Generating dark matter halo merger trees

Hannah Parkinson, Shaun Cole* and John Helly

Institute of Computational Cosmology, Department of Physics, University of Durham, South Road, Durham DH1 3LE

Accepted 2007 September 25. Received 2007 September 24; in original form 2007 August 10

ABSTRACT
We present a new Monte Carlo algorithm to generate merger trees describing the formation history of dark matter haloes. The algorithm is a modification of the algorithm of Cole et al. used in the GALFORM semi-analytic galaxy formation model. As such, it is based on the Extended Press–Schechter theory and so should be applicable to hierarchical models with a wide range of power spectra and cosmological models. It is tuned to be in accurate agreement with the conditional mass functions found in the analysis of merger trees extracted from the /Lambda cold dark matter Millennium N-body simulation. We present a comparison of its predictions not only with these conditional mass functions, but also with additional statistics of the Millennium Simulation halo merger histories. In all cases, we find it to be in good agreement with the Millennium Simulation and thus it should prove to be a very useful tool for semi-analytic models of galaxy formation and for modelling hierarchical structure formation in general. We have made our merger tree generation code and code to navigate the trees available at http://star-www.dur.ac.uk/~cole/merger_trees.

Key words: methods: numerical – cosmology: theory – dark matter.

1 INTRODUCTION
In hierarchical models of structure formation, such as /Lambda cold dark matter (ACDM), the formation of a DM halo through accretion and repeated mergers can be described by a merger tree (Lacey & Cole 1993). The merger trees, which list the progenitors of a given halo at a series of redshifts and describe the sequence in which they merge together, contain essentially all the information one needs about the DM when building models of the other processes involved in galaxy formation. Thus, the merger trees, whether extracted from N-body simulations such as the Millennium Simulation (MS; Springel et al. 2005) or generated by Monte Carlo (MC) algorithms (e.g. Sheth & Lemson 1999; Somerville & Kolatt 1999; Cole et al. 2000), provide the framework within which one can model the additional astrophysical processes of galaxy formation (Cole et al. 2000; Kauffmann & White 1993; Somerville & Kolatt 1999).

The statistical properties of MC merger trees based on the approximate Extended Press Schechter (EPS) theory (Bond et al. 1991; Bower 1991; Lacey & Cole 1993) are not in perfect agreement with those built from high-resolution, highly non-linear N-body simulations (e.g. Cole et al. 2007). We have found that when the same semi-analytic galaxy formation model is run first on MC trees and then with N-body trees there can be significant differences in the properties of their resulting galaxy populations. In some ways, these differences are minor as small changes in the uncertain parameters of the star formation and feedback prescriptions can often bring them back into alignment. However, it would be far better if MC and N-body merger trees were in much better agreement. For instance, this would allow galaxy formation models to be run first on MC trees and parameters including cosmological parameters tuned to match observed galaxy properties in advance of running an expensive N-body simulation which will furnish the positional information needed to make galaxy clustering predictions. Additionally, as N-body simulations always have poorer mass resolution than can be obtained with MC merger trees, one would like to be able to use MC trees of varying resolution to assess the impact of the limited resolution of the N-body simulation. This can be hard to achieve when the two sets of trees differ systematically.

In this paper, we present a modification to the MC merger tree algorithm of Cole et al. (2000) that we tune to be in accurate agreement with the statistical properties of the MS merger trees that were presented in Cole et al. (2007). In Section 2, we describe both the original EPS MC algorithm as implemented in Cole et al. (2000) and our modification. Section 3 compares the results of our modified algorithm with the original and with the statistics of merger trees from the MS. We briefly discuss relationship of our algorithm to other models in Section 4 and conclude in Section 5.

2 THE MONTE CARLO ALGORITHM
In the following sections, we briefly review the MC algorithm implemented in the GALFORM semi-analytic code (Cole et al. 2000) and then describe how we modify it to achieve more accurate agreement with simulation data.
2.1 The GALFORM algorithm

The merger tree algorithm employed in the GALFORM semi-analytic model uses as its starting point the conditional mass function

\[
f(M_1|M_2) \text{d} \ln M_1 = \frac{2}{\pi} \frac{\sigma_1^2}{\sigma_1^2 - \sigma_2^2} \left| \frac{\text{d} \ln \sigma_1}{\text{d} \ln M_1} \right| \text{d} \ln M_1, \quad (1)
\]

