Mock galaxy catalogues using the quick particle mesh method

Martin White,1,2,⋆ Jeremy L. Tinker3 and Cameron K. McBride4

1Lawrence Berkeley National Laboratory, 1 Cyclotron Road, Berkeley, CA 94720, USA
2Departments of Physics and Astronomy, University of California, Berkeley, CA 94720, USA
3Center for Cosmology and Particle Physics, Department of Physics, New York University, New York, NY 10003, USA
4Harvard-Smithsonian Center for Astrophysics, 60 Garden Street, Cambridge, MA 02138, USA

E-mail: mwhite@berkeley.edu

ABSTRACT
Sophisticated analysis of modern large-scale structure surveys requires mock catalogues. Mock catalogues are used to optimize survey design, test reduction and analysis pipelines, make theoretical predictions for basic observables and propagate errors through complex analysis chains. We present a new method, which we call ‘quick particle mesh’, for generating many large volume, approximate mock catalogues at low computational cost. The method is based on using rapid, low-resolution particle mesh simulations that accurately reproduce the large-scale dark matter density field. Particles are sampled from the density field based on their local density such that they have N-point statistics nearly equivalent to the haloes resolved in high-resolution simulations, creating a set of mock haloes that can be populated using halo occupation methods to create galaxy mocks for a variety of possible target classes.

Key words: galaxies: haloes – galaxies: statistics – cosmological parameters – large-scale structure of Universe.

1 INTRODUCTION
The study of the large-scale structure in the Universe is a cornerstone of modern cosmology. In addition to allowing us to understand the structure itself, such studies offer an incisive tool for probing cosmology and particle physics and set the context for our modern understanding of galaxy formation and evolution. Large galaxy surveys play a key role in this enterprise, and ever larger surveys have provided increasing insight and ever tighter constraints on cosmological models. For almost as long as there have been sky surveys, people have used mock catalogues in order to interpret them (e.g. Neyman, Scott & Shane 1953; Scott, Shane & Swanson 1954; Soneira & Peebles 1977, 1978; Shanks 1979). With time, surveys have become larger and the type of questions we ask from them have changed significantly, becoming steadily more quantitative. Driven by the need for ever more precise theoretical predictions and the use of increasingly complex and sophisticated data-analysis algorithms, simulations and synthetic data sets have become increasingly important.

Within the modern paradigm, wherein galaxies and quasars form and evolve within the haloes of the cosmic web, the natural technique for creating mock catalogues is N-body simulation (see e.g. Springel, Frenk & White 2006, for a recent review). N-body simulations have been the workhorse of modern cosmology for several decades, and codes with high force resolution can accurately predict the abundance, spatial distribution, profiles and substructure of dark matter haloes in representative cosmological volumes (e.g. Springel et al. 2005; Kuhlenn, Vogelsberger & Angulo 2012). Such runs are often quite expensive, however, and while it is certainly possible (e.g. Fosalba et al. 2008; McBride et al. 2009; Heitmann et al. 2010) the practicality of running very large numbers of them for Monte Carlo studies is unclear.1 This is especially true if the majority of the information can be obtained more cheaply. In this vein there are two important considerations: the total runtime of any given method and the total memory requirement. The latter is in many ways the most stringent if we wish to enable the generation of many simulations on commonly available hardware.

Over the years a large number of approximate methods have been developed which produce halo catalogues of sufficient reliability for many tasks, such as predicting the very large-scale distribution of galaxies in simulations. Many of these methods started out as a means to highlight key characteristics of non-linear structure formation and were implemented numerically (this includes such methods as the adhesion approximation [Gurbatov, Saichev & Shandarin 1989; Weinberg & Gunn 1990; Kofman et al. 1992; Melott, Shandarin & Weinberg 1994b; Valageas & Bernardeau 2011]; the log-normal (LGN) model [Coles & Jones 1991]; the truncated Zel’dovich approximation [Coles, Melott & Shandarin 1993; Melott, Pellman & Shandarin 1994a]; the frozen flow approximation [Matarrese et al. 1992]; the free-particle approximation [Short & Coles 2006]; PThalos [Scoccimarro & Sheth 2002; Manera et al. 2013]; Pinocchio [Monaco, Theuns & Taffoni 2002; Monaco et al. 2013]; remapping Lagrangian perturbation theory [LPT; Leclercq et al. 2013]; PATCHY [Kitaura, Yepes & Prada 2013 and

1 Or at the very least, it limits the applicability to a small number of groups.
machine-learning techniques [Xu et al. 2013]). A recent summary of some of these methods can be found in Neyrinck (2013). One of the key observations is that much of the filamentary cosmic web of structures which arises within N-body simulations is also present in these more approximate methods. Indeed, much of the ‘work’ involved in running an N-body simulation in a large volume is evolving linear theory, or modes which are not far from linear. This gives some hope that approximate and cheap algorithms can capture many of the important properties of full N-body simulations.

The most widely used approximation is the LGN model, which has been used to create mock catalogues for many large-scale structure surveys in the past decade (e.g. Cole et al. 2005; Percival et al. 2010; Reid et al. 2010; Beutler et al. 2011, 2012; Blake et al. 2011, among others). By contrast, the Baryon Oscillation Spectroscopic Survey (BOSS) team used mocks based on second-order Lagrange perturbation theory in their recent cosmology analyses (PThalos; Manera et al. 2013).

In this paper we present a new, approximate, method for generating mock galaxy catalogues for the first step in a Monte Carlo simulation of redshift surveys. There are two particular arguments which pushed us to consider the rapid generation of mock catalogues as a means to Monte Carlo our errors. The first is that many steps in the analysis can be quite non-linear, for example the reconstruction procedure for baryon acoustic oscillations (BAO) which involves constrained realizations of interpolating fields and non-linear motions of both galaxy and random points prior to computing the two-point function. The interaction of these non-linearities with the complex observing geometry can be particularly difficult to model directly. The second is that we are primarily interested in quite large scales, where the complexities of galaxy formation and fully non-linear clustering are mitigated. On smaller scales internal estimates of the covariance matrix can be constructed (e.g. via bootstrap or jackknife), however, on large scales these methods do not perform as well (e.g. Norberg et al. 2009, and references therein). For these reasons we wish to investigate a procedure or set of procedures which allow us to generate point sets which mimic the observational samples, at least in terms of low-order statistical properties. As is often the case, increasing fidelity for each simulation comes at a price of increased computational complexity which implies fewer realizations (and more noise in the Monte Carlo; see e.g. Taylor, Joachimi & Kitching 2013) for a fixed computational effort.

The outline of this paper is as follows. In Section 2, we discuss different approaches to producing mock catalogues and the direct N-body simulation method which shall serve as our benchmark. Section 3 details the ‘quick particle mesh’ (QPM) method, the main substance of this paper. Section 4 compares the mock catalogues produced by a variety of different methods, including QPM, to those derived from an N-body simulation. We briefly describe the public implementation of this method, MOCKFACTORY, in Section 5. Finally, we discuss limitations, extensions and comparisons to other methods recently presented in Section 6. Unless otherwise stated, the assumed cosmology is flat Λ cold dark matter (ΛCDM) with $\Omega_m = 0.274$, $\Omega_b = 0.046$, $\Omega_\Lambda = 0.726$, $h = 0.7$, $n = 0.95$ and $\sigma_8 = 0.8$.

