HBT+: an improved code for finding subhaloes and building merger trees in cosmological simulations

Jiaxin Han,[1,2]★ Shaun Cole,[2] Carlos S. Frenk,[2] Alejandro Benitez-Llambay[2] and John Helly[2]

[1]*Kavli IPMU (WPI), UTIAS, The University of Tokyo, Kashiwa, Chiba 277-8583, Japan*
[2]*Institute of Computational Cosmology, Department of Physics, University of Durham, South Road, Durham DH1 3LE, UK*

ABSTRACT

Dark matter subhalos are the remnants of (incomplete) halo mergers. Identifying them and establishing their evolutionary links in the form of merger trees is one of the most important applications of cosmological simulations. The HBT (Hierachical Bound-Tracing) code identifies haloes as they form and tracks their evolution as they merge, simultaneously detecting subhaloes and building their merger trees. Here we present a new implementation of this approach, HBT+, that is much faster, more user friendly, and more physically complete than the original code. Applying HBT+ to cosmological simulations, we show that both the subhalo mass function and the peak-mass function are well fitted by similar double-Schechter functions. The ratio between the two is highest at the high-mass end, reflecting the resilience of massive subhaloes that experience substantial dynamical friction but limited tidal stripping. The radial distribution of the most-massive subhaloes is more concentrated than the universal radial distribution of lower mass subhaloes. Subhalo finders that work in configuration space tend to underestimate the masses of massive subhaloes, an effect that is stronger in the host centre. This may explain, at least in part, the excess of massive subhaloes in galaxy cluster centres inferred from recent lensing observations. We demonstrate that the peak-mass function is a powerful diagnostic of merger tree defects, and the merger trees constructed using HBT+ do not suffer from the missing or switched links that tend to afflict merger trees constructed from more conventional halo finders. We make the HBT+ code publicly available.

Key words: gravitational lensing: strong – methods: numerical – galaxies: haloes – dark matter.

1 INTRODUCTION

The process of cosmic structure formation, as revealed by numerical simulations, can be largely summarized by the growth of dark matter haloes and their interactions. In a universe with cold dark matter (CDM), small haloes merge hierarchically to form bigger haloes, a process that is often described by a halo merger tree. After a merger, the remnants of the progenitors are not erased immediately. Instead, they survive inside the descendant halo as subhaloes (Ghigna et al. 1998; Moore et al. 1998; Klypin et al. 1999; Moore et al. 1999). A complete list of haloes and subhaloes, together with their merger history, has become a standard data product of a simulation, whose calculation requires a (sub)halo finder and a merger tree builder.

Finding isolated dark matter haloes is relatively straightforward once a definition of halo is adopted. For example, a Friends-of-Friends (FOF) halo finder (Davis et al. 1985) works by connecting particles located within a linking length of each other to find clustered particles above a certain density threshold. A spherical over-density halo finder (e.g. Lacey & Cole 1994) works by growing a radius around a density peak until the average density inside the sphere matches a predefined value. By contrast, finding subhaloes are more complicated. Generally speaking, the process of finding a subhalo consists of two steps: (1) collecting a list of candidate particles to build a 'source' subhalo; and (2) pruning the source to remove unbound particles until a self-bound subhalo remains.

Depending on the way the source is defined, subhalo finders can be broadly categorized into three types: configuration space finders that examine the spatial clustering of particles (e.g. Ghigna et al. 1998; Springel et al. 2001; Knollmann & Knebe 2009); phase space finders that consider clustering in both spatial and velocity

★ E-mail: hanjiaxin@gmail.com

space (e.g. Maciejewski et al. 2009; Elahi, Thacker & Widrow 2011; Behroozi, Wechsler & Wu 2013); and tracking finders that build the source from past progenitors (Tormen, Diaferio & Syer 1998; Han et al. 2012). It has been shown that configuration space finders suffer from a 'blending' problem, the difficulty of resolving subhaloes embedded in the inner high-density region of the host halo due to spatial overlap (Han et al. 2012; Knebe et al. 2011; Muldrew, Pearce & Power 2011). In the case of major mergers, this problem is further manifest as a random switching of the masses of the merging haloes or of the presumed halo centre: once the two protagonists of the merger overlap substantially, the partitioning of mass between them can be arbitrary and inconsistent from snapshot to snapshot. Even phase space finders have difficulty dealing with this situation (Behroozi et al. 2015). These problems in identifying the main descendant of a merger propagate into the merger tree, giving rise to incorrect or missing links (Han et al. 2012; Srisawat et al. 2013; Avila et al. 2014; Jiang et al. 2014).

One way to solve these problems is by exploiting prior knowledge about the history of the subhalo particles. A tracking finder such as the HBT (Hierarchical Bound-Tracing; Han et al. 2012) achieves this by taking the list of particles in the progenitor as the source of the subhalo. This approach relies on the fact that a subhalo can be defined as the self-bound remnant of its progenitor halo after a merger. Since HBT does not rely on spatial or phase-space clustering to build the source, it is naturally immune to the blending and mass or centre-switching problems.

In this work, we present a new implementation of the HBT algorithm, HBT+ . HBT+ is written in C++ from scratch, and improves upon HBT in many respects including modularity, usability, performance, support for distributed architecture, applicability to hydrodynamical simulations and richness in output subhalo properties. The default output format is HDF5, which can be easily manipulated in scripting languages such as PYTHON. Besides the technical improvements, the most significant change in the physical prescription is that HBT+ can handle the merger of subhaloes due to dynamical friction. It is known that HBT catalogues include subhaloes that are located at nearly identical positions in phase space (Han et al. 2012; Behroozi et al. 2015). Although it may be desirable to track these overlapping objects separately for certain applications, their separate identities are not supported by the resolution of the simulation. In HBT+ , we introduce a prescription to detect and merge these overlapping pairs. As we will show, this mostly affects the population of surviving subhaloes at the high-mass end and in the very centre of the host halo.

The downside of the tracking approach is that it does not work on a single snapshot of the simulation. Instead, a sequence of snapshots has to be provided. This problem can be overcome by combining a tracking finder with the simulation code and carrying out halo finding and tree building on-the-fly. The optimized HBT+ would be a good candidate for such developments.

We apply HBT+ to cosmological and zoomed simulations to test the performance of the code. As an illustration, we consider the distribution of massive subhaloes. Recent lensing observations suggest an excess of massive subhaloes in galaxy clusters. Using a combined weak- and strong-lensing analysis, Jauzac et al. (2016) and Schwinn et al. (2017) compared the distribution of massive subhaloes inferred in Abell 2744 with those in the Millennium-XXL (Angulo et al. 2012) simulation. They could not find any halo in Millennium-XXL that hosts as many massive subhaloes as are observed in Abell 2744. Natarajan et al. (2017) carried out a strong-lensing analysis of the subhalo distribution in the inner region of several galaxy clusters in the Hubble Frontier Field, and compared their results with the Illustris (Vogelsberger et al. 2014) hydrodynamical simulations. They find relatively good agreement in the subhalo mass function, but the observed radial distribution of subhaloes is more concentrated than that in simulations. These comparisons are based on subhalo catalogues constructed from SUBFIND (Springel et al. 2001), a subhalo finder in configuration space. Besides selection effects and extreme number statistics, these authors interpret the discrepancy as due to overly efficient dynamical friction and tidal stripping in the simulation. Interestingly, it has been argued that configuration space finders such as SUBFIND significantly underestimate the subhalo mass function at the high-mass end (van den Bosch & Jiang 2016). This conclusion is mostly based on comparing SUBFIND results with those from ROCKSTAR (Behroozi et al. 2013) and SURV (Tormen, Diaferio & Syer 1998; Giocoli, Tormen & van den Bosch 2008; Giocoli et al. 2010). Unfortunately, these studies did include a comparison of the radial distribution of subhaloes. In this work, we compare both the mass and radial distributions of HBT+ and SUBFIND subhaloes in detail, by applying both finders to the same set of simulations.

Our analysis also eliminates some systematic uncertainties in the van den Bosch & Jiang (2016) comparison. One such uncertainty is in the definition of subhalo mass. While ROCKSTAR and SURV define the mass of a subhalo to include the contribution from its sub-subhaloes, SUBFIND follows an exclusive mass definition. In addition, the SUBFIND results used in that comparison were inferred from a fitting formula derived from a different simulation with a different convention for defining the host halo properties. In this work, we will make a direct comparison between SUBFIND and HBT+ by applying them to the same set of simulations. Since both SUBFIND and HBT+ adopt an exclusive mass definition for subhaloes, this allows a fair comparison.

We confirm the conclusion of van den Bosch & Jiang (2016) that SUBFIND underestimates the high-mass end of the mass function. We find that this deficiency is mostly caused by the difficulty SUBFIND has in resolving massive subhaloes near the host centre. This means that the excess of massive subhaloes in cluster centres may be attributed to systematics in the subhalo catalogues in the simulations, rather than posing a challenge to current ΛCDM cosmological simulations. This 'indigestion' of massive subhaloes leads to a hardening in the subhalo mass function at the high-mass end, which explains the flatter slope found by van den Bosch & Jiang (2016). However, at much lower subhalo masses, we find a slope for the subhalo mass function of -0.95, consistent with previous studies Springel et al. (2008b)

This paper is organized as follows. In Section 2, we explain the technical details of the HBT+ algorithm, with emphasis on the improvements over its predecessor, HBT . In Section 3, we apply HBT+ to simulations to test its performance, with special attention to the distribution of massive subhaloes. We summarize and conclude in Section 4.

2 ALGORITHM

In this paper, we make an explicit distinction between a halo and a subhalo. A halo is defined as an *isolated* virialized object, while a subhalo is a substructure embedded inside a halo. As an input to HBT+ , an existing halo catalogue containing the list of particles in each isolated halo at each snapshot must be provided. This halo-finding step can be done with any halo finder of the user's preference, and is independent of the subhalo finding step which is the main function of HBT+ .
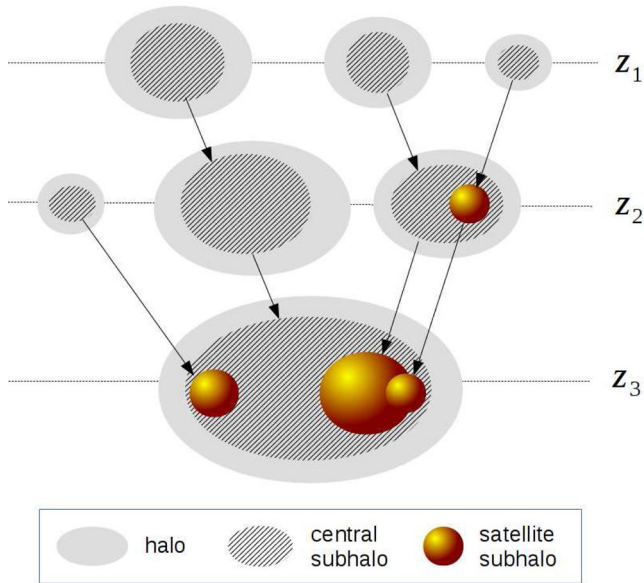
Figure 1. Illustration of a merger tree and the algorithm to find subhaloes through tracking. Each horizontal line represents a snapshot labelled by its redshift, with $z_1 > z_2 > z_3$. As haloes merge, satellite subhaloes are created as remnants of the progenitors, and can be identified by tracking the progenitor particles to subsequent snapshots and removing unbound particles. For each bound halo, a central subhalo is always defined as the one containing the majority of its bound particles, and is linked to the main progenitor.

