RADIATIVE TRANSPORT IN STAR- AND PLANET FORMATION

TROELS FROSTHOLM

A Diffuse Ray-Tracing Scheme for Adaptive Mesh Global Star Formation Modelling

Masters Thesis in Physics
Niels Bohr Institute
University of Copenhagen
Supervisors: Åke Nordlund and Troels Haugbølle
June 2014 – version 0.1

ABSTRACT

A long-characteristics ray tracing solver for radiative transport is developed and integrated into the adaptive mesh refinement (AMR) code RAMSES, coupling radiative heating with the magnetohydrodynamics. It can be applied to diffusive radiating gas as well as to point sources of radiation. The solver is tested against the analytical solution of a minimal one-dimensional problem, with results within floating point error using the integral method and on order per mil using Feautriers method. The coupling with hydrodynamics is tested in the context of a radiatively cooling hydrostatic disk atmosphere. RAMSES' Godunov solver is unable to keep the equilibrium stable, but the atmosphere's temperature profile evolves at least qualitatively as expected. A state-of-the-art photo-ionization algorithm[45] is implemented, integrated with the ray tracing, and tested on the expanding HII region around a newborn star, as part of the StarBench code comparison. A thin smooth spherical ionization front is observed, which expands slightly faster than expected from analytical calculations, albeit with no more deviation than other participating codes. Finally, the code is used for a simulation of the ionization by a high-mass star of the surrounding turbulent molecular hydrogen cloud, and features like champagne-flows are seen. Performance on a unigrid allows hundreds of rays through each cell without exceeding the computational time of the hydrodynamics, while in full AMR more work is needed to achieve such performance.

ACKNOWLEDGMENTS

I would like to thank my supervisors, Troels Haugbølle and Åke Nordlund for their great support and guidance during the writing of this thesis, and not least during the writing of amounts of Fortran code that has consumed most of my time during the past year. They have been available with lots of good ideas, explanations and help, even at odd hours near deadlines. They have also done a great job at including me in the research community, both at the local research group here in Copenhagen, and at conferences and summer schools abroad. I would like to thank them for giving me these opportunities, and for making this masters thesis possible.

I would also like to thank the people in the research group here at the Niels Bohr institute and at STARPLAN for providing an enjoyable working environment. I will be looking forward to more football matches and barbecues.

To the other kids in the masters thesis office, I would like to say thank you for injecting some fun into the sometimes long days. A special thanks to Michael Küffmeyer for sharing his knowledge and for reading and giving feedback on my thesis.

Finally I would like to thank my girlfriend for her invaluable patience and support.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LISTINGS

ACRONYMS

AMR Adaptive Mesh Refinement

MHD Magnetohydrodynamics

TSC Triangular Shaped Cloud (interpolation)

RAMSES the cosmological AMR Magnetohydrodynamics (MHD) code

STAGGER the stellar MHD code

MPI Message Passing Interface

CIC Cloud in Cell (interpolation)

FTT Fully Threaded Tree

ISM Interstellar Medium

KROME a non-equilibrium Interstellar Medium (ISM) chemistry code

IMF Initial Mass Function

SED Spectral Energy Distribution

IR Infrared

MRI Magneto-Rotational Instability

YSO Young Stellar Object

GMC Giant Molecular Cloud

Part I

INTRODUCTION

1

MOTIVATION

Stars form in collapsing cores in giant clouds of cold molecular gas. Turbulence, magnetic effects, radiative energy transport and chemistry are vital processes, and there is a complex interplay between these. Much progress has been made in understanding the process of star formation, but there are still many important open questions, including the question of how the magnetic flux is removed efficiently enough that the stars end up with much weaker magnetic flux per unit mass than the average interstellar medium out of which they are formed. Another issue is the so-called luminosity problem, namely that the heating expected from observed accretion rates and the measured luminosity from accretion disks do not match. Furthermore, there is still a long way to a detailed understanding of the transport and evolution of chemical species, which can be compared with current ongoing extra-solar star formation through line observations, as well as the formation of the solar system through measurements on meteorites.

To answer these and many other questions, detailed simulations of star formation in three dimensions are required and abundantly used. The computational group at NBI and STARPLAN tries to do this in the form of assumption-free forward modeling from first principles. To avoid imposing arbitrary initial- and boundary conditions, simulations start at galactic scales where statistically reasonable initial conditions are known empirically. Formation of dense molecular cloud complexes by the action of supernovae outbursts is followed, and used by detailed simulations starting on molecular cloud scales (typically tens of parsecs). These are used to zoom in on the details of the process of star formation. The huge range of size scales due to the hierarchical nature of the star forming environment pose a computational challenge, as does the range of physical conditions and relevant physical processes.

For this reason, until recently, in hyper-global simulations such as these, the effects of radiative energy transport and chemical evolution has been treated as secondary effects that could be approximated by empirically derived laws. Important effects, such as angle- and shadow effects on protostellar accretion disks, ionization from central and nearby stars among many others, cannot be treated exactly in this manner. Furthermore, a self-consistent first principles treatment of energy transfer by radiation is an important step towards a more assumption-free model, and towards a model with better quantitative predictive power.

2

AIM

This masters thesis aims to develop a solver for radiative energy transfer for use in, and integrated into, the cosmological AMR MHD code (RAMSES), which is used in a modified version in the Copenhagen computational astrophysics group for modeling star formation. In order to be both efficient and exact, it should be based on ray-tracing. To be able to treat many phenomena such as for example irradiation from jet-induced shocks, it must treat radiation from diffuse sources as well as point sources. The efficiency and validity of the solver should be documented using standard benchmarks from the field of computational astrophysics and star formation. Furthermore, it should be applied in a situation relevant for star formation.

3 METHOD

The radiative transfer solver is tailored to the adaptive mesh refinement (AMR) and adaptive time stepping of RAMSES, as well as its specific memory layout. A clean separation between the radiative transport solver and the rest of RAMSES is enforced, requiring only an initialization-, and a radiative solver procedure to be called from RAMSES. For efficiency and clarity of the code, the solver does however access RAMSES variables directly. Written as an extension patch, the solver, like RAMSES, is implemented using the Fortran 90 programming language extended with the Message Passing Interface (MPI) cluster parallelization framework combined with OpenMP shared memory parallelization.

The solver was implemented in three iterations. First, a solver was developed that works only in the unigrid case. The unigrid case is when the mesh is not adaptively refined, but is a regular Cartesian grid. This work had for the main part already been undertaken by Troels Haugbølle when work on the present thesis commenced, but the unigrid solution was made operational and tested as part of this work. These tests include checking the solution in minimal inherently one-dimensional cases, as well as examining the solution in the case of cooling of a simple model of a proto-planetary disk initially in isothermal hydrostatic equilibrium.

The second iteration was to implement the full adaptive mesh refinement solution, using the unigrid solution as a reference implementation. The same overall strategy is used in both implementations, but the details differ considerably. The method employed is a ray-tracing approach, where the equation of radiative transfer is solved along long characteristic rays through the whole computational domain. This is done efficiently by changing the decomposition of the problem from being centered on cells, to being centered on rays. In this way, the solution along rays can be found in parallel on a very large number of CPUs, doing so efficiently by using the capabilities of the CPU to vectorize and optimally use its memory cache. It is crucial that the solution along the rays is fast, because the large number of rays required to get an accurate overall solution is the reason that ray tracing is traditionally considered an expensive method. The full AMR solver is tested against the same problems as the unigrid solver. The details of the full AMR solver are covered in Chapter 8, but it is worth highlighting one important difference from the unigrid solver here. Where the unigrid solver handles batches of parallel rays that cover all cells in the simulation, the full AMR solver allows for all in-

dividual rays to be chosen arbitrarily, and so any ray geometry may be used. This is exploited to use the solver to model a point source.

The third iteration was to implement a solver for the radiative ionization of hydrogen. It was tested on the classical problem of an expanding HII ionization front around a star where fusion has just ignited. The experiment functions simultaneously as a test of the radiative transfer using a point source geometry, the photo-ionization solver and the interplay between radiation and hydrodynamics as the heating from the ionization creates a pressure shock-front.

Part II

THEORY

4

STAR FORMATION

Over the last couple of decades, the understanding of star formation has transformed, from considering the symmetrical collapse of structures in near-equilibrium to a much more dynamical picture where turbulence and the influence of magnetic fields play a crucial role. This has to a large extent been made possible by the advances in computer simulations in two and three dimensions, as well as in the spectral and spatial resolution of observations. This chapter attempts to give a brief overview of the vast field, with emphasis on the parts where radiative transport plays a particular role.

This chapter is mainly based on McKee and Ostriker [44], and Stahler and Palla [61].

4.1 GIANT MOLECULAR CLOUDS

Stars form in giant clouds of cold, molecular hydrogen and atomic helium (Giant Molecular Cloud (GMC)s), with masses in excess of $10^4 M_\odot$. They contain most of the molecular mass of the interstellar medium (ISM), and form mainly in galactic spiral arms. Clouds have a hierarchical structure, that extends from the scales of the whole molecular cloud down to the thermal Jeans mass in the case of gravitationally bound clouds. Overdense regions within clouds are called 'clumps'. Cores are the regions out of which individual stars (single or multiple systems) form. Molecular clouds are surrounded by a layer of atomic gas, which shields their interior from interstellar UV radiation. Clouds are supersonically turbulent. Turbulence in incompressible flows has long been known to be associated with power laws like the velocity-length scale relation $v(l) \propto l^q$ in Kolmogorov theory[44, 36]. In more recent time, with the aid of simulations, similar power laws has been found to hold even for strongly compressible gas, to which molecular clouds belong. The first such indication that molecular clouds are turbulent was found by Larson [40] and is expressed in Larson's first law, which states that in GMCs, a relation between length scale and spectral line width holds, equivalent with a velocity dispersion that follows a power law. Larson finds the exponent to be 0.38. The relation also holds within the cloud. Larson further found (second law) that GMCs are marginally gravitationally bound, that is their virial number $\alpha_{vir} \approx 1$. The virial theorem states that the rate of change of contraction (or expansion) of a cloud is determined by the total energy of the system, which consists of a kinetic, magnetic and gravitational term. The virial number α_{vir} is the

ratio of total kinetic to gravitational energy. Later research [e.g. 51, 33] however suggests that the virial parameter of GMCs must be around 5-10, which is also in better accordance with the low star formation rate that is observed. GMCs have filamentary structure, which is the expected outcome of gravitational collapse in three dimensions, but may also form at contact lines of shock sheets. GMC lifetimes are estimated at a few sound crossing times $t_{cross} \approx 10\,\mathrm{Myr} \cdot \left(\frac{M}{10^6 M_\odot}\right)^{\frac{1}{4}}$, which sets an upper limit on the time star formation takes.

The formation of GMCs has been explained using a top-down and a bottom-up approach. In bottom-up formation, cold HI clouds was thought to collide inelastically, until they would become self-gravitating. The time-scales for these collisions, however, would be so large that the process would be truncated by destructive star formation. The top-down approach relies on large-scale instabilities in the diffuse ISM to produce the denser regions. One is the Parker instability [52], which involves differences in buoyancy in regions of different density along magnetic field lines parallel to the mid-plane. Another is the magneto-Jeans instability[35], which is a gravitational instability in the presence of magnetic fields. This is a runaway process. Star-forming clouds will form if the Toomre parameter[66]

$$Q \equiv \frac{\kappa c_s}{\pi G \Sigma_{gal}} \tag{1}$$

is below $Q_{crit} \approx 1.5$, where κ is the epicyclic frequency and c_s is the mean sound speed.

The thermal structure of clouds depends on radiative heating from internal and external stars, as well as cosmic rays, in combination with the density- and chemical structure and mechanical effects like collapse and shocks due to winds. In the absence of external massive stars, in the outer layers of the cloud, heating is dominated by cosmic ray ionizations, and cooling happens in the infrared, mainly in O I and C II lines, both of which are optically thin in this region. H is generally dissociated, while further into the cloud, self-shielding allows H_2 to exist. The lower level of ionization also means that the temperature drops, and that the main coolant becomes CO_2. Inside the cloud, the most significant modification of cloud thermal structure is from nearby stars. This happens in the form of direct radiation from (especially massive) stars, as well as shocks associated with magnetically driven winds and outflows. Massive stars emit the bulk of their radiation in the ultraviolet region of the spectrum. A molecular cloud near such a star receives an ultraviolet flux orders of magnitude greater than from diffusive interstellar radiation. The cloud receives photons at all energies up to 13.6 eV, since those with higher energies are absorbed in the HII region surrounding the star. The remaining radiation in effect dissociates all the molecules in the clouds surface layer, producing a photo-dissociation region (PDR). It also heats the

dust, whose thermal radiation can be observed in the far-infrared, for example in the Orion Nebula[61].

4.2 CLUMPS AND CORES

GMCs are highly clumped. Most of these clumps are gravitationally unbound, and so do not obey Larson's laws. The most massive clumps, however, appear to be bound [5]. Observationally, clumps are generally seen as peaks in position-velocity data cubes. Analysis of simulations show that such clumps found from position-velocity may in fact correspond to different physical structures, and many single physical structures have two or more components when viewed in line-of-sight[54].

Observationally, there is a function that characterizes the distribution of the initial masses of stars, called the Initial Mass Function (IMF), which may be approximated by a sequence of piece-wise power laws. The pieces take the form

$$\xi\left(m_*\right) = \xi_0 m_*^{-\alpha}, \tag{2}$$

where ξ is the probability of finding a star of mass m_*. The single power-law slope α initially found by Salpeter [58] has since been extended, and the standard IMF [38] has three parts, with slopes given in table 1. A very similar distribution holds for cores (the Core Mass

$\frac{m_*}{M_\odot}$	α
0.5-50	1.3
0.3-0.08	0.5
0.08-0.01	-0.7

Table 1: Slopes of the initial mass function as given in [38].

Function, CMF), except it is shifted to higher mass by a factor of a few, which suggests that the mass of a star is controlled by the available reservoir of matter in the core from which it was created. Any coherent theory of star formation must account for the features of the IMF.

4.3 LOW-MASS STARS

Low-mass star formation is characterized by a formation time shorter than the Kelvin-Helmholtz time $t_{KH} = \frac{Gm_*^2}{RL}$, which specifies the contraction time of a uniform density gas parcel of mass m to radius R with heat due to contraction radiated away with luminosity L. This means that the luminosity in low mass star formation is dominated by heating due to accretion, while for high-mass star formation it is

dominated by nuclear fusion. Since this criterion would let any star with rapid enough accretion be categorized as low mass, a line is drawn at $8M_{\odot}$.

Low-mass stars appear to form from gravitationally bound cores. The time scale for the collapse of these cores determines both the time scale for the formation of a star and the accretion luminosity. Isothermal cores that undergo gravitational collapse become very centrally concentrated, with density profiles that are approximately $\rho \propto r^{-2}$. Collapse of such a marginally unstable core begins near the outer radius. The infall rate is $\dot{m}_{in} = \phi_{in}\frac{c_s^3}{G}$, where ϕ_{in} is a constant factor, typically below one. If the effects of magnetic- and thermal pressure along with turbulence are included, the relation still holds approximately with c_s^2 replaced by $c_{eff}^2 = c_s^2 + v_A^2 + v_{turb}^2$, where v_A is the velocity of plasma waves due to magnetic tension; the Alfvén speed. If on the other hand[60] the evolution to the r^{-2} profile is assumed to be quasi-static (likely due to magnetic fields), the initial configuration is a singular isothermal sphere, which is an unstable hydrostatic equilibrium. The resulting collapse is initiated in the center and propagates inside-out at the sound speed, with infall rate

$$\dot{m}_{in} = 0.975\frac{c_s^3}{G}. \tag{3}$$

In the inner-most regions of the collapsing core, due to the enhanced density, the opacity eventually (at a density $\rho \sim 10^{13}\mathrm{g\,cm^3}$) becomes large enough that the gas switches from approximately isothermal behavior to adiabatic behavior. The gas becomes hot enough to stop the collapse by thermal pressure, which forms a stable structure known as the first core[40]. Once the gas is hot enough to dissociate hydrogen and destroy dust, the opacity drops again, and a second collapse ensues, in which the protostar is formed. When opacity effects are included, the maximum infall rate is limited to about $13\frac{c_s^3}{G}$[41]. A detailed multi-group radiative transfer treatment in one dimension can be found in [70, 71]. These models are however spherically symmetric, and do not take into account the following.

There are two observational facts that star formation theory must explain, namely that the specific angular momentum and ratio of mass to magnetic flux in a star is much smaller than in the interstellar medium. Angular momentum may be transported radially by viscous forces, and an effective viscosity can be induced in a sufficiently ionized disk by the Magneto-Rotational Instability (MRI). This instability arises from the fact that the lower a Keplerian orbit is, the higher is the velocity. In an ionized medium, the magnetic field lines move with the matter, and magnetic tension is created when they are forced to bend. For matter that by some small perturbation falls to a lower orbit, this creates a braking force from the field line connected with neighboring matter, forcing it into an even smaller orbit and increasing the tension in a runaway process.

Poloidal magnetic fields prevent gravitational collapse when they are sufficiently strong, in which case the core is 'subcritical'. Otherwise they merely inhibit collapse, and the core is 'critical'. Calculations involving the evolution of a rotating, thin axi-symmetrical disk show that the magnetic field assumes a characteristic hourglass shape, in which the field is normal to the disk and flares above and below it [e.g. 4]. This shape of the field is also found observationally [e.g. 20].

Due to the initial angular momentum of the core, an accretion disk forms as the core collapses. The magnetic fields anchored in the large-scale structures around the core, frozen as they are into the ionized gas, are twisted by the rotation of the disk, causing a braking effect due to the increasing magnetic tension. This provides an efficient transport mechanism of angular momentum, which is funneled out in the shape of magnetically driven jets and winds perpendicular to the disk. It can be so efficient, that in symmetric simulations it leads to the magnetic braking catastrophe, where the braking prevents a rotationally supported disk from appearing at all. This problem however disappears in models that include the effects of turbulence[32] or ambipolar diffusion[2]. Ambipolar diffusion and magnetic reconnection are candidates for resolving the magnetic flux problem, but the problem is not yet resolved.

The growth of a protostar can be inferred observationally through the mass distribution surrounding it, the velocity distribution of this circumstellar gas and the non-stellar radiative flux. The protostellar Spectral Energy Distribution (SED) is classified according to what is believed to be an evolutionary order, given by Andre et al. [3] as follows.

- Class 0 sources are extremely faint in the optical and near Infrared (IR), have a significant submilimeter luminosity and have a central protostar. Protostars are believed to acquire most of their mass in this phase.
- Class I sources have a positive slope of the SED over the wavelength range between 2.2μ and $10 - 25\mu$. These are believed to be relatively evolved protostars with circumstellar disks and envelopes.
- Class II sources has negative SED slope above -1.5 in the infrared, and are believed to be pre-main sequence stars with significant circumstellar disks.
- Class III sources with slope less than -1.5 are pre-main sequence stars that are no longer accreting significantly.

4.4 DISKS

Because protostellar cores are rotating, collapse with conservation of angular momentum results in the formation of a centrifugally-supported disk. Observed sizes and rotation parameters for low mass dense cores predicts disk sizes ≤ 1000 AU, consistent with submilimeter observations that indicate average dust disk sizes around T Tauri stars of ≈ 200 AU. Disk lifetimes are inferred based on stellar ages combined with IR and mm/submm emission signatures, which are sensitive to warm dust. Spitzer observations show that for around 70% of stars, disks have become optically thin in the IR within 2-3 Myr, implying that the inner disk has been removed. Calvet et al. [8] and others find an approximate dependence of accretion rate on stellar mass $\dot{M}_{disk} \propto m_*^2$ with considerable scatter. The scatter may potentially be explained by Bondy-Hoyle accretion by stars moving through the ambient medium[50].

Accretion processes depend on a complicated interplay of MHD, radiative transfer and chemistry. The MHD is non-ideal, since the medium is partly ionized, and self-gravity is important in many circumstances. Self-gravity effects and electrical conductivity are very sensitive to thermal- and ionization properties, which in turn are determined by chemistry and radiative transfer. Additionally, there are the difficulties of determining initial- and boundary conditions, as well as the huge dynamical range of scales from parsecs for the whole molecular cloud to fractions of an AU required to resolve the thin disk. Pure hydrodynamical accretion has not been very promising as angular momentum transport tends to work inwards instead of outwards. Viscous accretion via the MRI provides outward angular momentum transport, and in combination with magnetic braking is thought to drive accretion. While a large portion of the mass in the disk ultimately accretes onto the star, what is left behind is either incorporated into planets or removed by photo-evaporation.