2 MOCK CATALOGUE METHODS

Modern galaxy formation models assume that galaxies form and remain in the potential wells of dark matter haloes. The four basic steps of creating mock galaxy catalogues are (1) predicting the evolution of the underlying mass field, (2) locating and characterizing the properties of dark matter haloes, (3) populating the haloes with mock galaxies and (4) applying survey characteristics to the box of galaxies. The halo to mock galaxy mapping is often trained by matching small-scale clustering statistics to the observational galaxy sample (e.g. Reid & Spergel 2009; White et al. 2011). The last step involves adding survey-specific realism to the mock galaxy distribution, such as applying radial and angular selection functions to match the geometry. The final goal is to produce a set of points that statistically matches the spatial distribution of observational galaxy samples, and in our case to be able to do this many times so as to characterize the probability distributions of observables.

Characterizing dark matter haloes, while not strictly required, is a useful step. By definition, haloes represent overdense regions of the mass field (typically 100–300 times the mean density or 200 times the critical density) that arise from non-linear gravitational collapse. Approximate methods of predicting the mass evolution do not attempt to accurately calculate this collapse. Instead the focus is on larger scales dominated by linear and weakly non-linear gravitational evolution. The strongly non-linear interactions within haloes, which are represented in the observational galaxy distributions, can be added with analytic prescriptions on top of the halo distribution. If adequate halo catalogues can be generated, the subsequent steps towards survey-specific mock galaxy catalogues are comparatively straightforward.

For this work, we will compare various fast methods of halo catalogue generation against the relevant properties of halo catalogues derived from periodic box N-body simulations. We have implemented three approximate methods of mock catalogue creation. Two of these represent recent methods used to model galaxy surveys, namely the LGN model and the method of Manera et al. (2013), based on second-order LPT, which we shall refer to as LPT. We discuss the details of our implementations in Appendix A. We compare both of these models to our new method which we refer to as QPM which we describe in detail in Section 3.

The focus of our investigation resolves halo masses and volumes appropriate to current modern redshift surveys, specifically galaxies in Sloan Digital Sky Survey (SDSS)-III BOSS (Dawson et al. 2012). This is a practical choice; the application of these methods is certainly not restricted to modelling these types of galaxies. Indeed, our method is flexible enough to create mocks for a variety of survey specifics and target types, even within the same mock galaxy distribution.

2.1 N-body simulations

We make use of several N-body simulations in this paper, each of the ΛCDM family with the same cosmology ($\Omega_m = 0.274$, $\Omega_b = 0.046$, $\Omega_\Lambda = 0.726$, $h = 0.7$, $n = 0.95$ and $\sigma_8 = 0.8$). These (high-resolution) simulations will form our benchmark and be the fiducial model of ‘truth’. They have also been used in White et al. (2011, 2012) and Reid & White (2011) and more details can be found in those papers.

Our high-resolution simulations resolve all of the haloes described throughout the paper. Briefly, we used an updated version of the TreePM2 code described in White (2002) to evolve 30003 particles ($5.9 \times 10^{10} h^{-1} M_\odot$) in a box of side 2750 $h^{-1}$ Mpc. We ran two realizations of this simulation, differing only in the random number

---

2 This TreePM code has been compared to a number of other codes and shown to perform well for such simulations (Heitmann et al. 2008). The code has been modified to use a hybrid MPI+OpenMP approach which is particularly efficient for modern computing platforms.
seed chosen for the initial conditions. We also ran 20 realizations of the same cosmology using 1500^3 equal mass \((7.6 \times 10^{10} h^{-1} M_{\odot})\) particles in a periodic cube of side length 1500 \(h^{-1}\) Mpc. This second set allows for better estimation of the sample variance at large scales. For each simulation, the initial conditions were generated by displacing particles from a regular grid using second-order LPT (Buchert 1989; Moutarde et al. 1991; Hivon et al. 1995) at \(z = 75\), where the rms displacement is 10 per cent of the mean interparticle spacing.

For each output we found dark matter haloes using a friends-of-friends (FoF) algorithm (Davis et al. 1985) with a linking length of 0.168 times the mean interparticle spacing. This partitions the particles into equivalence classes roughly bounded by isodensity contours of \(100 \times\) the mean density. The position of the most-bound particle and the centre of mass velocity are stored for each halo and used in the comparisons described below.

### 3 QPM: THE METHOD

The QPM method uses a low-resolution particle-mesh (PM) \(N\)-body solver to evolve particles within a periodic simulation volume as is common in the field. The time steps are set to be quite large and the mesh scale and mean interparticle spacing exceed the size of all but the largest dark matter haloes. In this manner we keep both the runtime and the memory requirements modest.

The PM evolution of \(N\) particles within a periodic cube employs fast Fourier transforms on a fixed Cartesian mesh to compute the force. In our configuration, the force mesh has as many nodes as the simulation has particles, and we use \(2 h^{-1}\) Mpc as the mean interparticle spacing as our default. We start each simulation at \(z = 25\) using second-order LPT. The rms particle displacement is 15 per cent of the mean interparticle spacing at this redshift. The code evolves the particles using a second-order leap-frog method, with time steps of \(\Delta \ln a = 15\) per cent. At each step the potential on the grid is computed using a \(1/k^2\) kernel and the force is derived from the potential using fourth-order differencing.

We experimented with different choices for the time step, number of particles and mesh spacing. The choice of time step is a major driver of runtime and the choice of mesh spacing is a major factor in memory requirements. Since we are primarily interested in quite large scales we found that even very large time steps and coarse force meshes were sufficient to get convergence to the required accuracy in our mock halo catalogues. Setting the mesh and particle numbers equal was convenient. Our choice of 15 per cent in \(\ln a\) provided a good trade-off between adequately resolving the change in the growth of structure during the matter-dominated to \(\Omega_m\) dominated transition and speed. Steps in \(a\) and in the linear growth rate produced similar behaviour, we decided on constant steps in \(\ln a\) partly out of simplicity.

In QPM we only resolve the density field on large scales and so must decide how to partition the mass, which in hierarchical theories forms in a bottom-up manner, into a spectrum of dark matter haloes using only large-scale information. We have chosen to do this using the local density for each particle at the time of relevance.\(^3\) Based on its density we anoint a subset of the simulation particles as mock dark matter haloes, recording the position and velocity of these ‘haloes’. Selecting particles based on their local density allows us to work with low particle numbers and hence low computational cost as well as offering a different avenue to explore compared to full simulation. This is similar in spirit to the method used in Cole et al. (1998) or Wechsler (2004), although the technical details differ somewhat.