given by the EPS theory (Bond et al. 1991; Bower 1991; Lacey & Cole 1993). Here \(f(M_1|M_2)\) represents the fraction of mass from haloes of mass \(M_2\) at redshift \(z_2\) that is contained in progenitor haloes of mass \(M_1\) at an earlier redshift \(z_1\). The linear density thresholds for collapse at these two redshifts are \(\delta_1\) and \(\delta_2\) (e.g. Eke, Cole & Frenk 1996). The rms linear density fluctuation extrapolated to \(z = 0\) in spheres containing mass \(M\) is denoted \(\delta(M)\) with \(\sigma_1 \equiv \sigma(M_1)\) and \(\sigma_2 \equiv \sigma(M_2)\). Taking the limit of \(f(M_1 | M_2)\) as \(z_1 \to z_2\), one finds

\[
\frac{df}{dz_1} \bigg|_{z_1=z_2} = \left| \frac{\text{d} \ln \sigma_1}{\text{d} \ln M_1} \right| \text{d} \ln M_1 \text{d}z_1, \quad (2)
\]

which implies that the mean number of haloes of mass \(M_1\) into which a halo of mass \(M_2\) splits when one takes a step \(dz_1\) up in redshift is

\[
\frac{dN}{dM_1} = \frac{1}{M_1} \frac{df}{dz_1} \left[ \frac{M_2}{M_1} \right] (M_1 < M_2). \quad (3)
\]

By specifying a required mass resolution, \(M_{\text{res}}\), for the algorithm, one can then integrate to determine

\[
P = \int_{M_{\text{res}}}^{M_2/2} \frac{dN}{dM_1} \text{d}M_1, \quad (4)
\]

which is the mean number of progenitors with masses \(M_1\) in the interval \(M_{\text{res}} < M_1 < M_2/2\) and

\[
F = \int_0^{M_{\text{res}}} \frac{dN}{dM_1} M_1 \frac{dM_1}{M_2} \text{d}M_1, \quad (5)
\]

which is the fraction of mass of the final object in progenitors below this resolution limit. Note that both these quantities are proportional to the redshift step, \(dz_1\), by virtue of equation (3).

The GALFORM merger tree algorithm then proceeds as follows. First, choose a mass and redshift, \(z\), for the final halo in the merger tree. Then, pick a redshift step, \(dz_1\), such that \(P \ll 1\), to ensure that the halo is unlikely to have more than two progenitors at the earlier redshift \(z + dz\). Next, generate a uniform random number, \(R\), in the interval 0 to 1. If \(R > P\), then the main halo is not split at this step. We simply reduce its mass to \(M_2(1 - F)\) to account for mass accreted in unresolved haloes. Alternatively, if \(R \leq P\), then we generate a random value of \(M_1\) in the range \(M_{\text{res}} < M_1 < M_2/2\), consistent with the distribution given by equation (3), to produce two new haloes with masses \(M_1\) and \(M_2(1 - F) - M_1\). The same process is repeated on each new halo at successive redshift steps to build up a complete tree. More details are given in Appendix A.

2.2 The modified algorithm

The binary merger algorithm described above fully respects a natural symmetry that whenever one fragment has mass \(M_1\) the other must have mass \(M_2 - M_1\) (at least in the limit of \(M_{\text{res}} \to 0\)). This means that it is not consistent with the EPS theory as equation (3) does not satisfy this symmetry of remaining unchanged when \(M_1 \to M_2 - M_1\) (Lacey & Cole 1993; Benson, Kimminkowski & Hassani 2005). To force the required symmetry, the algorithm only uses equation (3) for \(M_1 < M_2/2\) and ignores its predictions for \(M_1 > M_2/2\). The algorithm is also unsatisfactory because the EPS conditional mass functions, and also the original Press–Schechter mass function, do not accurately match what is found in N-body simulations (Sheth & Tormen 1999; Jenkins et al. 2001; Cole et al. 2007). However, many statistical properties of the merger trees produced by the above algorithm have trends with mass and redshift that agree well with those of merger trees constructed from high-resolution N-body simulations, but with increasing redshift they systematically underestimate the mass of the most massive progenitors (Cole et al. 2007). (In practice, the problem is ameliorated in GALFORM by starting the merger tree construction at higher redshift.) Here, our aim is to reduce these systematic differences.

