MNRAS **000**, 1–6 (2017) Preprint 13 September 2017 Compiled using MNRAS LATEX style file v3.0

TREVR: Tree-based Reverse Ray Tracing in Gasoline

R. M. Woods, J. J. Grond, J. Wadsley and H. M. P. Couchman
Department of Physics and Astronomy, McMaster University, Hamilton, Ontario L8S 4M1, Canada

Accepted XXX. Received YYY; in original form ZZZ

ABSTRACT

Key words: radiative transfer – methods: numerical – galaxies

1 INTRODUCTION

Does any other paper shorten Radiative Transfer to RT? To me it feels wrong to do that. Radiation is arguably the most important physical phenomenon in astrophysics. Almost all of the information we receive from space comes in the form of photons we detect on or around earth. Understanding the process of radiative transfer (RT) is key in interpreting this information, as the photons are affected by the medium they travel through on the way to our telescopes and detectors. Interactions between photons and the medium along their paths do not only affect the photons themselves, but the matter as well. These interactions with baryons exchange energy and momentum, and also modify their excitation and ionization states, thus impacting the chemical and thermodynamic properties of the matter. This in turn makes radiation a driving factor in many of the astrophysical processes we study.

Despite its importance, RT is treated rather poorly in many astrophysical simulations, often as an imposed uniform background. This is because RT is inherently a difficult problem. The full RT problem depends on 3 spatial dimensions, 2 angular dimensions, frequency and time, and has a characteristic speed of c, the speed of light. Gravity is also long range but RT is more complex because it depends on the intervening material. A typical ray-tracing solution for RT would scale with the number resolution elements N like $O(N^2)$. Ideally, the scaling would be similar to current gravity and hydrodynamics solvers ($O(NlogN)$ or better) so approximations are needed.

We can separate numerical methods into two broad categories by looking at the classical RT equation (Mihalas & Mihalas 1984),

JW: I don't think we need the arguments for the variables

$$\left[\frac{1}{c}\frac{\partial}{\partial t} + \hat{\mathbf{n}}\cdot\nabla\right] I\left(\mathbf{x},\hat{\mathbf{n}},t,\nu\right) = \epsilon\left(\mathbf{x},\hat{\mathbf{n}},t,\nu\right) - \alpha\left(\mathbf{x},\hat{\mathbf{n}},t,\nu\right) I\left(\mathbf{x},\hat{\mathbf{n}},t,\nu\right), \quad (1)$$

where I, ϵ and α are the intensity, emissivity and extinction coefficient respectively which all depend on position $\mathbf{x}$, unit direction of light propagation $\hat{\mathbf{n}}$, time t and frequency ν. The difference comes from how different methods treat the speed of light c. For methods that use a finite c, which is often a reduced speed of light, the partial time derivative remains in equation 1 and the radiation field is stored at all points in space and evolved. Methods that solve the RT equation in this way, which we will call evolutionary **alternate name: updated, or something that reflects fact that radiation field must be remembered/stored** methods, include moment methods like OTVET (Gnedin & Abel 2001) and RAMESE-RT (Rosdahl & Teyssier 2015) as well as photon packet propagation methods like TRAPHIC (Pawlik & Schaye 2008), SPHRAY (Altay et al. 2008) and SimpleX2 (Paardekooper et al. 2010).

On the other hand, in limit where c is taken to be infinite, the partial time derivative in equation 1 goes to zero and the radiation field can be computed instantaneously from knowledge of the sources. [Methods that solve the RT equation in this way, which we will refer to as instantaneous methods,] **JW: I am not aware of anything in this category that is not ray tracing. Let's call it ray tracing**. The most obvious approach to estimate the instantaneous radiation field is via forward ray tracing from sources. Forward ray tracers include C^{2}Ray (Mellema et al. 2006), Moray (Wise & Abel 2011) and Fervent (Baczynski et al. 2015). **limitations of forward ray tracers can go here – and advatnages, no need to store radiation field or be limited by RT timestep** Reverse reverse ray tracers include as TreeCol (Clark et al. 2012) and URCHIN (Altay & Theuns 2013). OTVET uses the instantaneous radiation field to estimate its Eddington tensor and is thus a hybrid method.