The clearest observable manifestations of Young Stellar Object (YSO) outflows are the central "Herbig-Haro" jets, which consist of knots of ionized gas with velocities in excess of $100\,\mathrm{km\,s^{-1}}$, and the larger-scale bipolar outflows consisting of expanding lobes of molecular gas with velocities around $10\,\mathrm{km\,s^{-1}}$. The high velocity jets indicate that they represent the primary wind from the inner part of the star-disk system, while the low velocities and large masses of broad molecular outflows indicate that they are made up of gas from the surrounding environment, accelerated by an interaction with the wind. Collimated outflows are ubiquitous in both high- and low mass star formation. In some cases jets are observed with the outflows, and in these cases the jets are powerful enough to drive the observed outflows[59]. Two main types of jets have been explored: "x-winds", which are driven at the interaction region between the inner accretion disk and the

magnetosphere of the rotating YSO, and disk winds, which assumes that a larger region of the disk is threaded by open field lines, giving a range of terminal wind speeds. Jets cause outflows, containing a series of shocks. The leading bow shock is created at the working surface of the head of the jet. Jets with internal shocks can create analogous bow shocks, which propagate into the wind that surrounds the jet.

5 RAMSES

The RAMSES code was developed in 2001 by Romain Teyssier [64, 18]. It was developed as a cosmological code, created for the study of large scale structure and galaxy formation. It has hierarchical adaptive mesh refinement, in order to capture the differences in length scales inherent in the problem[65]. Originally a pure fluid dynamics code with self-gravity and collision-less point particle dynamics, it was extended in 2006 with a magneto-hydrodynamics solver[18]. It was chosen by the computational group at NBI and STARPLAN for its ability to cover many orders of magnitude in physical size, and suitability to describe galactic dynamics, as a starting point for a code to study the formation of stars in a realistic galactic environment. To that end, it was extended with a module for sink particles representing stars, empirically based metalicity-depended heating and optically-thin cooling, as well as the ability to rerun and zoom in on an interesting region (e.g. a forming star) in a simulation snapshot. The present work of course aims to eventually give a more precise treatment of the radiative energy budget also in optically-thick and occluded regions. Another current project in the Copenhagen group aims to treat in more detail the dependence of heating and cooling on the chemistry, by linking RAMSES with a non-equilibrium ISM chemistry code (KROME).

The rest of this chapter describes the aspects of RAMSES needed to understand the Radiative Transport module, which is the focus of this thesis.

5.1 ADAPTIVE MESH REFINEMENT

RAMSES uses an adaptively refined hierarchical mesh. It is a Cartesian grid, where each grid cell may be recursively subdivided into the eight child cells made up of the octants of the cubical cell, which gives a structure called an octree. We shall call the number of such recursive subdivisions the level of refinement. A cell at level of refinement l has side length $dx = \left(\frac{1}{2}\right)^l$ in units of the size of the simulated domain. RAMSES has a base grid that is fully refined, i.e. all cells are subdivided up to a level of refinement l_{min}, which is a parameter of the simulation. At levels of refinement higher than l_{min}, refinement criteria determine which cells are subdivided. There is also an upper limit l_{max}, above which no further refinement takes place.

5.2 MESSAGE PARSING INTERFACE

For parallelization, RAMSES makes use of the Message Parsing Interface (MPI) framework for multi-process parallelization, in combination with OpenMP for in-process threaded parallelization. The difference is that MPI can parallelize on a cluster of independent machines, each running a self-contained process with its own local memory, whereas OpenMP can parallelize over a number of CPU cores in a number of threads that share memory on a single machine. MPI works by starting the same program in a number of processes, assigning each a unique id (called the process rank) from 0 to the number of processes minus one. The processes may be located on the same machine, or on different machines. While the program runs, processes may communicate with each other by sending messages, using the ranks as "postal addresses".

5.3 FULLY THREADED TREE

RAMSES makes use of the Fully Threaded Tree (FTT) structure proposed by Khokhlov [34]. It solves a number of common problems with octrees, including lookup of neighbor cells without tree traversal, parallelizability of all tree operations including creation and destruction of cells, and reduction of the memory overhead required to maintain the tree. Additionally, an adaptive time stepping strategy is proposed, which is also used in RAMSES. The basic unit is the oct, which is a structure that has a pointer to eight cells, each of which may in turn have a pointer to an oct. Because creation of all eight cells of an oct always happen at the same time, they may be stored contiguously in memory, so only one pointer is needed to access them. Since they are always stored in the same consecutive order, any one cell can always be found by adding a constant number to the pointer. Each oct has pointers to the six parent cells of its neighboring octs, and a pointer to its own parent cell (figure 1). These additional pointers are called threads, hence the term fully threaded.

Finding a neighbor cell without tree traversal just amounts to following the pointer to the neighboring parent cell. If the cell is a leaf, we are done. If the cell is split, i.e. it contains an oct, the address of the neighbor cell is found by pointer addition. Parallel creation and removal of cells is problematic if the cells contain pointers to their neighbors. The problem arises when two neighbors are created or removed at the same time in different processes. For removal, the two cells should update their pointer, but they are already removed. For addition, the cells should be born with pointers to a cell they don't yet know has been created. This problem is avoided by the FTT because only a pointer to the neighboring octs' parent cell is stored, which doesn't change at cell creation and removal. Lastly, because pointers

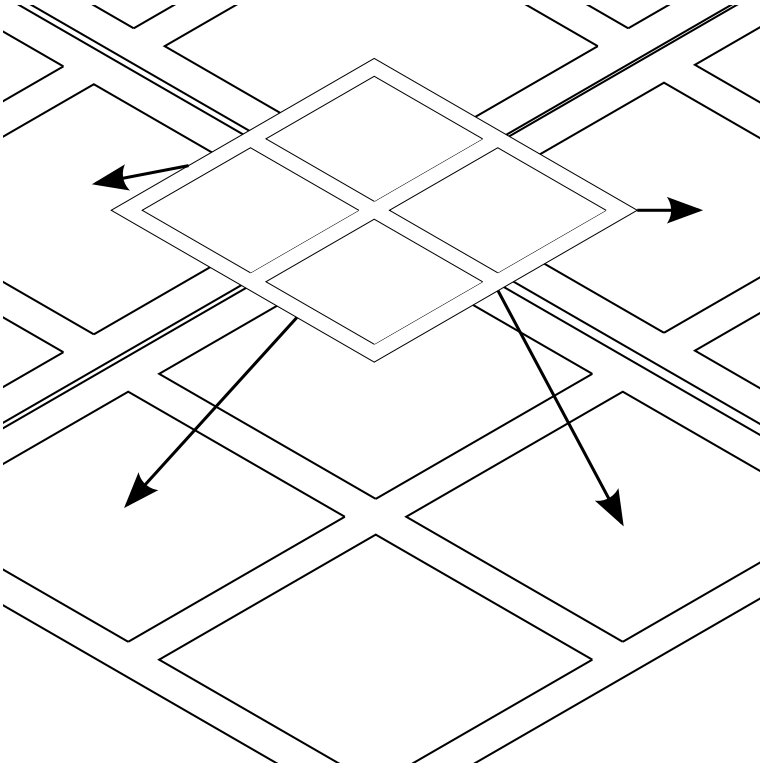

Figure 1: Illustration of neighbor relation in a Fully Threaded Tree (FTT) in two dimensions. The oct on the higher level points to the parent cells of the four neighbor octs. These cells reside at the level below.

are only stored at oct level, the memory usage per cell is smaller than for an ordinary unthreaded octree.

5.4 ADAPTIVE TIME STEP

An AMR simulation with point particles and hydrodynamics requires an independent time step on each level of refinement to be efficient (Teyssier [64]). In RAMSES this is achieved with a second order mid-point scheme, that reduces to second order leap frog for constant time steps. For the point particles, it involves a predictor step

$$v_p^{n+\frac{1}{2}} = v_p^n - \nabla\phi\frac{\Delta t^n}{2} \tag{4}$$

$$x^{n+1} = x_p^n + v_p^{n+\frac{1}{2}}\Delta t^n \tag{5}$$

followed by a corrector step

$$v_p^{n+1} = v_p^{n+\frac{1}{2}} - \nabla\phi^{n+1}\frac{\Delta t^n}{2} \tag{6}$$

where x_p and v_p are the position and velocity of particle p, and ϕ is the gravitational potential. The time step of a given level Δt_l is determined by the criterion that gives the smallest time step, out of the following:

1. A fraction of the smallest free fall time.
2. A fraction of the cell crossing time by point particles.
3. A fraction of the sound crossing time of a cell when including the motion of the gas (the Courant-Friederichs-Lewy stability condition).

Additionally the time steps at level $l+1$ must sum to the time step at level l for synchronization of the levels.

5.5 MAGNETOHYDRODYNAMICS

The magnetohydrodynamics (MHD) solver in RAMSES [18] uses a high order Godnunov scheme for the solution, and constrained transport (CT) to satisfy the solenoidal constraint. The set of equations it solves are the ideal MHD equations, here in conservative form:

$$\frac{\partial \rho}{\partial t} + \boldsymbol{\nabla} \cdot (\rho \boldsymbol{v}) = 0 \tag{7}$$

$$\frac{\partial \rho \boldsymbol{v}}{\partial t} + \boldsymbol{\nabla} \cdot (\rho \boldsymbol{v}\boldsymbol{v} - \boldsymbol{B}\boldsymbol{B}) + \boldsymbol{\nabla} P_{tot} = 0 \tag{8}$$

$$\frac{\partial E}{\partial t} = \boldsymbol{\nabla} \cdot \left[(E + P_{tot})\, \boldsymbol{v} - \boldsymbol{B}\, (\boldsymbol{B} \cdot \boldsymbol{v}) \right] = 0 \tag{9}$$

$$\frac{\partial \boldsymbol{B}}{\partial t} + \boldsymbol{\nabla} \cdot (\boldsymbol{v}\boldsymbol{B} - \boldsymbol{B}\boldsymbol{v}) = 0 \tag{10}$$

where ρ is the fluid density, $\boldsymbol{v}$ its velocity and $\boldsymbol{B}$ the magnetic field.

$$P_{tot} = P + \frac{\boldsymbol{B} \cdot \boldsymbol{B}}{2} \tag{11}$$

is the sum of thermal and magnetic pressure, and

$$E = \epsilon + \rho \frac{\boldsymbol{v} \cdot \boldsymbol{v}}{2} + \frac{\boldsymbol{B} \cdot \boldsymbol{B}}{2} \tag{12}$$

is the total energy density of the fluid, where ϵ denotes the internal energy density. The conserved quantities in the equations are mass (7), momentum (8), energy (9) and magnetic flux (10). A number of simplifying assumptions are made in deriving these, including that the pressure is isotropic and that the fluid is overall charge neutral. Perfect conductivity is also assumed, which gives these MHD equations the label 'ideal', and so resistivity is neglected. Derivations can be found [in e.g. 23, 37].

These equations assume the solenoidal constraint $\boldsymbol{\nabla} \cdot \boldsymbol{B} = 0$, which must be ensured to hold everywhere by the code. As mentioned this is done using constrained transport (Evans and Hawley [12]), which amounts to writing the induction equation

$$\frac{\partial \boldsymbol{B}}{\partial t} = \boldsymbol{\nabla} \times (\boldsymbol{v} \times \boldsymbol{B}) \tag{13}$$

in integral form

$$\frac{\partial \Phi_s}{\partial t} = \frac{\partial}{\partial t} \iint \boldsymbol{B} \cdot \mathrm{d}\boldsymbol{S} = \oint \boldsymbol{E} \cdot \mathrm{d}\boldsymbol{l}, \tag{14}$$

where $\boldsymbol{E} = \boldsymbol{v} \times \boldsymbol{B}$ is the electric field, and Φ_s the magnetic field integrated over the cell face surface. This approach requires a staggered mesh for the magnetic field, where the magnetic field is defined on the centers of the cell faces as an integral over the face, as opposed to the other state variables, which are cell centered values defined as an integral over the cell volume.

Following Londrillo and Del Zanna [42], the MHD equations are split into two sub-system: The 'Euler system' for the cell-centered variables, and the 'Induction system' for the induction equation. Both systems are solved with an adapted MUSCL-Hancock scheme [67, 68], which solves conservatively in a predictor step at $t^{n+\frac{1}{2}} = t^n + \frac{\Delta t}{2}$, and a subsequent corrector step at t^{n+1}.

5.6 MEMORY STRUCTURE

The state variables are stored in a vector over cells (index c) and variables (index i)

$$\boldsymbol{U}_{c,i} = \left(\rho, \rho v_x, \rho v_y, \rho v_z, E, B_{l,x}, B_{l,y}, B_{l,z}, \{\text{metals} \ldots\}, B_{r,x}, B_{r,y}, B_{r,z}\right)_i , \tag{15}$$

where ρ is density, v velocity, E total (kinetic+potential+magnetic) energy density, B_l and B_r the magnetic fields on the left and right face, and metals is a number of passive scalars, determined as a build parameter for 'make', that may be used to track the evolution of cloud and stellar metalicity, and chemical species. RAMSES may be built for a smaller number of dimensions than three, in which case the $\boldsymbol{U}$ vector is shorter. Due to the predictor-corrector nature of the time integration, both a $\boldsymbol{U}_{old}$ with the state before, and a $\boldsymbol{U}_{new}$ with the state after the last predictor step, needs to be maintained.

The octs are organized into a linked list, so apart from the parent- and child relations, a pointer is also stored to the previous and the next oct. This makes insertion and removal of octs a simple and cheap operation, albeit not vectorizable. Cells within an oct are not, as suggested by Khokhlov [34], kept contiguous in memory. Instead all upper left cells are blocked together in oct order, then all upper right cells and so on, such that the offset between cells from the same oct is the same as the number of octs that memory has been allocated for. In other words, the indices c_i, $i \in \{1, 2..8\}$ of all eight cells belonging to oct o are given by

$$c_i = o + (i - 1) n_{oct} \tag{16}$$

where n_{oct} is the maximal possible number of octs.

At every nth (typically~10'th) time step, the cells are reorganized along a Peano-Hilbert space filling curve (figure 2) as a part of load

balancing. This has the advantage that the mapping from one to three dimensions given by this curve preserves locality reasonably well. Therefore, when every process manages cells consecutive in Hilbert key, all these cells are guaranteed to be neighbors (or neighbors of neighbors etc.), and the surface-to-volume ratio of the domain is kept small, minimizing interprocess communication.

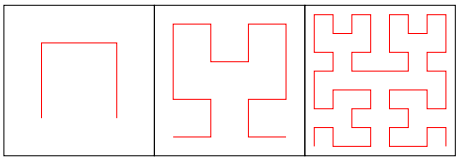

Figure 2: First three iterations of a Hilbert space-filling curve in two dimensions. Note that at each iteration, the figure of the previous iteration is repeated four times in 90° rotations. Adapted from Wikipedia.

5.7 SINK PARTICLES

A modification made in the Copenhagen group [51] is to include sink particles. Their purpose is to represent gravitationally bound and collapsing gas in a way that avoids the problem of the gravitational force going to infinity as radius goes to zero, in order to be able to model the formation of stars. Sink particles are created when the density is greater than a suitable threshold value. A number of additional constraints must be fulfilled, in order not to create spurious sink particles that do not represent collapsing gas.

1. The gravitational potential has a local minimum value.
2. The divergence of the velocity is negative, so the flow is contracting
3. No other sink particle is already present within a certain exclusion radius.

These criterion are similar to those in [14]. The sink particles are then modeled as massive point particles. Mass in excess of the density threshold, within a certain accretion radius from a sink, is accreted onto the closest sink particle, while conserving mass, linear- and angular momentum [49]. Massive stars may be followed until they explode as Type II supernovae.

5.8 HEATING AND COOLING

Another local modification is the addition of simple heating and cooling recepies, essentially implemented as table lookup. To cite from Vasileiadis et al. [69]:

"Heating and cooling is modeled as schematic heating by UV-photons Osterbrock [48] quenched in dense gas according to the recipe of Franco and Cox [16], and an optically thin cooling function consistent with Gnedin and Hollon [22]."

Part III

RADIATIVE TRANSFER

6 BACKGROUND

Radiative energy transport is a vast topic. This chapter aims to briefly lay down the basic concepts and outline some common approaches to solve the radiative transfer problem, especially those relevant in the context of hydrodynamical simulations.

6.1 INTENSITY AND MOMENTS OF THE RADIATION

The fundamental concept in astrophysical radiative transport is the intensity I_ν. A nice illustration of an idealized apparatus that would measure I_ν, is given in Castor [9], and the idea is repeated here in figure 3.

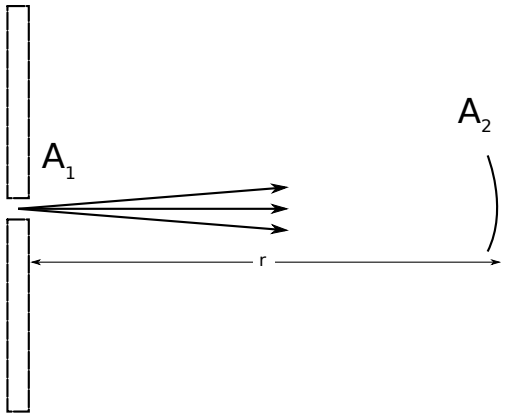

Figure 3: Illustration of idealized apparatus to define the specific intensity. Adapted from [9].

Imagine measuring the radiative energy at a point in space, by placing the opening of a pinhole camera there, with opening area A_1, open it for a time duration dt and catch the radiation with a detector of area A_2, a distance r into the box, which only detects radiation within a frequency bandwidth $d\nu$ of a frequency ν. The definition of intensity I_ν is that the energy collected by the detector is

$$dE = I_\nu \frac{A_1 A_2}{r^2} dt d\nu = I_\nu A_1 d\Omega dt d\nu, \tag{17}$$

where $d\Omega = \frac{A_2}{r^2}$ is the solid angle subtended by A_2 at the opening. Note that I_ν is a function of position $\boldsymbol{r}$ and orientation (given by the unit vector $\boldsymbol{n}$ perpendicular to the surface, with direction $A_1 \rightarrow A_2$) of our apparatus as well as frequency and time. That is,

$$I_\nu = I_\nu \left(\boldsymbol{r}, \boldsymbol{n}, \nu, t\right). \tag{18}$$

During the passage of the opening, the photons crossing A_1 occupy a volume of cA_1dt, where c is the speed of light. Dividing by this in equation 17 shows that the energy density per unit solid angle per unit frequency is $\frac{I_\nu}{c}$. Integrating over all directions gives the spectral energy density

$$E_\nu = \frac{1}{c}\int_{4\pi} I_\nu \, \mathrm{d}\Omega. \tag{19}$$

This is the zeroth angle moment of the intensity. The first moment is the vector flux

$$\boldsymbol{F}_\nu = \int_{4\pi} \boldsymbol{n} I_\nu \, \mathrm{d}\Omega, \tag{20}$$

which is indeed the energy flux field of the radiation. Since the radiation field consists of particles carrying momentum (photons), it can be described as a fluid. The ratio $\frac{\boldsymbol{F}_\nu}{E_\nu}$ gives the velocity field of this radiation fluid. The second angular moment of the radiation is defined by

$$\mathsf{P}_\nu = \frac{1}{c}\int_{4\pi} \boldsymbol{n}\boldsymbol{n} I_\nu \, \mathrm{d}\Omega, \tag{21}$$

which is equivalent to the pressure tensor of the radiation fluid. Further moments take the form of higher-rank tensors. [9]

6.2 THE EQUATION OF RADIATIVE TRANSPORT

Lets start with radiation in vacuum. Here the energy is conserved as the radiation moves through space, so

$$I_\nu\left(\boldsymbol{r} + \boldsymbol{n}c\Delta t, \boldsymbol{n}, t + \Delta t\right) = I_\nu\left(\boldsymbol{r}, \boldsymbol{n}, t\right). \tag{22}$$

Taylor-expansion around the point $\boldsymbol{r}$ and time t gives

$$I_\nu\left(\boldsymbol{r} + \boldsymbol{n}c\Delta t, \boldsymbol{n}, t + \Delta t\right) = I_\nu\left(\boldsymbol{r}, \boldsymbol{n}, t\right) + \left(\frac{\partial I_\nu}{\partial t} + c\boldsymbol{n}\cdot\nabla I_\nu\right)\frac{\Delta t}{2} + O\left(\Delta t^2\right). \tag{23}$$

Dropping the higher order terms and invoking energy conservation gives the time-dependent equation of radiative transfer in vacuum

$$\frac{1}{c}\frac{\partial I_\nu}{\partial t} + \boldsymbol{n}\cdot\nabla I_\nu = 0. \tag{24}$$

If the media absorbs radiation, the probability for a photon to be absorbed is proportional to the distance traveled cdt, so the specific intensity lost to absorption is

$$\mathrm{d}I_\nu = -\chi_\nu c I_\nu \mathrm{d}t, \tag{25}$$

where χ_ν is called the extinction coefficient. Similarly, if the medium emits radiation isotropically (e.g. by thermal radiation, or radiation caused by recombination of ionized gas), the accumulated extra intensity over $\mathrm{d}t$ is

$$\mathrm{d}I_\nu = j_\nu c \mathrm{d}t \tag{26}$$

where j_ν is the emission coefficient. Including these two terms in equation 24 gives

$$\frac{1}{c}\frac{\partial I_\nu}{\partial t} + \boldsymbol{n} \cdot \nabla I_\nu = j_\nu - \chi_\nu I_\nu. \tag{27}$$

The coefficients j_ν and χ_ν are local, and depend on the conditions of the medium, but they are taken to be independent of I_ν. Scattering of the radiation may also be taken into account, but its treatment is a bit more complicated, since the scattered radiation is redistributed in frequency and angle, and therefore depends on the whole radiation field at that point in space. In the case of isotropic scattering, an extra absorption term may be added to account for the radiation lost by scattering, and an extra emission term for the scattered radiation, proportional to the average intensity $J_\nu = \frac{E_\nu}{4\pi}$.

$$\frac{1}{c}\frac{\partial I_\nu}{\partial t} + \boldsymbol{n} \cdot \nabla I_\nu = j_\nu + \sigma_\nu J_\nu - (\chi_\nu + \sigma_\nu) I_\nu \tag{28}$$

Here σ_ν is called the scattering coefficient [9].