There is no reason, in principle, why we cannot skip the halo creation step and generate a sample of mock ‘galaxies’ directly by sampling particles. However we have found it very useful to have halo information for our samples, and it allows tighter connection with halo-occupation modelling, as well as the ability to model multiple target samples within the same mock – e.g. red and blue galaxies, or bright and faint galaxies – essentially, and target population that can be modelled within the halo occupation context. Thus we shall always go through haloes in this paper.

To keep the runtime as small as possible the density is estimated using Fourier methods on the same mesh as is used for the force calculation. The density is interpolated on to and off of the mesh using cloud-in-cell interpolation, and the density is smoothed with a Gaussian kernel of 1 mesh cell in width \((i.e. \sigma = 2 h^{-1}\) Mpc). If the smoothing kernel is too large then a scale-dependent bias is introduced\(^4\) near the acoustic peak in the correlation function at \(100 h^{-1}\) Mpc. For a \(2 h^{-1}\) Mpc Gaussian smoothing the scale dependence is small enough that it is not a concern. Once the density field is computed using all of the particles, the position, velocity and density of a random subset of the particles are saved.

As we intend these particles to stand in for haloes, it might be better to store the average velocity smoothed on some scale rather than the particular velocity of the particle. However, at the scales of interest and the resolutions we choose, we are relatively insensitive to ‘virial’ motions within haloes, so we use the particle velocity for simplicity. In fact, comparing the pairwise velocity dispersion of particles in the PM simulations with those in a higher resolution simulation we see that the small-scale velocity field is not well resolved by the PM simulation. Thus we add an additional, Gaussian random velocity of \(125\) km s\(^{-1}\) to each component of the halo velocity. As this value is dependent on the details of the simulation we make it a free parameter in the model and the software described below. It affects the measured quadrupole moment of the correlation function on small scales and can in principle be adjusted to improve agreement with measurements on small scales, if desired.

The mock halo catalogue at each output time is constructed in post-processing. Our goal is to choose particles from the saved subset, with a density-dependent probability, and have them stand in for haloes of a given mass. We select the particles and assign halo masses so as to match the mass function and large-scale bias of haloes, as determined from high-resolution simulations (e.g. Tinker et al. 2008, 2010). The choice of sampling function is arbitrary, and we tried several. A convenient form is a Gaussian in \(\ln(1 + \delta_{\text{sub}})\), with a mean \(\mu\) and a width \(\sigma\) which we can adjust to get the desired clustering. Higher \(\mu\) in general leads to a higher large-scale bias, as shown in Fig. 1.

In our fiducial models we held \(\sigma = 0.1\) fixed and adjusted \(\mu(b)\) so as to reproduce the large-scale bias, \(b\), as a function of halo mass.
It is of course possible to simultaneously adjust both $\mu$ and $\sigma$ as a function of halo mass to better match the scale dependence of halo bias at smaller scales, but we did not find that to be useful for our purposes.

Recently Tassev, Zaldarriaga & Eisenstein (2013) introduced a rapid simulation method which they referred to as ‘COLA: Comoving Lagrangian Acceleration’. This method accelerates a PM code by solving for the evolution of large-scale structure in a frame that is comoving with trajectories calculated in LPT. While our procedure is quite similar in spirit, we have chosen not to implement this speed-up for three reasons. The first is that the runtime of our simulations is already short (taking only a few more steps than COLA). The second is that the COLA method carries with it memory overhead (holding the first- and second-order Lagrangian displacements for each particle) and memory is the primary driver for simulation size or volume in our situation. Finally the COLA method, like any PM scheme, resolves haloes and halo positions reliably only when the mesh scale is significantly finer than the mean interparticle separation and the halo virial radius. If run in this regime over very large volumes the code requires a large amount of memory to hold the force mesh, and this severely limits the machines upon which it can be run.

4 COMPARISON OF METHODS

In this section we compare several different methods of making mock catalogues with those based upon the haloes found in high-resolution N-body simulations. We shall treat the high-resolution simulations as ‘truth’ for the purposes of the comparisons.

4.1 Mass comparison

While we will primarily be concerned with halo and galaxy catalogues in this paper, we begin by comparing the matter fields produced by LPT and QPM with that of the T PikePM simulation (TPM). For simplicity we use one of the larger box simulations and in making this comparison we ensure that all of the methods use the same phases in the linear theory realization, so we expect the structure to match in position across simulations.

A visual comparison of the density field produced by these three methods is given in Fig. 3, which shows as a grey-scale image the mass density in a thin (4 $h^{-1}$ Mpc) slice through the box. The slice is centred on the third most massive halo in the 2.75 $h^{-1}$ Gpc box and zooms in from a region 256 $h^{-1}$ Mpc on a side to 16 $h^{-1}$ Mpc on a side. The 256 $\times$ 256 $\times$ 4 $h^{-1}$ Mpc slice (top row) shows that all three methods produce the same filamentary structure on large scales (with the random phases matched). The eight times coarser mass resolution in the QPM (middle column) simulation compared to the TPM simulation (left-hand column) and the lack of small-scale power in the LPT simulation (right-hand column) is already evident in the middle row. These trends are more apparent in the bottom row, where we see the QPM code has merged a number of haloes together while the LPT code does not produce a bound structure at all. In the original T Pikehalos algorithm (Scoccimarro & Sheth 2002) the mass field was ‘corrected’ by superposing the profiles of bound dark matter haloes as measured in high-resolution simulations. In the LPT method of Manera et al. (2013) this is not done (for the mass field) because the focus is on populating haloes with galaxies. Were a more accurate mass field desired, it would be straightforward to modify the algorithm to include this step.

In general the matter density field produced by the QPM simulation is highly correlated with that of the TPM simulation on scales.(taken from Tinker et al. 2010). The same large-scale clustering can be achieved by simultaneously increasing $\mu$ and $\sigma$ (see Fig. 2), but we are more interested in the models with lower $\sigma$ and holding $\sigma$ fixed is a convenient choice.

We divided the range of halo masses we wish to produce into $N_h$ bins spaced equally in log $M_h$. The results are insensitive to the value of $N_h$, as long as $h(M_h)$ does not vary significantly over the bin. For each mass bin we know how many particles we need to sample to match the mass function (Tinker et al. 2008) and we know the Gaussian we need to sample from to match the desired large-scale bias (from the calibration of $h(M)$ above). Thus we simply loop over the particles and select haloes in a mass bin from them with the necessary probability. The resulting set of mock haloes has the proper number density and large-scale bias as a function of mass. Note that this procedure is equivalent to using Bayes’ theorem in the form $P(M_h|\delta) \propto P(\delta|M_h)P(M_h)$ with $\delta = \ln (1 + \delta)$.

![Figure 1](https://example.com/figure1.png)

**Figure 1.** The large-scale bias, $h$, of particles selected based on their local density. The probability of selection is a Gaussian in $\ln (1 + \delta)$ with mean $\mu$ and width $\sigma$. We show the bias as a function of $\mu$ for two values of $\sigma$: 0.1 (blue squares and dotted lines) and 0.2 (red triangles and dashed lines).