All of these methods use approximations to the full RT problem to make it feasible to solve that also limit the kinds of problems they are well-suited for. This makes is harder to compare these methods on an equal footing. Some authors state how their method scales with resolution elements in the context of the problem their RT method is meant to solve. **Think key is spatial resolution elements vs rays vs no. sources. Clarke+ is not a good starting point – start with a simpler example. For example, standard ray tracing.**.

suggested place to start to make scaling discus-

sion easy to follow: To facilitate comparisons we will consider scaling for a volume with uniform resolution such as a regular three-dimensional grid with N resolution elements. A naive ray-tracing approach with each element as both a source and sink would scale as $O(N^2)$. However, this involves casting N rays from each source whereas only $O(N^{2/3})$ rays (proportional to the surface area of the volume) are required to hit every other element at least once. With absorption, each ray intersects $O(N^{1/3})$ elements along the way for a standard ray tracing scaling of $O(N^2)$.

out of the blue – needed more context For example, Clark et al. (2012) state that their reverse ray tracer TreeCol scales with resolution elements like $O(N \log N)$. The specific problem they are trying to solve is the calculation of column densities via RT from background radiation, and thus the number of sources of radiation they simulate is some small fixed number. They treat the number of sources as a constant factor in their scaling function, and so what is an $O(N^2)$ method in the general case looks like a much more feasible method at first glance. For this reason we will be very careful with scaling statements, using the number of radiation sources $N_{\rm source}$ and sinks $N_{\rm sink}$ where appropriate to make the limitations and possible uses of a given method clear.

Instantaneous methods are typically ray tracers. Ray tracers are the most simple, natural way to go about solving the RT problem. Forward ray tracers trace many rays outward from sources of radiation, similarly to the actual phenomena, in hope that they will intersect resolution elements for which the radiation field will be computed. Each source needs to compute a number of rays comparable to the number of resolution elements to ensure accuracy, meaning forward ray tracers scale with number of resolution elements like $O(N_{\rm source}N_{\rm sink})$. This scaling limits forward ray tracers to problems with few sources to avoid $O(N^2)$ like scaling. This also rules out the inclusion of scattering in the method as scatterings are treated as re-emission events and thus all sinks would have to be treated as sources as well.

No repeats! – introduce this once only so the intro flows. Recently there has been some focus on reverse ray tracing methods (Clark et al. 2012; Altay & Theuns 2013). Reverse ray tracers trace rays from the sink particle directly to the sources. This way of tracing the rays has a couple of benefits over forward ray tracing. Firstly, tracing from the sinks guarantees the density distribution is well sampled as apposed to forward ray tracing where one would have to increase the number of rays per sink to guarantee accuracy. Put simply, radiation is computed exactly where it is needed. This is especially advantageous in smoothed particle hydrodynamics (SPH) simulations, as low density regions are represented by few SPH particles, and thus extra work is not done to resolve said regions. Another benefit is that sub time steps can be used. However, just performing a naive reverse ray trace does not negate $O(N_{\rm source}N_{\rm sink})$ scaling with resolution elements, and so the inability to model many sources remains the most significant barrier current instantaneous methods face when trying to solve the general RT problem.

Evolutionary methods do not suffer from the linear dependence on number of sources. The main benefit of evolutionary methods is that they have no dependence on the number of sources, and just scale like $O(N)$ with the number of resolution elements, allowing them to handle large numbers of sources and scattering. Moment methods are limited to the optically thick diffusive limit. They also lack sharp directionality, resulting in poor shadows behind optically thick objects. This also makes moment methods reliant on the partitioning of space into uniform grids. If implemented in a smooth particle hydrodynamics (SPH) like scheme, the lack of resolution elements in less dense regions would only exacerbate the directionality problem.