Given the simplicity and zeroth order success of the original GALFORM algorithm, it seems reasonable to try modifying it by perturbing the basic function that drives the algorithm. Namely we consider replacing the function defined in equation (3) by making the substitution

\[
\frac{dN}{dM_1} \to \frac{dN}{dM_1} G(\sigma_1/\sigma_2, \delta_2/\sigma_2). \quad (6)
\]

Here \(G(\sigma_1/\sigma_2, \delta_2/\sigma_2)\) is the ‘perturbing’ function which we expect to be of the order of unity for most of the range of interest. The choice that the function \(G\) should only depend on the ratios \(\sigma_1/\sigma_2\) and \(\delta_2/\sigma_2\) is motivated by the desire that the algorithm should preserve self-similarity if used in a flat \(\Omega_m = 1\) cosmology with scale-free initial conditions (e.g. see Efstathiou et al. 1988). The dependence on \(\delta_2/\sigma_2\) allows the halo-splitting rate to be modified as a function of \(M_2/M_\ast\), where the characteristic non-linear mass, \(M_\ast\), is defined by \(\sigma(M_\ast) = \delta\), while the dependence on \(\sigma_1/\sigma_2\) allows the mass distribution of the resulting fragments to be modified. Restricting the dependence of the function to only these parameters is necessary to preserve self-similarity, but on its own does not guarantee self-similarity. The additional unwanted freedom we hope to remove by fitting to the statistical properties of the MS merger trees, as presented in Cole et al. (2007). Note that since the merger tree algorithm described above only makes use of equation (6) for progenitor masses \(M_1 < M_2/2\) it is only the behaviour of \(G(\sigma_1/\sigma_2, \delta_2/\sigma_2)\) for \(M_1 < M_2/2\) (\(\sigma_1 > \sigma_2\)) that is constrained by comparison to the MS merger trees. Consequently, the predictions of equation (6) for \(M_1 > M_2/2\) are of no relevance.

To simplify the problem still further, we make the assumption that

\[
G(\sigma_1/\sigma_2, \delta_2/\sigma_2) = G_0 \left( \frac{\sigma_1}{\sigma_2} \right)^{n_1} \left( \frac{\delta_2}{\sigma_2} \right)^{n_2}, \quad (7)
\]

which can be considered as a first-order Taylor series approximation for \(\ln G\) in terms of \(\ln (\sigma_1/\sigma_2)\) and \(\ln (\delta_2/\sigma_2)\). This functional form is particularly convenient. The two terms \(G_0\) and \((\delta_2/\sigma_2)^{n_2}\) have no dependence on \(M_1\) and so just enter the integrals in equations (4) and (5) as multiplicative constants. The term \((\sigma_1/\sigma_2)^{n_1}\) alters the distribution \(dN/dM_1\) and the integrands in both equations (4) and (5), but has simple analytic properties that allow a very fast implementation of the splitting algorithm (see Appendix A).

3 COMPARISON WITH THE MILLENNIUM SIMULATION

The MS (Springel et al. 2005) is, to date, the largest N-body simulation of a cosmologically representative volume. It uses \(N = 2160^3\) particles.
particles in a comoving cube of side \( L = 500 \, \text{h}^{-1} \, \text{Mpc} \) to follow the non-linear gravitational evolution of a Gaussian random density field drawn from a power spectrum consistent with cosmological constraints from 2dFGRS (Percival et al. 2001) and the first-year WMAP data (Spergel et al. 2003). The cosmological density parameters are \( \Omega_m = 0.25, \Omega_b = 0.045 \) and \( \Omega_{\Lambda} = 0.75 \), the Hubble parameter \( h = H_0/100 \, \text{km} \, \text{s}^{-1} \, \text{Mpc}^{-1} = 0.73 \) and the linear amplitude of the density fluctuations in spheres of radius \( 8 \, \text{h}^{-1} \, \text{Mpc} \) is \( \sigma_8 = 0.9 \). At each of over 60 output times, a catalogue of friends-of-friends (FOF; Davis et al. 1985) groups was constructed and the descendant of each group found at the subsequent time-step. Details of the construction of these merger trees and and their statistical properties can be found in Cole et al. (2007). Below we compare a variety of the statistics they estimated with the results of the original Cole et al. (2000) GALFORM MC algorithm and our new modified algorithm.

### 3.1 Conditional mass functions

In Fig. 1, we compare the conditional mass functions of the MS merger trees with those of the MC algorithms. Here, for haloes of various masses \( M_1 \) at redshift \( z_2 = 0 \) we find what a fraction of their mass is in progenitor haloes of mass \( M_1 \) at various earlier redshifts \( z_1 \). The histograms show the results of the MS while the solid curves show an analytic fit described in Cole et al. (2007). The results of the original GALFORM algorithm are shown by the dotted curves. As has been noted by Cole et al. (2007), these conditional mass functions evolve more rapidly than those of the simulation. Thus, the ‘GALFORM 2000’ algorithm strongly underpredicts the number of high-mass progenitors at high redshift. The fact that the EPS theory gives predictions that evolve more rapidly with redshift than is found in N-body simulations has been noted previously (e.g. van den Bosch 2002; Wechsler et al. 2002; Lin, Jing & Lin 2003). Giocoli et al. (2007) have shown that average halo formation times agree better with the elliptical collapse model of Sheth & Tormen (1999) than with the spherical collapse EPS formalism. Furthermore, Giocoli et al. (2007) find that scaling the time-variable in the EPS theory by the factor \( \sqrt{q} = \sqrt{0.707} = 0.84 \) that comes from fitting the elliptical collapse model to N-body data results in formation-time predictions that better match the N-body data for a wide range of final masses and redshifts. By reference to equation (3), it can be seen that our factor \( G(\sigma_1/\sigma_2, \delta_1/\delta_2) \) of equation (6) can be viewed as a modification to the time-step \( \delta z_1 \). Thus, the elliptical collapse modelling of Giocoli et al. (2007) suggests that we should find \( G(\sigma_1/\sigma_2, \delta_1/\delta_2) \approx 0.84 \). In fact, we expect a somewhat lower value as the Monte Carlo trees of the original GALFORM algorithm evolve even more rapidly than the analytic predictions of the EPS formalism (Cole et al. 2007). To find the best-fitting parameters, we have minimized the rms difference

