A Faster Fourier Transform?
Computing Small-Scale Power Spectra and Bispectra for Cosmological Simulations in $O(N^2)$ Time

Oliver H. E. Philcox\textsuperscript{1,2,*}
\textsuperscript{1}Department of Astrophysical Sciences, Princeton University, Princeton, NJ 08544, USA
\textsuperscript{2}Harvard-Smithsonian Center for Astrophysics, 60 Garden St., MA, 02138, USA

Accepted XXX. Received YYY; in original form ZZZ

ABSTRACT
We present $O(N^2)$ estimators for the small-scale power spectrum and bispectrum in cosmological simulations. In combination with traditional methods, these allow spectra to be efficiently computed across a vast range of scales, requiring orders of magnitude less computation time than Fast Fourier Transform based approaches alone. These methods are applicable to any tracer; simulation particles, halos or galaxies, and take advantage of the simple geometry of the box and periodicity to remove almost all dependence on large random particle catalogs. By working in configuration-space, both power spectra and bispectra can be computed via a weighted sum of particle pairs up to some radius, which can be reduced at larger $k$, leading to algorithms with decreasing complexity on small scales. These do not suffer from aliasing or shot-noise, allowing spectra to be computed to arbitrarily large wavenumbers. The estimators are rigorously derived and tested against simulations, and their covariances discussed. The accompanying code, HIPSTER, has been publicly released, incorporating these algorithms. Such estimators will be of great use in the analysis of large sets of high-resolution simulations.

Key words: methods: statistical, numerical – Cosmology: large-scale structure of Universe, theory – galaxies: statistics

1 INTRODUCTION
In the modern epoch, we have access to a large variety of cosmological datasets, both real and simulated, and through their analysis one can probe fundamental particle physics, investigate the nature of gravitation, and expose the composition of the Universe. To robustly extract such information, we require summary statistics that can be easily computed, modeled and interpreted. Considering the late Universe, the most useful tool for the analysis of surveys has historically been the isotropic two-point correlation function (2PCF) and its Fourier counterpart, the power spectrum $P(k)$. Both of these describe clustering as a function of scale, and, if the underlying density field is Gaussian, encapsulate all available cosmological information. Whilst recent galaxy survey analyses have focused on consideration of the prominent Baryon Acoustic Oscillation peak (e.g., Eisenstein et al. 2005; Anderson et al. 2014; Gil-Marín et al. 2016; Beutler et al. 2017), additional information is encoded in its full shape (recently demonstrated for the galaxy power spectra in D’Amico et al. 2019, Ivanov et al. 2019a,b and Philcox et al. 2020a). Power spectra are not simply limited to galaxy surveys, however; additional usages can be found for example in weak lensing analyses (e.g., Miyatake et al. 2015; More et al. 2015; Hikage et al. 2019; Hamana et al. 2020).

At low redshifts, the assumption of Gaussianity fails in the real Universe, and we require statistics beyond the power spectrum, most notably the isotropic bispectrum (or the equivalent three-point correlation function; 3PCF), encoding non-linear clustering effects, and anisotropic power spectra, describing the deviation from isotropy due to redshift-space distortions (RSD; Kaiser 1987) and the Alcock-Paczynski effect (Alcock & Paczynski 1979). A wealth of literature exists on each quantity, describing algorithms for their measurement, theoretical descriptions and observational measurements (e.g., Feldman et al. and references therein).
In order to measure the power spectra of discrete objects such as galaxies and simulation particles, cosmologists have long relied on methods based on Fast Fourier Transforms (FFTs), first assigning the particles to a grid of side-length \( N \), then transforming to Fourier space in an operation whose complexity scales as \( N \log N \). This scaling is particularly favorable on large scales, and naturally translates into other observables such as correlation functions (e.g., Slepian & Eisenstein 2016). On small scales, these computations become expensive since a large grid must be used to avoid the effects of aliasing, with a runtime scaling as \( k_{\text{max}} \log k_{\text{max}} \). In many cases, in particular machine learning applications, we are interested in computing statistics across a broad range of scales, encompassing both the largest structures of the Universe as well as distributions of matter inside clusters. To this end, it is important to find a more efficient manner of computing small-scale spectra.