Photon packet propagation methods, specifically TRAPHIC (Pawlik & Schaye 2008), perform better in the optically thin regime. TRAPHIC introduces virtual particles (ViPs) to propagate their photon packets in less dense, optically thin regions lacking in SPH particles. They also preserve directionality quite well, however Monte Carlo aspects of how they propagate their photon packets introduce significant noise into their computed radiation field. Monte Carlo resampling is shown to reduce this noise but is quite expensive and deteriorates the initially sharp shadows. Both of these methods scale linearly with resolution elements as mentioned before, but are also forced to operate on every resolution element. In moment methods the radiation field for every grid cell needs to be computed, and in photon packet propagation methods the photon packets hop particle to particle. In the case of TRAPHIC, their N is even greater than the number of SPH particles including the addition of ViPs. Regardless, TRAPHIC is arguably the best general RT method due to its ability to handle both the optically thick and thin regimes with feasible scaling.

We hope from this introduction to the state of the art in RT methods it is apparent that there is room for improvement. Although promising work has been done with reverse ray tracers like TreeCol, a general implementation of one has yet to be published. There is also the problem of scaling with sources in instantaneous methods. If this could be solved, instantaneous ray tracers could compete with the feasibility and improve upon the accuracy of evolutionary codes like TRAPHIC.

In this paper we introduce TREVR (Tree-based Reverse Ray Tracing), a novel method that hopes to address all of the issues presented in the previous paragraph. TREVR is a reverse ray tracer based on the Barnes & Hut (1986) tree gravity solver. TREVR is implemented in the SPH code Gasoline (Wadsley et al. 2004), but we would like to note that the basic algorithm is not SPH or Gasoline specific, TREVR only requires that the simulation volume be divided hierarchically in space.

This is not intro material as written. It might be in intro if it were more general – about reverse tree rather than /acro. You should not be not saying why acro is better – you are setting up the existence of gaps in prior methods that could be fixed by a "New method" which we introduce in what follows. You do detailed "why we are better' in the summary/discussion section.

A main feature of TREVR is that sources can be merged via an opening angle criteria similarly to the Barnes and Hut gravity method. This means that TREVR scales with sources like $O(\log N_{\rm source})$, alleviating the linear scaling with sources characteristic of other instantaneous RT methods. Being a reverse ray tracer enables us to easily merge sources, as sources interact with a single sink at a time. This is the first time source merging has been implemented in a reverse

ray tracer, and thus TREVR is the first general reverse ray tracer that can feasibly operate on all sources in a simulation and not just a small, fixed amount of background sources as in TreeCol and URCHIN. **mischaracterization of treecol and urchin – they don't have any sources, just assume rays coming from infinity**

TREVR traces rays per each sink because it is a reverse ray tracer, this adds a linear dependence on the number of sinks. This coupled with source merging means TREVR can solve the optically thin case $O(N_{\rm sink} \log N_{\rm source})$ operations. The fact that TREVR depends on a tree-data structure allows it to trace rays through an optically thick medium by traversing the sub-tree between the sink and source which is at most $\log N$ tree cells. This results in scaling with resolution elements in the optically thick case that goes like $O(N_{\rm sink} \log N_{\rm source} \log N)$. This scaling in both the optically thin and thick case makes TREVR a feasible method as it scales similarly to modern Gravity and Hydro solvers.

In the coming sections we will further explain the general method in section 2 and its implementation in section 3. Tests of scaling with resolution elements and the method's accuracy as a function of opening angle and refinement criteria will be presented in section 4. At the end of section 4 we will do our best to compare TREVR to other methods in common RT tests such as the Strömgren sphere test and dense clump tests used in the Iliev et al. (2006) RT method comparison project. Finally, in section 5 we will discuss the method's merits and drawbacks to conclude where TREVR fits in with other RT methods and what scientific problems it can solve.