\[
\sigma_{\text{rms}} = \left( \frac{1}{12} \sum (f_{\text{MS}}(M_1 \mid M_2) - f_{\text{MC}}(M_1 \mid M_2))^2 \right)^{1/2}
\]

between MS data and the results of the MC algorithm over all 12 panels in Fig. 1. With the exception of the lowest two mass bins plotted in each panel, which were discarded as they are influenced by the mass resolution of the MS, equal weight was given to each bin. Initially, we kept \( \gamma_1 = \gamma_2 = 0 \) fixed and allowed only \( G_0 \) to vary. For \( G_0 = 1 \), the original GALFORM algorithm, the rms difference \( \sigma_{\text{rms}} = 0.27 \). The best-fitting value of \( G_0 \) is 0.79, as anticipated, somewhat smaller than the factor 0.84 from Giocoli et al. (2007), and this reduces the rms difference significantly to \( \sigma_{\text{rms}} = 0.12 \). However, this is a compromise and the data from the different panels of Fig. 1 prefer different values of \( G_0 \). This can be accommodated by allowing \( \gamma_1 \) and \( \gamma_2 \) to vary. As \( \delta_1/\sigma_2 \) is an increasing function of the final mass \( M_2 \), a positive \( \gamma_2 \) would give a relatively higher merger rate for high mass, \( M > M_* \), haloes (where the character mass, \( M_* \), has the usual definition of \( \sigma(M_*) = \delta \)). Choosing \( \gamma_1 > 0 \) skews the shape of the progenitor mass functions by boosting the ratio of low mass to high mass progenitors. Since \( \sigma_1 > \sigma_2 \), setting \( \gamma_1 > 0 \) boosts the overall merger rate and so needs to be compensated for by the lower value of \( G_0 \). Allowing all three parameters to vary, consistently good fits are found with \( G_0 = 0.57, \gamma_1 = 0.38 \) and \( \gamma_2 = -0.01 \) with a reduced rms deviation from the MS data of \( \sigma_{\text{rms}} = 0.055 \). Over most of the range of which it is employed \( G(\sigma_1/\sigma_2, \delta_1/\delta_2) \) remains less than but of the order of unity. The conditional mass functions produced by this new set of trees are shown by the dashed lines in Fig. 1. We see that this minor change to the merger tree algorithm has resulted in merger trees that are in good agreement with the N-body simulation results over a wide range of masses and redshifts. The only mass bins where the MC and MS conditional mass functions are not in good agreement are the bins with \( M_1 > M_2 \). In a truly hierarchical model such as is produced by our algorithm, \( M_1 \) can never be greater than \( M_2 \). In the MS data, Cole et al. (2007) noted that \( M_1 > M_2 \) happens occasionally for low-mass haloes due to the temporary, premature linking of FOF groups.

### 3.2 Main progenitor mass functions

The parameters of our new MC merger tree algorithm were tuned to produce good agreement with the conditional mass functions plotted in Fig. 1 and so the level of agreement is perhaps not surprising. However, we can go further and test the success of the algorithm by testing other statistical properties of the merger trees. Fig. 2 plots the mass functions for the first and second most massive progenitors for the same selection of redshifts and final masses as in Fig. 1. This is an interesting test of the merger trees as often in galaxy formation applications it is the most-massive progenitors and mergers between them that are most important in determining the properties of the galaxies hosted by the haloes. It also tests an aspect of the merger trees that cannot be predicted by the EPS theory alone as it involves how often one has a progenitor of a given mass and not just the mean number of such progenitors. As noted in Cole et al. (2007), the mass functions for the first and second most-massive progenitors given by the original GALFORM algorithm do a good job of matching the shape and relative positions of the two distributions, but systematically underestimate both masses with increasing redshift. This is completely remedied in the new algorithm which matches the positions and shapes of the N-body distributions extremely accurately.