Here, we address this problem by presenting a different set of spectral estimators, first suggested in Jing & Börner (2001) and Li et al. (2016) and rigorously developed in Philcox & Eisenstein (2020, hereafter PE20). These stem from the idea that, since the power spectrum and bispectrum are fundamentally just the Fourier transforms of correlation functions, and correlation functions can be computed by simply counting groups of particles in space, we can compute spectra directly by counting groups of particles with Fourier weights \( e^{i k \cdot r} \). Although this strictly requires counting all possible groups of particles in the survey or simulation, counts can be truncated at a finite radius with negligible impact on the small-scale spectra. Power spectrum (bispectrum) computation is thus reduced to counting pairs (triplets) of galaxies and random particles in space. For a fixed truncation radius, this does not scale with the \( k \)-scale considered and naturally avoids affects such as aliasing and shot-noise.

Whilst PE20 focused on the application of this method to the power spectrum of non-uniform survey data, in this paper we specialize to cosmological simulations, since these contain readily available information across a range of scales, without the added complexities afforded by observational data (e.g., fiber collisions and blending). Given that simulations (both N-body and hydrodynamical) are usually made with periodic boundary conditions, a number of simplifications are possible; in particular this obviates the need for a random particle catalog, since the random particle integrals can be performed analytically (see Pearson & Samushia 2019 for an analogous calculation for the 3PCF). This dramatically increases the efficiency. A large swathe of this paper is also dedicated to computation of the bispectrum; using spherical harmonic decompositions, this can also be formulated as a pair count and evaluated in comparable time to the power spectrum. In this instance, our work is similar to the work of Slepian & Eisenstein (2015) and Slepian et al. (2017) which considered the corresponding configuration-space 3PCF.

This paper is structured as follows. We begin in Sec. 2 with a high-level overview of configuration-space algorithms and the main ideas of this work, before deriving the power spectrum and bispectrum estimators in full in Secs. 3 & 4. Sec. 5 contains a brief comment on window-function convolutions inherent to our method, before we derive the auto- and cross-covariances of our estimators in detail in Secs. 6, 7 & 8. In Sec. 9, we discuss our implementation of the algorithms into the public HIPSTER code, and give a number of examples of its usage, including measuring the power spectrum and bispectrum on a broad range of scales. We conclude with a summary in Sec. 10, with supplementary mathematical material presented in appendices A to C. For the reader whose prime interest lies in understanding the algorithm rather than diving into detailed mathematics, we recommend skipping all but Secs. 2 & 9.

1 Sefusatti et al. (2016) provides an interesting method to ameliorate this, making use of interlaced FFT grids.

2 HIPSTER.readthedocs.io
2 OVERVIEW OF CONFIGURATION-SPACE SPECTRAL ESTIMATORS

We begin with a discussion of configuration-space estimators in cosmological simulations, acting both as a summary of PE20 and a broad overview of this work. Initially, we will consider the anisotropic power spectrum $P(k)$, defined as the Fourier transform of the two-point correlation function (2PCF) $\xi(r)$. Using the Landy & Szalay (1993) estimator, this can be written in terms of data-data (DD), data-random (DR) and random-random (RR) counts:

$$P(k) = \int \frac{dr}{4\pi} e^{-i k \cdot r} \xi(r) = \int \frac{dr}{4\pi} e^{-i k \cdot r} \frac{DD(r) - 2DR(r) + RR(r)}{RR(r)} .$$

where ‘data’ and ‘randoms’ refer to the cosmological particles (which can be simulation particles, halos or galaxies) and a group of randomly placed points. For a uniform simulation of volume $V$ with periodic boundary conditions and particle density $n$, we can replace the denominator by the RR counts for an ideal survey, $n^2 V$;

$$P(k) = \frac{1}{n^2 V} \int dr \ e^{-i k \cdot r} [DD(r) - 2DR(r) + RR(r)] \equiv \frac{1}{n^2 V} \left[ \overline{DD}(k) - 2 \overline{DR}(k) + \overline{RR}(k) \right].$$

where the above relation defines the functions $\overline{DD}$, $\overline{DR}$, and $\overline{RR}$. Inserting the definition of the pair-counts as an integral of the number density fields over the full simulation volume, these are defined as

$$\overline{XY}(k) = \int dr \ e^{-i k \cdot r} XY(r) = \int dr \ e^{-i k \cdot r} \int dx_1 dx_2 \ n_X(x_1)n_Y(x_2)D_D(r - (x_1 - x_2))$$

\[= \int dx_1 dx_2 n_X(x_1)n_Y(x_2)e^{-i k \cdot (x_1 - x_2)} ,\]