The overall algorithm of HBT+ is the same as that of HBT, and can be summarized in Fig. 1. Starting from the earliest snapshot at redshift z_1 of a simulation, each halo is screened to identify bound particles and eliminate unbound ones. The particle list of each bound halo is then passed to the next snapshot z_2 to identify a descendant halo. When multiple progenitors are linked to the same descendant, a main progenitor is determined (according to mass and dynamical consistency, see Section 2.2), while the others are unbinded to create satellite subhaloes. The main progenitor is then updated to include the particles of the current host halo (excluding satellites) and unbinded to create a central subhalo. The distinction between centrals and satellites is used to reflect that a central subhalo is accreting mass, while a satellite is subject to mass stripping. The particle lists of these central and satellite subhaloes are further propagated to the next snapshot z_3, and the iteration continues until all the snapshots have been processed.

The detailed implementation of this algorithm has been improved in many aspects in HBT+, and is describe below.

2.1 A simple and intuitive merger tree format

Conventional approaches to subhalo finding and merger tree building works by first finding subhaloes at each snapshot, then linking them across snapshots. As a result, each subhalo is regarded as a different object and a tree is represented by the links between subhaloes at different snapshots. HBT also follows this scheme in tree building, by assigning separate subhalo IDs to subhaloes in different snapshots, and recording the progenitor ID for each subhalo, as shown in Fig. 1. Such a representation is also commonly found in popular data bases (e.g. the Millennium data base, Lemson
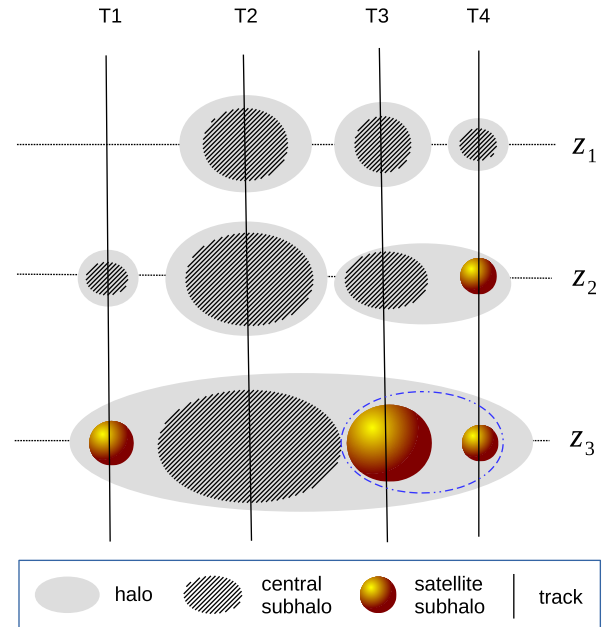

Figure 2. The merger tree in Fig. 1 represented as a list of tracks ($T1$, $T2$, $T3$ and $T4$) grouped by different host haloes at each snapshot. The dash-dotted ellipse at z_3 marks a subhalo group.

& Virgo Consortium 2006)[1] and in a community proposed merger tree format (Thomas et al. 2015), where additional auxiliary links are further provided to facilitate tree walking.

In HBT+, we switch to an alternative representation of subhaloes and trees by organizing them in terms of tracks, that are native to the tracking algorithm. Each track is the entire evolution history of a subhalo, while a subhalo is a snapshot of a track. This is equivalent to treating a subhalo as a Lagrangian object, which is labelled by a single Lagrangian ID throughout time. Thus, a merger tree can be completely specified by a list of tracks associated with haloes at each snapshot (e.g. by recording a host halo ID for each track). Fig. 2 shows such an example. Properties of each subhalo at each snapshot can still be added as a local property of the track at different times. This approach essentially flattens the merger tree into a table, which is much more convenient and flexible to store. Such a 'track table' is more convenient to query and sample as well. For example, one can directly obtain the progenitor or descendant of a subhalo at any other snapshot by searching for the given track ID, without having to walk the tree snapshot by snapshot. One can also freely remove arbitrarily selected snapshots from the catalogue without having to rebuild the merger tree. As in HBT, the merging hierarchy is propogated to subsequent snapshots to record subhalo groups, so that some subhaloes can be satellites of another subhalo.

Once a track ID is created, it persists through all following snapshots. When a subhalo's mass drops below the mass resolution of the simulation, we use the most-bound particle to represent the track. This can be useful for galaxy formation models that place 'orphan galaxies' on top of these most-bound particles. It can also be used to identify subhalo mergers by identifying the host subhalo of this most-bound particle when the subhalo disappears.

---

[1] http://gavo.mpa-garching.mpg.de/Millennium/,
http://galaxy-catalogue.dur.ac.uk:8080/Millennium/

2.2 Tracking

Host finding. For each subhalo, its host halo at the next snapshot is simply determined to be the host halo of its most-bound particle.[2] We have checked that such a tracking is robust enough compared with tracking multiple most-bound particles. This is much cleaner than the original HBT treatment that splits the progenitor particles into different hosts, which mostly introduces short-lived noisy tracks (splitter tracks) into the catalogue.

Main progenitor determination. Inside each host halo, the main progenitor is typically selected to be the most-massive one. However, when other progenitors have masses close to the most-massive one, such a choice becomes less justified. In this case, we further compare the kinetic energy of the progenitors with respect to the bulk motion of the host halo. Out of all the progenitors whose mass exceeds two-thirds of the most-massive progenitor mass, the one that has the smallest specific kinetic energy is chosen to be the main progenitor. As we further justify in equation (2) in Section 2.3, this choice yields the highest total binding energy when all the halo particles are accreted by the main progenitor. [3]

Source subhalo update. An important step for robust tracking is selecting a set of particles – a source subhalo – for each subhalo that are passed to the next snapshot for unbinding (Section 2.3). As is shown in Han et al. (2012), the definition of the source subhalo has to be precise enough so as to avoid too many unbound particles, while at the same time it has to be conservative enough to allow for reaccretion of previously stripped particles. In HBT this is achieved by adaptively choosing a progenitor at some previous snapshot according to the current mass of the subhalo. In this work, we do this in a more flexible way by updating the source continuously. After each unbinding step, source particles are sorted according to binding energy. Less bound particles are excluded, to leave a source subhalo with at most $3N_{bound}$ particles, where N_{bound} is the number of bound particles. This updated source is then passed to the next snapshot for unbinding.

2.3 Stripping

The stripping of mass from subhaloes is determined by unbinding.

Reference frame. Unbinding is the process of removing particles whose kinetic energy exceeds their potential energy. To calculate the kinetic energy, a reference frame must be defined, which we choose to be the one that minimizes the total kinetic energy of the subhalo particles. Since

$$K_{tot} = \frac{1}{2} \sum_{i=1}^{N} (\boldsymbol{v}_i - \boldsymbol{v}_c)^2, \tag{1}$$

$$= \frac{N}{2} [(\langle \boldsymbol{v} \rangle - \boldsymbol{v}_c)^2 + (\langle \boldsymbol{v}^2 \rangle - \langle \boldsymbol{v} \rangle^2)], \tag{2}$$

minimizing the kinetic energy is equivalent to minimizing the distance between the centre velocity and the average velocity vectors, $(\langle \boldsymbol{v} \rangle - \boldsymbol{v}_c)^2$. When Hubble flow is considered, the distance becomes $(\langle \boldsymbol{v} \rangle - \boldsymbol{v}_c + H \langle \boldsymbol{r} - \boldsymbol{r}_c \rangle)^2$. A natural choice is thus the centre of mass

---

[2] In some rare cases, a subhalo does not find any host halo but remains bound. This could happen when the host halo is occasionally missed by the halo finder (e.g. FoF) near the resolution limit. We keep these types of objects as field subhaloes and assign the background universe as a special host halo for them.

[3] After unbinding, the subhaloes are sorted in mass one more time to ensure that the most-massive bound subhalo is assigned as the central subhalo.

---

frame, centred at $\boldsymbol{r}_c = \langle \boldsymbol{r} \rangle$ with bulk velocity $\boldsymbol{v}_c = \langle \boldsymbol{v} \rangle$. Once unbound particles are removed, we update the reference frame and calculate the binding energies using the gravitational potential from the remaining particles and unbind again. This process continues until the bound mass converges.

Fast unbinding. The calculation of potential energy during the unbinding iteration is expensive even with a tree code. The majority of the computation time of HBT is spent on unbinding. We introduce two optimizations to speed up this process.

The first optimization is to apply a differential potential update. During every step of the unbinding iteration, the change in potential is due to the removal of unbound particles. When the number of removed particles between two iterations becomes smaller than that of the remaining particles, the potential energy can be efficiently obtained by applying a correction to the potential in the previous iteration, that is, by subtracting the contribution from the removed particles.

For the purpose of unbinding, a very accurate potential energy is not required. Thus, we further optimize this step by calculating the potential using a small sample of randomly selected subhalo particles. As we show analytically in Appendix A, the bound density profile of a subhalo can be recovered to percent level accuracy or better over the entire radial range when the mass distribution is sampled with only 1000 particles. With this algorithm, the potential calculation for all the N particles in the subhalo becomes an $O(N)$ operation, compared to the $O(N\log(N))$ complexity of a tree-code.

Because the potential energy is less accurate in the centre when calculated with a sampled mass distribution, it becomes difficult to select the most-bound particle which is a commonly adopted reference frame of a subhalo. To overcome this problem, we further calculate an 'inner binding energy' adopting only the potential from the 1000 most-bound particles, and select the most-bound particle thereafter.

We also tried unbinding using a potential estimate that assumes spherical symmetry by binning the mass distribution radially, which can also speed up the calculation significantly. However, such an unbinding tends to fail when spherical symmetry is not a good approximation, such as near pericentric passage where the tidal shear is strong.