### 3.3 Major mergers

Another important aspect of the merger trees is the occurrence of major mergers. In galaxy formation models, major mergers between galaxies, which occur after halo mergers, are often deemed to be responsible for initiating bursts of star formation and for converting disc galaxies to spheroidal systems. Thus, it is interesting to see what level of agreement our new algorithm has with estimates from the MS. Fig. 3 compares the redshift distribution of the most-recent major merger for haloes of various final masses. Here, a major merger has been defined as a merger between two haloes where the smaller is at least fraction \( f_{\text{major}} = 0.3 \) of the mass of the larger. At each
Figure 1. The fraction of mass in progenitor haloes of mass $M_1$ in bins of $\log_{10} M_1/M_2$ at redshifts $z = 0.5, 1, 2$ and 4 as indicated, for three different masses $M_2$ (indicated at the top of each column). The histograms show the results from the MS while the dotted and dashed curves are the corresponding conditional mass functions given by the original GALFORM MC algorithm and our new modified algorithm, respectively. The solid curve shows an analytic fit to the whole set of conditional mass functions as described in Cole et al. (2007). The vertical dotted line indicates the 20-particle mass resolution of the MS.

redshift, we find the most massive progenitor of the final halo and record the lowest redshift at which one of these progenitors is undergoing a major merger.

It was found in Cole et al. (2007) that the original GALFORM algorithm, shown by the dotted line in Fig. 3, significantly overestimated the number of recent major mergers. Fig. 3 shows that this shortcoming is very largely overcome by our new algorithm. The redshift distributions of the most-recent major mergers match accurately the overall shape of those from the MS including their dependence on final halo mass.
Figure 2. The mass distributions of the first and second most-massive progenitors. The plotted quantities $f_{1st}$ and $f_{2nd}$ are the contributions to the overall conditional mass functions plotted in Fig. 1 provided by the first and second most-massive progenitors, respectively. The panels correspond directly to those of Fig. 1 and are labelled by the final halo mass $M_2$ and redshift $z_1$ of the progenitors. The histograms show the results from the MS with distribution $f_{1st}$ plotted with the heavy lines and $f_{2nd}$ with the light lines. The corresponding predictions of the GALFORM and new MC algorithms are shown by the heavy ($f_{1st}$) and light ($f_{2nd}$) dotted and dashed curves, respectively. The 20-particle mass resolution of the MS is shown by the vertical dotted line, but only plays a role for the $z = 4$ progenitors of the lowest mass, $M_2 = 10^{12} \, M_\odot$, haloes.

### 3.4 Overall mass functions

The above comparisons to the results of the MS test the mass range of the merger trees that is most important for galaxy formation applications, but it is also interesting to probe whether our new algorithm remains plausible for much larger ranges in mass. Sheth & Tormen (1999) (see also Jenkins et al. 2001) have shown that for a wide range of initial conditions and redshifts that the halo mass...
Figure 3. The redshift distribution of the most-recent major mergers of haloes with different final masses $M_2$. Here, a major merger is defined as a merger of the most-massive progenitor of the final halo with a second halo whose mass is at least $f_{\text{major}} = 0.3$ times that of the main progenitor. The histogram shows the $N$-body results and the dotted and dashed lines show the results from the original GALFORM and new MC algorithm, respectively.

matches the Sheth–Tormen prediction quite accurately over a wide range of redshifts. Note that for each merger tree mass function, the turnover at low masses (low $\nu$) is due to the specified mass resolution of the trees. At higher redshift, a fixed mass implies higher $\nu = \delta(z)/\sigma(M)$ and so the mass resolution causes deviations at higher and higher $\nu$.

4 DISCUSSION

It is perhaps surprising that an algorithm motivated by the EPS theory, which is only a function of the smoothed linear theory overdensity at a point, is able to accurately describe the complete merger histories of DM haloes in a fully non-linear $N$-body simulation. The EPS theory, as derived by Bond et al. (1991), makes the following series of assumptions none of which can rigorously be true. It assumes that virialized haloes form when the linear theory overdensity equals the threshold for collapse given by the pure spherical collapse model; the linear overdensity at a given point in space is assumed to vary with the smoothing scale as an uncorrelated (Brownian) random walk (the sharp $k$-space filtering approximation); when assigning mass points to haloes of mass $M$, no condition is set to require that these mass points should lie in spatially localized regions capable of forming haloes of that mass.