6.3 EQUATIONS OF THE ANGLE MOMENTS

Multiplying the equation of radiative transport by $\boldsymbol{n}$ and integrating over angles, equations for the moments of the radiation can be found.

$$\frac{\partial E_\nu}{\partial t} + \nabla \cdot \boldsymbol{F}_\nu = 4\pi j_\nu - k_\nu c E_\nu \tag{29}$$

$$\frac{1}{c}\frac{\partial \boldsymbol{F}_\nu}{\partial t} + c\nabla \cdot \mathsf{P}_\nu = -k_\nu \boldsymbol{F}_\nu. \tag{30}$$

Integrated over frequencies, equation 29 expresses radiative energy conservation, and equation 30 expresses conservation of momentum of the radiation.

6.4 FORMAL SOLUTION

The processes modeled by RAMSES are non-relativistic, which means the rate of change of intensity will everywhere be small compared to the speed of light. The left-most term in equation 28 can in this case be neglected. The dependence on J_ν in the scattering term means that a ray-tracing scheme cannot simply solve along each ray independently, since the rate of change of intensity in a given direction depends on the radiation field in all directions at that point in space. A common solution to this problem is to solve iteratively until the radiation field converges to a self-consistent solution, by a method called accelerated lambda iteration [see e.g. 53, 31]. This is much more computationally expensive than just solving for the radiation field once, and is left out of the scope of this work. The equation considered here is therefore the time-independent steady state equation of radiative transport,

$$\boldsymbol{n} \cdot \nabla I_\nu \left(\boldsymbol{r}, \boldsymbol{n}, \nu\right) = j_\nu - \chi_\nu I_\nu \left(\boldsymbol{r}, \boldsymbol{n}, \nu\right). \tag{31}$$

Along a straight line in a given direction $\boldsymbol{n}$, given by $\boldsymbol{r} = \boldsymbol{r}_0 + \boldsymbol{n}x$ (a ray), equation 31 reduces to

$$\frac{dI_\nu}{dx} = j_\nu - \chi_\nu I_\nu. \tag{32}$$

Changing parameter to the optical depth, defined by $\tau(x_0, x) = \int_{x_0}^{x} \chi_\nu dx$, and defining the source function $S_\nu = \frac{j_\nu}{\chi_\nu}$, equation 32 has the formal solution

$$I_\nu \left(\tau\right) = I_\nu(\tau_0) e^{\tau - \tau_0} + \int_{\tau_0}^{\tau} S\left(\tau'\right) e^{\tau - \tau'} d\tau'. \tag{33}$$

The solution along a ray is the central theme of this thesis. Most of what follows is dedicated to develop an efficient method for solving this equation for many rays through a domain simultaneously, and incorporating the effects of this solution in the adaptive mesh refinement code RAMSES.

7

NUMERICAL METHODS

7.1 FLUX LIMITED DIFFUSION

The diffusion approximation treats the radiation as an ideal fluid, with a small correction. It is accurate in the limit where the photon mean free path is small compared to the typical length scales L in the problem, $\chi_\nu L \gg 1$. If we rearrange equation 27 as follows

$$I_\nu = \frac{j_\nu}{\chi_\nu} - \frac{1}{\chi_\nu}\left(\frac{1}{c}\frac{\partial I_\nu}{\partial t} + \boldsymbol{n}\cdot\nabla I_\nu\right), \tag{34}$$

we see that if the extinction coefficient χ_ν is large, the intensity can be approximated to first order as

$$I_\nu^0 = \frac{j_\nu}{\chi_\nu}. \tag{35}$$

This corresponds to local thermodynamical equilibrium, where $\frac{j_\nu}{\chi_\nu} = B_\nu(T)$ is the Planck function. To go to second order, we may use this expression in the right hand term of equation 34 to get

$$I_\nu^1 = \frac{j_\nu}{\chi_\nu} - \frac{1}{\chi_\nu}\left(\frac{1}{c}\frac{\partial\left(\frac{j_\nu}{\chi_\nu}\right)}{\partial t} + \boldsymbol{n}\cdot\nabla\frac{j_\nu}{\chi_\nu}\right). \tag{36}$$

The angle moments of the radiation in this approximation are

$$E_\nu = \frac{4\pi}{c}\frac{j_\nu}{\chi_\nu} - \frac{4\pi}{\chi_\nu c^2}\frac{\partial}{\partial t}\left(\frac{j_\nu}{\chi_\nu}\right) \tag{37}$$

$$\boldsymbol{F}_\nu = -\frac{4\pi}{3\chi_\nu}\nabla\frac{j_\nu}{\chi_\nu} \tag{38}$$

$$\mathsf{P}_\nu = \frac{1}{3}\mathsf{I}\left[\frac{4\pi}{c}\frac{j_\nu}{\chi_\nu} - \frac{4\pi}{\chi_\nu c^2}\frac{\partial}{\partial t}\left(\frac{j_\nu}{\chi_\nu}\right)\right] = \frac{1}{3}E_\nu\mathsf{I} \tag{39}$$

where I is the unit matrix, and the last relation is Eddington's approximation.

In Eddington's approximation, the equation for the radiative momentum (equation 30) becomes

$$\frac{1}{c}\frac{\partial \boldsymbol{F}_\nu}{\partial t} + \frac{c}{3}\nabla E_\nu = -k_\nu \boldsymbol{F}_\nu \tag{40}$$

If the leftmost term is discarded, the radiative flux can be substituted into the equation for the radiative energy (equation 29) to obtain

$$\frac{\partial E_\nu}{\partial t} - \boldsymbol{\nabla}\cdot\left(\frac{c}{3k_\nu}\nabla E_\nu\right) = 4\pi j_\nu - k_\nu c E_\nu. \tag{41}$$

This is a parabolic equation like the one for diffusion of heat. In the process of discarding the rate of change of the flux, the finite propagation speed is lost. Ad hoc methods for preventing numerical simulations from giving unphysical results on account of this are referred to as flux limiters[9]. The flux limited diffusion scheme in the gray approximation has been already implemented in RAMSES, as described in Commercon et al. [10].

7.2 MOMENT METHODS

Flux limited diffusion is one instance of many attempts to recast the radiative transfer equation as a diffusion-like problem using the angular moments of the radiation, in order to obtain a more computationally manageable problem. These approaches go under the name of Moment Methods, since the series of moment equations converges on an accurate description of the photon distribution. In order to get a finite solvable set of equations, an ad-hoc closure equation must be imposed (for flux limited diffusion, equation 40). It is in the choice of closure that the moment methods differ. Two important classes of closures are P_N and M_N. The basic idea of P_N is to expand the photon distribution into a polynomial of degree N in the direction variable[17]. Often many moments are required to give reasonable results, especially for situations with beams and shadows, since the distribution function for a beam is the Dirac delta function. Also they may give unphysical results if the distribution function becomes negative. The M_N closure is constructed by choosing the distribution that minimizes entropy[11]. This ensures a finite speed of light and a positive photon distribution. They can however produce unphysical shocks if they are of odd order[15].

An M_1 moment method has also previously been implemented in RAMSES, under the name RAMSES-RT[57]. It uses an explicit advective solver, so the time step is restricted by the speed of light, and an approximation where the speed of light is reduced is required.

7.3 MONTE CARLO METHODS

The idea in Monte Carlo methods is to sample the source function, and place particles according to this distribution across the simulated domain in space, directions and frequencies. The particles are followed until they leave the domain or are destroyed. The number of particles found in a given zone in space is then a measure of the radiation energy density. The Doppler and aberration effects are easily taken into account, by transforming particle frequencies according to the local cell fluid velocity. Compton scattering may be included in full generality by sampling a relativistic electron velocity distribution and the Klein-Nishina cross section can be used to de-

termine if scattering occurs. In short, the main advantage of Monte Carlo methods is that it is often straight-forward to include otherwise complicated optical phenomena, with just some factor in difference in performance characteristics. The main disadvantage however, is that in order to obtain good statistics, a reasonable number of photons must be placed in each space×direction×frequency bin, which easily makes the method computationally expensive[9]. Monte Carlo radiative transfer has in fact been applied to ionization radiation hydrodynamics[27], and the radiative transfer calculation here dominates the computational time.

7.4 DETERMINISTIC RAY-TRACING

Ray tracing usually refers to methods that solve the one-dimensional equation of radiative transfer along a number of rays, using a number of direction bins. The result is the full radiation field in terms of the intensity discretized in space, angle and frequency. It is sometimes termed deterministic because it is not an approximation in the sense of the moment methods (given enough rays, convergence of the solution is guaranteed), and it doesn't carry the noise from the random photon sampling of the Monte Carlo methods. However, randomness and sampling of e.g. directions can be useful with ray tracing, and is also used a little in this work. A distinction between long- and short characteristics ray tracing is usually made. With short characteristics is meant that the radiative transfer equation is solved independently along short rays within each cell. These rays do not have to line up with the rays of neighboring cells, and the boundary intensities can be interpolated from the neighboring rays. Long characteristics rays, however, extend all the way to the boundary of the domain. If used cleverly, the solution along the whole long-characteristic may be used locally at all the intermediate points, to avoid repeating the calculation from the boundary at each point. This is equivalent to using short characteristics that do line up over cell boundaries. This is straightforward to do in Cartesian coordinates, but may be less so in other coordinates. This in general requires some interpolation scheme to line up cell values with the straight rays.

As noted by Nordlund [46], flux-limited diffusion can produce inaccurate results in regions with $\tau = 1$, where ray tracing with as little as 8 direction bins provides better accuracy. These are usually the regions of interest, since they represent optical surface layers that heat or cool the most. More generally, the moment methods have a degree of diffusivity that will lead to inaccurate solutions when sharp features like shadows are involved. One such situation could be a star in a fragmented proto-planetary disk. In addition, the elliptical equations that needs to be solved in a moment method are substantially more complicated than the solution along a ray, so the gain in remov-

ing the angular dependence from the equations may not substantially outweigh this cost.

Part IV

RAY-TRACING IN RAMSES

8

RADIATIVE TRANSFER IMPLEMENTATION IN RAMSES

In this chapter is a description of the method of solution of the radiative transfer problem and its coupling with the magnetohydrodynamics in the RAMSES code, as well as some relevant details on its implementation. The radiative transfer problem is solved using ray tracing, which has the advantage over moment methods that it is the direct explicit solution to the radiative transfer equation, and so will converge on the correct radiation field as resolution in space, angles and frequencies is increased. Using a pure Monte-Carlo method would not be feasible in terms of computation time, since a radiative transfer step must not take much longer to compute than a hydrodynamics step. Use of random sampling weighted by e.g. hydrodynamics state could however be a useful component, and the method here described is designed to allow for this kind of improvements. The crux of the method is that it solves the problem of using long characteristics on an adaptive mesh, without oversampling the domain, while reaping the benefits that comes from the fact that the solution along long characteristics rays are independent, and therefore can be perfectly parallelized and vectorized. In addition the choice of rays is flexible, so the same method can be used for point sources and diffuse radiation alike. A reservation must be taken on this point, though. At present it requires that the angular resolution is high enough to properly account for the geometric attenuation from point sources.

8.1 RADIATIVE TRANSFER WITH LONG CHARACTERISTICS IN A UNIGRID SETTING

Radiative transfer with long characteristics in a unigrid geometry is a well studied problem, and we started with this implementation to compare it to another MHD code, which has been in widespread use in Copenhagen, namely the the stellar MHD code (STAGGER) code. It also illustrates, in a simpler setting, how we build up a scalable parallel algorithm for radiative transfer in RAMSES.

We solve the radiative transfer equation along parallel rays distributed such that every cell is penetrated by one ray (figure 4). This is repeated for a set of directions. This constraint on the ray geometry is relaxed in the general implementation. If we follow the rays to a common plane (which we shall call the ray plane), each ray can be uniquely identified by its coordinates in this plane. A ray can then be fully described by four numbers: Its coordinates in the ray plane

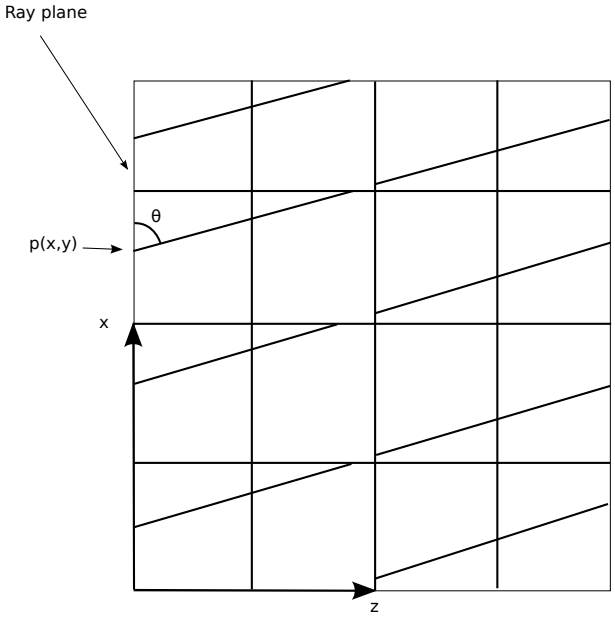

Figure 4: Geometry of rays in the unigrid implementation.

p and its two angle cosines with the ray plane. A ray that lies in the ray plane can of course not be described like this. This is easily circumvented by choosing the ray plane to be the coordinate-axis plane perpendicular to the axis i where the ray travels the fastest; $i = \max_{i \in \{x,y,z\}} \left(\frac{d\boldsymbol{r}}{di} \right)$. Therefore a fifth (3-bit) parameter is needed to specify the ray plane. This also ensures that the angles may not be at less than 45° with respect to the ray plane.

The cells are of course the ones where RAMSES does its calculations. They are distributed by RAMSES over a number of processes running on a number of CPUs. This decomposition of the domain of the hydro problem in terms of cells distributed across CPUs, we shall call the cell domain. The first step in the solution is to go through the cells in the cell domain, find out which ray crosses the cell, and interpolate the density and temperature of the cell to the ray. The next step is to assign to every process an equal fraction of the rays, and gather the interpolated values for these rays from the other processes. The radiative transfer problem has now been decomposed in terms of they rays, and we call this decomposition the ray domain. The radiative transfer problem is now solved along the rays in parallel. At the end, the calculated heating rates are transferred back to the cell domain, where they are interpolated back to the cell centers.

8.2 STRATEGY

The full AMR implementation of radiative transfer has quite a few steps and details, so I will begin by describing the overall strategy. Recall that the octs are ordered by a Peano-Hilbert space filling curve,

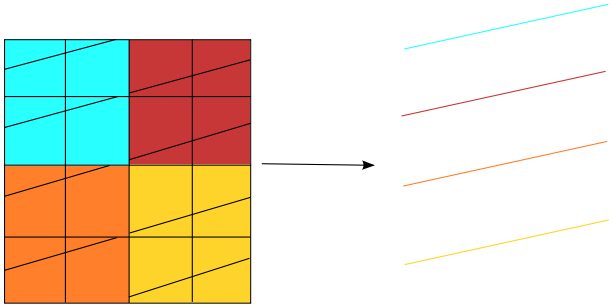

Figure 5: The cell domain and the ray domain. On the left is shown the cell domain, where blocks of cells belong to four CPUs marked by four different colors. On the right the problem is moved to the ray domain, where again the CPU to which the rays belong are marked by the colors.

and distributed out running on processes on a number of CPUs (sections 5.2 and 5.6). Consequently the physical space of the simulation is decomposed in coherent clumps, and neighboring octs tend to reside on the same process.

The essence of the strategy is to transfer the state relevant for radiative transfer from the cell domain to the ray domain, in which neighboring points along a ray are guaranteed to reside on the same process. In the ray domain it is straightforward to solve the radiative transfer equation in one dimension along each ray with e.g. Feautriers method. This calculation is vectorized, has optimal cache usage, and is therefore very fast. The result must then be transferred back to the cell domain, where it may be used to update the state, e.g. as a source term in the energy equation or by affecting ionization fractions.

The star formation simulations RAMSES is used for cover a huge range of length scales ($\approx 2^{30} \approx 10^9$). The gas structure is in general filamentary, and likewise covers a huge range of densities and temperatures. Consequently, is is hard to ensure that optical surfaces (regions over which the optical depth $\tau \approx 1$) are always resolved. It is therefore important that the solution is photon-conserving, since otherwise energy may not always be conserved. This is done by, instead of solving for the intensity, we solve for the solid-angle-frequency-specific heating rate Q_ν defined as $Q_\nu \equiv I_\nu - S_\nu$ (in units of intensity), where I is the intensity and S the source function.

Recall that the structure of a RAMSES time step is a predictor step, a recursive call with the next level of refinement, and a corrector step. This way predictor steps happens from the coarsest, all the way up to the finest level of refinement, after which corrector steps happen all the way back down to the coarsest level. Recall also that RAMSES has dynamic time stepping, in which an oct has twice as long time step as its parent. Since rays may pass through octs of several different levels of refinement, and radiative energy transfer is taken to be instantaneous, the ray tracing algorithm has to employ the shortest time

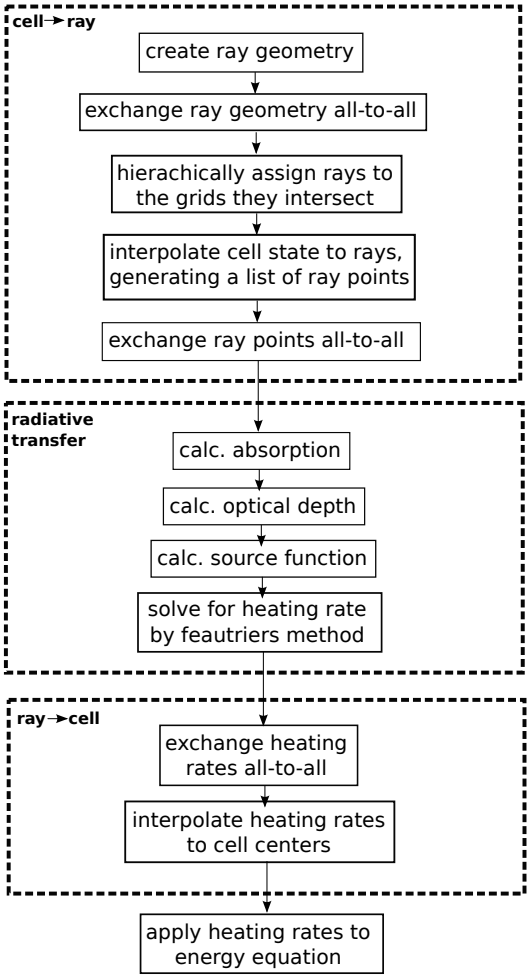

Figure 6: Flowchart of radiative transfer algorithm in full AMR. Every step happens simultaneously in every process. Steps in the middle dotted box happen in the ray domain, everything else in the cell domain. Once created, the ray domain exists alongside the cell domain until the end of the radiative transfer step.

step of the intersected cells. The simplest solution, which is the one employed here, is to calculate radiative transfer for all rays once per hydro time step, at the globally shortest time step. This corresponds to calling the radiative transfer routine only when the finest level of refinement is reached by the recursive time step routine.