![Figure 2](https://example.com/figure2.png)

**Figure 2.** Contours of the likelihood for fitting the large-scale correlation function as a function of scale $\mu$ and width $\sigma$. Contours are spaced by $\sqrt{2}$ in likelihood, from the peak. In this example we matched the correlation function of the selected particles to $2^7$ times the real-space, matter correlation function. The fit was over the range $30 < r < 60 h^{-1}$ Mpc with 10 bins and 10 per cent (uncorrelated) errors per bin.
The density field in three different simulations with matched initial conditions. The grey-scale is an arcsinh mapping of the density (linear at low density and logarithmic at high density) in a slice $4 h^{-1}$ Mpc thick through a $2750 h^{-1}$ Mpc box. The top row shows a $256 \times 256 h^{-1}$ Mpc region around the third most massive cluster in the simulation ($M \simeq 1.6 \times 10^{15} h^{-1} M_{\odot}$), the middle row a zoom in to a $64 \times 64 h^{-1}$ Mpc region and the bottom row a further zoom in to a $16 \times 16 h^{-1}$ Mpc region. The three columns are for the TPM, the QPM simulation (with $1/8$ the particle number) and the LPT simulation, respectively.

The power spectrum of the matter fields in the TPM and QPM simulations agree to better than 5 per cent to $k \simeq 0.35 h$ Mpc$^{-1}$ beyond which the QPM simulation has less power than the TPM simulation.\(^5\) The cross-correlation is above 95 per cent in Fourier space for $k < 1 h$ Mpc$^{-1}$ and above 95 per cent in configuration space for cubic cells larger than $2.7 h^{-1}$ Mpc (it falls to 87 per cent on the mesh scale of the QPM simulation). The cross-correlation\(^6\) between the initial density field and the field at $z \simeq 0.55$ in the TPM simulations is also well reproduced by the QPM and LPT runs, as shown in Fig. 4.

The distribution of the counts-in-cells is very similar for the QPM and TPM runs on scales above $5 h^{-1}$ Mpc. For the LPT runs there is a pronounced lack of high-density cells, as might be expected from Fig. 3. The discrepancy is an order of magnitude at $\rho = 10 \bar{\rho}$ even for $10 h^{-1}$ Mpc cells.

One way of characterizing the filamentary nature of large-scale structure is through the configuration dependence of the three-point function (in real space). As an example, the three-point function for triangles with $r_2 = 2 r_1$ is shown in Fig. 5. We see that the three-point function of the QPM simulation agrees very well with that of the TPM simulation on all scales plotted. The LPT density field is also in good agreement with TPM on larger scales, but appears to be slightly more circular (i.e. less filamentary) for the smaller triangles.

These comparisons suggest that the QPM simulation is well resolving the large-scale structure in which the mock haloes are to be placed, validating its use as the input to the mock halo creation step.

### 4.2 Halo comparison

Each of our approximations provides us with a catalogue of halo masses, positions and velocities and galaxy (luminosities) positions and velocities. While our primary interest is in the galaxy catalogues, we first discuss the haloes upon which they are built since such galaxy catalogues are limited by the accuracy of the input halo catalogues.

---

\(^5\) The power spectrum of the LPT mass field has much less small-scale power, as expected, and departs significantly from the TPM power spectrum by $k \simeq 0.1 h$ Mpc$^{-1}$.

\(^6\) This is sometimes referred to as the ‘propagator’.
The Fourier-space cross-correlation coefficient, $r = P_{ij}/\sqrt{P_{ii}P_{jj}}$, between the initial conditions and density field at $z \approx 0.55$. The lines are for TPM (red solid), QPM (blue dashed) and LPT (green dotted). Note that the cross-correlation, sometimes called the 'propagator', seen in the TPM simulation is well reproduced by the QPM and LPT approximations on all scales that are of interest here.

The real-space three-point function of the matter density at $z \approx 0.55$ from the three simulations. We show three-point configurations with side lengths in the ratio $r_2 = 2r_1$ as a function of the cosine of the included angle ($\mu$). The upper panel shows $r_2 = 10 h^{-1}$ Mpc and the lower panel $r_2 = 20 h^{-1}$ Mpc. The lines are for TPM (red solid), QPM (blue dashed) and LPT (green dotted). Note that the matter distribution is more circular, or less filamentary, on small scales for LPT than for QPM and TPM, but on large scales all methods agree very well.

The halo catalogues are necessarily sparser than the particles which defined the mass density field, so in order to have better statistics across a wide range of scales we make use of the 20 lower resolution simulations, with a corresponding 20 mock simulations for each. Throughout we shall plot the average of each statistic over the 20 realizations to reduce sample variance. Fig. 6 shows the real-space halo correlation function (i.e. configuration space two-point function) for the four mock catalogues for a sample of haloes with $M > 10^{13} h^{-1} M_\odot$, chosen to roughly match the number density of BOSS CMASS galaxies (White et al. 2011; Anderson et al. 2012; Dawson et al. 2013) though other mass ranges look qualitatively similar.

Fig. 6 shows that the large-scale bias of the haloes produced by all of the methods is approximately correct. (We are not concerned here with the level of agreement below $10 h^{-1}$ Mpc scales, though we shall come back to that in the next section.) On large scales we see that the LGN mocks agree almost perfectly with the TPM mocks, as expected. Those mocks are designed to produce a given two-point function, in this case produced using convolution Lagrangian perturbation theory (CLPT; Carlson, Reid & White 2013). Since CLPT provides a good fit to the halo correlation function we fully expected our LGN mocks to fit as well, and this expectation is born out. The QPM mocks provide an adequate fit to the real-space correlation function, with a slight excess clustering near $20 h^{-1}$ Mpc. This excess appears visually quite significant due to the choice of $r^2/\xi$ as the $y$-axis, but is only 3 per cent in $\xi$ at $20 h^{-1}$ Mpc and is completely acceptable for our purposes. Finally, we see that the LPT mocks tend to undershoot the correlations over a wide range of scales and have a slightly different shape near the acoustic peak ($r \sim 100 h^{-1}$ Mpc). This is due to our choice of a broad range of haloes masses and a single linking length. For a narrower range of haloes we can adjust the linking length to obtain a better match, but we found it difficult to simultaneously match the bias and mass function across a wide range of halo masses with a single value of the linking length.

The situation in redshift space is shown in Fig. 7 which presents the monopole and quadrupole moments of the correlation function. For the monopole all of the mocks agree relatively well. Once again the LPT mocks show a slight deficit of clustering across a broad range of scales, the QPM mocks overshoot near $20 h^{-1}$ Mpc and the LGN mocks match very well. For the quadrupole the agreement is less good, but still quite acceptable given that the observations tend to have larger errors on the quadrupole than the monopole.

An alternate view of the redshift-space statistics is given in Fig. 8, which presents the monopole and quadrupole power spectra divided by a smooth model. The trends are very similar to those in Fig. 7.

Fig. 9 shows the configuration-space three-point correlation function for triangles with $r_2 = 2r_1 = 20 h^{-1}$ Mpc as a function of the cosine of the included angle. The TPM and LPT mocks agree quite well in the amplitude and shape of the three-point function. The
Figure 7. The redshift-space monopole (upper) and quadrupole (lower) two-point correlation functions for haloes with $M > 10^{13} h^{-1} M_\odot$. The shaded red band shows the $1\sigma$ error on the mean of the correlation function from the 20 TPM mocks. The uncertainty on the other lines (QPM: dashed blue; LPT: dotted green; LGN: dot–dashed magenta) is of the same size and is discussed in the text.