One might have thought that a more natural starting point for developing an accurate merger tree algorithm would have been the ellipsoidal collapse model of Sheth & Tormen (1999) as its mass function much more accurately matches that of $N$-body
simulations (although a free parameter, $q$, is adjusted to achieve this fit). However, it is not easy to work with this model as there is no simple analytic expression for the conditional mass function for small time-steps. Furthermore, the results of this more complicated model can often be approximated by minor modifications of the formulae that are derived using the EPS formalism. For example, Giocoli et al. (2007) have shown the inserting a factor of $\sqrt{q} = 0.84$ into the EPS formation time formula of Lacey & Cole (1993) results in a reasonable match to the predictions of the ellipsoidal collapse model. Our algorithm was motivated by the EPS formalism, but the modification we introduce in equation (6) means that its predictions are no longer those of the EPS formalism. If instead one were trying to come up with an algorithm based on the ellipsoidal collapse model, then the end result might well be very similar. In fact, as noted in Section 3.1, the $\sqrt{q} = 0.84$ factor advocated by Giocoli et al. (2007) is equivalent to our $G_0$ factor. The other assumptions of the EPS theory, listed above, and not addressed in the ellipsoidal collapse model must also play a role in determining merger histories. By adopting the modification defined in equation (6) and fitting directly to $N$-body simulation results, our model is fitting the net effect of all the additional physics and not just that due to the shape of the density perturbation.

After we completed this project Neistein & Dekel (2007) presented an alternative algorithm to generate DM halo merger trees based on fitting lognormal distributions to progenitor mass functions expressed in scaled mass and time variables. Their algorithm, which is of very comparable speed to ours, is also tuned to fit the conditional mass functions of merger trees extracted from the MS. There will be some differences in the results of the two algorithms because the MS merger trees used by Neistein & Dekel (2007) are not the simple FOF merger trees constructed for this project, but instead the ‘DHALO’ merger trees that were constructed by the Durham Group and used in the semi-analytic galaxy formation model of Bower et al. (2006). Both sets of trees are based on the same catalogues of FOF groups, but the ‘DHALO’ algorithm uses additional information concerning substructures identified using SUBFIND (Springel et al. 2001). (There is some discussion of the additional criteria useful for galaxy formation calculations in Harker et al. (2006).) We opted not to use these trees since a criterion that delays the time at which the merger is deemed to take place has the side effect of causing some haloes to lose mass prior to the merger. This artificially increases the occurrence of progenitor haloes that are more massive than their descendants and so slightly distorts the conditional mass functions.

5 CONCLUSIONS
We have presented a new MC algorithm to generate DM halo merger trees. The algorithm is a modification of the EPS algorithm described in Cole et al. (2000). The change we have made to the algorithm was motivated empirically and tuned to match the conditional mass function of haloes extracted from the MS (Springel et al. 2005). We find that not only can we get a very accurate match to these conditional mass functions over a wide range of mass and redshift, but that the other statistical properties of the new trees match very well those from the MS. The improvement in accuracy over the algorithm previously used in the GALFORM semi-analytic code (Cole et al. 2000) is very significant and should make the new algorithm a very useful tool.

While our algorithm has been tuned to match the results of MS, which is a particular $\Lambda$CDM model, we would expect it to be a significant improvement over EPS-based algorithms for quite a wide range of CDM-like initial conditions. The overly rapid evolution in the typical mass of progenitor haloes was a generic problem with the old algorithm and the reduced merger rate of the new algorithm should be an improvement in all cases. We have made a FORTRAN90 implementation of algorithm available at http://star-www.dur.ac.uk/~cole/merger_trees.

ACKNOWLEDGMENTS
We thank the referee, Ravi Sheth, for comments that improved this paper. We thank Yu Lu for spotting an error that was present in an earlier version of our code. The MS used in this paper was carried out as part of the programme of the Virgo Consortium on the Regatta supercomputer of the Computing Centre of the Max-Planck-Society in Garching. Data for the halo population in this simulation, as well as for the galaxies produced by several different galaxy formation models, are publicly available at http://www.mpa-garching.mpg.de/millennium and under the ‘downloads’ button at http://www.virgo.dur.ac.uk/new. This work was supported in part by the PPARC rolling grant for Extragalactic and cosmology research at Durham.