In Fig. 3, we show the performance improvement achieved by the various optimizations. We use a test simulation of 270^3 particles with a boxsize of $62.5\,\mathrm{Mpc}\,h^{-1}$, run in the same cosmology as that of the Millennium simulation (Springel et al. 2005). The tests are done on a single computational node of the COSMA machine in Durham with 12 cores. The performance is improved significantly with the differential potential update optimization, and further when the sampled potential estimate is also used. To compare against the performance of HBT, we also run HBT+ using the same level of optimization as HBT, which is by terminating the unbinding iteration when the bound mass, M_i, at iteration i converges with $M_{i+1}/M_i > \epsilon_M$, where $\epsilon_M = 0.995$ is the mass precision. The performance difference between HBT and HBT+ adopting this common unbinding optimization can be mostly attributed to a change in the central-determination step in Section 2.2. In HBT, the centrals are selected by comparing the bound mass of the progenitors at the current snapshot, which are obtained by one extra unbinding step. By contrast, in HBT+, this is done by comparing the progenitor mass at the previous snapshot, together with their current kinetic energies in the host halo frame, avoiding the extra unbinding. Because the ϵ_M optimization introduces similar improvement as the differential potential update optimization, we no longer rely on the ϵ_M optimization in HBT+, although this parameter is still available.
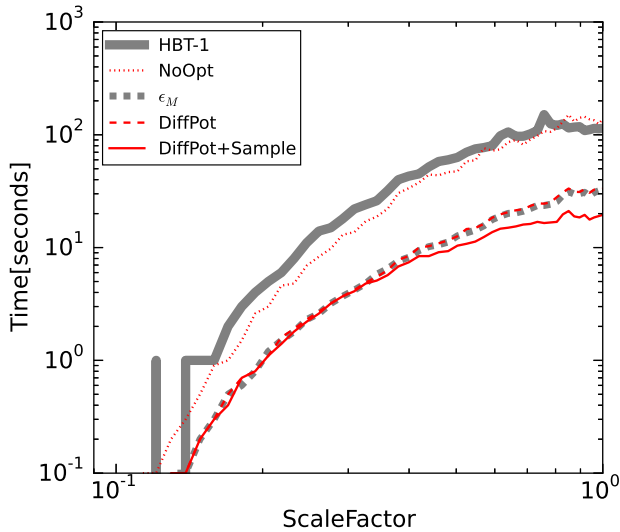
Figure 3. The runtime of HBT+ on a test simulation. Different curves show the performance with different levels of optimizations: NoOpt: HBT+ with no optimization; DiffPot: HBT+ with only differential potential update; DiffPot+Sample: HBT+ with both differential potential update and sampled potential estimate. For comparison, the thick grey line (HBT-1) shows the timing of HBT, and the dashed grey line (ϵ_M) shows the timing of HBT+ adopting the same optimization as HBT, which is by relaxing the mass convergence criterion.

Overall, the performance is already increased by a factor of ∼6 for this small test simulation, and we expect even higher improvements for larger simulations, given the change in complexity from $O(N\log(N))$ to $O(N)$.

Recursive unbinding. As in HBT, the unbinding is done recursively, by unbinding the deepest nested subhaloes first and then feeding the stripped particles to their host subhaloes for unbinding. This ensures that the particles in each subhalo do not include any bound particles contained in its sub-subhaloes, leading to an exclusive mass definition.[4] The subhalo nesting hierarchy is propagated from the merging hiearchy of their progenitor haloes.

2.4 Merging

Due to dynamical friction and heating, subhaloes could lose their orbital energy and eventually sink to the centre of their host. When the trajectories of two subhaloes overlap and evolve together without being separated thereafter, the satellite is trapped at the host centre, and the two subhaloes can be defined as having merged. After a merger, tidal stripping ceases to take effect due to the concentric configuration, and the bound mass of the trapped subhalo remains constant until, it is heated up by mass accretion or stripped by other haloes. We show an example in Fig. 4. We identify trapped mergers by comparing the spatial and velocity separations, Δ_x and Δ_v, of the two subhaloes with the resolution at the centre of the host subhalo. To estimate the resolution, we use the spatial and velocity dispersion

---

[4] Note that physically, the gravitational potential of sub-subhaloes could also contribute to binding the particles in the subhalo. As we stick to a strict self-bound mass definition, we do not consider the contribution of sub-subhaloes when computing the potential energy of the subhalo. Given the small expected mass fraction in sub-subhaloes (∼10 per cent), this should lead to only a small mass difference in most cases.
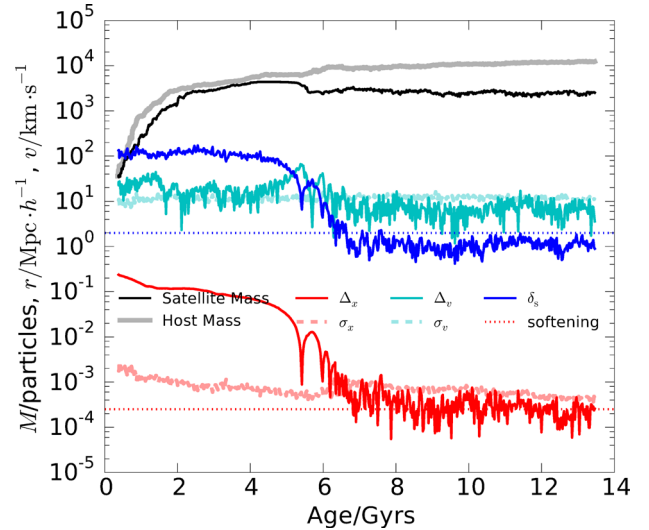

Figure 4. A resolved merger of two subhaloes from the Aquarius simulation of a Milky-Way sized halo with a particle mass of $2.9 \times 10^5 \, \mathrm{M_\odot} \, h^{-1}$. We show the evolution of mass, separation (Δ_x) and relative velocity (Δ_v) of the two objects. The spatial and velocity dispersions (σ_x and σ_v) of the 20 most-bound particles of the host subhalo are also shown; these measure the spatial and velocity resolution at the centre of the halo. The merger of the two objects can be identified with $\delta_s < 2$, shown as a blue dotted line.

of the 20 most-bound particles of the host subhalo, σ_x and σ_v. Let

$$\delta_s = \frac{\Delta_x}{\sigma_x} + \frac{\Delta_v}{\sigma_v}. \tag{3}$$

When $\delta_s < 2$, the two objects are regarded as merged giving the current numerical resolution. We use the position and velocity of the most-bound particle of each subhalo to measure Δ_x and Δ_v, so this merger criterion can be interpreted as when the most-bound particles of the two objects cannot be separated in phase space. By default, we merge the trapped subhalo with its host once $\delta_s < 2$ and only track the most-bound particle thereafter. In Appendix B, we provide more information about the distribution of these trapped subhaloes for the case when we do not implement this merging criterion.

2.5 Parallelization

HBT+ comes in two flavours of parallelization: one pure OPENMP version to be used on a shared memory machine, and one MPI/OPENMP hybrid version to be used on distributed servers which can also be run in pure MPI mode.

For the OPENMP version, the parallelization is automatically determined by the symmetry of the workload. When the most-massive halo in a snapshot exceeds 10 per cent the total mass of all haloes, the parallelization is done inside each halo, by calculating the binding energy of individual particles in parallel. Otherwise, the parallelization is done by processing different haloes in parallel.

In the current MPI version, the workload is decomposed by dividing the simulation box into spatial grids that are assigned to different computational nodes. Haloes and subhaloes are then distributed to the grids according to their spatial coordinates. On each node, the computation is then performed in the same way as the OPENMP version. Because the particles loaded from each halo typically only contain the particle IDs, they must be matched to the particles in the snapshot files to obtain their coordinates and other properties. This is done in parallel by first distributing the snapshots

to different nodes according to the particle IDs. The particles in each halo are then split and passed to different nodes according to their IDs. For all the halo particles received on each node, we then sort both the halo particles and the snapshot particles. The sorted halo particles are then matched to the snapshot particles with a batched binary search, which successively narrows down the search range of each particle by the search result of the previous particles. The same is done to query subhalo particles.

2.6 Support for hydrodynamical simulations

The same tracking and unbinding procedure can be applied to hydrodynamical simulations no matter how many types of particles exist in the halo catalogue, although additional routines are needed to handle the creation of particles due to star formation and the destruction of particles due to accretion by black holes. Inside each subhalo, the binding energy of each particle is calculated under the gravitational potential of all types of particles in the subhalo adopting a common reference frame. By default, we do not include the thermal energy of gas particles in the binding energy calculation in order to reflect the instantaneous dynamical state of the system. The effect of thermal energy on the system will be automatically revealed by the instantaneous dynamical state of the system in subsequent snapshots, once the thermal energy is converted into kinetic energy. Technically, however, the code can be configured to output both the binding energy and thermal energy of each particle, so that one can always switch to an alternative binding energy definition including the thermal energy in post-processing.

3 TESTS AND APPLICATION

Previous works have already revealed a few features of HBT, as summarized below:

(i) The subhaloes found by HBT have more extended density profiles compared to the truncated outer density profile typical in configuration space finders, leading to a larger mass estimate in HBT (Han et al. 2012). This difference can be much more significant for massive subhaloes.

(ii) HBT easily overcomes the blending problem of subhaloes, and successfully recovers subhaloes even when a subhalo is deeply embedded in the halo centre (Han et al. 2012; Srisawat et al. 2013).

(iii) HBT maintains consistent link between subhalo progenitor and descendant by construction, and is free from the mass or centre-switching problem in merger tree construction (Srisawat et al. 2013; Behroozi et al. 2015).

However, the mass difference, the blending and mass or centre-switching problems are demonstrated only through case studies of individual objects or through idealized simulations of a single pair of objects. In this section, we aim to investigate these differences statistically using cosmological simulations. In particular, we will compare the distribution of massive subhaloes found by HBT+ and SUBFIND, in order to understand the systematics in the distribution of these objects, and to shed light on the observed excess of massive subhaloes in clusters. In fact, the mass difference also can be understood as a blending problem that obscures the outskirts of a subhalo, while the mass or centre-switching is created by partial or total obscuration of the subhalo in the merger tree. It is thus natural to expect that all three issues are significant for massive subhaloes.
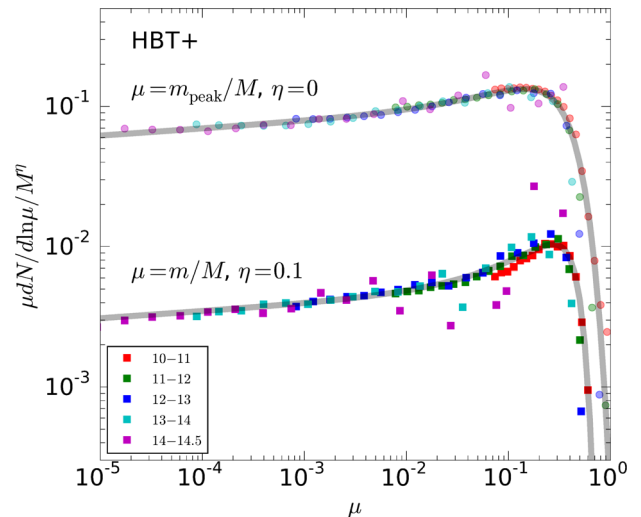

Figure 5. The peak and final subhalo mass functions in Millennium-II haloes, normalized by M^η, where M is the host halo mass. η and the mass variable μ are specified in the figure for the peak and final mass functions, respectively. Data points and lines of different colour represent different host halo mass bins, as listed in the legends in terms of $\log(M/M_\odot \, h^{-1})$. The light thick lines are the fits of equation (4), with best-fitting parameters listed in Table 1 (200Crit rows). The results for other virial definitions are qualitatively similar.