(2.3)

where each of $X$ and $Y$ is either $D$ or $R$, where the second line follows by integrating over $k$ and applying the Dirac delta $\delta_D$. For a discrete set of tracer particles (e.g., galaxies or simulation particles) the number densities can be written as a sum over Dirac deltas, giving

$$\overline{XY}(k) = \sum_{i \in X} \sum_{j \in Y, i \neq j} \ e^{-i k \cdot (x_i - x_j)} ,$$

(2.4)

where the indices run over all particles $i, j$ in fields $X, Y$, at $x_i, x_j$. For $X = Y$ we exclude self-counts (defined as pairs of particles for which $x_i = x_j$), since these contribute only to shot-noise and are not expected to have cosmological relevance. The computation of the power spectrum thus reduces to a summation over pairs of points with a specific weighting function. Note however that this requires a count over all pairs in the data-set for any given $k$-bin, unlike in 2PCF analyses where we count only up to the maximum radial bin. To ameliorate this, we truncate the pair counts at some radius $R_0$, with a smooth window function as in PE20;

$$W(r; R_0) = \begin{cases} 1 & \text{if } 0 \leq x < 1/2 \\ 1 - 8(2x - 1)^3 + 8(2x - 1)^2 & \text{if } 1/2 \leq x < 3/4 \\ -64(x - 1)^3 - 128(x - 1)^2 & \text{if } 3/4 \leq x < 1 \\ 0 & \text{else} \end{cases}$$

(2.5)

with $x = r/R_0$. This is introduced for computational tractability and has negligible effect on the measured spectra for $kR_0 \gg 1$. This explains why our estimator is optimized for small scales. In addition, we decompose the function into Legendre multipoles about the (local) line-of-sight which, following a somewhat involved computation, yields

$$\tilde{P}_\ell(k; R_0) = \frac{1}{n^2 V} \left[ \overline{DD}_\ell(k; R_0) - 2 \overline{DR}_\ell(k; R_0) + \overline{RR}_\ell(k; R_0) \right],$$

(2.6)

3 In this paper, we define the forward and inverse Fourier transforms as

$$\hat{X}(k) = \int dx \ e^{-i k \cdot x} X(x) , \quad X(x) \equiv \int \frac{dk}{(2\pi)^3} e^{i k \cdot x} \hat{X}(k) .$$

and the Dirac function $\delta_D$ via

$$\int dx \ e^{i (k_1 - k_2) \cdot x} \equiv (2\pi)^3 \delta_D(k_1 - k_2) .$$

The correlation function and power spectrum of the density field are defined as

$$\xi(r) = \langle \delta(x)\delta(x + r) \rangle , \quad (2\pi)^3 \delta_D(k + k')P(k) = \langle \tilde{\delta}(k)\tilde{\delta}(k') \rangle ,$$

with the power spectrum as the Fourier transform of the correlation function and higher order correlators being defined similarly.

4 Unlike PE20, we make the simplification that the unclustered number densities are uniform and the particles are unweighted; this is a valid assumption for a cosmological simulation. This also leads to a ‘survey correction function’ $\Phi(r)$ of unity everywhere.

5 Note that particle weights could be easily added to this formalism by introducing an additional $w_i w_j$ factor in Eq. 2.4 This may be useful when considering the power spectrum of multiple species of particle for example.
where the modified pair counts are given in discrete and continuous form by

\[
\begin{align*}
\overline{XY}_\ell(k; R_0) &= (-i)^\ell (2\ell + 1) \int dx_1 dx_2 n_X(x_1) n_Y(x_2) j_\ell(k|x_1 - x_2|) L_\ell(\hat{x}_{12} \cdot \hat{n}_{12}) W(x_1 - x_2; R_0) \\
&= (-i)^\ell (2\ell + 1) \sum_{i \in X} \sum_{j \in Y, i \neq j} j_\ell(k|x_i - x_j|) L_\ell(\hat{x}_{ij} \cdot \hat{n}_{ij}) W(x_i - x_j; R_0)
\end{align*}
\tag{2.7}
\]

(PE20), where \(x_{ab} \equiv x_a - x_b\) and \(n_i j\) points along the line-of-sight direction (fixed for a simulation, or equal to \((x_i + x_j)/2\) for survey data). This uses spherical Bessel functions \(j_\ell\) and Legendre multipoles \(L_\ell\). We may thus compute the power spectrum from a set of pair counts with the above weighting functions. In actual analyses it is beneficial to bin in \(|k|\)-space; we defer this complexity to later sections. It is a key point of this paper that the counts involving randoms, \(DR\) and \(RR\), can be performed analytically. This reduces the power spectrum estimator to a simple pair-count over all tracer particles up to a maximum radius \(R_0\).