2 METHOD

3 IMPLEMENTATION

Temporarily commented out We now present the TREVR algorithm. TREVR prioritizes the ability to deal with a large number of sources over high accuracy, though we still insist equilibrium behaviour be correct. In order to accomplish this, we start by making some simplifying assumptions to the radiative transfer equation

$$\frac{dI_\nu}{ds} = -(\alpha_\nu + \sigma_\nu)(I_\nu - S_\nu)\,, \tag{2}$$

where α_ν and σ_ν are the specific absorption and scattering coefficients respectively, S_ν is the combined source function for absorption and scattering, and I_ν is the specific intensity.

We can simplify TREVR to only include absorption. This still allows us to treat scattering, as it can be expressed as an absorption followed by an emission. For now though, we only account for absorption by treating star particles as emitters and gas particles as absorbers. In the future one could implement scattering by simply treating gas particles as emitters *and* absorbers (more discussion on this in section 5). This reduces equation 2 to

$$\frac{dI_\nu}{d\tau_\nu} = -I_\nu + S_\nu, \tag{3}$$

where τ_ν is absorption-only optical depth and is defined as

$$d\tau_\nu = \alpha_\nu ds = \rho\kappa_\nu ds, \tag{4}$$

where ρ is density and κ_ν is specific opacity due to absorption.

It is useful to consider only absorption, as many astrophysical simulations only model a single or few sources. This causes the emission coefficient to be zero at most points. In our case we only consider star particles to be emitters, meaning we have a non emitting medium along a ray. This reduces equation 3 by setting the source function to 0 and setting I_ν to be the initial intensity from a single star particle $I_\nu(0)$

$$I_\nu(\tau_\nu) = I_\nu(0)e^{-\tau_\nu}. \tag{5}$$

This now allows us to turn the initial integral over all sources to a sum of diminished contributions from each star particle.

We compute the radiation field as a flux magnitude, as this is the quantity most useful to applications of our algorithm. When expressed as a flux 5 becomes,

$$F_\nu = \frac{L_\nu}{4\pi s^2} e^{-\tau_\nu}, \tag{6}$$

where L_ν is the specific luminosity of the star particle and s is the distance between the star particle and the absorbing gas particle. We can then sum this equation over all contributing star particles to find the flux at the gas particle in question. This summation is what TREVR computes for all gas-type SPH particles in a simulation, thus approximating the radiation field.

Please note that TREVR has been implemented in the Smoothed Particle Hydrodynamics (SPH) code Gasoline (Wadsley et al. 2004). However, TREVR is *not* specific to Gasoline or SPH. We only require that the simulation volume can be hierarchically partitioned in space.

3.1 The Optically Thin Regime

In the absence of absorbing material, the optical depth is zero and equation 6 becomes just $L/4\pi s^2$. This problem is almost identical to gravity, and so we use the same tree-based technique as gravity to solve it. The tree-based gravity solver of Barnes & Hut (1986) has become commonplace in astrophysical simulations (Hubber et al. 2011; Wadsley et al. 2004; Springel et al. 2001; Vine & Sigurdsson 1998; Benz 1988). Like the Barnes & Hut algorithm our optically thin method should scale with resolution elements like $O(N \log N)$. In our case the N factor represents the number of gas particles ($N_{\rm sink}$), and the $\log N$ factor represents the number of emitting particles ($N_{\rm source}$). This means that for the optically thin case we should see scaling with number of sinks and sources go as $O(N_{\rm sink} \log N_{\rm source})$.

In our Gasoline implementation we use a binary tree. During the tree build process we can compute useful average properties of tree cells such as total luminosity, centre of luminosity, average density and average opacity (the latter two are used in the optically thick regime). Computing the radiation field is accomplished by traversing the tree structure. Receiving gas particles which live in the leaf nodes of the tree are looped over. An opening angle criterion, just as in gravity, is used to decide on how the gas particles interact with the emitters.

Note that we already have used early versions of this algorithm to investigate the effects of ionizing feedback on gas cooling in galaxies (Kannan et al. 2014).