3.1 Simulations

We make use of two simulations in this section. The first is the Millennium-II simulation (Boylan-Kolchin et al. 2009) in the ΛCDM cosmology with $\Omega_m = 0.25$ and $\sigma_8 = 0.9$. It resolves 2160^3 particles in a cubic box of $100 \, \text{Mpc} \, h^{-1}$ on each side, with a particle mass of $6.89 \times 10^6 \, M_\odot \, h^{-1}$. The other simulation is a zoomed in simulation of a Milky-Way sized halo from the Aquarius simulation set (Springel et al. 2008a) with the same cosmology as Millennium-II. The Aquarius set consists of several Milky-Way sized haloes each simulated with a series of resolutions. We mainly use the first halo simulated at the second highest resolution level, which we call halo AqA2 hereafter. It has a particle mass of $10^4 \, M_\odot \, h^{-1}$ in the high-resolution region, corresponding to $\sim 10^8$ particles resolved in the main halo.

The Millennium-II simulation provides a large sample of haloes to study the average distribution of subhaloes in host haloes of different mass. Most importantly, it allows us to study the distribution of massive subhaloes statistically, which is impossible with a single host halo due to the rarity of massive subhaloes. We will study three aspects of the subhalo population: the final subhalo mass, the peak subhalo mass (i.e. the maximum mass attained by a subhalo over its entire history) and the location inside the host halo. As we will show, combining the final and peak subhalo mass function allows us to assess the quality of the merger tree statistically, which is further demonstrated in a side-by-side comparison of the merger trees of the AqA2 halo.

3.2 The subhalo mass function

It is well known that the subhalo mass function follows a simple power-law behaviour at the low-mass end, $\frac{\mathrm{d}N}{\mathrm{d}\ln m} \propto m^{-\alpha}$, with $\alpha \approx 0.9$ (e.g. Gao et al. 2004). It has also been shown that the slope of α is conserved between the unevolved and evolved subhalo mass function (Han et al. 2016). In Fig. 5, we show the subhalo mass functions from the Millennium-II simulation. Both the evolved mass

Table 1. Fits to the subhalo mass functions of the form equation (4). We show the results for three different definitions of the host halo mass M, corresponding to spherical overdensities of 200 times the critical density (200Crit, our default choice in this work), 200 times the mean mass density (200Mean) of the universe and that given by the spherical collapse model (Virial). The abundances are computed inside the radius of the corresponding spherical overdensity. m and m_{peak} are the final and peak mass of the subhalo.

Host halo definition	μ	a_1	α_1	a_2	α_2	b	β	η
200Crit	m/M	0.0055	0.95	0.017	0.24	24	4.2	0.1
	m_{peak}/M	0.11	0.95	0.20	0.30	7.6	2.1	0
Virial	m/M	0.0072	0.95	0.017	0.26	54	4.6	0.1
	m_{peak}/M	0.11	0.95	0.32	0.08	8.9	1.9	0
200Mean	m/M	0.0090	0.95	0.055	0.16	36	3.2	0.1
	m_{peak}/M	0.11	0.95	0.64	-0.20	11	1.8	0

function and the unevolved mass functions are shown, which we call the final and peak subhalo mass functions, respectively. These functions are computed as follows. For each host halo, we identify all the branches that are currently located within its virial radius according to the position of the most-bound particle of each branch. After that, the evolved mass function is defined as the distribution of the final subhalo mass of these branches, and the unevolved mass function is defined as the distribution of their peak bound masses. Both surviving and disrupted branches contribute to the peak-mass function.

With the large sample of haloes, we are able to well resolve the high-mass end of the mass function. Both distributions are well fitted by a double Schechter function of the form

$$f(\mu) \equiv \frac{dN}{d\ln\mu},$$
$$= \left(\frac{M}{10^{10}M_\odot\,h^{-1}}\right)^\eta \left(a_1\mu^{-\alpha_1} + a_2\mu^{-\alpha_2}\right)\exp\left(-b\mu^\beta\right), \quad (4)$$

where μ is the ratio of the final (peak) mass of the subhalo to the host halo mass. By default, we adopt the virial definition corresponding to a spherical overdensity of 200 times the critical density of the universe. However, we list the best-fitting parameters for three common virial definitions in Table 1. The first power-law component in equation (4) describes the low-mass end behaviour of the mass function, while the second component is necessary to fit the shoulder at $\mu > 0.1$. Note that fitting the low-mass-end slope can be tricky depending on the weights given to the data points, the available mass range that is trusted to have converged, and the functional form adopted to describe the high-mass end behaviour. Thus, we refrain from fitting this slope. Instead, we fix $\alpha_1 = 0.95$ according to the result of Han et al. (2016) using much higher resolution data. As shown in Fig. 5, such a choice is well supported by the data.

Despite the apparently different parameter values, the peak-mass function depends only weakly on the virial definition. Consistent with previous studies (van den Bosch et al. 2005; Giocoli, Tormen & van den Bosch 2008), the peak-mass function is independent of the host halo mass. On the other hand, the final mass function scales with the host halo mass as $M^{0.1}$ in the host mass range probed by our simulation. This is consistent with the expectation that more massive haloes are younger, thus possessing a higher mass fraction in subhaloes. Overall, the peak mass and final mass functions have similar shapes, while the presence of a shoulder at $\mu > 0.1$ is more prominent in the final mass function. In Fig. 6, we show the ratio between the two fitted mass functions. For Milky-Way sized and cluster-sized haloes, the ratio is around 0.1 at the low-mass end, consistent with the findings of Han et al. (2016). If tidal stripping of the subhaloes is independent of subhalo mass, then the peak mass
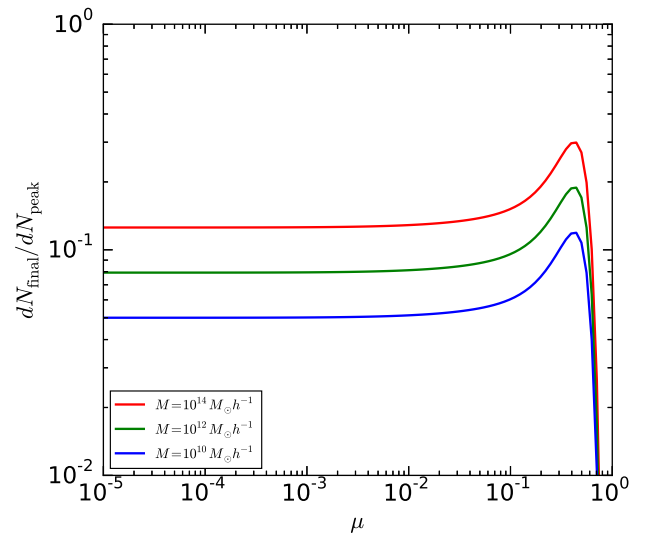

Figure 6. The ratio between the final mass and peak-mass subhalo mass functions, for different host halo masses (10^{14}, 10^{12} and 10^{10} M$_\odot$ h^{-1} from the top to bottom).

and final mass functions are expected to have the same shape. At the high-mass end, however, there is a peak in the ratio, indicating that the massive satellites are less stripped than the low-mass ones. This is not surprising because when the satellite mass is comparable to the mass of the host halo, the tidal force from the host becomes less important compared with the self-gravity of the satellite, and hence it is more difficult for the tidal force to accelerate subhalo particles to their escape velocities. We defer the detailed modelling of this effect to a future paper.

3.3 The radial distribution of subhaloes

The peak in Fig. 6 can be further understood by reference to the spatial distribution of the subhaloes. As shown in Fig. 7, the radial distributions of subhaloes of different relative mass have the same shape near the virial radius, where the subhaloes are barely affected by tidal stripping and are expected to follow the host halo density profile (Han et al. 2016). At smaller radii, however, massive subhaloes have a steep profile, while less massive subhaloes are depleted at the centre. This pattern is a consequence of both dynamical friction and tidal stripping. The former is more important for massive subhaloes, making them sink to smaller radii. At the same time, tidal stripping is less efficient for massive subhaloes, and even more so at the centre of the host when the satellite largely overlaps with the host, thus failing to eliminate these objects.
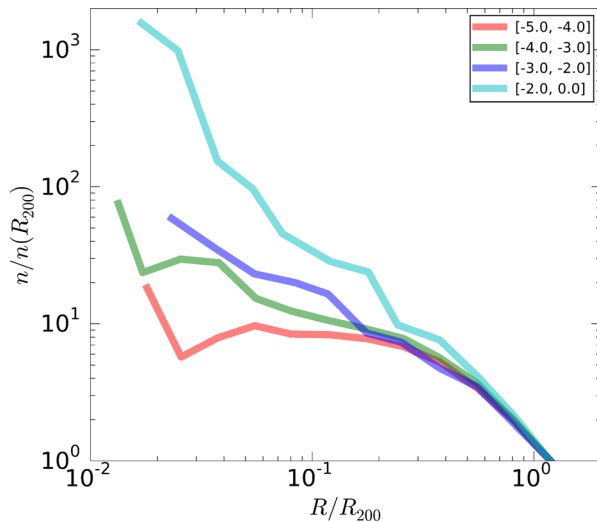
Figure 7. The radial distribution of subhaloes in host haloes of 10^{13}–10^{14} M$_\odot$ h^{-1}. The subhaloes are binned in $\log(m/M)$ as labelled. The profiles are normalized by their values at the host virial radius, R_{200}. For reference, the gravitational softening of the simulation is about $0.002R_{200}$ for these host haloes.

Being free from tidal stripping, the relative abundance of subhaloes near the virial radius is expected to follow the peak-mass function, i.e. $n(R_{200}, m) \propto f_{\text{peak}}(m)$. The subhalo mass function within the virial radius is simply an integral of subhalo abundance inside R_{200},

$$f_{\text{final}}(\mu) = \int_0^{R_{200}} n(r, m) \mathrm{d}^3 r, \tag{5}$$

$$= n(R_{200}, m) \int_0^{R_{200}} \frac{n(r, m)}{n(R_{200}, m)} \mathrm{d}^3 r, \tag{6}$$

$$\propto f_{\text{peak}}(m) \int_0^{R_{200}} \frac{n(r, m)}{n(R_{200}, m)} \mathrm{d}^3 r. \tag{7}$$

The cuspier radial profile for massive subhaloes leads to an increase in the integral of the relative profile in equation (7). As a result, the ratio between the final mass and peak-mass profiles is also higher for massive subhaloes. Note that the data in Han et al. (2016) probe only the subhalo distribution at $m/M < 10^{-3}$, where this effect is smaller and further suppressed by the use of a different subhalo finder, as we show explicitly in Section 3.4.1. In a follow-up paper, we will extend the model of Han et al. (2016) to study these distributions in detail.