The phases of radiative transfer are shown in figure 6. They are described in detail in the rest of this chapter.

8.3 MEMORY LAYOUT

The main data structures involved in going from the cell domain to the ray domain and back, are the following:

- A structure that holds arrays of parameters for affine lines describing a set of rays. It also holds the number of points along each ray.
- A structure that holds references to the rays which intersects a given cell. It consists of an array of ray indices, arrays of cached properties of the rays for efficiency, and the number of rays that intersects the cell in question.
- A structure that represents a single point along a ray. It holds the index of the ray, the level of refinement of the cell it happens to reside in, the dynamical variables required for solving the radiative transfer equation, and its one-dimensional position in ray space.
- An array of heating rates, corresponding to points along the rays.

There are also data structures to orchestrate the communication of ray points to the ray domain, and heating rates to the cell domain.

- A lookup table ($process\ id \rightarrow \#\ of\ ray\ points$) that specifies how many ray points should be sent to any given process from this process, when going from the cell- to the ray domain.
- A similar lookup table for offsets in memory of the beginning of blocks of ray points to send.
- Two similar lookup tables, describing the ray points this process expects to receive from every other process.

The contents of these lookup tables can be reused when going back from the ray domain to the cell domain, since the number of ray points in each domain does not change.

Finally, there are data structures involved in solving the radiative transfer equation in the ray domain.

- An array for each dynamical variable required for solving the radiative transfer equation (e.g. ρ, T).
- An array for each optical variable ($s, \chi, \tau, \Delta\tau$).
- An array with the distances between neighboring points along the rays.

Since each ray may encounter cells of different sizes on its way through the simulation box, the number of points along each ray may vary. Each value at each point along each ray are stored, as described above, in a flat array. The values are ordered primarily by the id of the ray they belong to, and secondarily by their position along that ray. Additionally three tables exist, which for a given ray id gives

- The number of points along the ray.
- The index of the first point along the ray.
- The index of the last point along the ray.

See appendix for more details.

8.4 RAY GEOMETRY

8.4.1 *Parameterizations*

Rays are parameterized as affine lines in 3D, by a direction vector and a displacement vector, $\mathbf{x} = \mathbf{r}t + \mathbf{p}$, where $\boldsymbol{r}$ is a unit vector. This form is the most convenient for the intersection test currently used, which tests intersection in a projection space, using what is known as Plücker coordinates. We note that an affine line may be described more compactly with displacement given by a 2-space coordinate on a fixed plane (hereafter called the ray plane), and direction given by two angle cosines with that plane. Here we choose as ray planes the planes perpendicular to the three coordinate axes, given by $x = 0$,$y = 0$ and $z = 0$ respectively. This parameterization thus consists of four real numbers and an integer in the range [1;3]. It is used as a compressed form, to reduce bandwidth usage when sending rays between processes. Another intersection algorithm is currently in development, for which the latter parameterization is more convenient. It is better suited for periodic rays, and is as efficient as the one using Plücker coordinates.

8.4.2 *Ray schemes*

Rays are initially created independently in every process. In principle rays may be created freely without restriction. Currently, two schemes are implemented. The first scheme simply mirrors the rays used by

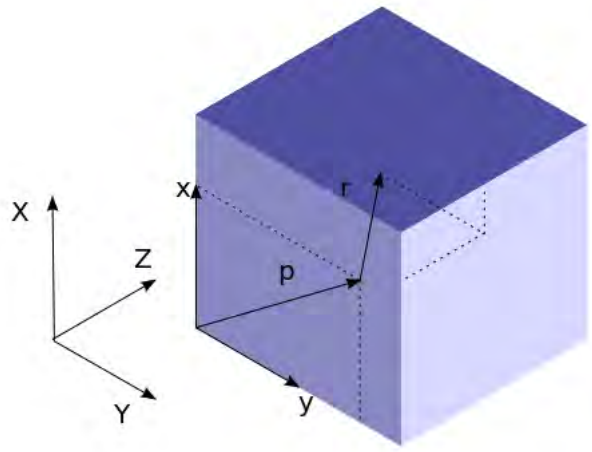

Figure 7: Ray parameterization. Rays are parameterized by a displacement vector $\boldsymbol{p}$ in the ray plane (x,y), and a direction unit vector $\boldsymbol{r}$. An additional parameter is used to fix which axis is perpendicular to the ray plane. X, Y and Z are the coordinate axes of the whole 3-dimensional domain, while x and y are the axes of the ray plane.

the unigrid solution. Its purpose is to enable direct comparison of the unigrid- and AMR implementations. The second scheme models a point source at the simulation box center, by making a number of randomly oriented rays, that all pass though the center.

An improvement on the point source scheme will be to use the HEALPix algorithm to produce rays with an angular distribution that divides the area of a sphere in a perfectly uniform manner, such that the intensity along each ray is exactly $I_{ray} = \frac{L_{source}}{N_{ray}}$.

There is a reason to create rays independently in every process, apart from parallelizing the work. It enables us to let the ray distribution depend on local conditions, such as the distribution of cells, density gradients, temperatures, optical depth gradients, and the position of star particles (point sources). For a reasonable coverage of rays when calculating diffuse radiation, for example, we may want some fixed number of rays to pass through a cell on average. A way to achieve this is to create a ray in a given cell with a probability proportional to the inverse cube root of the number of cells. Some regions will be of little interest, like regions enclosed by high optical depth regions, and cold, optically thin regions. In other regions it will be more important to have a good coverage of rays, such as regions with large temperature and optical depth gradients. The coverage may be tuned in this way, by using a probability weighted by such parameters, to determine which rays are created.

8.4.3 *Communication of ray geometry*

Once created locally, the rays must be communicated between processes, so that every process has access to at least all rays that intersects cells owned by it. In the current implementation, the simplest possible approach is used, where all rays are passed to all processes. This solution will not scale to a large number of rays and processes, since

1. The communication is not sparse.
2. Every process must consider intersection of its domain with all rays.

The communication side of things may be improved a lot, by using the fact that the cell domain owned by a process tend to be confined to a clump in physical space. An axis-aligned box enclosing this clump may be calculated from the location on the Peano-Hilbert space filling curve of the first and last oct owned by the process. The rays intersecting this box will be a superset of the rays actually intersecting cells in the domain of that process, but it will be a much better approximation on the correct number. Sending to a process only the rays that intersects its bounding box, the communication will be sparse. However the total number of intersection tests will on average remain roughly the same, because intersection tests are also required to determine which rays intersects each domain.

8.5 INTERSECTION AND CELL-RAY PAIRING

8.5.1 *Intersection test using Plücker coordinates*

Plücker coordinates parameterize a directed line in a manner convenient for describing relations such as incidence and relative orientation of two such lines. A directed line is the line through two points $\boldsymbol{a}$ and $\boldsymbol{b}$, with direction from $\boldsymbol{a}$ to $\boldsymbol{b}$, so the definition is the same as for a ray. Consider two directed lines, l from $\boldsymbol{a}$ to $\boldsymbol{b}$, and r from $\boldsymbol{o}$ to $\boldsymbol{p}$, with direction vector $\boldsymbol{d} = \boldsymbol{p} - \boldsymbol{o}$. The normal vector to the plane that contains $\boldsymbol{a}$, $\boldsymbol{b}$ and $\boldsymbol{o}$ is $\boldsymbol{n} = (\boldsymbol{a} - \boldsymbol{o}) \times (\boldsymbol{b} - \boldsymbol{o})$. Define the side relation $side(r, l) = -\boldsymbol{d} \cdot \boldsymbol{n}$. It determines the relative orientation, in that it is positive if $\boldsymbol{d}$ points above the plane, negative if below, and zero if it lies in the plane. Expanding the side relation and rearranging, we get

$$
\begin{aligned}
side(r,l) = -((\boldsymbol{p}_1 - \boldsymbol{o}_1)(\boldsymbol{a}_2\boldsymbol{b}_3 - \boldsymbol{a}_3\boldsymbol{b}_2) + \\
(\boldsymbol{p}_2 - \boldsymbol{o}_2)(\boldsymbol{a}_3\boldsymbol{b}_1 - \boldsymbol{a}_1\boldsymbol{b}_3) + \\
(\boldsymbol{p}_3 - \boldsymbol{o}_3)(\boldsymbol{a}_1\boldsymbol{b}_2 - \boldsymbol{a}_2\boldsymbol{b}_1) + \\
(\boldsymbol{b}_2 - \boldsymbol{a}_2)(\boldsymbol{p}_1\boldsymbol{o}_3 - \boldsymbol{p}_3\boldsymbol{o}_1) + \\
(\boldsymbol{b}_3 - \boldsymbol{a}_3)(\boldsymbol{p}_2\boldsymbol{o}_1 - \boldsymbol{p}_1\boldsymbol{o}_2) + \\
(\boldsymbol{b}_1 - \boldsymbol{a}_1)(\boldsymbol{p}_3\boldsymbol{o}_2 - \boldsymbol{p}_2\boldsymbol{o}_3)) \\
= R_2L_3 + R_5L_1 + R_4L_0 + R_1L_5 + R_0L_4 + R_3L_2 \quad (42)
\end{aligned}
$$

Notice that each L_i is a determinant of a 2×2 minor of the matrix

$$
\begin{pmatrix} a_1 & a_2 & a_3 & 1 \\ b_1 & b_2 & b_3 & 1 \end{pmatrix}
$$

and the numbering of the indices of L follows from systematically combining columns from left to right. The same goes for R. L and R are Plücker coordinates for the lines l and r.

To test for intersection of a ray r with a cell with center $\mathbf{c}$, first we change coordinates $\boldsymbol{x} \rightarrow \boldsymbol{x} - \boldsymbol{o}$, so the ray origin coincide with the coordinate system origin. We note that to intersect the cell, the ray must intersect one of three faces, depending on the signs of the components of the ray direction vector. If e.g. all three components are negative, either the top, front or right face must be intersected. Lets call the three bit value encoding the sign of the three components the ray's class. Given a ray class, we can write up the six edges that constitute the silhouette of the three faces using Plücker coordinates. We can then use the side relation six times to determine if the ray is on the right side of each edge for intersection to occur. [43, 63] A performance measurement of the Plücker-based intersection implementation is given in figure 8. As can be seen, the intersection test costs 8-16 nsecs per ray, meaning that we can calculate at least several thousand intersections per cell in a time step, given that the typical RAMSES time step costs $50 - 200\mu s$ per cell. This is crucial for our radiative transfer scheme. In every time step we need to know exactly which rays cross inside a given cell, and therefore have to do many tests per cell per time step for a realistic ray geometry with 20-100 rays per cell.

8.5.2 *Calculating distance to intersections*

The distance calculation is simple geometry. Consider a ray with equation $\boldsymbol{x} = \boldsymbol{r}t + \boldsymbol{p}$ and a cell with center $\boldsymbol{c}$ and side length dx. The cell is aligned with the axis, so the planes containing the faces can be indicated with just their coordinate along the axis perpendicular to them. Lets call the plane first intersected by the ray, the plane of the

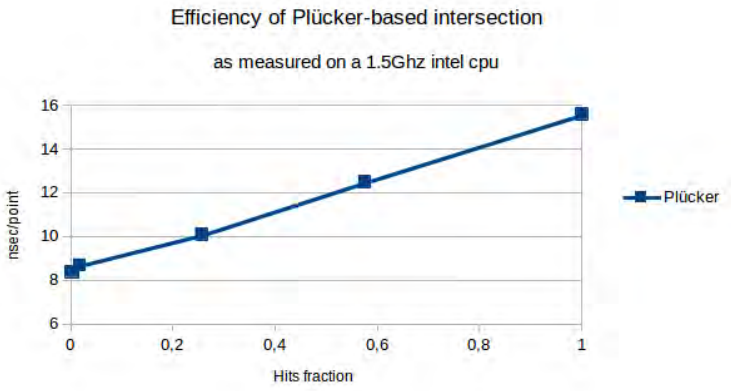

Figure 8: Measured efficiency of the Plücker-based intersection implementation

front face f, and the plane last intersected the plane of the back face b. Their coordinates are

$$f_i = \boldsymbol{c}_i + sign(\boldsymbol{r}_i)\frac{dx}{2} \tag{43}$$

$$b_i = \boldsymbol{c}_i - sign(\boldsymbol{r}_i)\frac{dx}{2} \tag{44}$$

The distance to the intersected front face along the ray is the greatest distance to a front face plane, so

$$d_f = \max_i \left(\frac{f_i - \boldsymbol{p}_i}{\boldsymbol{r}_i}\right) \tag{45}$$

and similarly

$$d_b = \min_i \frac{b_i - \boldsymbol{p}_i}{\boldsymbol{r}_i}. \tag{46}$$

The coordinate of the ray point $c_{rp} \equiv \frac{d_f + d_b}{2}$ is defined as the midpoint between the two intersections.

8.5.3 *Periodic rays*

In many cases, including the star formation simulations done by the group, periodic boundary conditions are used. Usually the purpose is to get the effect of having the simulated domain surrounded by a similar environment, e.g. when simulating part of a Giant Molecular Cloud (GMC). Periodic ray tracing, however, means that a single ray can essentially cover the whole domain if it has the right direction (figure 9).

This is not an easy problem, if not for other reasons, then because it is not clear how to efficiently parallelize the solution. Here the simpler formulation that a ray is only periodic on the two axes with

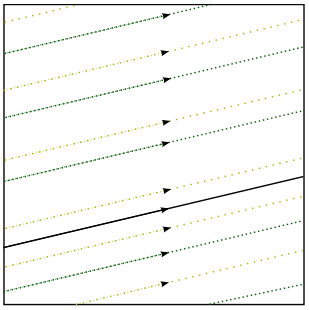

Figure 9: A fully periodic ray.

smallest component in its direction vector is adopted. A ray may in other words not cross the length of the simulation box more than once (figure 10).

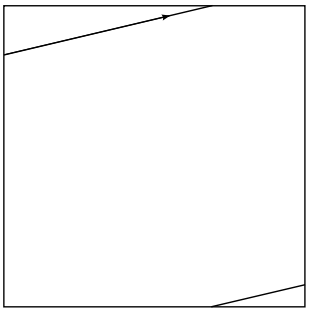

Figure 10: A simpler approach. Here is used a periodic ray that only crosses the box length once.

This will not give periodicity for the radiation hitting a point close to a boundary within 45° in each direction towards the boundary. A strictly speaking incorrect, but given the purpose of having periodic boundaries, still useful, way to get around this, is to identify the end points of the ray when solving along it.

Periodicity as described above is achieved in the following way. Consider a ray $\boldsymbol{x} = \boldsymbol{r}t + \boldsymbol{p}$ and a cell with center $\boldsymbol{c}$ and side length dx. The cell may be transported along the direction of the ray without changing whether it is intersected. Transport the cell to the ray plane $\boldsymbol{c}' = \frac{\boldsymbol{c}}{\boldsymbol{r}}$, and find the distance vector $\boldsymbol{d}$ to the rays origin, which lay in the ray plane.

$$\boldsymbol{d} = \boldsymbol{c}' - \boldsymbol{p}$$

Given that the largest possible cell size is $dx_{max} = \frac{l_{box}}{2}$, if on either axis $i \in \{x, y\}$, $|\boldsymbol{d}_i| > \frac{l_{box}}{2}$, then the ray cannot hit the cell. But it might still hit due to periodicity, in which case we need to take $\boldsymbol{p}_i \rightarrow \boldsymbol{p}'_i = -sign(\boldsymbol{d}_i) l_{box}$. We can now use the above intersection test and distance calculation with the original cell center $\boldsymbol{c}$ and the translated ray offset $\boldsymbol{p}'$.

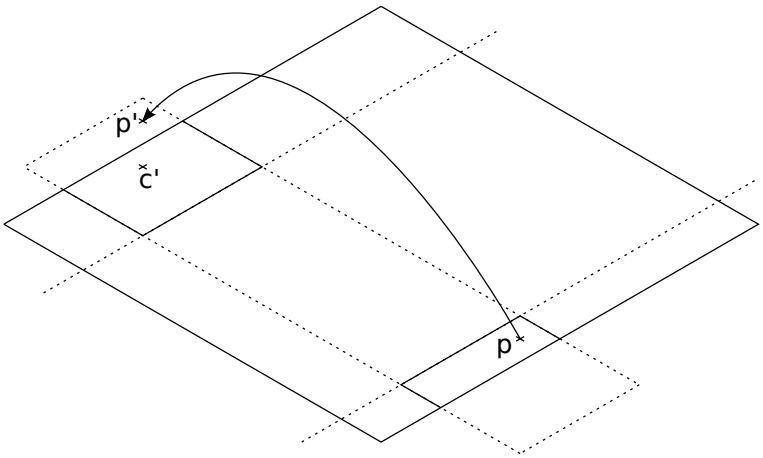

Figure 11: Translation of ray origin

The calculation can be simplified by always taking

$$\boldsymbol{p}_i \rightarrow \boldsymbol{p}'_i = \boldsymbol{p}_i + \text{round}(\frac{\boldsymbol{c}'_i - \boldsymbol{p}_i}{l_{box}})l_{box} \tag{47}$$

. A further optimization is to reject the ray before using the Plücker-based intersection test, if on any ray-plane axis i,$|\boldsymbol{c}'_i - \boldsymbol{p}'_i| > dx$, since this is an upper bound on the distance as no rays have an angle of more than 45° with the primary axis in each direction.

8.5.4 *Improved time complexity with hierarchical filtering*

The Plücker-based intersection test takes up to $t_{int} = 15$ nsec per point when run on a single 1.5GHZ Intel processor, depending on the fraction of rays that intersects the cell (figure 8). To put this number into context, lets look briefly at the kind of simulations that was the initial motivation for developing the scheme. A typical global star formation run simulates a cube with side length between 3 and 40 parsec, a coarse grid of 128^3-512^3 cells, and a maximum octree depth of 13-17 levels of refinement. This gives a spatial resolution of 10-100 AU on the finest grid. The temporal resolution is around 0.1-10 years. The total number of cells of course depends on the specific details of what happens to be going on in the simulation. A concrete example is one run with 17 levels of refinement, and in total $2.2 * 10^8$ cells (the distribution is shown in figure 12). It was run with 512 MPI processes and 32 OpenMP threads on a BlueGene architecture machine, which performs roughly equivalent to 512 MPI $\times$ 4 OpenMP threads on an x86 architecture, like the one where t_{int} was measured. [25]. Lets adopt an absolute lower limit of one ray crossing each cell, to give a reasonable coverage of rays. Intersection testing is embarrassingly parallel, and may easily be OpenMP parallelized. The average number of intersection tests per thread is in this case $n_{int} = \frac{n_{cell}^2}{n_{thread}} = \frac{\left(2.2*10^8\right)^2}{512*4} \approx 2.4 * 10^{13}$,

Algorithm 8.1 Hierarchical intersection

Listing 1 Hierachical intersection

```
minlevel <- lowest refinement level with active octs
maxlevel <- highest refinement level
candidate-rays <- all rays

for level from minlevel to maxlevel
  for icpu from 1 to ncpu
    if icpu is this cpu then
      octs <- active-octs(level)
    else
      octs <- ghost-octs(icpu, level)

    for oct in octs
      if ilevel > minlevel then
        candidate-rays <- intersecting-rays(father-cell(oct))
      for each cell in oct
        c <- center-coordinate(icell)
        intersecting-rays(icell) <- intersections(c, candidate-
            rays)
```

which takes around $3.5 * 10^5$ sec. This should be compared with a total coarse time step in this simulation of around 60 sec, and is clearly unreasonably large.

A standard approach (e.g. [21, 13]) to intersection testing, is to partition space into an octree, and use the fact that a ray that does not intersect a given oct, cannot intersect any of its child cells . We can then walk the tree recursively, each time on average cutting the number of rays in one-eighth. RAMSES happens conveniently to already partition space in an octree. It does however not guarantee that a given oct's parent cell is located in the same process, but it does guarantee that a copy of the cell is available as a ghost cell. The approach chosen here is a non-recursive version of the tree walk, where, in a given process, we start at the lowest refinement level that contain any octs belonging to the process. All active cells and ghost cells are iterated through, testing intersection, and storing the intersections in a table with the cell id as key. When proceeding to the next level of refinement, at each cell the intersections of its parent is retrieved from the table as candidates for intersection. A pseudo-code listing is given in algorithm 8.1.