A similar line of reasoning applies for the bispectrum, \(B\). Here we consider only the isotropic bispectrum, which is integrated over all orientations of the \(k\)-space triangle with respect to the line-of-sight, and hence does not carry RSD information. Similar conclusions apply however for the anisotropic function. Writing this in terms of its Fourier transform, the three point correlation function (3PCF) \(\zeta\), and inserting the Szapudi & Szalay (1998) estimator gives

\[
B(k_1, k_2) = \int \frac{d^3 r_1 d^3 r_2 e^{-i k_1 \cdot r_1} e^{-i k_2 \cdot r_2}}{r_1 r_2} \zeta(r_1, r_2)
= \int \frac{d^3 r_1 d^3 r_2 e^{-i k_1 \cdot r_1} e^{-i k_2 \cdot r_2}}{r_1 r_2} \frac{DDDD(r_1, r_2) - 3 \overline{DR}(r_1, r_2) + 3 \overline{RR}(r_1, r_2) - \overline{ RR}(r_1, r_2)}{\overline{RR}(r_1, r_2)}.
\tag{2.8}
\]

As before, we rewrite the denominator in terms of the idealized random triple counts, \(6\overline{r}^3 V\), to give

\[
B(k_1, k_2) = \int \frac{d^3 r_1 d^3 r_2 e^{-i k_1 \cdot r_1} e^{-i k_2 \cdot r_2}}{r_1 r_2} \zeta(r_1, r_2)
= \frac{1}{6\overline{r}^3 V} \int \frac{d^3 r_1 d^3 r_2 e^{-i k_1 \cdot r_1} e^{-i k_2 \cdot r_2}}{r_1 r_2} \left[ DDDD(r_1, r_2) - 3 \overline{DR}(r_1, r_2) + 3 \overline{RR}(r_1, r_2) - \overline{ RR}(r_1, r_2) \right]
= \frac{1}{6\overline{r}^3 V} \left[ DDDD(k_1, k_2) - 3 \overline{DR}(k_1, k_2) + 3 \overline{RR}(k_1, k_2) - \overline{ RR}(k_1, k_2) \right],
\tag{2.9}
\]

where the modified triple counts are defined in continuous and discrete form as

\[
\overline{XYZ}(k_1, k_2) = \int \frac{d^3 x_1 d^3 x_2 d^3 x_3}{n_X(x_1) n_Y(x_2) n_Z(x_3)} e^{-i k_1 \cdot (x_1 - x)} e^{-i k_2 \cdot (x_2 - x)} e^{-i k_3 \cdot (x_3 - x)} + 5 \text{ perms.}
= \sum_{i \in X} \sum_{j \in Y} \sum_{k \in Z} e^{-i k_1 \cdot (x_i - x_j)} e^{-i k_2 \cdot (x_j - x_k)} e^{-i k_3 \cdot (x_k - x)} + 5 \text{ perms.},
\tag{2.10}
\]

where ‘perms.’ refers to permutations of the set \(\{x_1, x_2, x_3\}\). The asterisks indicate that we exclude self-counts for identical fields (hence avoiding shot-noise contributions). Analogously to the power spectrum, we introduce pair-separation window functions, \(W(r; R_0)\) that will allow us to count particles only up to radius \(R_0\). Using this (and denoting the angular part of \(d\mathbf{k}\) by \(d\Omega_k\), the Legendre moments of \(B\) are defined by

\[
B_{\ell}(k_1, k_2) = \frac{(2\ell + 1)}{4\pi} \int \frac{d\Omega_{k_1}}{4\pi} \frac{d\Omega_{k_2}}{4\pi} B(k_1, k_2) L_\ell(\hat{k}_1 \cdot \hat{k}_2)
\Rightarrow B_{\ell}(k_1, k_2; R_0) = \frac{(2\ell + 1)}{6\overline{r}^3 V} \left[ DDDD(k_1, k_2; R_0) - 3 \overline{DR}_\ell(k_1, k_2; R_0) + 3 \overline{RR}_\ell(k_1, k_2; R_0) - \overline{ RR}_\ell(k_1, k_2; R_0) \right].
\tag{2.11}
\]