REFERENCES
Benson A. J., Kamionkowski M., Hassani S. H., 2005, MNRAS, 357, 847
Bond J. R., Cole S., Efstathiou G., Kaiser N., 1991, ApJ, 379, 440
Bower R. G., 1991, MNRAS, 248, 332
Bower R. G., Benson A. J., Malbon R., Helly J. C., Frenk C. S., Baugh C. M., Cole S., Lacey C. G., 2006, MNRAS, 370, 645
Cole S., 1991, ApJ, 367, 45
Cole S., Lacey C. G., Baugh C. M., Frenk C. S., 2000, MNRAS, 319, 168
Cole S., Helly J. C., Frenk C. S., Parkinson H., 2007, MNRAS, in press (this issue, doi:10.1111/j.1365-2966.2007.12516.x)
Davis M., Efstathiou G., Frenk C. S., White S. D. M., 1985, ApJ, 292, 371
Efstathiou G., Frenk C. S., White S. D. M., Davis M., 1988, MNRAS, 235, 715
Eke V. R., Cole S., Frenk C. S., 1996, MNRAS, 282, 263
Jenkins A., Frenk C. S., White S. D. M., Colberg J. M., Cole S., Evrard A. E., Couchman H. M. P., Yoshida N., 2001, MNRAS, 321, 372
Giocoli C., Moreno J., Sheth R. K., Tormen G., 2007, MNRAS, 376, 977
Harker G., Cole S., Helly J. C., Frenk C., Jenkins A., 2006, MNRAS, 367, 1039
Kauffmann G., White S. D. M., 1993, MNRAS, 261, 921
Lacey C., Cole S., 1993, MNRAS, 262, 627
Lacey C., Cole S., 1994, MNRAS, 271, 676
Lin W. P., Jing Y. P., Lin L., 2003, MNRAS, 344, 1327
Neistein E., Dekel A., 2007, MNRAS, in press (arXiv:0708.1599)
Percival W. J. et al., 2001, MNRAS, 327, 1297
Press W. H., Schechter P., 1974, ApJ, 187, 425
Sheth R. K., Lemson G., 1999, MNRAS, 305, 946
Sheth R. K., Tormen G., 1999, MNRAS, 308, 119
Sheth R. K., Tormen G., 2002, MNRAS, 329, 61
Somerville R. S., Kolatt T. S., 1999, MNRAS, 305, 1
Springel V., White S. D. M., Tormen G., Kauffmann G., 2001, MNRAS, 328, 726
Springel V. et al., 2005, Nat, 435, 629
van den Bosch F. C., 2002, MNRAS, 331, 98
Wechsler R. H., Bullock J. S., Primack J. R., Kravtsov A. V., Dekel A., 2002, ApJ, 568, 52

© 2007 The Authors. Journal compilation © 2007 RAS, MNRAS 383, 557–564

This parameter is denoted by $a$ in Sheth & Tormen (1999) and by $q$ in Giocoli et al. (2007).
APPENDIX A: THE SPLIT ALGORITHM

Given a halo of mass $M_2$ at redshift $z$, the task of the split algorithm is to take a small step, $\Delta z$, to higher redshift, determine the mass accreted in this interval in unresolved haloes with masses less than $M_{res}$ and determine whether or not the halo undergoes a binary split. If the halo does split, then it must determine the masses of the two fragments.

In the description below, we make use of the following notation. We denote minus logarithmic slope of the $\sigma(M)$ relation by $\alpha(M) = -\frac{d\ln \sigma}{d\ln M}$ and its values at masses $M_2$, $M_2/2$ and $M_1 = qM_2$, by $\alpha_2$, $\alpha_3$ and $\alpha(q)$, respectively. Similarly, we denote the values of $\sigma(M)$ at $M_2$, $M_2/2$, $M_{res} = q_{res}M_2$ and $M_1$ by $\sigma_2$, $\sigma_3$, $\sigma_{res}$ and $\sigma(q)$, respectively. With this notation, the expression in equation (3) for the number of fragments produced per unit interval of $q$ produced in a redshift step $\Delta z$ can be written as

$$\frac{dN}{dq} = S(q) \, R(q) \, \Delta z,$$

where

$$S(q) = \sqrt{\frac{2}{\pi}} \, B \, \sigma_3 q^{n-1} \, \frac{G_0}{2\nu_{\gamma}} \left( \frac{\delta}{\sigma_2} \right)^{\nu_{\gamma}} \left( \frac{\sigma_2}{\sigma_3} \right)^{\gamma_{\nu}} \, d\delta \, dz,$$

$$R(q) = \frac{\alpha(q)}{\alpha_h} \left( \frac{V(q)}{B q^p} \right) \left( \frac{2q^p \sigma(q)}{\sigma_h} \right)^{\gamma_{\mu}},$$

$$V(q) = \frac{\sigma(q)^2}{[\sigma(q)^2 - \sigma_{\gamma\nu}^2]^{1/2}},$$

and $\eta = \beta - 1 - \gamma_{\nu}/\mu$. We have written the expression for $dN/dq$ in this form so that, as detailed below, we can choose the parameters $B$, $\beta$ and $\mu$ such that $R(q) < 1$ for $q_{res} < q < 1/2$ and $S(q) \propto q^{-n-1}$ is a simple power law. This results in several very useful properties. First,