3.4 Comparison with other works

3.4.1 Direct comparison with SUBFIND

In Fig. 8, we compare our mass functions and radial distributions with that found by SUBFIND (Springel et al. 2001). To compute the peak-mass function, we use the merger tree built by the Durham merger tree code DTREE (Jiang et al. 2014). DTREE has been developed with efforts to overcome halofinder pitfalls that could cause missing links or frequent switching of links, and is the default *N*-body merger tree used by the GALFORM semi-analytic models of galaxy formation (Bower et al. 2006). We extend the merger tree by tracking the most-bound particle after termination of a branch, so that the peak-mass function can be computed in the same way as our code.

As in HBT (Han et al. 2012), our final mass function is ~10 per cent above that of the SUBFIND at the low-mass end. At the high-mass

end; however, the difference is dramatic, corresponding to a factor of two to three difference in subhalo mass or an order of magnitude difference in abundance. This has been pointed out as a problem of SUBFIND through comparison with some other halo finders (including ROCKSTAR and SURV , van den Bosch & Jiang 2016) for these massive subhaloes.

In the right-hand panel of Fig. 8, we explore this difference in the spatial distribution of subhaloes. In the outer halo, our subhaloes are slightly more abundant, which can be understood as our subhaloes being slightly more massive. The overall shape of the distributions are still quite consistent with each other. In the inner halo, however, SUBFIND shows a deficiency of subhaloes compared with our result, which is most significant for more massive subhaloes. This can be understood as a reflection of the 'blending problem' exhibited by configuration space subhalo finders: when a subhalo overlaps with the host halo, it is difficult to separate it from the host using only density information. It is easy to understand that this issue is more severe for larger subhaloes.

The deficiency of massive subhaloes near the centre of the host halo in catalogues constructed using SUBFIND explains, at least in part, the disagreement noted by Schwinn et al. (2017) and Natarajan et al. (2017) between the distribution of subhaloes identified using SUBFIND in ΛCDM simulations and the distribution of subhaloes inferred from lensing studies in galaxy clusters. Mao et al. (2017) have argued that the mismatch between the simulation result of Millennium-XXL (Angulo et al. 2012) and the lensing results of Jauzac et al. (2016) and Schwinn et al. (2017) is also affected by the use of different masses in the comparison: SUBFIND masses in the simulation but projected aperture masses from lensing that also include mass contributions from the host. Even with the use of deblended subhalo mass estimates in the strong lensing analysis of Natarajan et al. (2017), however, the observed spatial distribution of massive subhaloes is still found to be more centrally concentrated than that from the SUBFIND catalogues of the Illustris simulations (Vogelsberger et al. 2014), which can be attributed to this radial-dependent blending issue in the SUBFIND catalogues.

In contrast to the final mass function, the peak-mass function from SUBFIND+DTREE is systematically above ours. Together with the lower final mass function, this means more branches are produced in the merger tree of SUBFIND+DTREE . As we will show explicitly in Section 3.5, this difference can be attributed to broken branches associated with missing links and the switching of subhalo masses in SUBFIND+DTREE .

3.4.2 Comparison with fitting functions and the low-mass end slope

In Fig. 9, we compare our results against a few fitting functions in the literature. In the left-hand panel, we have switched to the same definition of virial quantities ('Virial' as in Table 1) as in Giocoli et al. (2008), Jiang et al. (2016) and van den Bosch & Jiang (2016) when computing the mass functions. The model of Jiang et al. (2016) is a semi-analytical model that evolves progenitor haloes generated from extended Press–Schechter merger trees according to an empirical average mass stripping rate. Their model is calibrated against ROCKSTAR subhaloes from the Bolshoi (Klypin, Trujillo-Gomez & Primack 2011) and MultiDark (Prada et al. 2012) simulations, and the resulting subhalo mass function is fitted using a Schechter function. The normalization of the mass function is predicted from the dynamical age of the host halo, which we find can be equivalently fitted with a power-law in the mass range 10^{10}–10^{15}M$_\odot$ h^{-1} for the
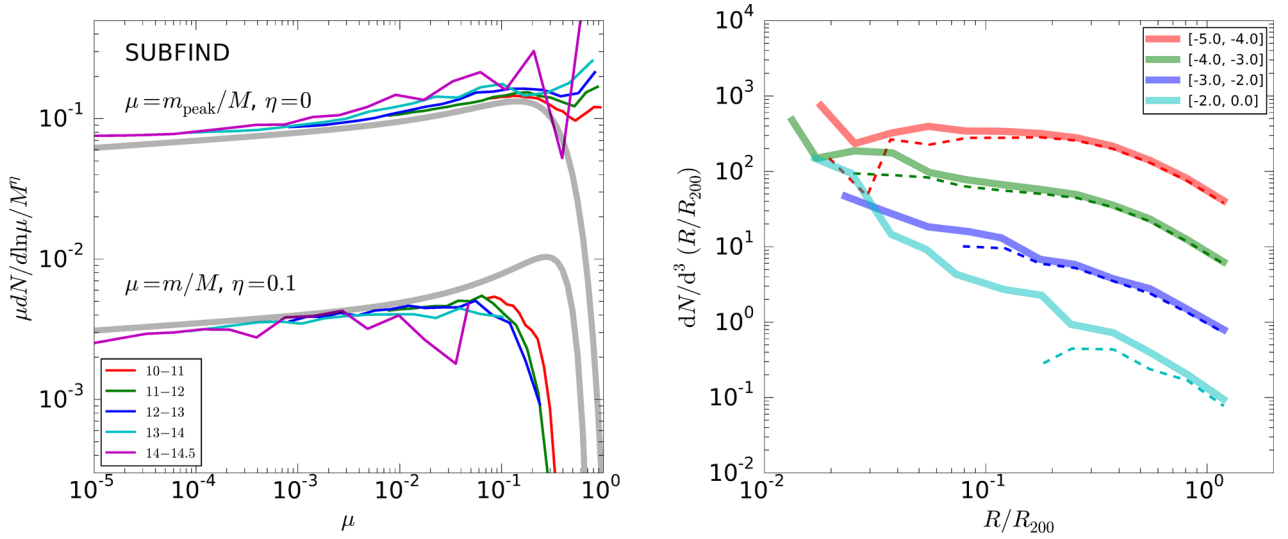
Figure 8. Left-hand panel: The peak-mass and final subhalo mass functions for SUBFIND and DTREE. The thick grey lines are fits to that of HBT+ subhaloes (same as in Fig. 5). The thin lines with different colours are SUBFIND+DTREE results in host haloes of different masses (labelled by log ($M/M_\odot\ h^{-1}$)). Right-hand panel: the radial profile of SUBFIND subhaloes (dashed lines) compared with that of HBT+ subhaloes (solid lines). The host halo mass and the subhalo mass bins are identical to those in Fig. 7, except that the profiles are not normalized at R_{200} in order to compare the relative amplitude of the two data sets.
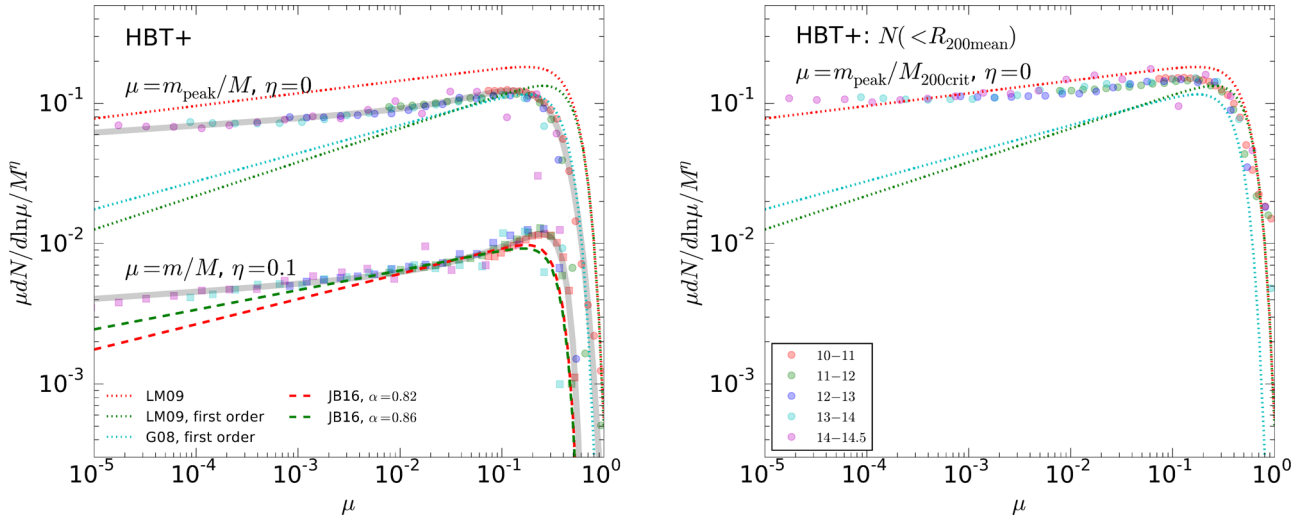
Figure 9. The subhalo mass functions compared with fitting functions in previous works. As in Fig. 5, the data points are results from our code, with different colours representing different host halo mass bins. The squares and circles show the peak-mass and final subhalo mass functions, respectively. The thick lines are fits to the data points as listed in Table 1. The peak-mass function is compared with the fitting functions of Li & Mo (2009) for all the subhaloes and first-order subhaloes, as well as with the fit of Giocoli et al. (2008) for first-order subhaloes. The final mass function is compared with the fitting function of Jiang et al. (2016) and van den Bosch & Jiang (2016), which are calibrated using the ROCKSTAR halo finder. In the left-hand panel, we adopt the 'Virial' definition for both host mass and radius. In the right panel, the host halo mass is defined according to the '200Crit' definition, while the host halo radius is defined according the '200Mean' definition.

concordance cosmology, consistent with our scaling in Table 1. For the purpose of comparison with our results, the final fitting function of the Jiang et al. (2016) model can then be summarized as

$$\frac{dN}{d\ln\mu} = a\left(\frac{M}{10^{10}M_\odot\ h^{-1}}\right)^\eta \mu^{-\alpha}\exp\left(-b\mu^\beta\right), \qquad (8)$$

with $a = 0.014$, $\eta = 0.1$, $\alpha = 0.82$, $b = 50$ and $\beta = 4$ according to Jiang et al. (2016), and slightly different parameters $a = 0.012$ and $\alpha = 0.86$ in van den Bosch & Jiang (2016). Our results are quite consistent with their fitting function with $\alpha = 0.86$ in the mass range $10^{-3} < \mu < 10^{-1}$. At the high-mass end, our data show a

higher shoulder around $\mu = 0.3$, indicating that the mass stripping rate of the most-massive subhaloes differs from the average mass stripping rate of low-mass ones in the framework of the Jiang et al. (2016) model. At the very low-mass end, our data clearly support a slope higher than $\alpha = 0.86$. While van den Bosch & Jiang (2016) argued that their data is in contradiction with a low-mass-end slope of $\alpha = 0.95$, our results suggest that their conclusion is caused by the limited mass range in their data ($\mu > 10^{-3}$). The local slope is an increasing function of subhalo mass due to the decrease in tidal stripping efficiency at the high-mass end, and the asymptotic α is still consistent with 0.95 down to $\mu \sim 10^{-5}$.