Following some of the kernels may be written as

\[
\overline{XYZ}_{\ell}(k_1, k_2; R_0) = (-i)^\ell (2\ell + 1) \int \frac{d^3 x_1 d^3 x_2 d^3 x_3}{n_X(x_1) n_Y(x_2) n_Z(x_3)} j_\ell(k_1|x_1|) j_\ell(k_2|x_3|) L_\ell(\hat{x}_{12} \cdot \hat{x}_{13}) W(x_1; R_0) W(x_3; R_0) + 5 \text{ perms.}
= (-i)^\ell (2\ell + 1) \sum_{i \in X} \sum_{j \in Y} \sum_{k \in Z} \left[ j_\ell(k_1|x_i|) j_\ell(k_2|x_k|) L_\ell(\hat{x}_{ij} \cdot \hat{x}_{ik}) W(x_j; R_0) W(x_k; R_0) + 5 \text{ perms.} \right],
\tag{2.12}
\]

where \(x_{ab} \equiv x_a - x_b\) as before (PE20). Whilst a naive implementation of this estimator (and the one suggested in PE20) is a count over triples of particles, scaling as \(O(N^3)\) for \(N\) particles, it is in fact possible to reduce this to a pair count, making use of spherical harmonic theorems and carefully considering self-count terms. Schematically, we obtain

\[
\overline{XYZ}_{\ell}(k_1, k_2; R_0) = (-i)^\ell (2\ell + 1) \sum_{i \in X} \left[ \frac{4\pi}{2\ell + 1} \sum_{m=-\ell}^\ell \left( A_{\ell m}(x_i; k_1, R_0) A_{\ell m}^*(x_i; k_2, R_0) - C_{\ell}(x_i; k_1, k_2, R_0) \right) \right],
\tag{2.13}
\]

where the functions \(A\) and \(C\) can be written as a sum over all points separated by less than \(R_0\) from particle \(i\), and \(C\) is included

---

6 Note that constraining two triangle sides to be shorter than \(R_0\) constrains the third to be shorter than \(2R_0\) via the triangle inequality. This does not break symmetry however, since we sum over permutations of the three sides.

7 Note that this differs from PE20 by a factor of \((2\ell + 1)\), correcting an earlier oversight.
to capture the $j = k$ term if $Y = Z$. We further note that any term involving a random field can be computed analytically, without use of a large random catalog, though in practice, it is faster to compute one of the terms using such a catalog. We refer the reader to Sec. 4 for a detailed discussion of such effects. For the reader who is less interested in detailed mathematics, the above should be sufficient proof that such estimators exist, and we encourage them to skip directly to the applications to data in Sec. 9.

3 THE POWER SPECTRUM IN THE PERIODIC LIMIT

For a cosmological simulation, we may substantially simplify the expressions for $P_\ell(k)$ (Eq. 2.6) by removing all dependence on the random particle catalogs. Firstly, since the number density of particles in the simulation is known precisely, we may use the alternative 2PCF estimator $\xi(r) = D(r)/R_{\text{RR}}(r) − 1$ rather than the Landy & Szalay (1993) form. (It can be shown that $DR(r) ≡ R_{\text{RR}}(r)$ by translational invariance.) We thus obtain the simpler power spectrum form

$$P_\ell(k; R_0) = \frac{1}{n^2 V} \left[ \overline{DD}_\ell(k; R_0) - \overline{RR}_\ell(k; R_0) \right].$$

(3.1)

Next, we note that the modified $RR$ count (from Eq. 2.7) becomes analytic in the limit of infinite randoms. To see this, we first write the $\overline{RR}_\ell$ function in continuous form using line-of-sight (LoS) vector $\hat{n}$:

$$\overline{RR}_\ell(k; R_0) \equiv (-i)^\ell \int d\mathbf{x}_1 d\mathbf{x}_2 n_\ell(\mathbf{x}_1) n_\ell(\mathbf{x}_2) L_\ell(\mathbf{x}_1 - \mathbf{x}_2) |\mathbf{v}(\mathbf{x}_1)| W(\mathbf{x}_1; R_0).$$

(3.2)