At the lowest level of refinement, all the rays will be tested. With a uniform distribution of rays in space, and an equal distribution of cells over processes, where a number α of rays hit each cell, on average $n_{ray,local} = \frac{\alpha}{n_{cpu}} n_{cell}^{\frac{2}{3}}$ rays will hit at least one oct on the lowest level. Let $n_{ray,i}$ denote the number of candidate rays at level i, and

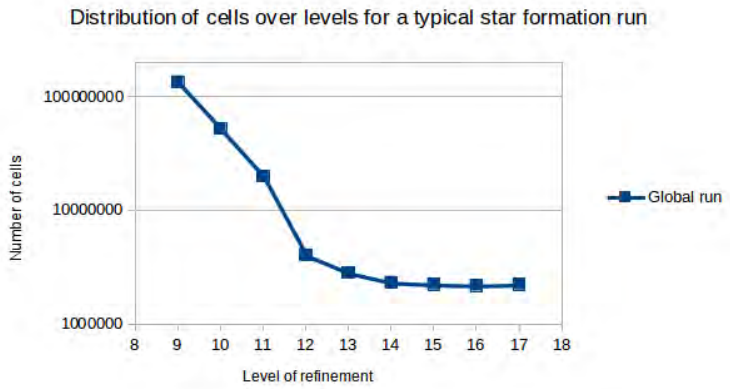

Figure 12: Distribution of cells over levels for a typical global star formation run

$n_{cell,i}$ be the number of cells at level i. The number of intersection tests at level i is $n_{int,i} = n_{ray,i}n_{cell,i}$, and the number of intersection tests at the subsequent level is $n_{int,i+1} = 8n_{cell,i}\frac{n_{ray,i}}{8} = n_{int,i}$, assuming none of the cells at level i were leaf cells. The total number of intersection tests with the hierarchical algorithm is therefore $n_{int} = n_{ray,global} + n_{ray,local}(l_{max} - l_{min} - 1)$ in the unigrid case, where l_{max} and l_{min} are the highest and lowest levels of refinement with active cells in a given process. With mesh refinement, there will be leaf cells on levels lower than l_{max}, and N_{int} will in general be smaller. By assuming that no intermediate layer cells are leaf cells, we can find an upper bound on number of intersection tests in the global simulation discussed above. We get $n_{int} = 2.2 * 10^8 + \frac{(2.2*10^8)^{\frac{2}{3}}}{512*4} * 17 = 2.2 * 10^7$ intersections, which takes around 3.3 seconds. This number should very much be taken with a grain of salt since the tree walk will incur some overhead, it is assumes that RAMSES distributes cells perfectly over processes, the number is sensitive to the ray geometry and spatial distribution of cells, etc. It does however tell us that intersection testing could become a performance bottleneck, but probably not prohibitively so.

8.6 INTERPOLATION AND RAY POINT CREATION

The previous step created a table where, given a cell, the intersecting rays can be looked up. Furthermore, the indexing used is the same that RAMSES uses for its Fully Threaded Tree (FTT). The interpolation of cell state to the rays simply consist of looping over all cells, looking up the intersecting rays, and for each ray interpolating the cell-centered state variables to the midpoint of the ray within the cell. A second order conserving Triangular Shaped Cloud (Triangular Shaped Cloud (interpolation) (TSC)) interpolation (see e.g. Hockney and Eastwood [29]) is used.

When looping over cells, only leafs should generate ray points, so non-leaf cells are skipped. TSC interpolation has the advantage of being second order, so it ensures a continuous first derivative. Recall that in the fully threaded tree, only pointers to the six neighboring oct's parent cells are maintained. An alternative to TSC would be to use linear so-called Cloud in Cell (interpolation) (CIC) interpolation between the 8 nearest cell neighbors to a point. But still a neighbor in a diagonal direction may be required, so up to two pointers must be followed. In addition to this, either some logic must be set up to select the correct neighbor, or all 27 neighbor points could be selected and the unneeded masked out. Therefore we get all 27 neighbor cells required for TSC interpolation without considerably higher cost than if linear interpolation was used. Before interpolation begins, the total number of cell-ray intersections, which is also the total number of ray points over all rays, is known, so a ray point array may be pre-allocated. During the interpolation loop over cells, the ray points are added to the array in the order the cells and rays are encountered, so the order will be more or less random. This is not a problem, because the index of the ray to which the point belongs, and the distance along the ray to the intersection with the cell back face are stored on the ray. Also the index of the intersected cell is stored in a table with the ray point index as key. These may later be used to sort the ray points into the required order.

The FTT ensures that there can be a difference of at most one level of refinement between two adjacent grids. Where a ray crosses the boundary to a grid of lower refinement however, not all 27 points required for TSC are available, and missing points should be interpolated from the parent grid.

8.7 COMMUNICATION OF RAY POINTS

When solving the radiative transfer equation, each ray is solved independently. Therefore, when constructing the ray domain, it does not matter which ray will belong to which process, as long as the rays are distributed evenly. The arbitrarily chosen convention is that rays will be assigned to processes in the order of their index, in such a way that every process will have the same number of rays if possible, but a process with lower rank may have one more ray than a process with higher rank. The convention gives rise to a relation $proc(i_{ray})$ from ray index to process rank, and $ray(i_{proc})$ from process rank to the last ray index that belongs to it. Let $f = n_{ray} \backslash n_{cpu}$ and $r = n_{ray} \% n_{cpu}$, where $\backslash$

denotes integer division (division followed by flooring), and % is the modulus operator. Then

$$proc(i_{ray}) = \begin{cases} (i_{ray} - 1) \setminus (f+1) + 1 & if\ i_{ray} \leq r(f+1) \\ r + (i_{ray} - r(f+1) - 1) \setminus f + 1 & otherwise \end{cases} \tag{48}$$

and

$$ray(i_{proc}) = (f+1)\min(i_{proc}, r) + f * \max(i_{proc} - r, 0). \tag{49}$$

All ray points that will be sent to the same process must lie in a contiguous block in memory. With the chosen domain decomposition, this is achieved simply by sorting the ray points by the index of the ray they belong to, using the Quicksort algorithm. The original order is saved for later in the global array order_cell. The number of points to send to each other process is found using the above relations. This number is communicated all-to-all in order so the process knows how many ray points it will receive from the other processes. The ray points themselves can now be communicated all-to-all. Finally the received ray points are sorted into the order required for solving along the rays, namely primarily after ray index, and secondarily after depth along the ray. Again the order before the sort is saved for later. This time the global array is named order_ray. We are now in the ray domain.

8.8 SOLVING THE RADIATIVE TRANSFER EQUATION

The absorption coefficient α_ν and source function S_ν are first calculated from the gas state at the ray points. This may be done in an arbitrarily complex and realistic manner, taking into account the line spectra due to the chemistry of the matter. There is a wide range of densities and temperatures to cover, and computational time will generally set the limit. A lookup table generated using more complex and time-consuming models is a good option. An implementation using the non-equilibrium chemical network solver KROME[24] is in development. Currently, only a gray Local Thermodynamical Equilibrium (LTE) model with constant opacity, and a simple photoionization model with HI and HII, have been implemented.

Disregarding ionization, which is covered in Chapter 9, the quantity of interest is the power deposited in each point due to radiation, which for time-independent radiative transport is equivalent to the divergence of the energy flux $\boldsymbol{\nabla} \cdot \boldsymbol{F}$. By the definition of intensity, the radiative energy flux is

$$\boldsymbol{F} = \oint_{4\pi} d\Omega \int_0^\infty d\nu \boldsymbol{n} I_\nu(\boldsymbol{n}) \tag{50}$$

with divergence

$$\boldsymbol{\nabla} \cdot \boldsymbol{F} = \oint_{4\pi} d\Omega \int_0^\infty d\nu \boldsymbol{n} \cdot \boldsymbol{\nabla} I_\nu (\boldsymbol{n}) . \tag{51}$$

In the non-relativistic limit the radiative transfer equation can be stated as

$$\boldsymbol{n} \cdot \boldsymbol{\nabla} I_\nu = -\chi_\nu (I_\nu - S_\nu) , \tag{52}$$

where χ is the optical extinction, and S is the source function[28]. Consequently we need to calculate the quantity $Q_\nu = I_\nu - S_\nu$. The integral over solid angle is done explicitly by the radiative transfer implementation, by summing the solution of rays with different directions. The integral over frequency is similarly done as a weighted sum over solutions for a set of frequency bins.

In the following, the solution along a single ray in a single frequency bin is considered, so the frequency index is dropped. When solving for Q, at large optical depth we have $I \approx S$, which can lead to round off errors dominating when taking the difference. A photon-conserving formulation is therefore necessary, which can be obtained by solving for Q directly at each point along the ray, instead of solving for I.

For this, Feautriers method is used, as it is has proven to be the fastest and most robust method. The onset of Feautriers method is to split the intensity into I^+ in direction of increasing τ, and I^- in the opposite direction, and define $P = \frac{I^+ + I^-}{2}$ and $R = \frac{I^- - I^+}{2}$. Then $\frac{dI^\pm}{d\tau} = \pm (S - I^\pm)$, giving

$$\frac{dP}{d\tau} = \frac{S - I^+ + I^- - S}{2} = R \tag{53}$$

and

$$\frac{dR}{d\tau} = \frac{I^+ - S + I^- - S}{2} = P - S \tag{54}$$

so

$$\frac{dP^2}{d\tau^2} = P - S. \tag{55}$$

To ensure photon conservation, we instead consider $Q = P - S$, for which we have

$$\frac{d^2 Q}{d\tau^2} = \frac{d^2 P}{d\tau^2} - \frac{d^2 S}{d\tau^2} = Q - \frac{d^2 S}{d\tau^2}. \tag{56}$$

This equation can be solved using a triliniear matrix with coefficients

$$a_{i,i-1} = \frac{1}{\Delta\tau_{i-\frac{1}{2}} \Delta\tau_i} \tag{57}$$

$$a_{i,i+1} = \frac{1}{\Delta\tau_{i+\frac{1}{2}} \Delta\tau_i} \tag{58}$$

$$a_{i,i} = 1 - a_{i,i-1} - a_{i,i+1} \tag{59}$$

where

$$\Delta\tau_{i\pm\frac{1}{2}} = \pm\left(\tau_{i\pm1} - \tau_i\right) \tag{60}$$

$$\Delta\tau_i = \frac{\Delta\tau_{i-\frac{1}{2}} + \Delta\tau_{i+\frac{1}{2}}}{2}. \tag{61}$$

Once Q_ν is known, we must multiply with the extinction $\chi_\nu = \rho\alpha_\nu$, and sum over frequencies to get the heating rate contribution $Q_{rad} = \sum_\nu (\chi_\nu Q_\nu)$ from each ray point (equation 52).

8.9 INTERPOLATION OF HEATING RATES TO GRID

In order to send the heating rate contributions back to the cell domain, they are placed in the same order as the ray points were received in, which was stored in the array order_ray (section 8.7). The same offsets and counts that were computed at that point can then be reused for the all-to-all communication.

Once received, the heating rate contributions are placed in the same order as the ray points were originally created in during the TSC interpolation from cell centers to ray points, which followed the order of the cell indices. It was stored in the array order_cell (section 8.7). Recall that the index of the cell to which each ray point belongs were stored in a table (section 8.6). The cells are looped over in the same order as before, and at each cell, heating rate contributions are taken off the array as long as the recorded cell index of the next ray point matches the current cell index. TSC interpolation is used again, this time in reverse, so each of the 27 neighboring cells get their part of the heating rate contribution from the ray point according to the interpolation weights.

8.10 UPDATING THE ENERGY EQUATION

Recall that RAMSES uses an energy-conserving formulation, in which it is the total energy density that is stored. The radiative energy transport is a source term in the equation stating energy conservation, so the total energy density is updated simply by adding to it the time-integrated heating rate over the time step.

9 IONIZATION

The implementation of photo-ionization into RAMSES was done in close collaboration with my supervisor Troels Haugbølle. Most of the implementation work regarding ionization was done by him, and most of the integration with the ray tracing was done by myself.

For transport of ionizing radiation an explicitly photon conserving method proposed by Mellema et al. [45] is used. With an ordinary finite-differencing method, in order to obtain accurate results the spatial resolution must be high enough that individual cells are optically thin, and the time step must be short enough that the ionization front does not cross more than one cell in a time step. By conserving photons explicitly, and by using time-averaged quantities, the requirements on the resolution in optical depth and time can be greatly reduced. We (like Mellema et al. [45]) model a point source embedded in a pure hydrogen medium, and take into account photo-ionizations, collisional ionizations and recombination.

Consider a point source with spectral luminosity L_ν. At a distance r from the source, photons arrive optically and geometrically attenuated, so the flux of photons is given by

$$F_\nu(r) = \frac{1}{4\pi r^2}\frac{L_\nu e^{-\tau_\nu(r)}}{h\nu}.$$

The total flux of arriving hydrogen-ionizing photons is

$$F(r) = \int_{\nu_{th}}^{\infty} F_\nu(r)d\nu,$$

where $h\nu_{th} = 13.6$ eV is the energy threshold for ionization of hydrogen. The corresponding local ionization rate for hydrogen atoms with a cross-section σ_ν is

$$\Gamma_{local}(r) = \int_{\nu_{th}}^{\infty} F_\nu \sigma_\nu d\nu.$$

The optical depth is defined as $\tau_\nu = \sigma_\nu N_{HI}$, where N_{HI} is the column density of neutral hydrogen.

Consider a spherical shell of width Δr about the distance r. Taking Γ to be constant through the shell is incorrect if the shell is optically thick radially, essentially because it neglects the effect that self-shielding gives a lower ionization rate at the outer edge of the shell, and the number of absorptions will not equal the number of ionizations. Mellema et al. [45] follows Abel et al. [1] and instead forces the ionization rate to equal the absorption. Let $\dot{N}(r - \frac{\Delta r}{2})$ be the number of ionizing photons arriving at the shell, and let $\dot{N}(r + \frac{\Delta r}{2})$ be the

number of ionizing photons leaving the shell. The ionization rate is then

$$\Gamma = \frac{\dot{N}(r - \frac{\Delta r}{2}) - \dot{N}(r + \frac{\Delta r}{2})}{n_{HI} V_{shell}} \tag{62}$$

in units of ionizations per neutral hydrogen atom per unit of time, with

$$V_{shell} = \frac{4\pi}{3}\left[\left(r + \frac{\Delta r}{2}\right)^3 - \left(r - \frac{\Delta r}{2}\right)^3\right].$$

Define τ_ν as the optical depth from the source to $r - \frac{\Delta r}{2}$, and $\Delta\tau_\nu$ as the optical depth between $r - \frac{\Delta r}{2}$ and $r + \frac{\Delta r}{2}$. Then equation 62 can be rewritten as

$$\Gamma = \int_{\nu_{th}}^{\infty} \frac{L_\nu e^{-\tau_\nu} - L_\nu e^{-(\tau_\nu + \Delta\tau_\nu)}}{h\nu * n_{HI} V_{shell}} d\nu = \int_{\nu_{th}}^{\infty} \frac{L_\nu e^{-\tau_\nu}}{h\nu} \frac{1 - e^{-\Delta\tau_\nu}}{n_{HI} V_{shell}} d\nu. \tag{63}$$

Lets now consider temporal discretization. Over a time step Δt, the photo-ionization rate Γ causes the density of neutral hydrogen to change, changing in turn the optical depth. The photo-ionization rate is really the average rate over Δt, so using the time-averaged value for $\Delta\tau_\nu$ in equation 63 should relax the requirement on the time step. In the pure hydrogen case, the evolution of the ionized fraction can be written

$$\frac{dx}{dt} = (1 - x)\left(\Gamma + n_e C_H\right) - x n_e \alpha_H, \tag{64}$$

where x is the fraction of ionized atoms, n_e is the electron density, C_H is the collisional excitation coefficient, and α_H is the recombination coefficient. The On-The-Spot approximation is used (Osterbrock [47]), where recombinations to the ground state are taken to be locally reabsorbed. The temperature dependence of the recombination coefficient consistent with this assumption is

$$\alpha_H = 2.59 \times 10^{-13} \left(\frac{T}{10^4 \mathrm{K}}\right)^{-0.7} \mathrm{cm}^3 \mathrm{s}^{-1}.$$

If we take Γ, n_e, C_H and α_H to be constant, then equation 64 has the analytical solution

$$x(t) = x_{eq} + \left(x_0 - x_{eq}\right) e^{-\frac{t}{t_i}}$$

where

$$t_i = \frac{1}{\Gamma + n_e + C_h + n_e \alpha_H}$$

and

$$x_{eq} = \frac{\Gamma + n_e C_H}{\Gamma + n_e C_H + n_e \alpha_H}.$$

The average ionized fraction over the time step is then

$$\langle x \rangle = x_{eq} + \left(x_o - x_{eq}\right)\left(1 - e^{-\frac{\Delta t}{t_i}}\right)\frac{t_i}{\Delta t}$$

which gives a time-averaged optical depth of

$$\langle \Delta\tau_\nu \rangle = (1 - \langle x \rangle)\, n_H \sigma_\nu \Delta r.$$

To be consistent, the time-averaged optical depth and number density of neutral hydrogen $\langle n_{HI} \rangle = (1 - \langle x \rangle)\, n_H$, should be used in equation 63 , which becomes

$$\Gamma = \int_{\nu_{th}}^{\infty} \frac{L_\nu e^{-\tau_\nu}}{h\nu} \frac{1 - e^{-\langle \Delta\tau_\nu \rangle}}{\langle n_{HI} \rangle\, V_{shell}} d\nu. \tag{65}$$

This leads to an iterative process, which is continued until the neutral fraction converges.

9.1 INTEGRATION WITH THE AMR RAY TRACING METHOD

The HI and HII fractions are added to RAMSES' state vectors as passive scalars, and in principle a detailed equation of state calculation could follow the exact evolution of the temperature using e.g. the KROME framework. In the present version we have instead opted for a simpler locally isothermal approach, where the neutral and ionized fractions have given fixed temperatures, e.g. 100 K and 10.000 K respectively. Then the temperature of a mixed medium can be calculated from their partial pressures. The total pressure in a cell is

$$P = n_{HI} k_b T_{HI} + n_{HII} k_b T_{HII} \tag{66}$$

where n_i is the number density and T_i the temperature of species i. Using that $n_i = \rho_i / \mu_i m_p = \rho X_i / \mu_i m_p$, where X_i is the fractional abundance and μ_i the molecular weight of species i, we can find the total pressure to be

$$P = \frac{\rho}{\mu_{eff} m_p} k_B T_{eff} = \rho \sum_i \frac{X_i}{\mu_i m_p} k_B T_i \,. \tag{67}$$

The resulting effective temperature and molecular weight are then

$$\frac{1}{\mu_{eff}} = \sum_i \frac{X_i}{\mu_i} \tag{68}$$

$$T_{eff} = \mu_{eff} \sum_i \frac{X_i}{\mu_i} T_i \,. \tag{69}$$

In the ray domain, the step of solving the radiative transfer along the ray with Feautriers method (section 8.8) is replaced by a solution to the the time-averaged and photon conserving photo-ionization rate as described above. We solve along the ray, starting at the point source, solving outwards in both directions, and using the accumulated τ_ν to calculate

$$L(\tau) = \frac{L_0}{n_{ray}} e^{-\tau}, \tag{70}$$

where n_{ray} is the number of traced rays, and we assume that each ray covers the same solid angle; e.g. that it is either using a random angle scheme or a scheme based on HealPix angles. The photo-ionization rate is thus calculated for a single iteration over all rays. The resulting photo-ionization rates are interpolated back to the cell domain grid as usual. After the ray tracing step, the new ionization fractions in each cell is computed as described above. The error in the ionized fraction is calculated, to check for convergence. The entire ray tracing step is repeated until convergence. Only then can we proceed to the next hydro step. Tracing all rays once in each iteration ensures that the solution upon convergence is consistent over rays that cross the same cell. This also means that adding more point sources or adding diffuse sources of ionizing photons is essentially trivial.

Part V

EXPERIMENTS

10

1D TESTS

The purpose of this chapter is to exercise the two solutions - the Unigrid, and the AMR solution - on some simple, and inherently one-dimensional, radiative transport problems where an exact analytical solution is known. This is mainly intended to make sure that all the steps in the solutions behave as expected. In RAMSES the problems are solved in 3D, but with all variables constant along the y- and z axes, and only variations in the x-direction. The first tests examines only the radiation field. The second tests examine also the heating of the gas, but, in order to compare with an analytical solution, in a situation where only thermodynamics play a role, and hydrodynamics can be neglected.