3.2 The Optically Thick Regime

In the presence of absorbing material along the ray, we need to compute the optical depth along said ray. To do this we traverse the tree from the interacting nodes to their common parent node to build up the optical depth along the ray. This is possible because the tree is partitioned in space, thus all intervening material should be contained in the sub-tree we traverse. Using the average properties computed in the tree build we can compute the optical depth of a piece of the ray using the geometry of the cell and ray, and the average density and opacity

$$\tau_i = \bar{\rho}_i \bar{\kappa}_i s_i . \tag{7}$$

The total optical depth is then summed up during the tree walk,

$$\tau = \sum_i \tau_i , \tag{8}$$

giving us everything needed to evaluate equation 6. Because of this extra tree walk another $\log N$ factor can be added to the scaling equation and so we would expect the scaling with resolution elements to now look like $O(N_{\rm sink} \log N_{\rm source} \log N)$. This algorithm is depicted in the left half of figure 1

Since we are calculating the radiation field at the receiving cell we are doing a process similar to a reverse ray trace, like URCHIN (Altay & Theuns 2013). **too much credit given to urchin here**. One advantage of reverse ray tracing is that rays are associated with with sinks rather than the source. Dense regions near the sink and source are therefore automatically well sampled and radiative transfer is computed exactly where it needs to be. Another benefit is that the simulation can make use of sub time steps. The problem with this simple method is that as the tree is traversed upwards the volume elements become larger and accuracy can be lost. That makes this algorithm very efficient at computing the radiation field for uniform density and opacity distributions, but highly inaccurate when dealing with sharp density and opacity gradients along the ray. To handle these situations a method of refinement is needed.

3.3 Refinement

Refinement is a straightforward addition to the algorithm. At a point in the tree walk where the average properties of the cell would be considered, we check to see if the current cell passes some refinement criteria. If the cell passes the criteria to refine, rather than using the average properties we recursively check the cell's children until the criteria fails building a better resolved section of the ray. This addition to the algorithm is depicted in the right half of figure 1

Difficulty comes in choosing a refinement criteria that is both accurate and efficient. Ideally, the criteria should be true when an average optical depth in a region may not be accurate to the true distribution, such as a clumpy medium where the average opacity is much higher than the "effective" opacity (Hegmann & Kegel 2003; Városi & Dwek 1999).

Our choice of refinement criteria is based on optical depth, and is unique to the TREVR algorithm. Consider two rays through a large cell (see figure 2, note that this description is simplified to 2D). These rays represent what the case would be if the properties of the children were used instead of the parent cell. We can calculate the minimum and maximum absorption coefficients $\alpha_{\rm min}$ and $\alpha_{\rm max}$, via their average density and opacity values computed during the tree build. This multiplied by the intersection l, gives us the minimum and maximum optical depths, $\tau_{\rm min}$ and $\tau_{\rm max}$. We can then test the following refinement criteria

$$\tau_{\rm refine} < \tau_{\rm max} - \tau_{\rm min}, \tag{9}$$

and refine if it is true. The fractional error in flux for a chosen value of $\tau_{\rm refine}$ is

$$\text{FractionalError} = \frac{F_1 - F_2}{F_1} \leq 1 - e^{(-\tau_{\rm max} - \tau_{\rm min})} < \tau_{\rm refine}, \tag{10}$$

for small τ, making the refinement criteria a convenient choice of parameter for guaranteeing accuracy. This criteria is conservative, as it assumes the worst case difference in optical depth. We suspect there is room for improvement in terms of efficiency.

If very high accuracy is required, sub-leaf node refinement is possible. If a leaf was reached during refinement and still passes the refinement criteria, the individual particles in the leaf can be considered. A ray tracing scheme through the cell similar to SPHRay (Altay et al. 2008) can be performed. The machinery to do this is implemented in TREVR.