For the peak-mass function, Giocoli et al. (2008) have measured the first-level unevovled mass function, that is, the mass distribution of progenitors that fall directly into the host halo (instead of being accreted as a satellite of another infalling halo). At the high-mass end, the peak-mass function is dominated by these first-level progenitors, and our measurement agrees very well with that of Giocoli et al. (2008). At the low-mass end, higher level contributions become more important, and the Giocoli et al. (2008) fit falls below our measurement. Li & Mo (2009) measured both the first-level and all level unevolved mass function, which have been used in Jiang et al. (2014) and Jiang et al. (2016) as benchmarks to calibrate Monte Carlo merger trees from extended Press-Schechter theories. Their results lie mostly above our measurements. However, it should be noted that Li & Mo (2009) adopted a somewhat peculiar combination of the mass and spatial extension of a host halo. Their merger trees are based on FoF haloes, while the host halo mass is defined as $M_{200\mathrm{Crit}}$ corresponding to a spherical overdensity that is 200 times the critical density of the universe. To make a better comparison, in the right-hand panel of Fig. 9, we compute the peak-mass function inside $R_{200\mathrm{Mean}}$, which is expected to be closest to the size of a FOF halo, while adopting $M_{200\mathrm{Crit}}$ as the host halo mass. Such a combination leads to a mass function that is closer to the Li & Mo (2009) results, but still lower at the high-mass end. This can be further attributed to the fact that they rely on FOF haloes to build their merger trees. In this case, haloes that are temporarily linked together and subhaloes that have been ejected from the host can both contribute to the progenitor mass function. The situation can become even worse if these objects fall back into the host and are counted multiple times (Benson 2017). In contrast, our approach of selecting branches located inside the final virial radius produces a progenitor population that can be unambiguously compared against the final mass function inside the virial radius. Due to these complications, the Li & Mo (2009) results should be quoted with caution in future analytical studies.

3.5 The persistence of tracks

To further investigate the difference in the peak-mass function between SUBFIND+DTREE and HBT+ , we carry out a detailed study focusing on a single high-resolution halo, AqA2 from the Aquarius simulation set. Fig. 10 shows the peak mass computed from our code and that from SUBFIND+DTREE . Consistent with Fig. 5, the peak-mass function from SUBFIND+DTREE is higher than ours. According to whether the descendant subhalo at $z = 0$ is still resolved, we decompose the peak-mass function into a surviving and a disrupted component. The peak-mass functions of surviving subhaloes agree well with each other, meaning that both codes have identified the same population of final subhaloes. On the other hand, the disrupted peak-mass functions can differ up to a factor of 2, with SUBFIND+DTREE having more disrupted branches.

In the (r, m_{peak}) plane, we have identified some of these extra branches that exists in SUBFIND+DTREE but not in HBT+ . Fig. 11 shows the mass evolution history of the three most-massive branches (B1, B2, B3) selected this way. Correspondingly, we have also identified branches in HBT+ that best match the selected branches in orbital and mass evolution. Interestingly, these branches are related to two major merger events that happened to the host halo. In the SUBFIND+DTREE case, the B1, B2 and B3 branches all temporarily become the most-massive branch in the host at some stage, and then get disrupted almost immediately after that. The final central subhalo emerges abruptly from inside the host halo, as shown by the B1-1 branch. In contrast, the corresponding T2 and T3 branches
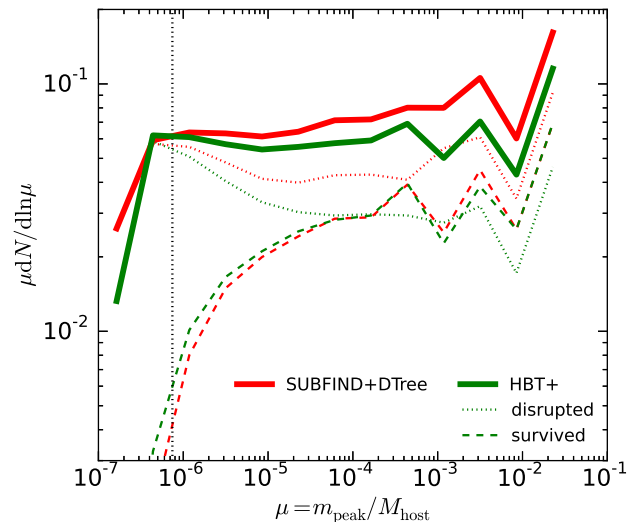

Figure 10. The peak-mass function (solid line) of tracks in Aquarius halo A2, decomposed into disrupted and surviving populations (dotted and dashed lines). The red and green colours show the results of SUBFIND+DTREE and HBT+ , respectively.
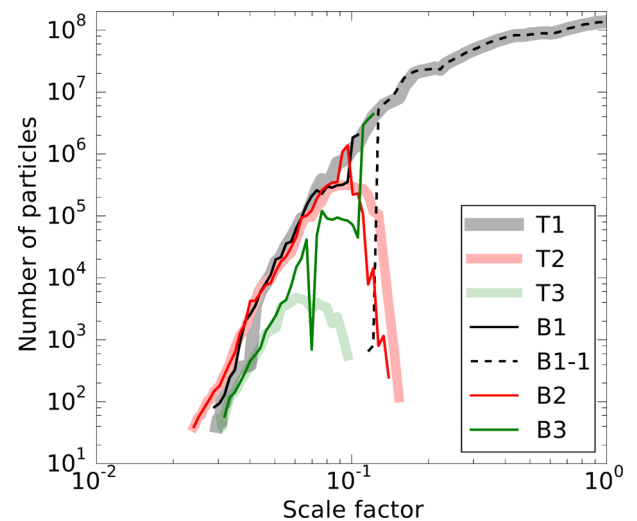

Figure 11. The mass evolution of the central subhalo of AqA2. The thin lines show the mass evolution history of subhaloes that are identified as the central subhalo of AqA2 at different times according to SUBFIND+DTREE . The thick lines show the corresponding branches as identified by HBT+ .

in HBT+ remain as less massive branches than T1 after merger until they are fully disrupted. The T1 branch remains as the most-massive branch in the host halo until the final time. In Fig. 12, we visualize this evolution as a track table, where an additional branch, B4 (T4), is included to show the major merger with B3 (T3) that leads to the strange mass growth in B3. In the HBT+ case, the central and satellite subhaloes are tracked consistently and persistently, while the SUBFIND+DTREE tree suffers a few switches in the mass and in the central-satellite determination, as well as a broken link that fragmented B1-1 from B1. The switching problem leads to an overestimate of the peak mass, while the broken links create extra progenitor branches. Both of these lead to an overestimate of the peak-mass function.
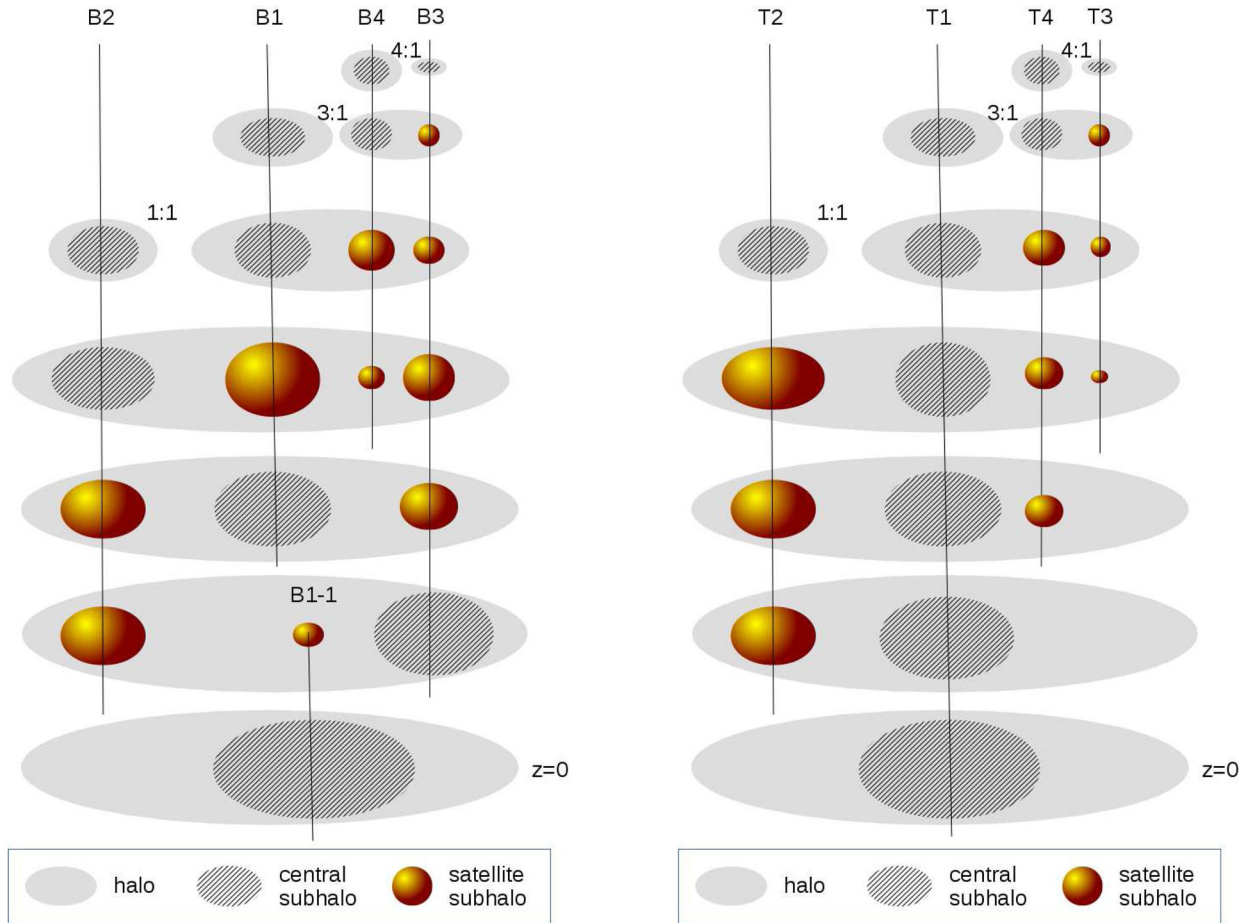
Figure 12. The major merger history of halo AqA2 as resolved by SUBFIND+DTREE (left-hand panel) and HBT+ (right-hand panel). The ratios next to haloes list the mass ratio of the progenitor haloes just before the merger. Each track (vertical lines) terminates after the disruption of the subhalo. For illustration purpose, only the snapshots of interests are plotted.