10.1 RADIATION FIELD

A source function and opacity is prescribed as a function of depth, and a few instances of the same family of solutions is used to test the Unigrid and AMR implementations. The solutions are monochromatic with constant opacity and a linear source function $S(x) = \alpha_S x$. The optical depth is then $\tau(x) = \chi x$. The transfer equation is solved for two oppositely directed pencils simultaneously, with ingoing radiation on the lower boundary $I_+(0) = I_0$ and ingoing radiation on the upper boundary $I_-(\tau_1) = I_1$. The radiative transfer equation reads

$$\frac{dI}{d\tau} = S - I \tag{71}$$

with solution in the positive direction

$$I_+(\tau) = I_+(\tau_1)e^{\tau-\tau_1} + \int_\tau^{\tau_1} S_+(\tau')e^{\tau-\tau'}d\tau' \tag{72}$$

and in the negative direction

$$I_-(\tau) = I_-(0)e^{-\tau} + \int_0^{\tau} S_-(\tau')e^{\tau'-\tau}d\tau' \tag{73}$$

Inserting the source function and boundary intensities, we have

$$I_+(\tau) = I_1e^{\tau-\tau_1} + \int_\tau^{\tau_1} \alpha_S\tau e^{\tau-\tau'}d\tau' = I_1e^{\tau-\tau_1} - \alpha_S\{(\tau_1+1)e^{\tau-\tau_1} - \tau - 1\} \tag{74}$$

and

$$I_-(\tau) = I_0e^{-\tau}\int_0^{\tau} \alpha_S\tau e^{\tau'-\tau}d\tau' = I_0e^{-\tau} + \alpha_S(\tau - 1 + e^{-\tau}) \tag{75}$$

The total heating rate $Q \equiv I - S$ due to the two rays is given by

$$2Q(\tau) = I_+(\tau) + I_-(\tau) - 2S(\tau) \tag{76}$$

The idea is to solve for a finite number of directions, and weight the results such that the total weight is a whole sphere of solid angle (4π). In this case we only solve for two directions (positive and negative direction along the same line), so each of them are weighted by 2π. The power deposited (the difference btw. incoming and outgoing flux) is then

$$\frac{dE}{dt} = 4\pi Q(\tau). \tag{77}$$

The implementations are checked in two cases:

1. $I_0 = 1$, $I_1 = 0$,$\alpha_S = 0$; An incoming intensity at the left boundary, and no source function (red curves in the plots).
2. $I_0 = 0$, $I_1 = 0$, $\alpha_S = 1$; No incoming radiation at the boundaries, but a linear source function $S(x) = x$ imposed (blue curves in the plots).

In both cases, extinction coefficient $\chi = 1$ is used.

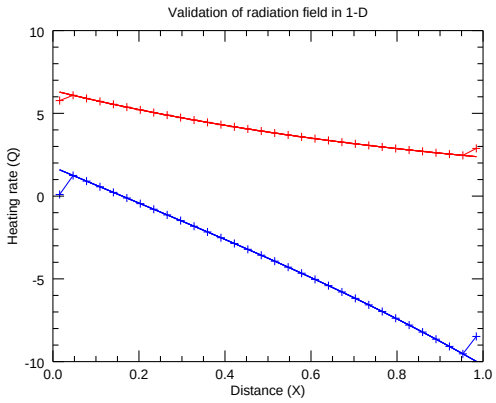

Figure 13: Validation of radiation field in one dimension. The red curves are case 1, and the blue curves are case 2. The unigrid implementation using both the integral- Feautriers method are indistinguishable from the analytical solution. The AMR solution using Feautriers method is marked with the + symbol.

As seen in figure 13, the AMR implementation has some weird behavior on the boundaries. The reason is that interpolation in three dimensions is used when transferring heating rates to the cells. This means that values are interpolated across the boundaries, which are here periodic.

The errors are shown in figure 14. For the AMR solution, the points near the boundaries are left out. The integral method has errors on

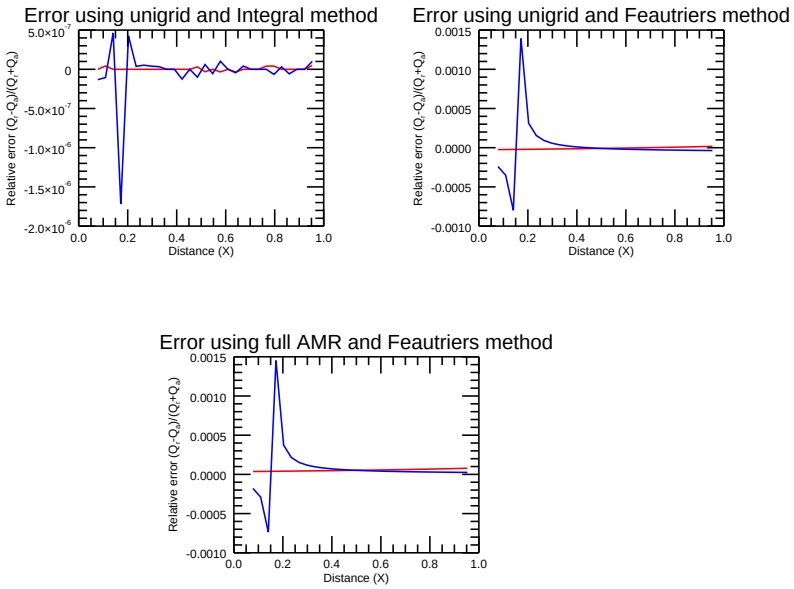

Figure 14: Relative errors when compared with the analytical solution. Again, red is case 1 and blue is case 2. For the AMR solution, the points near the boundaries are left out.

the order expected from floating point precision, while the Feautrier method has errors on the order of per mil. The reason is again found at the boundaries. This time it is an assumption in the Feautrier solver about perfect transparency, when calculating derivatives from I_0 and I_1 that causes the difference.

10.2 PERFORMANCE

The important question when it comes to performance is how many directions and frequencies can be afforded without exceeding the computational time of the hydrodynamics. The number of directions and frequencies that can be calculated in the same time as the hydrodynamics is shown in figure 15. The peak is at 300 frequencies and 1 direction, and the total number falls to with number of directions to 230, with 23 frequencies and 10 directions. The measurement was performed with 32^3 cells, but since the number of rays is proportional to the number of cells in this setup, the numbers should be roughly the same for larger simulations.

The performance with the full AMR implementation is so far less impressive, as it already at one direction and frequency takes roughly a quarter of the computational time of the hydrodynamics for the

same experiment. The discussion on why this is the case, and what can be done about it, is postponed to chapter 14.

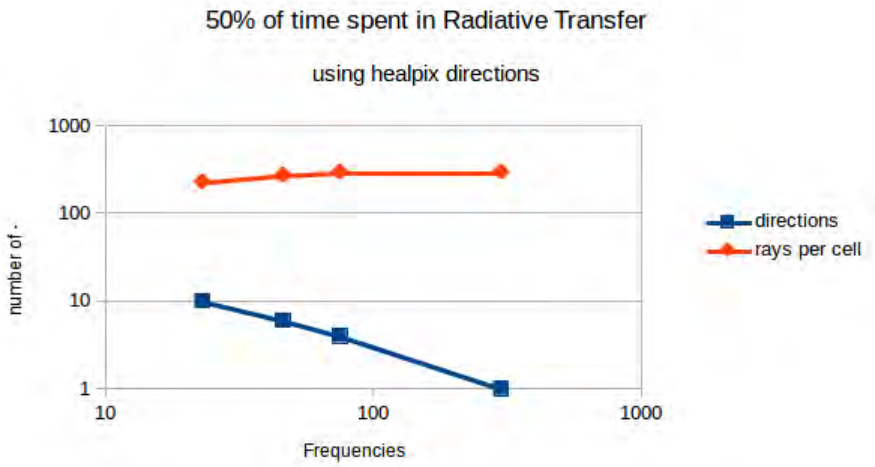

Figure 15: Performance measurement for the inflowing intensity test using 32^3 cells with the unigrid radiative transfer implementation.

11

RADIATIVE COOLING OF HYDROSTATIC DISK

11.1 EXPERIMENT SETUP

This experiment consists of a cubical column of a proto-planetary disk, co-moving in a Keplerian orbit at radius R from a central object of mass $M_\circledast$. The disk is taken to be thin, so the gravitational acceleration g is given by.

$$g = \frac{GM_\circledast}{R^2}\frac{y}{R} = g_0 y \tag{78}$$

where $g_0 = \frac{GM_\circledast}{R^3}$, G is Newtons constant of gravity and y is the vertical distance from the mid-plane. Shearing forces due to the radial extent of the column are neglected.

The box is assumed to contain an adiabatic perfect fluid at constant temperature $T = T_0$. This gives equation of state

$$P(\rho) = \frac{nk_b}{V}T_0 = \frac{\rho k_b}{\mu m_p}T_0 \tag{79}$$

where μ is the average molecular mass in proton masses (m_p). The speed of sound c_s in a perfect fluid is given by $c_s^2 = \frac{\gamma P}{\rho}$, so the equation of state may also be written

$$\rho(P) = \frac{\gamma P}{c_s^2} \tag{80}$$

where c_s is constant. In hydrostatic equilibrium, the gravitational force is balanced by the pressure, so

$$\frac{\partial P}{\partial y} = -\rho g = -\frac{\gamma g_0}{c_s^2} y P \tag{81}$$

This is a separable differential equation with solution

$$P(y) = P_0 e^{-\frac{\gamma g_0}{2c_s^2}y^2} = P_0 e^{-\frac{y^2}{2H_s^2}} \tag{82}$$

where $H_s = \sqrt{\frac{c_s^2}{\gamma g_0}}$ is the scale height and P_0 is the pressure at the disk mid-plane. To fix $P_0 = \frac{c_s^2}{\gamma}\rho_0$, the mass density at distance R is required. In the literature, it is usually given as a planar density $\Sigma(R) = \int_0^\infty \rho(R, y)dy$ with units $\frac{g}{cm^2}$. Using equation 80,

$$\Sigma = 2\int_0^\infty \rho_0 e^{-\frac{y^2}{2H^2}}dy = \sqrt{2\pi}H_s\rho_0 \tag{83}$$

where ρ_0 and P_0 are related by equation 80.

11.2 BOUNDARY CONDITIONS

On the lower boundary, reflective boundary conditions are used. The upper boundary is an outflow boundary, with vanishing gravity at the boundary, to maintain the equilibrium here.

11.3 PARAMETERS

A column at a distance of 1 AU from a solar mass object is considered. As a typical disk temperature, $T_0 = 100K$ is chosen. The adiabatic index is that of a diatomic gas $\gamma = 1.4$ and the average molecular mass $\mu = 2.37$ is that of a 75%/25% mix of $H_2$and He with solar metalicity. As a typical [6] planar density at one AU, $\Sigma = 3000$ is chosen. From these values, g_0, c_s, ρ_0 and H_s may be found from the relations in section 11.1. A constant absorption coefficient of $\alpha = 0.1\frac{cm^2}{g}$ is used.

For the simulation, the following units are chosen

One unit of ..	is chosen to be	and is in cgs units
length, u_l	$5H$	2.58×10^{12}cm
velocity, u_v	c_s	12.14×10^5 $\frac{\text{cm}}{\text{s}}$
density, u_ρ	ρ_0	2.32×10^{-9} $\frac{\text{g}}{\text{cm}^3}$

The initial density distribution is shown in figure 16.

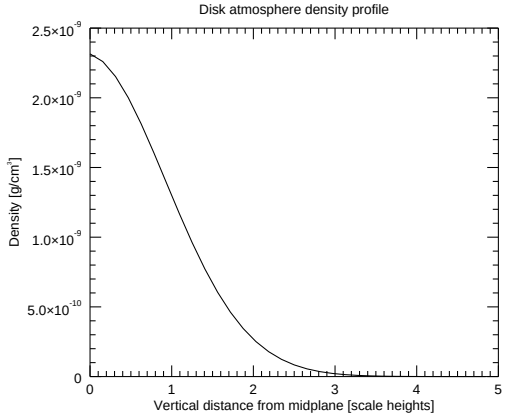

Figure 16: Initial density profile of the disk atmosphere

To test the capabilities of the radiative transfer code in three dimensions, an alternative second initial condition is also used. It is arrived at by adding noise to the pressure distribution of the hydrostatic disk. This is done in the form of white noise, which is smoothed by a Gaussian filter. The noise signal takes the form

$$\frac{P}{P_0} = g\left(e^{pr}\right), \tag{84}$$

where r is a random number in the range $[-1;1]$, p is a parameter, here chosen to be 1 and the filter is a normal Gaussian filter

$$g(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}, \tag{85}$$

where σ is the standard deviation.

11.4 RESULTS

The initial cooling rate is shown in figure 17. It shows a radiating surface layer at around 3.5 scale heights.

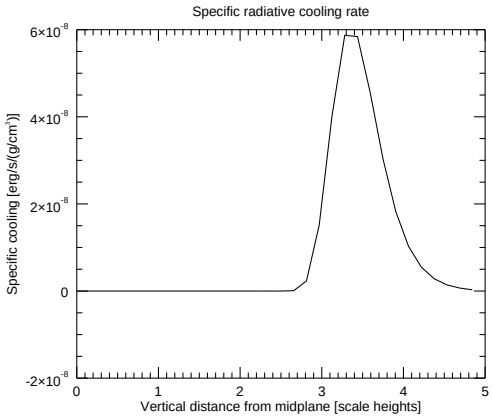

Figure 17: Initial specific radiative cooling rate in the disk atmosphere

The temperature evolution is shown in figure 18.

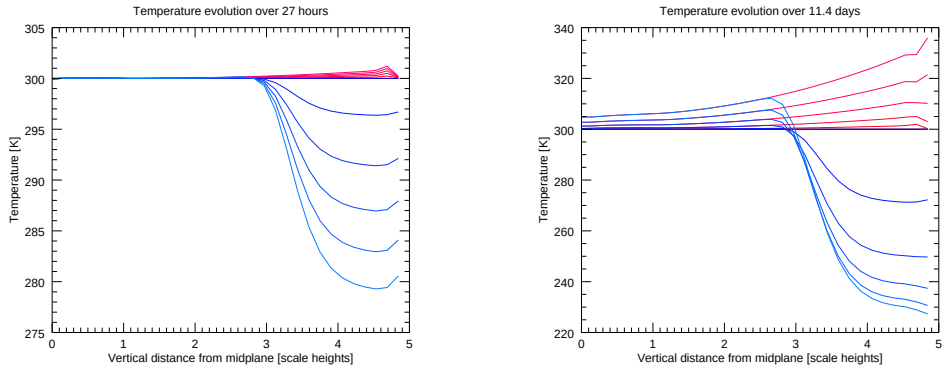

Figure 18: Temperature evolution in the disk atmosphere. The red curves show the (supposed-to-be) hydrostatic disk without cooling. The blue curves show the temperature evolution with radiative cooling. Time progresses from darker to lighter colors, with equal time spacing between the curves.

In the short term (left figure), the temperature falls at, and in front of, the surface layer, and goes towards a sharp transition at the surface. It can also be seen (red curve) that the equilibrium is not entirely stable even without cooling. Over a bit longer period (right figure), the instability becomes more prominent, unfortunately on the same timescale as the temperature profile begins to approach an equilibrium shape. After a long investigation, and attempts with several different approaches for making stable equilibrium boundary conditions, the instability turned out to be related to a common problem with Godunov-type solvers in regions with a fine balance between opposing forces (here gravity and pressure).

The result with the smoothed white noise signal in the pressure are shown in figure 19 and 20.

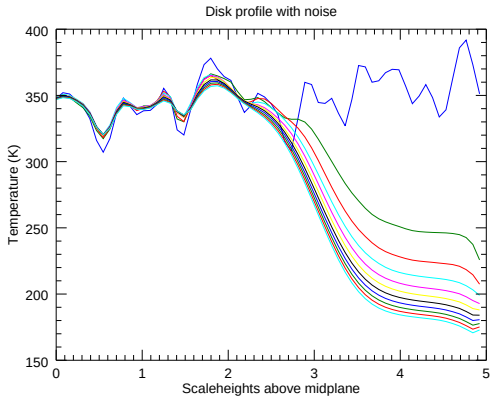

Figure 19: Evolution of the disk temperature with a smoothed white noise signal in the pressure distribution.

Figure 20: Temperature profile in a horizontal cut through the disk atmosphere, using the noisy pressure. Time evolution is from left to right.

As can be seen, the noise is quickly smoothed by the radiation in the optical surface layer, and the optically thin part above it, where the cooling happens the fastest. In the optically thick part at the bottom of the atmosphere, the temperature difference is smoothed out much more slowly.

12 D-TYPE EXPANSION OF AN HII REGION

In this ionization benchmark, we model the D-type expansion of the HII region around a newly formed star, where fusion has just been ignited. The results have been submitted to be part of the Starbench star formation code comparison [7, 26], in the test this chapter is named after. There are several reasons to take part in this benchmark:

1. The full AMR implementation will be tested thoroughly while using its ability to handle arbitrary collections of rays to model a point source.
2. Photo-ionization can be an important factor for determining the temperature in star forming regions.
3. The problem is well described, with several analytical approximations available, as well as numerical solutions with different codes.

Citing from [7], the objectives of the benchmark are

- To reproduce the equation of motion of the ionization front.
- To reproduce the equation of motion of the shock front.
- To measure the time evolved mass of the ionized medium.

12.1 BACKGROUND

The benchmark considers the idealized situation of a point source emitting $\dot{\mathcal{N}}_{LyC}$ Lyman continuum photons per unit time, all of energy $h\nu = 13.6$ eV. The source is embedded in a pure H cloud of uniform density ρ_{cl}. The mean photo-ionization cross section is taken to be $\bar{\sigma} = 6.3 \times 10^{-18}\text{cm}^2$, and the case-B recombination coefficient α_B is used, assuming the on-the-spot approximation. The HII region is taken to be isothermal at $T_i = 10^4\,K$, in which case $\alpha_B \approxeq 2.7 \times 10^{-13}\,\text{cm}^3\text{s}^{-1}$. Strömgren [62] showed that the photons will initially ionize a spherical region of radius

$$R_{st} = \left(\frac{3\dot{\mathcal{N}}_{LyC} m_p^2}{4\pi\alpha_B \rho_o^2} \right)^{\frac{1}{3}}, \tag{86}$$

where m_p is the mass of a proton and $\rho_o = \rho_{cl}$ is the initial neutral hydrogen density. The ionization front expands in a rarefied (R-type)

ionization front until it reaches R_{st} in timescales on the order of the recombination time

$$t_D = \frac{m_p}{\alpha_B \rho_o} \approxeq 19.6\,\text{yrs}\left(\frac{\rho_o}{10^{-20}\,\text{g cm}^{-3}}\right). \tag{87}$$

The HII region then further expands due to the large temperature difference between the ionized- and the neutral medium, which causes a large pressure difference. This time it expands in a dense (D-type) ionization front. Due to the temperature difference, the sound speed is highest in the ionized medium, which causes the propagation speed of the ionization front to be subsonic in the ionized medium and supersonic in the neutral medium. The ionization front is therefore preceded by a shock front, and the two fronts define a shell of the dense gas between them[7].

12.2 ANALYTICAL APPROXIMATIONS

By assuming balance between the ionizing photons produced by the star and the recombination events, and taking the dense gas shell to be small, a differential equation of the motion can be found for the time-dependent radius $R_{IF}(t)$ of the spherical ionized region Bisbas et al. [7], Raga et al. [55]:

$$\frac{1}{c_i}\frac{dR_{IF}(t)}{dt} = \left\{\frac{R_{St}}{R_{IF}(t)}\right\}^{\frac{3}{4}} - \frac{\mu_i T_o}{\mu T_i}\left\{\frac{R_{St}}{R_{IF(t)}}\right\}^{-\frac{3}{4}}. \tag{88}$$

Here c_i denotes the sound speed in the ionized medium, and μ_i, μ_o, T_i and T_o are the mean molecular weight and the temperature of the ionized and neutral medium respectively. The ratio $\frac{\mu_i T_o}{\mu T_i}$ is generally small $\left(\sim 200^{-1}\right)$, so at early times the second term on the right hand side can be neglected. This leads to the Spitzer solution

$$R_{IF}(t) = R_{St}\left(1 + \frac{7}{4}\frac{c_i t}{R_{St}}\right)^{\frac{4}{7}}. \tag{89}$$

At later times the second term on the right hand side becomes important, and eventually the ionization front stagnates at a radius R_{STAG}, that can be found by setting the left hand side of equation 88 to zero.

$$R_{STAG} = \left(\frac{c_i}{c_o}\right)^{\frac{4}{3}} R_{St}. \tag{90}$$

An expression for the shock front velocity can also be found

$$V_S = \dot{R}_{SF}(t) = \frac{1}{2}\left\{\dot{R}_{IF}(t) + \sqrt{\dot{R}_{IF}(t)^2 + 4c_o^2}\right\}, \tag{91}$$

and it may be integrated to get the position of the shock front[7, 55].