Transforming variables to $r = \mathbf{x}_1 - \mathbf{x}_2$ and integrating over $\mathbf{x}_2$ gives

$$\overline{RR}_\ell(k; R_0) = n^2 V (-i)^\ell \int dr L_\ell(\hat{r} \cdot \hat{n}) W(r; R_0)$$

$$= 4\pi n^2 V a^K_0 \int r^2 dr \, j_\ell(kr) W(r; R_0) = n^2 V \delta^K_0 \overline{W}(k R_0),$$

where we have used that the angular integral of $L_\ell$ is simply $4\pi \delta^K_0$ for Kronecker delta $\delta^K$, and that $W(r; R_0)$ is isotropic with spherical Fourier transform $\overline{W}(k R_0)$. (Note that, for the polynomial window function used in PE20, $W(k R_0)$ can be expressed analytically in terms of incomplete gamma functions.) The full power spectrum estimator thus becomes

$$\hat{P}_\ell(k; R_0) = \frac{1}{n^2 V} (-i)^\ell (2\ell + 1) \sum_{ij} W(\mathbf{x}_i - \mathbf{x}_j; R_0) j_\ell(k |\mathbf{x}_i - \mathbf{x}_j|) L_\ell(\hat{\mathbf{x}}_{ij} \cdot \hat{n}) - \delta^K_0 \overline{W}^2(k R_0),$$

(3.4)

where $(i, j)$ run over all galaxies and we exclude self-counts to avoid shot-noise. Note that this does not require a random catalog to compute.

In practice, it is usually preferable to bin the power in $k$-space. For a $k$-bin $a$ with volume $v_a$, we obtain

$$\overline{P}_\ell^a(R_0) = \frac{4\pi}{v_a} \int k^2 dk \, \Theta^a(k) \hat{P}_\ell(k; R_0),$$

(3.5)

where $\Theta^a(k)$ is a binning function that is unity if $k$ is in $a$ and zero else. Practically, $k$-space binning simply modifies the pair-count kernels and the analytic $RR$ term; computation proceeds identically using the bin-integrated forms. Including this, our estimator becomes

$$\overline{P}_\ell^a(R_0) = \frac{1}{n^2 V} (-i)^\ell (2\ell + 1) \sum_{ij} W(\mathbf{x}_i - \mathbf{x}_j; R_0) j_\ell^a(\mathbf{x}_i - \mathbf{x}_j) L_\ell(\mathbf{w}_{ij} \cdot \hat{n}) - \delta^K_0 \overline{W}^2(k R_0),$$

(3.6)

where the superscript $a$ indicates an average over the $k$-space bin $a$ (which has volume $v_a$). The $j_\ell^a$ functions are in fact analytic, with

$$j_\ell^a(\mathbf{u}) = \frac{4\pi}{v_a} \int k^2 dk \, \Theta^a(k) j_\ell(k \mathbf{u})$$

$$= \frac{3}{x^3} \left[ D_\ell(k x_{a,\text{max}}) - D_\ell(k x_{a,\text{min}}) \right],$$

(3.7)

where $D_\ell(u) = \int u^2 du j_\ell(u)$ can be evaluated via a recursion relation, as shown in Appendix C. Using this notation, the binned pair-separation windows are thus

$$\overline{W}^a(R_0) = \frac{4\pi}{v_a} \int k^2 dk \, \Theta^a(k) \overline{W}(k R_0) = 4\pi \int_0^{R_0} r^2 dr \, j_\ell^a(r) W(r; R_0).$$

(3.8)

4 THE BISPECTRUM IN THE PERIODIC LIMIT

A less trivial extension of the above is to the bispectrum. We begin by rewriting the windowed bispectrum of Eq. 2.11;

$$B_\ell(k_1, k_2; R_0) = \frac{1}{6 V n^2} \left[ \overline{D D D}_\ell(k_1, k_2; R_0) - 3 \overline{D D R}_\ell(k_1, k_2; R_0) + 3 \overline{D R R}_\ell(k_1, k_2; R_0) - \overline{R R R}_\ell(k_1, k_2; R_0) \right].$$