4 SUMMARY AND CONCLUSION

We have presented an improved version of the original HBT algorithm of Han et al. (2012) that tracks haloes through time to find subhaloes and build merger trees. A series of improvements are implemented, including:

(i) Treatment of subhaloes as Lagrangian objects and organization of the merger tree as a table of tracks. This allows intuitive and flexible storage and retrieval of subhaloes and trees.

(ii) Significant improvement in speed. This is made possible by a physically motivated yet simple algorithm for the identification of the main progenitor halo and a refined unbinding algorithm with a complexity of $O(N)$ compared to the $O(N\log N)$ complexity of a plain tree code.

(iii) Detection and merging of trapped subhaloes. These are massive satellites that sink to the centre of their host halo and remain there without being disrupted, leading to pairs of subhaloes that overlap in their orbit while remaining individually self-bound. We have developed a prescription to detect such pairs and merge the trapped satellite.

(iv) Support for distributed computation through MPI.

(v) Support for hydrodynamical simulations.

The code has been rewritten in C++ with user friendly configuration options, as well as HDF5 output format that allows direct post-processing with other software. The source code is publicly available at https://github.com/Kambrian/HBTplus and http://icc.dur.ac.uk/data/.

As an illustration, we applied the new code to a study of the distribution of subhaloes and tested the persistence of merger trees in the Millennium-II simulation and in one of the Aquarius project simulations. In contrast to previous studies that fit the mass functions with a single Schechter function, we find that both the final and peak-mass subhalo mass functions are well fitted by a double Schechter function (equation 4) with similar shapes. These mass functions harden towards the high-mass end before falling off exponentially. The hardening is most significant in the final mass function, reflecting an inefficiency of tidal stripping of massive subhaloes. This also reflects our finding that the radial distribution of massive subhaloes is more concentrated than the universal radial profile of low-mass ones, due to stronger dynamical friction and weaker tidal stripping. The detection of this hardening requires the ability to identify subhaloes in the inner regions of the host halo, which HBT+ does, but which subhalo finders that work in configuration space alone have difficulty identifying. Recent lensing observations of galaxy clusters have resulted in reports of discrepancies in the observed subhalo distribution compared to that of ΛCDM predictions, including the excess of massive subhaloes reported by Jauzac et al. (2016) and Schwinn et al. (2017) and the more concentrated subhalo radial distribution reported by Natarajan et al. (2017). These discrepancies can be explained, at least in part, by the blending problem present in

the Millennium-XXL and Illustris SUBFIND catalogues used in their comparisons.

The hardening of the subhalo mass function at the high-mass end means that single power-law fits to the mass function are inadequate. From our HBT+ subhalo catalogues constructed from the Millennium-II simulation, we find that, when the entire mass function is fitted with a double Schechter function, the low-mass-end slope (down to $m/M_{host} = 10^{-5}$) is consistent with a power-law exponent, $\alpha = 0.95$.

We have demonstrated that the peak-mass function, or the ratio between the peak mass and final mass functions, are good statistics to test the quality of merger trees. The existence of broken or false links in the trees introduces extra branches and inflates the peak-mass function, which can be overestimated by as much as a factor of 2 in an complex merger tree built from SUBFIND subhaloes, as a result of the 'blending' and 'mass or centre-switching' problems that are present in the latter. This issue is important for studies that focus on the remnants of the most-massive progenitors in a halo, such as studies of streams in the Milky-Way halo. It also has important implications for abundance matching models that match galaxies to simulated merger trees using peak masses: the peak mass function inflated by broken and false links could lead to false matches of galaxies to broken branches, subsequently biasing the inferred properties of massive satellite galaxies. In contrast, our algorithm is robust against these problems by design. It is able to track the tree branches persistently, and recovers a higher final mass function, as well as a lower and universal peak-mass function.

ACKNOWLEDGEMENTS

REFERENCES

Angulo R. E., Springel V., White S. D. M., Jenkins A., Baugh C. M., Frenk C. S., 2012, MNRAS, 426, 2046
Avila S. et al., 2014, MNRAS, 441, 3488
Behroozi P. S., Wechsler R. H., Wu H.-Y., 2013, ApJ, 762, 109
Behroozi P. et al., 2015, MNRAS, 454, 3020
Benson A. J., 2017, MNRAS, 467, 3454
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
Boylan-Kolchin M., Springel V., White S. D. M., Jenkins A., Lemson G., 2009, MNRAS, 398, 1150
Davis M., Efstathiou G., Frenk C. S., White S. D. M., 1985, ApJ, 292, 371
Elahi P. J., Thacker R. J., Widrow L. M., 2011, MNRAS, 418, 320

Gao L., White S. D. M., Jenkins A., Stoehr F., Springel V., 2004, MNRAS, 355, 819
Ghigna S., Moore B., Governato F., Lake G., Quinn T., Stadel J., 1998, MNRAS, 300, 146
Giocoli C., Tormen G., van den Bosch F. C., 2008, MNRAS, 386, 2135
Giocoli C., Tormen G., Sheth R. K., van den Bosch F. C., 2010, MNRAS, 404, 502
Han J., Jing Y. P., Wang H., Wang W., 2012, MNRAS, 427, 2437
Han J., Cole S., Frenk C. S., Jing Y., 2016, MNRAS, 457, 1208
Jauzac M. et al., 2016, MNRAS, 463, 3876
Jiang F., van den Bosch F. C., 2014, MNRAS, 440, 193
Jiang F., van den Bosch F. C., 2016, MNRAS, 458, 2848
Jiang L., Helly J. C., Cole S., Frenk C. S., 2014, MNRAS, 440, 2115
Klypin A., Gottlöber S., Kravtsov A. V., Khokhlov A. M., 1999, ApJ, 516, 530
Klypin A. A., Trujillo-Gomez S., Primack J., 2011, ApJ, 740, 102
Knebe A. et al., 2011, MNRAS, 415, 2293
Knollmann S. R., Knebe A., 2009, ApJS, 182, 608
Lacey C., Cole S., 1994, MNRAS, 271, 676
Lemson G., Virgo Consortium, 2006, preprint (arXiv:astro-ph/0608019)
Li Y., Mo H. J., 2009, preprint (arXiv:0908.0301)
Maciejewski M., Colombi S., Springel V., Alard C., Bouchet F. R., 2009, MNRAS, 396, 1329
Mao T.-X., Wang J., Frenk C. S., Gao L., Li R., Wang Q., 2017, preprint (arXiv:1708.01400)
Moore B., Governato F., Quinn T., Stadel J., Lake G., 1998, ApJ, 499, L5
Moore B., Ghigna S., Governato F., Lake G., Quinn T., Stadel J., Tozzi P., 1999, ApJ, 524, L19
Muldrew S. I., Pearce F. R., Power C., 2011, MNRAS, 410, 2617
Natarajan P. et al., 2017, MNRAS, 468, 1962
Prada F., Klypin A. A., Cuesta A. J., Betancort-Rijo J. E., Primack J., 2012, MNRAS, 423, 3018
Schwinn J., Jauzac M., Baugh C. M., Bartelmann M., Eckert D., Harvey D., Natarajan P., Massey R., 2017, MNRAS, 467, 2913
Springel V., White S. D. M., Tormen G., Kauffmann G., 2001, MNRAS, 328, 726
Springel V. et al., 2005, Nature, 435, 629
Springel V. et al., 2008a, MNRAS, 391, 1685
Springel V. et al., 2008b, Nature, 456, 73
Srisawat C. et al., 2013, MNRAS, 436, 150
Thomas P. A. et al., 2015, preprint (arXiv:1508.053880)
Tormen G., Diaferio A., Syer D., 1998, MNRAS, 299, 728
van den Bosch F. C., Jiang F., 2016, MNRAS, 458, 2870
van den Bosch F. C., Tormen G., Giocoli C., 2005, MNRAS, 359, 1029
Vogelsberger M. et al., 2014, MNRAS, 444, 1518

APPENDIX A: SAMPLING NOISE IN UNBINDING

Consider a singular isothermal sphere sampled with N particles out to a truncation radius r_{max}. Expressed in the dimensionless radius $\tilde{r} = r/r_{max}$, the cumulative number density profile of the halo particles is $N(<\tilde{r}) = N\tilde{r}$. Assuming Poisson fluctuations in the particle counts at each radius, the uncertainty in the estimated potential can be obtained as

$$\delta_\psi(r) \equiv \frac{\sigma_\psi(r)}{|\psi(r)|}, \tag{A1}$$

$$= \frac{\sqrt{\frac{N(r)}{r^2} + \int_r^{r_{max}} \frac{dN(R)}{R^2}}}{\frac{N(r)}{r} + \int_r^{r_{max}} \frac{dN(R)}{R}}, \tag{A2}$$

$$= \frac{1}{\sqrt{N}} \frac{\sqrt{2/\tilde{r} - 1}}{1 - \ln\tilde{r}}. \tag{A3}$$

At $r = r_{max}$, the uncertainty is the smallest with $\delta_\psi(r_{max}) = 1/\sqrt{N}$. The radial dependence of this Poisson noise is shown in Fig. A1.
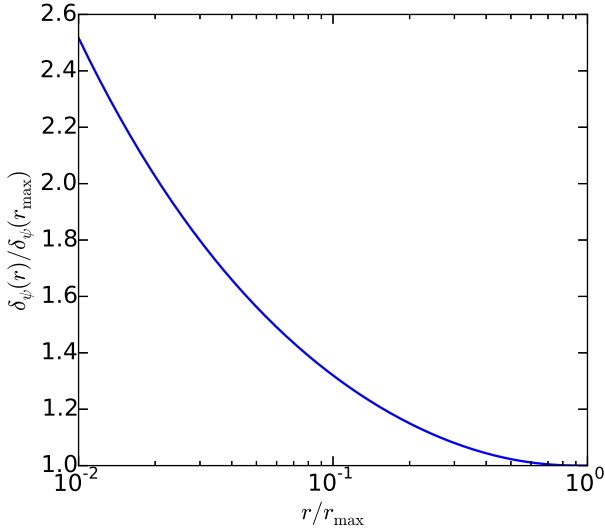
Figure A1. The radial dependence of the Poisson noise in the estimated potential for an isothermal sphere sampled with particles. The noise increases as the radius decreases.
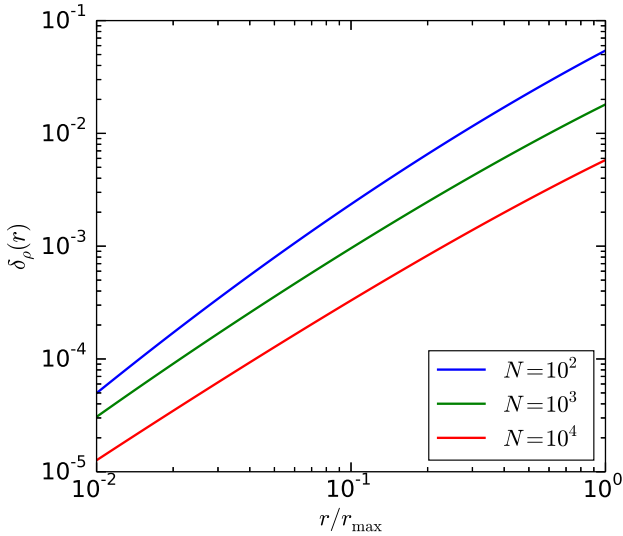

Figure A2. The relative uncertainty in the self-bound density profile due to Poisson noise in the potential, for an isothermal sphere sampled with N particles.