$$N_{\text{upper}} = \int_{q_{res}}^{1/2} S(q) \, dq \, \Delta z,$$

provides an upper limit on the expected number of resolved fragments split off the main halo in step $\Delta z$. We use this to choose the step size by taking $\Delta z$ to be the minimum of $\sigma_1 \, \sqrt{2(\sigma_2^2 - \sigma_{\gamma\nu}^2)}/d\delta/\, dz$ and the value given by equation (A5) when $N_{\text{upper}} = \epsilon_1$ (by default, we take $\epsilon_1 = \epsilon_2 = 0.1$). The first constraint ensures that the exponent in equation (1) is small so that equation (2) is correct to first order and the second constraint ensures that multiple splittings in one time-step are negligible.

Having determined $\Delta z$, the next step is to evaluate $F$ from equation (5) to determine the fraction of mass that is accreted in unresolved haloes in this time-step. The expression defining $F$ can be simplified to the form

$$F = \sqrt{\frac{2}{\pi}} J(u_{\text{res}}) \frac{G_0}{\sigma_2} \left( \frac{\delta}{\sigma_2} \right)^{\nu_{\gamma}} \, d\delta \, dz,$$

where we have made the substitution $u = \sigma_2/(\sigma_2^2 - \sigma_{\gamma\nu}^2)^{1/2}$ and the integral

$$J(u_{\text{res}}) = \int_0^{u_{\text{res}}} (1 + 1/u^2)^{\gamma_{\nu}/2} \, du,$$

with $u_{\text{res}} = \sigma_2/(\sigma_2^2 - \sigma_{\gamma\nu}^2)^{1/2}$. Since this integral has no dependence on $M_2, z$ or $\sigma(M)$ it can be tabulated as a simple look-up table for any chosen value of the parameter $\gamma_{\nu}$. In the original GALFORM algorithm, it reduces to $J(u_{\text{res}}) = u_{\text{res}}$.

The next step is to generate the first of three uniform random variables in the range $0–1$. If this first variable, $r_1$, is greater than $N_{\text{upper}}$ evaluated with the selected $\Delta z$, then no split occurs at this time-step and $M_2$ is just reduced to $M_2(1-F)$ to account for the accreted mass. If $r_1 < N_{\text{upper}}$, we generate a second random variable $r_2$ and transform it using $q = (q_{res}^2 + (2^{-n} - q_{res}^n)r_2)^{1/n}$, so that it is drawn from the power-law distribution $q^{n-1}$ in the range $q_{res} < q < 1/2$. Finally, we generate a third random variable $r_3$ and only accept $q$ if $r_3 < R(q)$. In the case that $q$ is rejected, we again simply reduce $M_2$ to $M_2(1-F)$, but if $q$ is accepted we generate two fragments with masses $qM_2$ and $M_2(1-F-q)$. This rejection step ensures that $q$ is being drawn with the correct normalization from the probability distribution defined by equation (A1).

For this algorithm to work, we require $R(q) < 1$ for $q_{res} < q < 1/2$. Referring to equation (A3), in all CDM models, $\alpha(M) > 0$ and $d(\alpha)/dM > 0$ and so the first term $\alpha(q)/\alpha_h$ is necessarily less than 1. Also these conditions imply that the function $V(q)$ is monotonically increasing and $\ln(V)$ versus $\ln(q)$ is concave upwards for $0 < q < 1/2$. This means that $V(q)$ is bounded from above by the power law $Bq^p$ chosen to satisfy $Bq^p = V(q)$ for $q = q_{res}$ and $q = 1/2$. In other words, with this choice of $B$ and $\beta$ the first term in equation (A3), $V(q)/Bq^p$, is less than or equal to 1. Finally, if we choose

$$\mu = \begin{cases} \frac{\sigma_h}{\ln(\sigma_{res}/\sigma_h)}, & \text{if } \gamma_{\nu} > 0, \\ \frac{\ln(\sigma_{res}/\sigma_h)}{2q_{res}}, & \text{if } \gamma_{\nu} < 0, \end{cases}$$

then regardless of the sign of $\gamma_{\nu}$, the last factor, $((2q^p \sigma(q)/\sigma_h)^{n-1}$, is also less than or equal to $1$ and so $R(q)$ is always less than 1 as required.

The merger tree produced by this algorithm has no directly imposed time/redshift resolution and comprises only binary mergers. However, we typically rebin each merger tree on to a discrete grid of predefined redshift snapshots. With this coarser time-resolution, the mergers occurring between snapshots can involve three or even more haloes.

This paper has been typeset from a TeX/LaTeX file prepared by the author.