Equation 88 does however not take into account the inertia of the shocked gas. Doing so results in an equivalent of the Spitzer solution of

$$R_{IF}(t) = R_{St}\left(1 + \frac{7}{4}\sqrt{\frac{4}{3}}\frac{c_i t}{R_{St}}\right)^{\frac{4}{7}}. \tag{92}$$

and a stagnation radius of

$$R_{STAG} = \left(\frac{c_i}{c_o}\right)^{\frac{4}{3}}\left(\frac{4}{3}\right)^{\frac{2}{3}} R_{St}. \tag{93}$$

[7, 30]

12.3 THE BENCHMARK

For the benchmark test is used an isothermal equation of state for both the ionized and the neutral medium. The test is purely hydrodynamical, so no gravitation is included. Due to the nature of the two equations 89 and 91, two tests are run. One to examine the early phase, and one to examine the late phase of D-type expansion. At the time of writing, only the early phase test has been run.

12.3.1 *Early Phase*

We consider a spherical cloud containing pure hydrogen $X = 1$ and an exciting source emitting $\dot{\mathcal{N}}_{LyC} = 10^{49}$ photons per second placed in the center of the cloud. The density of the cloud is $\rho_o = 5.21 \times 10^{-21}\,\mathrm{g\ cm^{-3}}$ and the total mass is $M_{cl} = 640\,M_\odot$. The radius is therefore $R_{cl} = 1.257\,\mathrm{pc}$, and the initial Strömgren radius is $R_{St} = 0.314\,\mathrm{pc} = \frac{1}{4}R_{cl}$.

The temperature of the ionized gas is $T_i = 10^4\,\mathrm{K}$ and the mean molecular weight $\mu_i = 0.5$, which gives a sound speed of $c_i = 12.85\,\frac{\mathrm{km}}{\mathrm{s}}$ in the ionized medium. The temperature of the neutral gas is $T_o = 10^2\,\mathrm{K}$, and mean molecular weight $\mu_o = 1$, so the sound speed in the neutral medium is $c_o = 0.91\,\frac{\mathrm{km}}{\mathrm{s}}$.

As mentioned, the equation of state is taken to be isothermal, and the adiabatic index is $\gamma = 1.0001$. We simulate the full cloud with the point source in the center of the simulation, in a resolution of 256^3 cells. Within $r < R_{St}$, the gas is ionized, which gives a pressure ratio of 200 between the ionized and the neutral gas.

The simulation is evolved for $t_{end} = 141\,\mathrm{kyr}$, when the ionization front has reached the boundary of the cloud $R_{IF}(t_{end}) \approxeq R_{cl}$.

12.3.2 *Results*

A slice through the resulting ionization front for the early phase run is shown in figure 21. The result is a smooth spherical, and sharp

ionization front. It is a bit too fast compared to the Spitzer solution, as can be seen from the last frame. At that point it should have only just reached the boundary of the cloud. It is however not a major effect, and the level of agreement is similar to other state-of-the-art ionization radiative transfer codes. Still it has to be investigated and resolved. Possible answers are that it may be due to solving the initial Strömgren sphere phase using the implicit method in essentially a single time step, or maybe due to the diffusivity inherent in the TSC interpolation. The evolution of the shock front as compared to the analytical solutions and a one-dimensional ionization code is shown in figure 22.

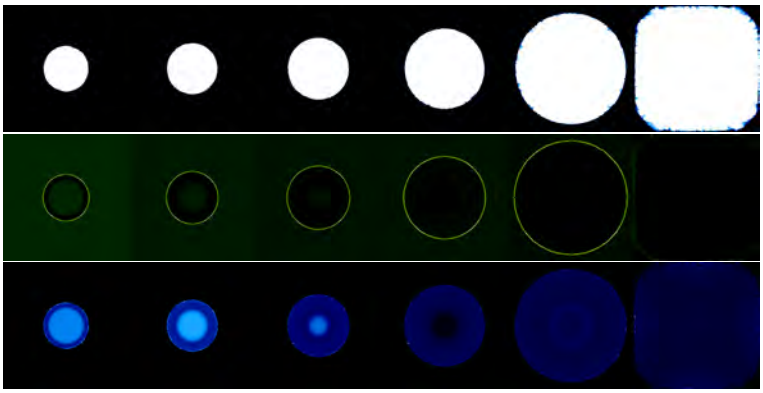

Figure 21: Slice through the middle of the ionized region. From above to below are shown the ionized fraction of H_{II}, the density and the pressure. The time of the snapshots from left to right are 5, 10, 20, 40,80 and 140 kyr.

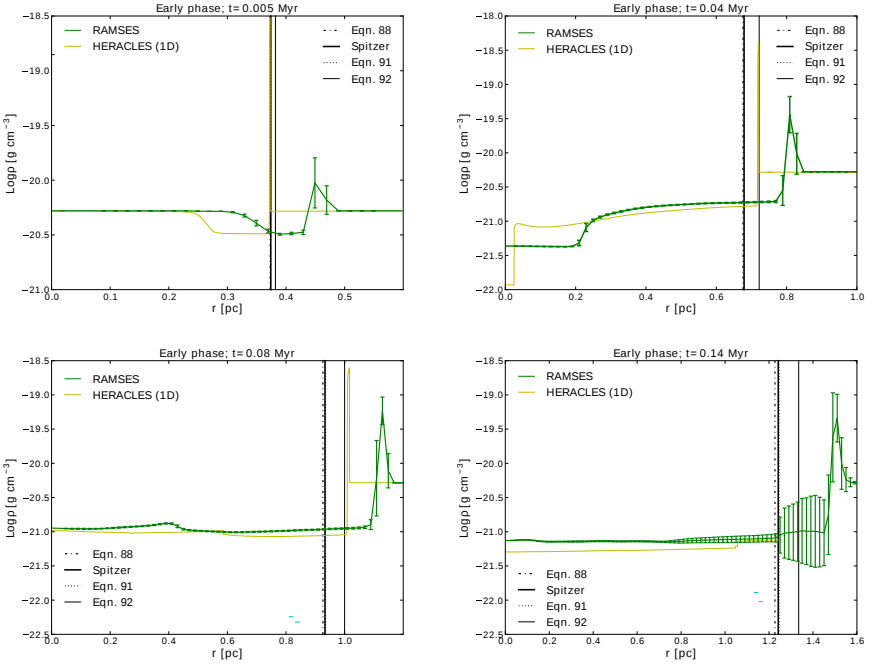

Figure 22: Evolution of the over-dense region at the shock front, compared to the analytical solutions and the 1D ionization code Heracles.

13 SIMULATION OF AN HII REGION

The radiation from short-lived high-mass stars ionizes the surrounding gas from which they were formed, resulting in large regions of ionized gas, primarily HII, where star formation is taking place, or has recently taken place. A large part of the reemission is seen in the optical, while the embedded stars can be observed in the infrared (figure 23). Due to the high temperatures in these regions, metallic ions of oxygen, nitrogen and carbon, rather than hydrogen, provide most of the cooling[61]. If we consider just a single massive star embedded in a uniform medium, a spherical ionized region will expand quickly out to the Strömgren radius, and then more slowly following the Spitzer solution and later the solution first derived by Hosokawa and Inutsuka [30], as described in section 12.2. In reality, stars usually form in clusters in a turbulent medium. A consequence of the latter is that before the ionization front slows down considerably, it will typically have hit an area of lower density, possibly at the boundary of the local molecular cloud core. The result is that the ionization shock front rushes out of the high-density region in a so-called champagne flow. [61]

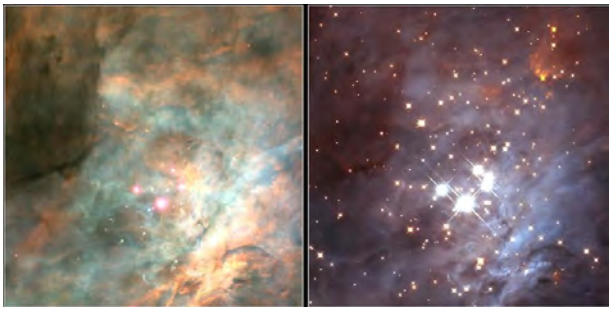

Figure 23: HII region in the Trapezium cluster in the Orion Nebula in the optical (left) and infrared (right). Image is public domain by NASA and available on Wikipedia.

In the following, an HII region is represented more realistically, by repeating the early phase D-type expansion (chapter 12) in a turbulent pure-hydrogen gas. In this way the evolution of the ionization front into the filamentary structures of the turbulent medium can be studied.

13.1 SETUP

The size of the simulation is 1.256 parsec in a 256^3 cell unigrid box. Turbulence is first induced in the gas, by driving it to Mach 10 in the 100K gas in an Ornstein-Uhlenbeck process [see e.g. 19] using a purely solenoidal forcing, over 5 turnover times. This means that the forcing happens with smoothly, randomly varying position and angle, and that no compressive forcing is applied, minimizing the induced density enhancements. The thermal- to magnetic pressure ratio $\beta = \frac{P_{th}}{P_M}$ is chosen to be 3.6, so the gas is marginally magnetized. After the turbulence has been induced, a point source with the same

luminosity as before is placed at the center. The temperature of the ionized- and neutral medium are 100K and 10^4 K respectively, also as before.

13.2 RESULTS

As can be seen from figure 24, the ionized region is only approximately spherical at the very earlies times. Instead two lobes develop, expanding into the low-density region below the radiation source.

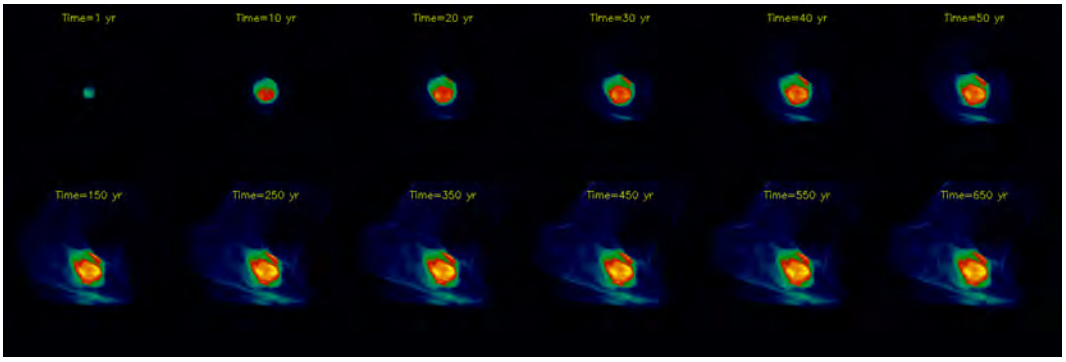

Figure 24: Evolution of the column density of ionized hydrogen in the XY plane.

This is seen better in figure 25, where the HII column density after 650 years is shown beside total column density in the three axis-planes. In the middle panel, it can be seen how the ionization front in the south-east is constrained by an impinging filament, which will be slowly swept away by the radiation. This structure is also illustrated in figure 26 with an isosurface mapping out the ionization front on top of a volume rendering of the logarithm of density. Here the champagne flow enclosed in the filaments going towards the top edge of the box is clearly visible.

This model has not evolved very far yet, and the ionziation front has not reached its final size. At later times, a second pressure wave would be launched in the medium where the ionization front collides with the filaments. The change in temperature induced by ionization has a profound impact on the kinematic structure. The originally highly supersonic gas will, due to the change in temperature, become trans-sonic. Therefore the turbulence is suppressed, and density peaks are smoothed on a sound-crossing time scale. The overall effect is a suppression of star formation inside the ionized medium, while the pressure wave at the edge may push critical prestellar cores above the density threshold for collapse and thereby induce new star formation.

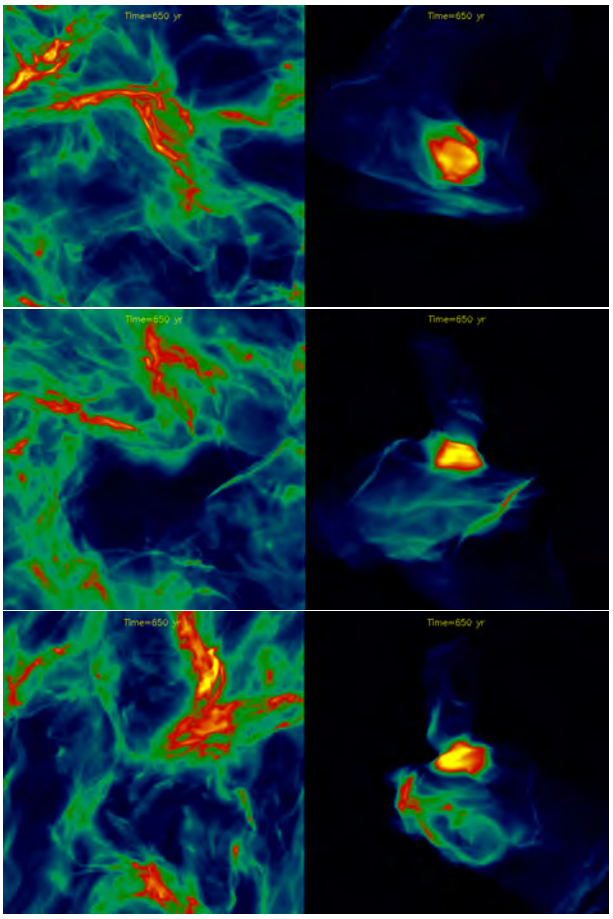

Figure 25: Column densities in the last snapshot. Total column density on the left, HII column density on the right. From top to bottom is shown the XY-, the YZ- and the XZ-plane.

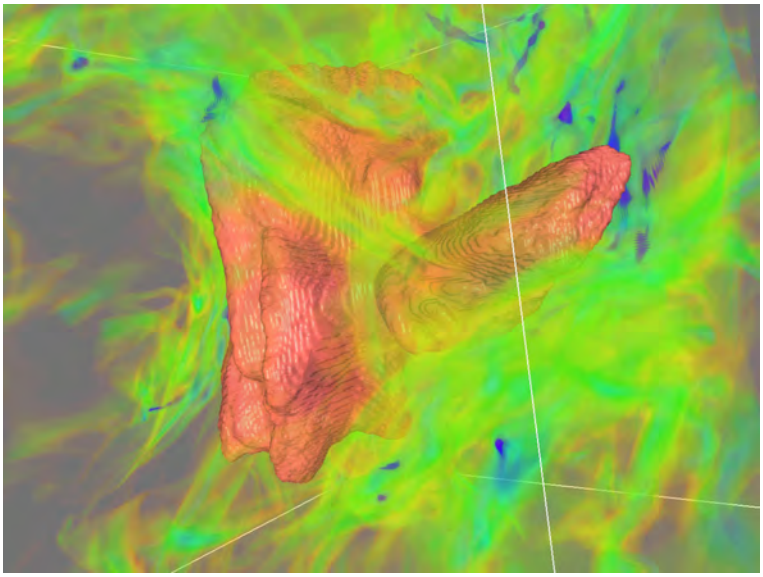

Figure 26: Iso-surface of ionization fraction $\rho_{HII}/\rho_{tot} = \frac{1}{2}$ in the last snapshot. The logarithm of the total density is shown in green/blue.

14 DISCUSSION AND CONCLUSIONS

Through the previous chapters a method for solving the radiative transfer equation on an hierarchical adaptive mesh with adaptive time stepping was developed. The method presented here has been implemented into the adaptive mesh refinement code RAMSES in two steps, first restricted to unigrid and then in full AMR. The implementation takes into account absorption and isotropic emission of radiation from the diffuse medium, and couples the radiative heating with the magnetohydrodynamical equations in RAMSES. It does so far not take into account scattering effects. The novelty of the method is that it, on an adaptive mesh, allows the use of multi-frequency long-characteristics ray tracing to directly integrate the radiative transfer equation in a way that is efficient enough that reasonable[1] directional- and frequency coverage can be obtained while maintaining the cost of the radiative transfer comparable to the hydrodynamics. Previous methods either use a diffusive approximation [e.g. 10, 57, 39], include only point sources and not emission from the diffuse medium [e.g. 72, 56], or have the computational time dominated by the radiative transfer calculation[27]. Importantly, the method is highly parallelizable, because the long-characteristic rays are independent, and the number of them is typically large, even compared to the number of cells in the simulation. Good enough performance characteristics has yet to be realized for the implementation in full AMR.

14.1 THE RADIATIVE TRANSFER SOLVER

The radiative transfer solver works by sending the relevant physical state of the cells to a ray-centered domain decomposition, using the second-order accurate TSC interpolation from cell centers to points along the long-characteristics rays. Here, the radiative transfer equation is solved along the rays in a photon-conserving formulation in terms of the direction-area-specific heating rate $Q_\nu = I_\nu - S_\nu$, using the second-order accurate Feautrier method for the integration. As opacity- and emission gradients in the turbulent and filamentary structures in star-formation simulations may be large and minimally resolved, the stability of a second order accurate method is preferred over the increased local accuracy of higher order methods. Once the heating rate is known, it is sent back to the cell domain decomposition, and interpolated back to the cell centers, where it finally enters

1 Hundreds of directions $\times$ frequencies. See section 10.2.

the magnetohydrodynamics equations as a source term in the energy conservation equation.

In one-dimensional tests where a fixed constant incoming intensity was prescribed at the boundaries, a constant opacity was used, and a linear relationship between source function and position was imposed, the radiative transfer solution gave the same results as the analytical solutions to within floating point precision when the integral method was used to solve along the rays (section 10.1). When Feautriers method was used, there were errors of order per mil. The reason is most likely the implementation of boundary conditions used. When finding values at the neighbor points to the boundary, a perfectly transparent medium between the boundary and the first point on the ray is assumed. This does not agree with the imposed constant opacity, and therefore there is a small discrepancy.

The coupling with hydrodynamics was tested in a simple setting, with conditions relevant to star formation, namely in a small midplane-centered parcel of a proto-planetary disk, initially in isothermal hydrostatic equilibrium. No radiation field was imposed on the upper boundary, and as a result the atmosphere should radiatively cool. The purpose of this test was originally to compare the radiative transfer solver with the STAGGER code, which is a code often used in solar physics, that include both MHD and radiative transfer. A stable hydrostatic equilibrium solution could however not be maintained with RAMSES, and the problem, which turned out to be an artifact of its Godunov hydrodynamics solver, has not yet been resolved. Lack of stable equilibria and exact comparisons aside, the cooling of the disk happens from a fairly thin surface layer, and the temperature profile approaches one where the temperature drops quickly at the surface layer, and only slowly above and below. This is qualitatively the solution one would expect. An exact comparison with a known solution is unfortunately not possible, because the result is disturbed by the imbalance due to the Godunov solver.

14.2 THE IONIZATION SOLVER

The photon-conserving time averaging photo-ionization solver due to Mellema et al. [45] was implemented and integrated into the radiative transfer solution for the case of a single point source embedded in a pure hydrogen medium (chapter 9). It was applied to the early phase of the D-type expansion of an HII region around a newly formed star, as part of the "StarBench" star formation code comparison (chapter 12). The simulation results feature an almost perfectly spherical ionization and pressure shock front. Both the ionization- and shock front are very sharp, with the gradient over just a couple of cells. For the early phase, the analytical Spitzer solution holds to a good approximation. In the simulation result, the ionization front moves a bit too

fast compared to the Spitzer solution. The reason for this is still unknown, but it might be due to the fact that the ionization front grows to the size of the initial Strömgren radius in just a single time step, or possibly because the interpolation between rays and cells allows ionized HII to diffuse out in front of the ionization front. The deviation is however comparable to other state-of-the-art ionization codes that participate in the benchmark.

14.3 PERFORMANCE CHARACTERISTICS

Performance test of the unigrid solution as a whole show that hundreds of ray directions $\times$ frequencies can be calculated in one time step of the hydro simulation. The isolated performance of the ray-cell intersection tests, which is an obvious potential bottleneck in the full AMR solution, is shown to also be adequate to cover hundreds of angle$\times$frequencies. Performance of the full AMR solution as a whole is however not good enough yet. The main bottleneck is that the ray-cell intersections are stored in a structure with indirect referencing to heap-allocated arrays of ray indices. With some restructuring of the code, this structure can be replaced by a flat cell-ray lookup table (just a two-dimensional integer array), which should decrease the time spent on memory allocation and lookup considerably. Regarding parallelization, there is also an important performance issue for large-scale simulations. Presently, all CPUs have knowledge of all rays, which implies a non-sparse all-to-all communication step. This is unnecessary. Only those rays that could potentially hit a box enclosing the domain owned by a given CPU needs to be known. The difference is mainly in bandwidth, since statistically every CPU will likely create rays that hit every other CPU's domain. The bandwidth difference is that between a true all-to-all communication step sending $O(n_{cpu}n_{ray})$ rays, and for the sparse communication sending $O\left(n_{cpu}^{\frac{1}{3}}n_{ray}\right)$ rays, where n_{cpu} is the number of CPUs and n_{ray} is the total number of rays used in the simulation.