3.4 Cosmological Background Radiation

In order to treat cosmological simulations properly we must account for the radiation coming from the rest of the universe outside of the simulation volume. Most current codes apply a constant UV field to the entire box, essentially the lowest order approximation possible. Some specialized codes like URCHIN (Altay & Theuns 2013) do a reverse ray trace to the edge of the box, where the background flux is assumed to be coming from. Others, such as TRAPHIC (Pawlik & Schaye 2008) allow their ray trace to be periodic. We believe that this periodic treatment is problematic for reasons we will explain at the end of this subsection.

Instead, we have implemented a method involving "background sources". "Background" particles are distributed in a spiral pattern on the surface of a sphere at the very edge of the simulation volume (or at a large distance if required) and the number of sources can be varied to match the required angular resolution of the background. Finding the flux at the centre of a sphere of sources is a problem akin to Newton's Shell Theorem. However, because the intensity does not cancel like force, the solution differs and is as follows:

$$F(r) = \frac{L}{8\pi R} \ln\left(\frac{R+r}{R-r}\right), \tag{11}$$

where L is the total luminosity of the emitting shell, R is the radius of the sphere and r is the radius the flux is being computed at. The shape of the function can be seen if Figure 3.4 where we have plotted the flux as a function of radius for a homogeneous, optically thin test volume.

We note that due to the logarithm in equation 11, the flux is nearly constant at small radii. Since most cosmological zoom in simulations only consider gas at a fairly small radius, this setup of background sources is an acceptable method to provide a cosmological background flux. A benefit of this method is that we can use all of the existing

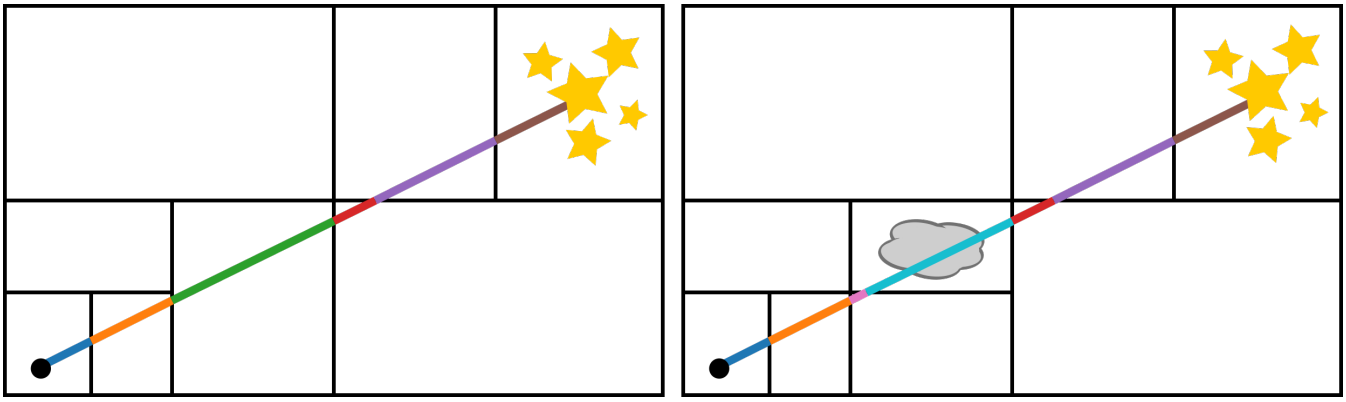

Figure 1. Depiction of the TREVR algorithm with and without the need for refinement (left and right respectively.)