(4.1)
As for the power spectrum, this expression can be simplified by noting that, due to translation invariance, the triple counts \( DRR \) are equivalent to \( RRR \). To see this rigorously, consider the triple counts in spatial bins \( a, b \) with associated volumes \( \delta x_a, \delta x_b \)

\[
RRR_{ab} = \int d\mathbf{x}_1 d\mathbf{x}_2 d\mathbf{x}_3 n_r(x_1)n_r(x_2)n_r(x_3) \left[ \Theta^a(x_1 - x_2)\Theta^b(x_2 - x_3) + 5 \text{ perms.} \right] \tag{4.2}
\]

\[
= 6n^3 \int d\mathbf{x}_1 d\mathbf{x}_2 d\mathbf{x}_3 \Theta^a(x_1 - x_2)\Theta^b(x_2 - x_3) = 6n^3 V\delta x_a\delta x_b \]

\[
DRR_{ab} = \int d\mathbf{x}_1 d\mathbf{x}_2 d\mathbf{x}_3 n_r(x_1)n_r(x_2)n_r(x_3) \left[ \Theta^a(x_1 - x_2)\Theta^b(x_2 - x_3) + 5 \text{ perms.} \right] \tag{4.3}
\]

\[
= 2n^2 \int d\mathbf{x}_1 d\mathbf{x}_2 d\mathbf{x}_3 n_r(x_3) \left[ 2\Theta^a(x_1 - x_2)\Theta^b(x_2 - x_3) + \Theta^a(x_1 - x_3)\Theta^b(x_1 - x_3) \right] \]

\[
= 2n^2 \int d\mathbf{x}_3 n_r(x_3) [\delta x_a\delta x_b + 2\delta x_a\delta x_b] = 6n^3 V\delta x_a\delta x_b \equiv RRR_{ab}.
\]

using the symmetries of the expressions under permutations of \( (a, b) \) and \( (x_1, x_2, x_3) \) and noting that \( \int d\mathbf{r} \Theta^a(r - s) = \delta r_a \), and \( \int d\mathbf{r} n_r(r) = nV \). This naturally applies also to the modified pair counts with the \( k \)-space kernels, allowing us to combine the \( DRR \) and \( RRR \) terms in the bispectrum estimator. Note that it is not correct to write the 3PCF estimator as \( \langle (r_1, r_2) = DDD(r_1, r_2)/RRR(r_1, r_2) \rangle - 1 \) (as one might expect from the simplified 2PCF estimator); this will contain additional contributions from the 2PCF.

4.1 RRR Term

We proceed by evaluating the modified \( RRR \) count analytically, in the limit of infinite randoms;

\[
RRR_t(k_1, k_2; R_0) = 6(-1)^{(2\ell + 1)} \int d\mathbf{x}_1 d\mathbf{x}_2 d\mathbf{x}_3 n_r(x_1)n_r(x_2)n_r(x_3) f_t(k_1, k_2, k_3) W(|x_1|; R_0)W(|x_2|; R_0)W(|x_3|; R_0) L_\ell(\mathbf{x}_1 \cdot \mathbf{x}_2 \cdot \mathbf{x}_3) \]

\[
= 6n^3 \int d\mathbf{r}_1 d\mathbf{r}_2 f_t(k_1, k_2) W(|r_1|; R_0) W(|r_2|; R_0) L_\ell(\mathbf{r}_1 \cdot \mathbf{r}_2) \]

\[
= 6n^3 \int d\mathbf{r}_1 d\mathbf{r}_2 f_t(k_1, k_2) W(|r_1|; R_0) \left[ 4\pi \int r_2^2 d\mathbf{r}_j f_t(k_j) W(|r_j|; R_0) \right] \delta_0^K \]

\[
= 6n^3 \int d\mathbf{r}_1 d\mathbf{r}_2 f_t(k_1, k_2) W(|r_1|; R_0) \delta_0^K \]

where we have inserted the \( k \)-space kernel with \( x_{ij} = \mathbf{x}_j - \mathbf{x}_i \) and noted that all six permutations of \( \{x_1, x_2, x_3\} \) give the same result. In the second line we have transformed variables and integrated over \( x_1 \) (which is trivial due to the periodicity), and in the third, we note that \( \int d\mathbf{r}_1 d\mathbf{r}_2 f_t(\mathbf{r} \cdot \mathbf{r}) = 4\pi \delta_0^K \). Given this simplification, the bispectrum may be written in somewhat friendlier form;

\[
B_r(k_1, k_2; R_0) = \frac{1}{6n^3} \left[ \tilde{D} \tilde{D} R_t(k_1, k_2; R_0) - 3 \tilde{D} \tilde{D} R_t(k_1, k_2; R_0) + 2 \delta_0^K \tilde{W}(k_1 R_0) \tilde{W}(k_2 R_0) \right]. \tag{4.4}
\]