14.4 CONCLUSION AND FUTURE WORK

In summary, a diffusive ray tracing radiative transfer solver for RAMSES, which handles absorption and reemission, as well as ionization has been successfully created and tested. There are still some issues that need to be taken care of regarding the validity of the solver. These include correct handling of boundary conditions on the Feautrier solver, and a closer look at the energy balance in the ionization solver to understand why the ionization front moves slightly too fast. There are also some performance improvements that need to be addressed, in-

cluding less reliance on indirect indexing of heap allocated arrays, and a sparser communication of ray geometry.

Using the solver for both diffusive radiation and point sources is presently possible, but in a large-scale simulation it requires a huge number of rays, as the geometrical attenuation of radiation from point sources needs to be represented statistically by the decreasing number of rays with distance. This situation can be improved, in part by varying ray directions randomly over time to get better coverage statistically. The signal might also be attenuated by using the HealPix algorithm to recursively define new equal-area rays as the distance to the source increases.

The work in this thesis is mainly concerned with the ray tracing and book-keeping parts of solving the radiative transfer problem. The interaction between the radiation and the medium has so far been treated minimally with constructs like constant opacities and gray blackbody radiation. However, the framework is in place for multi-wavelength radiative transfer. Since the non-equilibrium chemical network solver KROME is currently being integrated into RAMSES, the next obvious steps are to derive the optical properties from the chemical abundances, and to choose frequencies suitable for representing for example reprocessing of visible and UV light into infrared light, and / or to treat spectral lines with an opacity sampling method.

In the near future the integration of either pre-created lookup tables, or directly coevolution of the MESA stellar structure code, will make it possible to include correct protostellar luminosities, both as bolometric flux, and in the UV. In combination with the radiative transfer presented in this thesis it will open up a new chapter in the use of RAMSES in Copenhagen, with realistic radiative feedback, crucial in the correct description of high mass star forming regions.

Part VI

APPENDIX

A

DETAILS OF MEMORY LAYOUT

A.1 MODULE RAYDOMAIN

For changing between domain decompositions.

RAY_GEOM_T	RAY GEOMETRY	
r	real (:,3)	Ray direction vectors
p	real (:,3)	Ray affine displacements
ixr, iyr, izr	integer (:)	Virtual coordinate axes
iaxis	integer (:,3)	(ixr, iyr, izr)
nray	integer	Number of rays
ndep	integer (:)	Number of points along each ray

Table 2: Ray Geometry Structure

CELL_RAYS_T	CELL-RAY INTERSECTIONS	
irays	integer (:)	Indicies of intersecting rays
dist	real (:)	Distance along ray to intersection
rpc	real (:,3)	Coordinate of ray center within cell
class	integer	Classification of ray for intersection test
nrays	integer	Number of intersecting rays

Table 3: Cell-Ray Intersection Structure

RAY_POINT_T	A SINGLE POINT ALONG A RAY	
iray	integer	Index of the ray to which the point belongs
cell_level	integer	Level of refinement of the cell in which the point resides
d	real	Specific density
t	real	Temperature
s	real	Distance along ray to this point

Table 4: Cell-Ray Intersection Structure

GLOBAL OF THE MODULE		
ray_geom	Ray_Geom_t	Global ray geometry
cell_rays	Cell_Rays_t (:)	Local cell intersections by rays
rp_cell	Ray_Point_t	Points along rays in cell domain
rp_ray	Ray_Point_t	Points along rays in ray domain
q_cell	real (:)	Heating rates in the cell domain
np_ray	integer (:)	Number of ray points to send to each process when going to the ray domain
ray offsets	integer (:)	Offsets in memory for sending ray points
np_cell	integer (:)	Number of ray points to receive from each process when going to the ray domain[1]
cell_offsets	integer (:)	Offsets in memory for receiving ray points

Table 5: Globals of the Ray_Domain module

BIBLIOGRAPHY

[1] Tom Abel, Michael L Norman, and Piero Madau. Photon-conserving radiative transfer around point sources in multidimensional numerical cosmology. *The Astrophysical Journal*, 523 (1):66, 1999. (Cited on page 57.)

[2] Anthony Allen, Zhi-Yun Li, and Frank H. Shu. Collapse of magnetized singular isothermal toroids. ii. rotation and magnetic braking. *The Astrophysical Journal*, 599(1):363, 2003. URL http://stacks.iop.org/0004-637X/599/i=1/a=363. (Cited on page 15.)

[3] Philippe Andre, Derek Ward-Thompson, and Mary Barsony. From pre-stellar cores to protostars: The initial conditions of star formation. *arXiv preprint astro-ph/9903284*, 1999. (Cited on page 15.)

[4] Shantanu Basu and Telemachos Ch Mouschovias. Magnetic braking, ambipolar diffusion, and the formation of cloud cores and protostars. 1: Axisymmetric solutions. *The Astrophysical Journal*, 432:720–741, 1994. (Cited on page 15.)

[5] Frank Bertoldi and Christopher F McKee. Pressure-confined clumps in magnetized molecular clouds. *The Astrophysical Journal*, 395:140–157, 1992. (Cited on page 13.)

[6] T Birnstiel, CP Dullemond, and F Brauer. Gas-and dust evolution in protoplanetary disks. *arXiv preprint arXiv:1002.0335*, 2010. (Cited on page 68.)

[7] Thomas G Bisbas, Thomas Haworth, Alex C Raga, Jonathan Mackey, Pascal Tremblin, and Robin JR Williams. Starbench: The d-type expansion of hii regions. (Cited on pages 73, 74, and 75.)

[8] Nuria Calvet, James Muzerolle, César Briceno, Jesus Hernández, Lee Hartmann, José Luis Saucedo, and Karl D Gordon. The mass accretion rates of intermediate-mass t tauri stars. *The Astronomical Journal*, 128(3):1294, 2004. (Cited on page 16.)

[9] John I Castor. *Radiation hydrodynamics*. Cambridge University Press, 2004. (Cited on pages ix, 29, 30, 31, 34, and 35.)

[10] Benoit Commercon, Romain Teyssier, Edouard Audit, Patrick Hennebelle, and Gilles Chabrier. Radiation hydrodynamics with adaptive mesh refinement and application to prestellar core collapse. i methods. *arXiv preprint arXiv:1102.1216*, 2011. (Cited on pages 34 and 83.)

[11] Bruno Dubroca and Jean-Luc Feugeas. Etude théorique et numérique d'une hiérarchie de modèles aux moments pour le transfert radiatif. *Comptes Rendus de l'Académie des Sciences-Series I-Mathematics*, 329(10):915–920, 1999. (Cited on page 34.)

[12] Charles R Evans and John F Hawley. Simulation of magnetohydrodynamic flows-a constrained transport method. *The Astrophysical Journal*, 332:659–677, 1988. (Cited on page 22.)

[13] F. Saez, A. Llebaria, P. Lamy, and D. Vibert. Three-dimensional reconstruction of the streamer belt and other large-scale structures of the solar corona. *A&A*, 473(1):265–277, 2007. doi: 10.1051/0004-6361:20066777. URL http://dx.doi.org/10.1051/0004-6361:20066777. (Cited on page 51.)

[14] Christoph Federrath, Robi Banerjee, Paul C Clark, and Ralf S Klessen. Modeling collapse and accretion in turbulent gas clouds: implementation and comparison of sink particles in amr and sph. *The Astrophysical Journal*, 713(1):269, 2010. (Cited on page 24.)

[15] Martin Frank Axel Klar Florian Schneider, Graham Alldredge. Higher order mixed moment approximations for the fokker-planck equation in one space dimension. *arXiv:1405.5305 [math-ph] (unpublished)*, 2014. (Cited on page 34.)

[16] Jose Franco and Donald P Cox. Molecular clouds in galaxies with different z: fragmentation of diffuse clouds driven by opacity. *Publications of the Astronomical Society of the Pacific*, pages 1076–1079, 1986. (Cited on page 25.)

[17] Martin Frank, A Klar, EW Larsen, and S Yasuda. Approximate models for radiative transfer. *BULLETIN-INSTITUTE OF MATHEMATICS ACADEMIA SINICA*, 2(2):409, 2007. (Cited on page 34.)

[18] Sébastien Fromang, Patrick Hennebelle, and Romain Teyssier. A high order godunov scheme with constrained transport and adaptive mesh refinement for astrophysical mhd. *arXiv preprint astro-ph/0607230*, 2006. (Cited on pages 19 and 22.)

[19] Crispin W Gardiner et al. *Handbook of stochastic methods*, volume 3. Springer Berlin, 1985. (Cited on page 79.)

[20] Josep M Girart, Ramprasad Rao, and Daniel P Marrone. Magnetic fields in the formation of sun-like stars. *Science*, 313(5788): 812–814, 2006. (Cited on page 15.)

[21] Andrew S Glassner. *An introduction to ray tracing*. Morgan Kaufmann, 1989. (Cited on page 51.)

[22] Nickolay Y Gnedin and Nicholas Hollon. Cooling and heating functions of photoionized gas. *The Astrophysical Journal Supplement Series*, 202(2):13, 2012. (Cited on page 25.)

[23] Johan Peter Goedbloed and Stefaan Poedts. *Principles of magnetohydrodynamics: With applications to laboratory and astrophysical plasmas*. Cambridge university press, 2004. (Cited on page 22.)

[24] T. Grassi, S. Bovino, D. R. G. Schleicher, J. Prieto, D. Seifried, E. Simoncini, and F. A. Gianturco. Krome - a package to embed chemistry in astrophysical simulations. *Monthly Notices of the Royal Astronomical Society*, 439:2386–2419, April 2014. doi: 10.1093/mnras/stu114. (Cited on page 54.)

[25] Troels Haugbølle. personal communication, 2014. (Cited on page 50.)

[26] Thomas Haworth. Starbench: A workshop for the benchmarking of star formation codes, May 2014. URL http://www.astro.ex.ac.uk/people/haworth/workshop_bench/. (Cited on page 73.)

[27] Thomas J Haworth and Tim J Harries. Radiation hydrodynamics of triggered star formation: the effect of the diffuse radiation field. *Monthly Notices of the Royal Astronomical Society*, 420(1):562–578, 2012. (Cited on pages 35 and 83.)

[28] T Heinemann, W Dobler, A Nordlund, and Axel Brandenburg. Radiative transfer in decomposed domains. *arXiv preprint astro-ph/0503510*, 2005. (Cited on page 55.)

[29] Roger W Hockney and James W Eastwood. *Computer simulation using particles*. CRC Press, 1988. (Cited on page 52.)

[30] Takashi Hosokawa and Shu-ichiro Inutsuka. Dynamical expansion of ionization and dissociation front around a massive star. ii. on the generality of triggered star formation. *The Astrophysical Journal*, 646(1):240, 2006. (Cited on pages 75 and 79.)

[31] I Hubeny. Accelerated lambda iteration: an overview. In *Stellar Atmosphere Modeling*, volume 288, page 17, 2003. (Cited on page 31.)

[32] A Hujeirat, P Myers, M Camenzind, and A Burkert. Collapse of weakly ionized rotating turbulent cloud cores. *New Astronomy*, 4 (8):601–613, 2000. (Cited on page 15.)

[33] Jens Kauffmann, Thushara Pillai, and Paul F Goldsmith. Low virial parameters in molecular clouds: Implications for high-mass star formation and magnetic fields. *The Astrophysical Journal*, 779(2):185, 2013. (Cited on page 12.)

[34] Alexei M Khokhlov. Fully threaded tree algorithms for adaptive refinement fluid dynamics simulations. *Journal of Computational Physics*, 143(2):519–543, 1998. (Cited on pages 20 and 23.)

[35] Woong-Tae Kim and Eve C Ostriker. Amplification, saturation, and q thresholds for runaway: growth of self-gravitating structures in models of magnetized galactic gas disks. *The Astrophysical Journal*, 559(1):70, 2001. (Cited on page 12.)

[36] Andrey Nikolaevich Kolmogorov. Dissipation of energy in locally isotropic turbulence. In *Dokl. Akad. Nauk SSSR*, volume 32, pages 16–18, 1941. (Cited on page 11.)

[37] P Kominsky. Derivation of conservative mhd equations. 2006. [Online; accessed 16-May-2014]. (Cited on page 22.)

[38] Pavel Kroupa. On the variation of the initial mass function. *Monthly Notices of the Royal Astronomical Society*, 322(2):231–246, 2001. (Cited on pages xi and 13.)

[39] Rolf Kuiper, Hubert Klahr, Cornelis Dullemond, Wilhelm Kley, and Thomas Henning. Fast and accurate frequency-dependent radiation transport for hydrodynamics simulations in massive star formation. *arXiv preprint arXiv:1001.3301*, 2010. (Cited on page 83.)

[40] RB Larson. Turbulence and star formation in molecular clouds. *Monthly Notices of the Royal Astronomical Society*, 194:809–826, 1981. (Cited on pages 11 and 14.)

[41] Richard B Larson. The physics of star formation. *Reports on Progress in Physics*, 66(10):1651, 2003. (Cited on page 14.)

[42] P Londrillo and L Del Zanna. High-order upwind schemes for multidimensional magnetohydrodynamics. *The Astrophysical Journal*, 530(1):508, 2000. (Cited on page 23.)

[43] Jeffrey Mahovsky and Brian Wyvill. Fast ray-axis aligned bounding box overlap tests with plucker coordinates. *Journal of Graphics Tools*, 9(1):35–46, 2004. (Cited on page 47.)

[44] Christopher F McKee and Eve C Ostriker. Theory of star formation. *arXiv preprint arXiv:0707.3514*, 2007. (Cited on page 11.)

[45] Garrelt Mellema, Ilian T Iliev, Marcelo A Alvarez, and Paul R Shapiro. C ^{2}-ray: A new method for photon-conserving transport of ionizing radiation. *New Astronomy*, 11(5):374–395, 2006. (Cited on pages iii, 57, and 84.)

[46] Å. Nordlund. Ray Casting and Flux Limited Diffusion. In J. Alves, B. G. Elmegreen, J. M. Girart, and V. Trimble, editors, *Computational Star Formation*, volume 270 of *IAU Symposium*,

pages 207–214, April 2011. doi: 10.1017/S174392131100038X. (Cited on page 35.)

[47] Donald E Osterbrock. Astrophysics of gaseous nebulae. *Research supported by the Research Corp., Wisconsin Alumni Research Foundation, John Simon Guggenheim Memorial Foundation, Institute for Advanced Studies, and National Science Foundation. San Francisco, WH Freeman and Co., 1974. 263 p.*, 1, 1974. (Cited on page 58.)

[48] Donald E Osterbrock. *Astrophysics of gaseous nebulae and active galactic nuclei*. University science books, 2006. (Cited on page 25.)

[49] Paolo Padoan and Åke Nordlund. The star formation rate of supersonic magnetohydrodynamic turbulence. *The Astrophysical Journal*, 730(1):40, 2011. (Cited on page 24.)

[50] Paolo Padoan, Alexei Kritsuk, Michael L Norman, and Åke Nordlund. A solution to the pre-main-sequence accretion problem. *The Astrophysical Journal Letters*, 622(1):L61, 2005. (Cited on page 16.)

[51] Paolo Padoan, Troels Haugbølle, and Åke Nordlund. A simple law of star formation. *The Astrophysical Journal Letters*, 759(2):L27, 2012. (Cited on pages 12 and 24.)

[52] EN Parker. The dynamical state of the interstellar gas and field. *The Astrophysical Journal*, 145:811, 1966. (Cited on page 12.)

[53] Annamaneni Peraiah. *An Introduction to Radiative Transfer: Methods and applications in astrophysics*. Cambridge University Press, 2002. (Cited on page 31.)

[54] Bárbara Pichardo, Enrique Vázquez-Semadeni, Adriana Gazol, Thierry Passot, and Javier Ballesteros-Paredes. On the effects of projection on morphology. *The Astrophysical Journal*, 532(1):353, 2000. (Cited on page 13.)

[55] AC Raga, A Rodríguez-González, A Noriega-Crespo, and A Esquivel. New variable jet models for hh 34. *The Astrophysical Journal Letters*, 744(1):L12, 2012. (Cited on page 74.)

[56] Erik-Jan Rijkhorst, Tomasz Plewa, Anshu Dubey, and Garrelt Mellema. Hybrid characteristics: 3d radiative transfer for parallel adaptive mesh refinement hydrodynamics. *arXiv preprint astro-ph/0505213*, 2005. (Cited on page 83.)

[57] Joakim Rosdahl, Jeremy Blaizot, Dominique Aubert, Timothy Stranex, and Romain Teyssier. ramses-rt: radiation hydrodynamics in the cosmological context. *Monthly Notices of the Royal Astronomical Society*, 436(3):2188–2231, 2013. (Cited on pages 34 and 83.)

[58] Edwin E Salpeter. The luminosity function and stellar evolution. *The Astrophysical Journal*, 121:161, 1955. (Cited on page 13.)

[59] Debra Shepherd. Massive star outflows. *Proceedings of the International Astronomical Union*, 1(S227):237–246, 2005. (Cited on page 16.)

[60] Frank H Shu. Self-similar collapse of isothermal spheres and star formation. *The Astrophysical Journal*, 214:488–497, 1977. (Cited on page 14.)

[61] Steven W Stahler and Francesco Palla. The formation of stars. *The Formation of Stars, by Steven W. Stahler, Francesco Palla, pp. 865. ISBN 3-527-40559-3. Wiley-VCH, January 2005.*, 1, 2005. (Cited on pages 11, 13, and 79.)

[62] Bengt Strömgren. The physical state of interstellar hydrogen. *The Astrophysical Journal*, 89:526, 1939. (Cited on page 73.)

[63] Seth Teller and Michael Hohmeyer. Determining the lines through four lines. *Journal of graphics tools*, 4(3):11–22, 1999. (Cited on page 47.)

[64] Romain Teyssier. Cosmological hydrodynamics with adaptive mesh refinement: a new high resolution code called ramses. *arXiv preprint astro-ph/0111367*, 2001. (Cited on pages 19 and 21.)

[65] Romain Teyssier. The ramses code, May 2014. URL `http://irfu.cea.fr/Phocea/Vie_des_labos/Ast/ast_sstechnique.php?id_ast=904`. (Cited on page 19.)

[66] Alar Toomre. On the gravitational stability of a disk of stars. *The Astrophysical Journal*, 139:1217–1238, 1964. (Cited on page 12.)

[67] Eleuterio F Toro and Eleuterio F Toro. *Riemann solvers and numerical methods for fluid dynamics*, volume 16. Springer, 1999. (Cited on page 23.)

[68] Bram Van Leer. Towards the ultimate conservative difference scheme. iv. a new approach to numerical convection. *Journal of computational physics*, 23(3):276–299, 1977. (Cited on page 23.)

[69] Aristodimos Vasileiadis, Åke Nordlund, and Martin Bizzarro. Abundance of 26 al and 60 fe in evolving giant molecular clouds. *The Astrophysical Journal Letters*, 769(1):L8, 2013. (Cited on page 24.)

[70] Neil Vaytet, Edouard Audit, Gilles Chabrier, Benoît Commerçon, and Jacques Masson. Simulations of protostellar collapse using multigroup radiation hydrodynamics i. the first collapse. *Astronomy & Astrophysics/Astronomie et Astrophysique*, 543(1), 2012. (Cited on page 14.)

[71] Neil Vaytet, Gilles Chabrier, Edouard Audit, Benoît Commerçon, Jacques Masson, Jason Ferguson, and Franck Delahaye. Simulations of protostellar collapse using multigroup radiation hydrodynamics. ii. the second collapse. *arXiv preprint arXiv:1307.1010*, 2013. (Cited on page 14.)

[72] John H Wise and Tom Abel. Enzo+ moray: radiation hydrodynamics adaptive mesh refinement simulations with adaptive ray tracing. *Monthly Notices of the Royal Astronomical Society*, 414(4): 3458–3491, 2011. (Cited on page 83.)

COLOPHON

This document was typeset using the typographical look-and-feel `classicthesis` developed by André Miede. The style was inspired by Robert Bringhurst's seminal book on typography *"The Elements of Typographic Style"*. `classicthesis` is available for both LaTeX and LyX:

http://code.google.com/p/classicthesis/

Happy users of `classicthesis` usually send a real postcard to the author, a collection of postcards received so far is featured at:

http://postcards.miede.de/

Final Version as of June 6, 2014 (`Radiative Transport in Star- and Planet Formation` version 0.1).

DECLARATION OF AUTHORSHIP

I, Troels Frostholm, declare that this thesis titled 'Radiative Transport in Star- and Planet Formation' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Copenhagen, June 2014

Troels Frostholm