The uncertainty of the bound density profile due to Poisson noise can be estimated as

$$\delta_\rho(r) \equiv \frac{\sigma_\rho(r)}{\rho(r)}, \tag{A4}$$

$$= \frac{\int_{\sqrt{2|\psi|}}^{\sqrt{2(|\psi|+\sigma_\psi)}} \exp\left(-\frac{v^2/2+\psi}{\sigma^2}\right) \mathrm{d}^3 v}{\int_0^{\sqrt{-2\psi}} \exp\left(-\frac{v^2/2+\psi}{\sigma^2}\right) \mathrm{d}^3 v}, \tag{A5}$$

$$= \frac{f(b)}{f(a)} - 1, \tag{A6}$$

where $f(x) = \sqrt{\pi}\mathrm{erf}(x) - 2e^{-x^2}x$, $a = \sqrt{|\psi|/\sigma^2} = \sqrt{2(1-\ln\tilde{r})}$, $b = \sqrt{(|\psi|+\sigma_\psi)/\sigma^2} = \sqrt{2(1-\ln\tilde{r})(1+\delta_\psi(r))}$, and σ^2 is the one-dimension velocity dispersion of the halo. This is plotted in Fig. A2 for a few different sample sizes, N. For $N = 10^3$, the density profile can be recovered to percent level accuracy or better over the entire radial range.

Even though the potential estimate is less accurate at smaller radii, the unbinding of the particles is almost unaffected by Poisson noise at these radii. This is because the potential is much deeper at small r. Given the constant velocity dispersion in the isothermal sphere, most of the particles at a small r are tightly bound, making the unbinding insensitive to the accuracy in potential. This is also true for NFW haloes, in which the velocity dispersion is known to eventually decrease towards the halo centre.

APPENDIX B: TRAPPED SUBHALOES

Massive satellites are likely to sink to the centre of their host halo without getting disrupted. During this process, the orbital energy of the central-satellite pair is converted into the internal energy of each subhalo, and the satellite is trapped in the centre thereafter. To see that these satellites are indeed a distinct population, in Fig. B2 we show the distribution of satellites according to their position and velocity offset from their host subhaloes. Here $\delta_x = \Delta_x/\sigma_x$, where Δ_x is the separation of the satellite from its host subhalo and σ_x is the position dispersion of the 20 most-bound particles in the host subhalo. Similarly, $\delta_v = \Delta_v/\sigma_v$ is the normalized velocity offset. It is obvious that the satellites show a bimodal distribution in this plane, with the trapped subhaloes clustered around $(\delta_x, \delta_v) = (1, 1)$, consistent with them being draw from the most-bound particles in the host subhalo. Note that in this test we have not merged the trapped subhaloes in order to make this plot, but only tag them as trapped as soon as they reach $\delta = \delta_x + \delta_v < 2$ in their orbital evolution.

In Fig. B1, we show the distribution of σ_x and σ_v as a function of subhalo mass. When using the peak mass as a proxy of satellite mass, the distributions are quite similar for central and satellites. σ_x approaches the softening of the simulation in well resolved subhaloes, while it is generally bigger for subhaloes with less than 10^3 particles. The velocity scale σ_v, however, increases with subhalo mass, since more massive objects are dynamically hotter. The median relation can be well fitted by

$$\sigma_v = (-3.2 + 7.4 \ln N)\,\mathrm{km\,s^{-1}}, \tag{B1}$$

where N is the number of particles in the subhalo. However, we caution that the above fitting formula may not be applicable to simulations with different resolutions.

The mass functions of trapped subhaloes are shown in Fig. B3. These objects are mostly massive objects with $m/M > 0.1$, and they evolve only mildly since infall, with the peak and final mass functions differing by a factor of $2 \sim 3$. Note that these trapped subhaloes have already been removed in the figures in the main text of the paper. One might wonder whether the overabundance of subhaloes at the high-mass end in our result is contaminated by trapped subhaloes that are not completely removed. Since the trapped subhaloes are mostly located within $2\sigma_x$, they do not contaminate the radial profiles in Figs 7 and 8 in $10^{13}\,\mathrm{M_\odot}\,h^{-1}$ haloes (corresponding to more than 10^6 particles) where the softening is around $0.002R_{200}$. Subsequently, the existence of the flattening in the subhalo mass function is robust against contaminations from trapped subhaloes, as discussed in Section 3.3.

These merged subhaloes represent the case when numerical resolution of the simulation is no longer able to separate the subhalo from its host. We have carried out a test using a lower resolution simulation and found that the mass functions of the merged subhaloes seem to have converged (though noisily). Even so, whether the galaxies in the trapped subhaloes have merged with their central galaxy is a different problem.
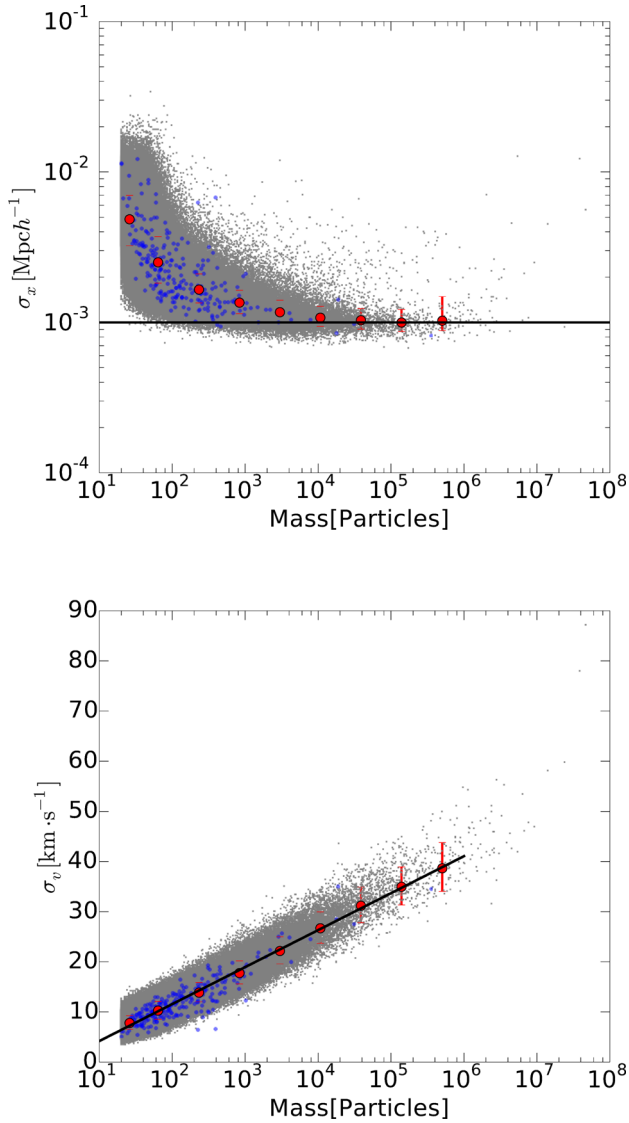
Figure B2. The distribution of satellites in the (δ_x, δ_v) plane at $z = 0$ in Millennium-II. We select satellites that are still resolved and whose host subhalo has more than 10^4 particles. δ_x and δ_v are the position and velocity offset of the satellite from its host subhalo, normalized by the position and velocity dispersions at the host centre, respectively. The red dots are satellites tagged as trapped, while the green dots are the remaining ones. The black line marks $\delta_x + \delta_v = 2$, the critical curve used to identify trapped subhaloes in the merger history.

Figure B1. The position and velocity resolution in subhaloes of different masses in Millennium-II. The grey dots are the results for central subhaloes, while the blue dots are those for satellites. In this figure, we use the peak mass of a satellite as its mass. For clarity, only 1/10 of the central subhaloes and 1/10 000 of the satellite subhaloes are plotted. Filled circles with errorbars show the median and $\pm 1\sigma$ percentiles of the distributions. In the top panel, the horizontal solid line marks the force softening of the simulation. In the bottom panel, the black solid line shows a fit (equation B1) to the median relation.
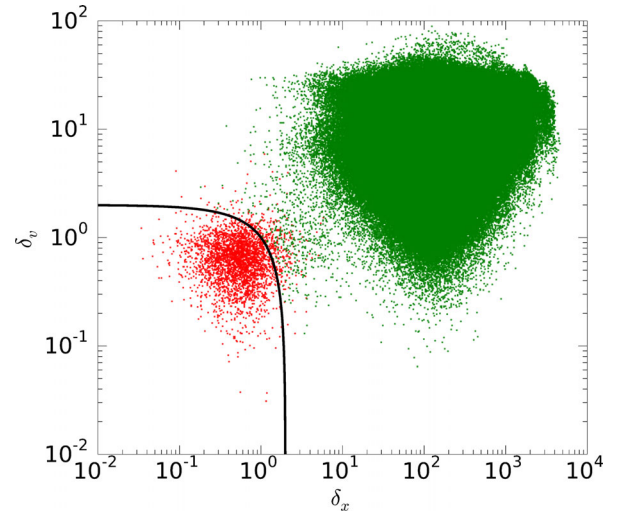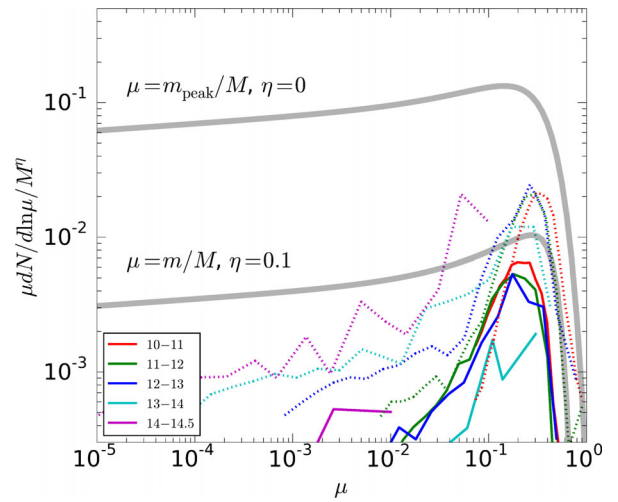

Figure B3. The mass distribution of trapped subhaloes that still survive at $z = 0$ in Millennium-II. The thick grey lines are the same as in Fig. 5 showing the fitted peak and final mass functions for normal subhaloes. The coloured lines show the mass functions of trapped subhaloes in different host halo mass bins (as labelled by $\log[M/M_\odot\,h^{-1}]$), with the solid and dotted lines showing the final and peak mass functions respectively.

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