Figure 8. The redshift-space monopole (upper) and quadrupole (lower) dimensionless power spectra for haloes with $M > 10^{13} h^{-1} M_\odot$. We have divided each spectrum by a smooth model which is the linear theory, real-space spectrum using the smooth fit to the linear theory transfer function of Eisenstein & Hu (1998) and a scale-independent bias of $b = 2$. The amplitude thus reflects the effects of redshift-space distortions as well as non-linearity, and the acoustic oscillations are highlighted. The shaded red band shows the $1\sigma$ error on the mean of the spectra from the 20 TPM mocks. The uncertainty on the other lines (QPM: dashed blue; LPT: dotted green; LGN: dot–dashed magenta) is of the same size.

QPM mocks agree slightly less well, but still have approximately the right shape, indicating the prevalence of filaments in the cosmic web. (Note that 10–20 $h^{-1}$ Mpc are the scales where the QPM mocks agree least well with the TPM simulation for the two-point function.) The clear outlier is the LGN mocks, where the remapping of the original Gaussian density field does not produce the same filamentary, beaded web as is produced by gravitational evolution and the shape dependence of the three-point function is very different from that of the other mocks. From this we infer that laying down haloes based on a Gaussian field does not produce the right structure in the halo density field. Laying down haloes based on the non-linearly evolved field does a slightly better job, but since the

mock haloes are selected at random based only on local density the filamentary nature of the halo distribution is slightly washed out. Finding overdense regions in the evolved field does best of all, but is the most expensive in terms of memory or computational cost.

Fig. 10 shows the fractional uncertainty in the real-space halo correlation function (upper) or monopole (middle) or quadrupole (lower) of the redshift-space correlation function as estimated from the 20 mock catalogues of each type. From 20 realizations we expect the error on the error to be about 30 per cent, so all of these methods agree within errors.
reduce the error and provide a stronger test. Near the acoustic peak the fractional error on the real-space correlation function for this volume and number density, assuming linearly biased linear theory and Gaussian statistics, is in good agreement with the numbers shown in Fig. 10 and the error is almost totally dominated by sample variance. Given that all of the methods are doing a good job of modelling the largest scales which dominate the sample variance, it is not unreasonable to expect they produce the same scatter in the correlation function.

4.3 Galaxy comparison

From the halo catalogues\(^7\) we create galaxy catalogues using a halo occupation distribution (HOD) approach (Benson et al. 2000; Peacock & Smith 2000; Seljak 2000; Scoccimarro et al. 2001; White, Hernquist & Springel 2001; Berlind & Weinberg 2002; Cooray & Sheth 2002). We use the HOD parametrization of Tinker et al. (2012). Once a set of HOD parameter values has been chosen, we populate each mock halo in a given output with mock ‘galaxies’. The HOD provides the probabilities that a halo will contain a central galaxy and the number of satellites. The central galaxy is placed at the centre of the halo (which is the position of the most bound particle in the halo for the FoF haloes and the position of the particle for the mock haloes), and we place satellites assuming a spherical Navarro, Frenk & White (1996) profile with a concentration determined using the method of Munoz-Cuartas et al. (2010).

We follow standard practice and adjust the HOD parameters for each mock to fit the small-scale projected two-point clustering. As an illustrative example we have chosen to use the projected correlation function of BOSS CMASS galaxies reported in White et al. (2011). In each case we are able to match the clustering between the mocks and the observational measurements, as shown in Fig. 11.

The real-space correlation function for each of the mock galaxy catalogues is shown in Fig. 12 which is the analogue of Fig. 6. As in that case we see that the mocks agree with the high-resolution \(N\)-body simulation very well over a broad range of scales. The QPM mocks have slightly more power near 20 \(h^{-1}\) Mpc but only at the few per cent level. This indicates that we are placing haloes correctly, in a statistical sense, within the large-scale mass field and any higher order correlations between haloes that we are missing by such a random assignment do not impact the low-order galaxy statistics.

The situation in redshift space is shown in Fig. 13 which is the analogue of Fig. 7 and presents the monopole and quadrupole moments of the correlation function. The level of agreement is very much as we saw in Fig. 7, indicating that we are modelling fingers-of-god and satellite fractions comparably in all three cases. The complementary view is given in Fig. 14, which shows the monopole and quadrupole of the redshift-space power spectrum.

Finally, we show the fractional uncertainty in the real space, monopole and quadrupole correlation functions versus scale in Fig. 15. As in Fig. 10 the small number of high-resolution \(N\)-body simulations means we only have a \(\sim\)30 per cent prediction of the fractional error with which to compare. Within this uncertainty the methods agree well, as expected since they are approximately linear tracers of the large-scale matter field.

4.4 Computational cost

For each of the mock catalogue methods described above the most time consuming steps involve fast Fourier transforms (FFT). We use the parallel, real Fourier transforms in the FFTW package.\(^8\) The FFT scales with problem size as \(N \log N\), so that doubling the linear dimension of the grid increases the runtime by approximately an order of magnitude (in addition to requiring about an order of magnitude more memory). The additional cost of moving the particles is small, though with more particles and more clustered particles the time taken to find haloes with the FoF algorithm and halo properties

\(^7\) Since the LGN mocks do not produce a halo catalogue with assigned masses, we do not include them in this section. The way in which LGN mocks are typically used is to directly produce the galaxies from the LGN density field, rather than going through an intermediate step. The comparison is then qualitatively similar to that for the haloes which we discussed above and we gain little insight by repeating it here.

\(^8\) www.fftw.org
Figure 13. The redshift-space monopole (upper) and quadrupole (lower) two-point correlation functions for mock galaxies. The shaded red band shows the $1\sigma$ error on the mean of the correlation function from the 20 TPM mocks. The uncertainty on the other lines (QPM: dashed blue; LPT: dotted green) is of the same size and is discussed in the text.

Figure 14. The redshift-space monopole (upper) and quadrupole (lower) dimensionless power spectra for mock galaxies. We have divided each spectrum by a smooth model which is the linear theory, real-space spectrum using the smooth fit to the linear theory transfer function of Eisenstein & Hu (1998) and a scale-independent bias of $b = 2$. The amplitude thus reflects the effects of redshift-space distortions as well as non-linearity, and the acoustic oscillations are highlighted. The shaded red band shows the $1\sigma$ error on the mean of the spectra from the 20 TPM mocks. The uncertainty on the other lines (QPM: dashed blue; LPT: dotted green) is of the same size.

Figure 15. The fractional uncertainty in the real-space correlation function (upper) or monopole (middle) or quadrupole (lower) of the redshift-space correlation function as estimated from the 20 mock catalogues of each type. From 20 realizations we expect the error on the error to be about 30 per cent, so we are unable to statistically distinguish these different approximations from the simulations we have.

