teyssier. A scheme for radiation pressure and photon diffusion with the M1 closure in RAMSES-RT. *MNRAS*, 449:4380–4403, June 2015. doi: 10.1093/mnras/stv567.

J. Rosdahl, J. Blaizot, D. Aubert, T. Stranex, and R. Teyssier. RAMSES-RT: radiation hydrodynamics in the cosmological context. *MNRAS*, 436: 2188–2231, December 2013. doi: 10.1093/mnras/stt1722.

G. B. Rybicki and A. P. Lightman. *Radiative Processes in Astrophysics*. June 1986.

P. R. Shapiro, I. T. Iliev, and A. C. Raga. Photoevaporation of cosmological minihaloes during reionization. *MNRAS*, 348:753–782, March 2004. doi: 10.1111/j.1365-2966.2004.07364.x.

M. A. Skinner and E. C. Ostriker. A Two-moment Radiation Hydrodynamics Module in Athena Using a Time-explicit Godunov Method. *ApJS*, 206:21, June 2013. doi: 10.1088/0067-0049/206/2/21.

L. Spitzer. *Physical processes in the interstellar medium*. 1978.

B. Strömgren. The Physical State of Interstellar Hydrogen. *ApJ*, 89:526, May 1939. doi: 10.1086/144074.

A. G. G. M. Tielens. *The Physics and Chemistry of the Interstellar Medium*. August 2005.

N. J. Turner and J. M. Stone. A Module for Radiation Hydrodynamic Calculations with ZEUS-2D Using Flux-limited Diffusion. *ApJS*, 135:95–107, July 2001. doi: 10.1086/321779.

J. W. Wadsley, J. Stadel, and T. Quinn. Gasoline: a flexible, parallel implementation of TreeSPH. *New Astron.*, 9:137–158, February 2004. doi: 10.1016/j.newast.2003.08.004.

S. Wolf. MC3D-3D continuum radiative transfer, Version 2. *Computer Physics Communications*, 150:99–115, February 2003. doi: 10.1016/S0010-4655(02)00675-6.

K. Wood, J. S. Mathis, and B. Ercolano. A three-dimensional Monte Carlo photoionization code for modelling diffuse ionized gas. *MNRAS*, 348:1337–1347, March 2004. doi: 10.1111/j.1365-2966.2004.07458.x.