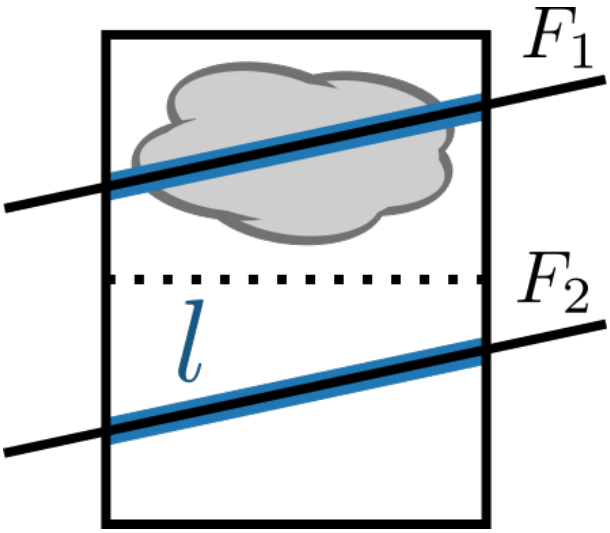

Figure 2. refine.pdf

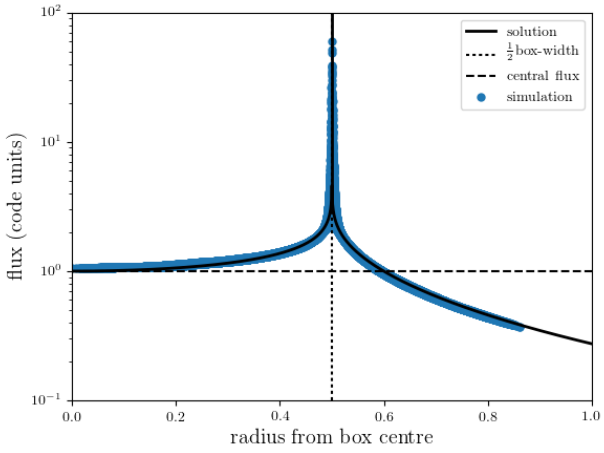

Figure 3. The distribution of flux that particles receive due to the cosmological background sources when distributed in a spherical shell on at the edge of the simulation box. Note that the value of the flux at the centre can be easily scaled by simply scaling $\boldsymbol{L}$, the luminosity of all sources on the sphere. The important property is the near constant flux at small radii. In this example, we have used 1024 background sources. The number of sources determines the width of the peak.

machinery described in the methods section, and only have to add temporary background star particles as the source of the background radiation. This way, there is no need to create periodic copies of the simulation volume. **explain <-this here**

4 CODE TESTS

5 DISCUSSION AND CONCLUSION

ACKNOWLEDGEMENTS

REFERENCES

Altay G., Theuns T., 2013, MNRAS, 434, 748
Altay G., Croft R. A. C., Pelupessy I., 2008, MNRAS, 386, 1931
Baczynski C., Glover S. C. O., Klessen R. S., 2015, MNRAS, 454, 380
Barnes J., Hut P., 1986, Nature, 324, 446
Benz W., 1988, Computer Physics Communications, 48, 97
Clark P. C., Glover S. C. O., Klessen R. S., 2012, MNRAS, 420, 745
Gnedin N. Y., Abel T., 2001, New Astron., 6, 437
Hegmann M., Kegel W. H., 2003, MNRAS, 342, 453
Hubber D. A., Batty C. P., McLeod A., Whitworth A. P., 2011, A&A, 529, A27
Iliev I. T., et al., 2006, MNRAS, 371, 1057
Kannan R., et al., 2014, MNRAS, 437, 2882
Mellema G., Iliev I. T., Alvarez M. A., Shapiro P. R., 2006, New Astron., 11, 374
Mihalas D., Mihalas B. W., 1984, Foundations of radiation hydrodynamics
Paardekooper J.-P., Kruip C. J. H., Icke V., 2010, A&A, 515, A79
Pawlik A. H., Schaye J., 2008, MNRAS, 389, 651
Rosdahl J., Teyssier R., 2015, MNRAS, 449, 4380
Springel V., Yoshida N., White S. D. M., 2001, New Astron., 6, 79
Városi F., Dwek E., 1999, ApJ, 523, 265
Vine S., Sigurdsson S., 1998, MNRAS, 295, 475
Wadsley J. W., Stadel J., Quinn T., 2004, New Astron., 9, 137
Wise J. H., Abel T., 2011, MNRAS, 414, 3458

This paper has been typeset from a TEX/LATEX file prepared by the author.