LGN simulations run a few times faster than the LPT simulations, which are comparable to the QPM simulations which is more than two orders of magnitude faster than the TPM simulation.

5 Mock Factory

So far we have dealt with producing a periodic cube populated with mock galaxies according to a specified HOD. In many instances it is also important to reproduce other aspects of the data, such as applying redshift-space distortions and trimming to the survey geometry. Our implementation of these steps has three separate parts, any of which can be used independently: (1) creation of mock halo files from local densities; (2) application of an HOD to map mock galaxies to haloes and (3) transformation of the periodic box of objects into a realistic catalogue on the sky. We make this code, referred to as MockFactory, publicly available.9

The first step was the topic of the previous sections. The code selects, from a random sample of dark matter particles with densities, a set of mock haloes with masses. The haloes match a specified mass function and large-scale bias versus mass relation, for which we use the fits of Tinker et al. (2008, 2010) for overdensity $\Delta_v = 200$ haloes. The result of this stage is a periodic box of haloes, stored as a HALOFILE. The operation of the code is controlled through a parameter file, which is ASCII text. The performance and memory considerations of the code depend on the size of the PM simulation and the fraction of particles that are retained in the subsample file from which the mock haloes are culled. For test runs on a PM simulation with $1280^3$ particles, of which 10 per cent were retained in the subsample file, the runtime on a typical workstation core is approximately 50 min to create the mock haloes, requiring 5.5 GB in memory (effectively no overhead relative to the size of the input subsample file). The runtime scales linearly with the size of the subsample file.

The second part of the code populates a halo catalogue (typically the HALOFILE described above, but it is possible to use any halo file provided it is in the proper format) with artificial galaxies. This applies a HOD as described in Section 4.3. The central galaxy is

9 http://github.com/mockFactory
placed at the centre of the halo and satellites are placed assuming a spherical Navarro et al. (1996) profile with a concentration determined using the method of Munoz-Cuartas et al. (2010), which can be scaled by a factor in the input parameter file. In the examples used in this paper, we use the central-satellite HOD parametrization used in Tinker et al. (2012), although the code has a number of options for these parametrizations. The code can also adjust the input HOD parameters to match a desired galaxy number density by rescaling all the halo mass values by the same factor. To fill the HaloFile with mock galaxies requires substantially less time and memory: approximately 3 min for the example introduced above and only the HaloFile is held in memory.

The third part of the code, called make_survey, takes a periodic box of objects (mock galaxies, quasars, haloes or particles from a simulation box) and projects them on the sky, optionally applying various layers of realism to represent a mock survey. Our implementation can perform a number of steps which include the following:

(a) volume remapping of the periodic box using BoxRemap (Carlson & White 2010),
(b) box translation and rotation,
(c) sky projection using cosmological distances;
(d) modelling redshift distortions using peculiar velocities;
(e) trimming to a survey footprint using a MANGLE mask (Swanson et al. 2008);
(f) downsampling based on angular sky completeness and
(g) downsampling based on radial selection.

A configuration file controls which of these steps are applied, as well as giving any required parameters. In addition, several tools exist within the code base to assist the user in determining and testing the parameters. The make_survey code is written to be relatively efficient. It reads the input catalogue one object at a time, processes each object through all required steps and outputs any object that makes all cuts before moving on to the next. This results in minimal memory usage as the input catalogue is never fully read into memory, nor is the output stored. The runtime is sufficiently fast such that it should not require significant computing resources even for a large number of mock catalogues.

6 SUMMARY

In the rising age of large-scale galaxy redshift surveys, the need for fast and accurate methods to create mock galaxy catalogues is at a premium. In this paper we have presented a new method for creating such mocks, which we call the QPM method. Our method is based on using rapid, low-resolution PM simulations that accurately reproduce the large-scale dark matter density field. Particles are sampled from the density field based on their local density such that the sampled particles have N-point statistics nearly equivalent to the haloes resolved in high-resolution simulations. Thus our method creates a set of mock haloes that can be populated using halo occupation methods to create galaxy mocks for a variety of possible target classes.

We compare the real- and redshift-space clustering statistics of our mock haloes and mock galaxies to those of second-order LPT and the LGN method. These two methods are currently the most widely used for creating large sets of mocks. We use high-resolution N-body simulations (TPM) as ‘truth’ in these comparisons, which focus on halo mass ranges applicable to those probed by BOSS-type samples. The halo clustering produced by the QPM method agrees quite well with TPM results. For two-point statistics, our implementation of LPT slightly underpredicts the halo clustering. The LGN method offers an excellent match to the TPM halo clustering, but cannot reproduce the behaviour of the density field for higher order statistics.

There are several benefits of our method compared to other current methods. In comparison to some perturbation theory methods, QPM offers improvement in terms of both runtime and memory requirements. Although a PM simulation performs more FFTs than LPT, with our particle sampling scheme the grid size required to achieve the same halo mass resolution is less, thus the transforms run significantly faster and require less memory. Although we restrict our comparisons to halo mass ranges \( \gtrsim 10^{12} \ h^{-1} \ M_\odot \), the lower mass limit of the QPM HaloFile can be set to any required value. Increasing the mass resolution of LPT requires increasing the size of the density grid on which the FFTs are performed. We note, however, that our current implementation of QPM is designed for BOSS-type galaxy samples, thus achieving accurate halo catalogues at \( M_{\text{halo}} \lesssim 10^{12} \ h^{-1} \ M_\odot \) requires additional calibration. When comparing TPM to the LGN catalogues we find the LGN method reproduces the two-point clustering better than either QPM or LPT. The filamentary nature of the field is not well reproduced however. In addition, a single QPM HaloFile can be used to create galaxy mocks for a variety of targets with varying bias values; the LGN simulation is tuned to a specific value of bias, thus must be re-run for any variation of target class. In some situations this is a disadvantage.

The QPM method, as discussed here, uses a simple scheme of determining mock haloes from local particle densities. While successful, there were a few discrepancies in the halo clustering that were not present in the mass fields. It is possible that an improved mapping could be found, perhaps using more information from the density field (or initial conditions) or conditionally selecting particles based on previous ‘draws’, and this would mitigate some of these discrepancies. We have not explored this avenue here as we do not believe the discrepancies to be a serious limitation at present—the differences are small and often reduced further once mapped to mock galaxies.

We have released our QPM implementation as part of the mockFactory software package. Users supply their own PM simulations, and the code can create the mock halo file, create the mock galaxy catalogue and convert this galaxy catalogue from a periodic cube to a cut-sky angular mock in redshift space. We do not supply the PM code, but there are many public codes that have been thoroughly tested and used throughout the community: e.g. the code of Klypin & Holtzman (1997),\(^{11}\) PMFAST\(^{12}\) (Merz, Pen & Trac 2005), as well as TREEPM codes that can be run as PM-only, such as GADGET\(^{13}\) (Springel 2005) and the TPM code\(^{14}\) of Bode & Ostriker (2003). Additionally, if the reader is sufficiently motivated, instructions for creating one’s own PM code are also available.\(^{15}\)

QPM offers a method that is fast, requires relative little memory and is highly flexible. Its low computational requirements and high flexibility make QPM optimal for the next generation of large-scale structure surveys that will push the limits of current methods both

---

\(^{10}\) http://mwhite.berkeley.edu/BoxRemap/

\(^{11}\) http://astro.nmsu.edu/~aklypin/pmcode.html

\(^{12}\) http://www.cita.utoronto.ca/~merz/pmfast/

\(^{13}\) http://www.mpa-garching.mpg.de/gadget/

\(^{14}\) http://www.astro.princeton.edu/~bode/TPM/

\(^{15}\) http://astro.uchicago.edu/~andrey/talks/PM/pm_slides.pdf
in terms of the volume they probe and the bias of the galaxies they target.

ACKNOWLEDGEMENTS

The simulations used in this paper were analysed at the National Energy Research Scientific Computing Center. MW is supported by the NSF and NASA.

REFERENCES

Anderson L. et al., 2012, MNRAS, 427, 3435
Benson A. J., Cole S., Frenk C. S., Baugh C. M., Lacey C. G., 2000, MNRAS, 311, 793
Berlind A., Weinberg D. H., 2002, ApJ, 575, 587
Beutler F. et al., 2011, MNRAS, 416, 3017
Beutler F. et al., 2012, MNRAS, 423, 3430
Blake C. et al., 2011, MNRAS, 415, 2892
Bode P., Ostriker J., 2003, ApJS, 125, 1
Bouchet F. R., Colombi S., Hivon E., Juszkiewicz R., 1995, A&A, 296, 575
Buchert T., 1989, A&A, 223, 9
Buchert T., Melott A. L., Weiss A. G., 1994, A&A, 288, 349
Carlson J., White M., 2010, ApJS, 190, 311
Carlson J., Reid B. A., White M., 2013, MNRAS, 429, 1674
Cole S., Hatton S., Weinberg D. H., Frenk C. S., 1998, MNRAS, 300, 945
Cole S. et al., 2005, MNRAS, 362, 505
Coles P., Jones B., 1991, MNRAS, 248, 1
Coles P., Melott A. L., Shandarin S. F., 1993, MNRAS, 260, 765
Cookay A., Sheth R., 2002, Phys. Rep., 372, 1
Davis M., Efstathiou G., Frenk C. S., White S. D. M., 1985, ApJ, 292, 371
Dawson K. et al., 2013, AJ, 145, 10
Eisenstein D. J., Hu W., 1998, ApJ, 511, 5
Fosalba P., Gaztanaga E., Castander F., Manera M., 2008, MNRAS, 391, 431
Gurbatov S. N., Saichev A. I., Shandarin S. F., 1989, MNRAS, 236, 385
Hockney R. W., Eastwood J. W., 1980, Computer Simulation Using Particles. McGraw-Hill, New York
Kaiser N., 1987, MNRAS, 227, 1
Kitaura F.-S., Gallerani S., Ferrara A., 2012, MNRAS, 420, 61
Kofman L., Pogosyan D., Shandarin S. F., 1989, MNRAS, 236, 385
Koerner J. L., Robertson B. E., Kravtsov A. V., Klypin A., Warren M. S., 2013, MNRAS, 433, 2389
Kuhlen M., Vogelsberger M., Angulo R., 2012, Phys. Dark Universe, 1, 50
Kuhlen M., Vogelsberger M., Angulo R., 2012, Phys. Dark Universe, 1, 50
Lacey C., Cole S., 1994, MNRAS, 271, 676
Leclercq F., Jasche J., Gil-Marin H., Wandelt B., 2013, J. Cosmol. Astropart. Phys., preprint (arXiv:1305.4642)
McBride C., Berlind A., Scoccimarro R., Wechsler R., Busch M., Gardner I., van den Bosch F., 2009, BAAS, 41, 253
Manera M. et al., 2013, MNRAS, 428, 1036
Matarrese S., Moscardini L., Saez D., 1992, MNRAS, 259, 437
Melott A. L., Shandarin S. F., 1994a, MNRAS, 269, 626
Melott A. L., Shandarin S. F., Weinberg D. H., 1994b, MNRAS, 269, 626
Merz H., Pen U.-L., Trac H., 2005, New Astron., 10, 393
Monaco P., Bode P., Taffoni G., 2002, MNRAS, 331, 587
Monaco P., Sefusatti E., Borgani S., Crocce M., Fosalba P., Sheth R. K., 2013, MNRAS, 433, 2389
Moulin F., Alimi J.-M., Bouchet F. R., Pellat R., Ramani A., 1991, ApJ, 393, 437
Munoz-Cuartas J. C., Maccio A., Gottlober S., Dutton A., 2010, in Proceedings of Cosmic Radiation Fields: Sources in the Early Universe (CRF 2010). Proceedings of Science, p. 16, available at: http://pos.sissa.it/cgi-bin/reader/conf.cgi?confid=121
Tassev S., Zaldarriaga M., Eisenstein D. J., 2013, J. Cosmol. Astropart. Phys., 06, 36
Taylor A., Joachimi B., Kitching T., 2013, MNRAS, 432, 1928
Tinker J., Kravtsov A. V., Klypin A., Abazajian K., Warren M., Yeges, Gottlober S., Holz D. E., 2008, ApJ, 688, 709
Tinker J. L., Robertson B. E., Kravtsov A. V., Klypin A., Warren M. S., Yeges, Gottlober S., 2010, ApJ, 724, 878
Tinker J. et al., 2012, ApJ, 745, 16
Valageas P., Bernardeau F., 2011, Phys. Rev. D, 83, 043508
Wechsler R. H., 2004, in Mulchaey J. S., Dressler A., Oemler A., eds, Carnegie Observatories Astrophysics Series, Vol. 3, Clusters of Galaxies: Probes of Cosmological Structure and Galaxy Evolution. Carnegie Observatories, Pasadena
Weinberg D. H., Gunn J. E., 1990, MNRAS, 247, 260
White M., 2002, ApJS, 143, 241
White M., Hernquist L.,Springel V., 2001, MNRAS, 328, 519
White M. et al., 2001, ApJ, 550, L129
White M. et al., 2011, ApJ, 728, 126
White M., 2002, ApJS, 143, 241
White M., 2002, ApJS, 143, 241
White M., 2002, ApJS, 143, 241
White M., Hernquist L., Springel V., 2001, ApJ, 550, L129
White M. et al., 2011, ApJ, 728, 126
White M. et al., 2012, MNRAS, 424, 935
Xu X., Ho S., Trac H., Schneider J., Poczos B., Ntampaka M., 2013, ApJ, 772, 147

APPENDIX A: MOCK CATALOGUES

We have compared the QPM method described in this paper to a number of commonly used approximate methods, in addition to high force and mass resolution simulations. In this appendix we give the technical details for the various mock catalogue schemes, other than QPM, presented in this paper.

A1 Log-normal model

A popular method for creating mock catalogues is the LGN model of Coles & Jones (1991). This model has been used to generate mock catalogues for analyses of the 2dF (e.g. Cole et al. 2005), SDSS (most recently, Percival et al. 2010; Reid et al. 2010), WiggleZ (Blake et al. 2011) and 6dF (Beutler et al. 2011, 2012) surveys, among others.

In this model a Gaussian density field is generated with a correction function $\xi_G$ and then the field is exponentiated to obtain a LGN field which is then sampled (with probability proportional
to density) to produce a set of particles. In order to obtain any desired $\xi_{LN}$ the correlation function for the Gaussian field should obey $\xi_G = \ln(1 + \xi_{LN})$. The density distribution of the resulting particle set is LGN, while its two-point function matches the desired function by construction. These particles thus behave in their lower-order statistics very similarly to the haloes in N-body simulations or galaxies in observational surveys.

Often the LGN model is used to produce fields in real space, ignoring redshift-space distortions. There are two ways of introducing redshift-space distortions into the model. First, one can follow the reasoning in Coles & Jones (1991) who point out that the LGN model can be thought of as a kinematical model in which the velocity field is assumed to remain always linear. In this case the continuity equation can be solved for the density, which is an exponential in the velocity divergence, $\theta$. In linear theory $\theta$ is simply proportional to $\delta$, which is assumed to obey Gaussian statistics. Thus we can draw our velocities from the Gaussian field, $\delta_G$ which is exponentiated to give $\delta_{LN}$ by treating it as (proportional to) the velocity divergence and generating $v(\mathbf{r}) \propto (k^2 / k^2) \delta_{LN}(\mathbf{k})$. Such an approach has been used in Kitaura, Gallerani & Ferrara (2012) for example.

A second method assumes a model for the effect of redshift-space distortions on the correlation function (or power spectrum) of the Gaussian field and simply generates a random realization of a Gaussian field from the anisotropic two-point function. This approach is followed in Butler et al. (2012) using the Kaiser approximation (Kaiser 1987) for the power spectrum. We follow the first approach rather than the second, because there are some situations where it is useful to have the velocity information explicitly.

Our implementation of a LGN code takes as input a target, real-space correlation function and a value of $\beta = f(\Omega)/b$. We generate this correlation function using CLPT, as described in Carlson et al. (2013). The power spectrum of the Gaussian field is computed semi-analytically by integrating $\ln(1 + \xi_{LN})$ against $\sin(kr)/kr$ and then a Gaussian random field is generated from this power spectrum in the usual way. A large number of points are thrown randomly throughout the volume, and kept with probability proportional to the density. We interpolate the density field to the trial position using ‘cloud-in-cell’ interpolation (e.g. Hockney & Eastwood 1980).

We find that the Gaussian power spectrum damps strongly at $k \approx 1 \text{h Mpc}^{-1}$ but that if we wish to sample the density field with tracers so as to obtain a point set then using grids with a finer spacing provides better agreement between the two-point function of the generated point set and the target than a coarser grid (for further discussion see Fig. A1). If the analysis is done directly on the gridded field, or if the interpolation scheme does not ‘clip the peaks’, then the coarser grid is adequate. As the velocity field is generated from the Gaussian velocity divergence the field in redshift space approaches the prediction of Kaiser (1987) on large scales.

A2 LPT model

The use of second-order LPT (Buchert 1989; Moutarde et al. 1991; Hivon et al. 1995; Scoccimarro 1999) to generate density distributions in which haloes are placed is the basis of the PThalos algorithm (Scoccimarro & Sheth 2002). A variant of this approach, in which the haloes in the LPT density field are found using the FoF algorithm (Davis et al. 1985), was introduced in Manera et al. (2013).

We follow the method outlined in Manera et al. (2013). From an initial power spectrum we generate displacements using LPT. We then displace particles from an initially Cartesian grid along these displacement vectors to form a particle-based realization of the density field. We identify haloes using the FoF algorithm and assign masses to the haloes by abundance matching to the mass function of an N-body simulation (or a fit to it).

There are several choices which must be made in this algorithm (see also Neyrinck 2013; Leclercq et al. 2013, for related discussion). First we need to decide whether to damp the input power spectrum at high $k$. While there are good arguments for such a truncation in some contexts, we found that using an undamped spectrum is better in our implementation of this algorithm. This is because we displace particles from a Cartesian grid, and a smoothened spectrum distorts the grid on large scales but leaves vestiges of it clearly visible in the particle distribution on small scales. When coupled with a FoF halo finder working with a spherical linking criterion this introduces undesirable artefacts. A natural choice is to set the grid resolution to match the mean interparticle spacing. If the particle loading and grid are too coarse we lose both mass resolution and linear resolution. Increasing the resolution of the grid too much can also cause problems however, as at high resolution LPT fails\textsuperscript{16} and leads to artefacts like haloes in voids (Buchert, Melott & Weiss 1994; Buchet et al. 1995; Sahni & Shandarin 1996; Neyrinck 2013; Leclercq et al. 2013). A good compromise is a grid spacing and mean interparticle spacing of $O(1 \text{h}^{-1} \text{Mpc})$ (see also Neyrinck 2013). We use a 3000$^3$ grid for the 2.75 $h^{-1}$ Gpc boxes and a 1500$^3$ grid for the 1.5 $h^{-1}$ Gpc boxes, with an equal number of particles as grid points.

\textsuperscript{16}This is particularly true using a Fourier-based estimator such as we employ.
formation. These complexities are worse for hierarchical models such as CDM (as compared with e.g. hot dark matter; Melott et al. 1994b; Monaco et al. 2002). We follow Manera et al. (2013) and use the FoF algorithm. This algorithm has a single parameter, the linking length and produces a unique partition of the particles into groups (of multiplicity one particle or higher). For a model such as CDM it is natural to express the linking length in terms of the mean interparticle separation, with typical numbers being 0.1–0.2 (Davis et al. 1985; Lacey & Cole 1994). In our situation the appropriate linking length to use (and whether it should scale only with the mean interparticle separation) is less clear. We follow Manera et al. (2013) and empirically adjust the linking length to match the clustering strength of a constant number density sample of haloes seen in our high-resolution N-body simulation. Specifically, for the halo catalogue produced at any linking length we rank order the haloes by particle number and choose those whose number densities lie within a narrow range determined from the abundance of haloes in the high-resolution simulation. For haloes spanning an octave in mass centred on \( \lg M = 12.5, 13 \) and \( 13.5 \) \( (h^{-1} M_{\odot}) \) we find number densities of \(-3.24 < \lg n < -2.95, -3.82 < \lg n < -3.45, -4.09 < \lg n < -4.54 \) \( (h^{-3} \text{ Mpc}^3) \) in the \( 2.75 h^{-1} \text{ Gpc} \) simulation. As the linking length is increased the large-scale bias of each sample, as determined by the amplitude of the halo-mass cross-spectrum at low \( k \), decreases. We were unable to find a single linking length which matched the large-scale bias for all three mass/abundance bins, so we chose the linking length which best matched the \( 10^{13} h^{-1} M_{\odot} \) sample. This linking length was the same as found by Manera et al. (2013), \( b = 0.36 \).

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