4.2 DDD Term

The \( DDD \) triple counts are formally given as the weighted triple count over the survey;

\[
\tilde{D} \tilde{D} D_t(k_1, k_2; R_0) = (-1)^{(2\ell + 1)} \sum_{i \neq j \neq k} \left[ f_t(k_i, x_j, x_k) W(|x_j|; R_0) W(|x_k|; R_0) L_\ell(\hat{x}_j \cdot \hat{x}_k) + 5 \text{ perms.} \right] \tag{4.5}
\]

\[
= 6 \sum_{i \neq j \neq k} A_{ij} (x_i, x_j, x_k; k_1, k_2; R_0)
\]

(as in Eq. 2.10), defining the kernel \( A_{ij} \), where the sum runs over all distinct triples of galaxies at \( (x_i, x_j, x_k) \) and the six permutations are identical by symmetry. A naive estimate of this term would involve counting all weighted triplets of particles with \( i - j \) and \( i - k \) separations up to \( R_0 \). However, in this case, we may do considerably better and reduce this to a simple pair count, following an analogous computation to that performed for the 3PCF in Slepian & Eisenstein (2015). We first isolate the \( j = k \) term of the summation;

\[
\tilde{D} \tilde{D} D_t(k_1, k_2; R_0) = 6(-1)^{(2\ell + 1)} \sum_{i \neq j} \left[ f_t(k_i, x_j) W(|x_j|; R_0) W(|x_k|; R_0) L_\ell(\hat{x}_j \cdot \hat{x}_k) \right] \tag{4.6}
\]

\[
-6(-1)^{(2\ell + 1)} \sum_{i \neq j} \left[ f_t(k_i, x_j) W^2(|x_j|; R_0) \right] \]

}\[\]
where the second line follows as $L_\ell (\hat{x}_j \cdot \hat{x}_j) \equiv 1$. The first term may be rewritten by expanding $L_\ell$ in terms of spherical harmonics as

$$L_\ell (\hat{a} \cdot \hat{b}) = \frac{4\pi}{2\ell + 1} \sum_{m=-\ell}^{\ell} Y_{\ell m}(\hat{a})Y_{\ell m}^*(\hat{b})$$  \hspace{1cm} (4.7)$$

(NIST DLMF, Eq. 14.30.9) and separating out the $j$ and $k$ summations;

$$\overline{DDD}_\ell(k_1, k_2; R_0) = 6(-1)^\ell (2\ell + 1) \frac{4\pi}{2\ell + 1} \sum_{m=-\ell}^{\ell} \sum_{i \neq j} \left[ j_r(k_1|x_{ij}|W(|x_{ij}|; R_0)Y_{\ell m}(\hat{x}_{ij})) \right. \sum_{k \neq i} \left[ j_r(k_2|x_{ik}|W(|x_{ik}|; R_0)Y_{\ell m}^*(\hat{x}_{ik})) \right]$$  \hspace{1cm} (4.8)$$

$$= 6(-1)^\ell (2\ell + 1) \sum_{i \neq j} \left[ \frac{4\pi}{2\ell + 1} \sum_{m=-\ell}^{\ell} A_{\ell m}(x_i; k_1, R_0)A_{\ell m}^*(x_j; k_2, R_0) - C_\ell(x_i; k_1, k_2; R_0) \right].$$

defining

$$A_{\ell m}(x_i; k, R_0) = \sum_{j \neq i} j_r(k|x_{ij}|W(|x_{ij}|; R_0)W_{\ell m}(\hat{x}_{ij})$$  \hspace{1cm} (4.9)$$

$$C_\ell(x_i; k_1, k_2, R_0) = \sum_{j \neq i} j_r(k_1|x_{ij}|)j_r(k_2|x_{ij}|)W^2(|x_{ij}|; R_0).$$

To estimate the $\overline{DDD}$ term, we thus need to compute (and sum) the $A_{\ell m}$ and $C_\ell$ functions at the location of each particle in the simulation. Since both are simply sums over one set of points, we are essentially only counting pairs of particles rather than triples; this section of the algorithm thus has complexity $O(NnR_0^6)$ rather than $O(Nn^2R_0^6)$ (as would be expected for a triple count). This affords a significant speed boost.

### 4.3 DDR Term

For the $DDR$ counts, simplification is possible by analytically performing the integral over the random field, though this is non-trivial. Firstly, we rewrite the term as

$$\overline{DDR}_\ell(k_1, k_2; R_0) = \int d\mathbf{x}_1 d\mathbf{x}_2 d\mathbf{x}_3 n_g(x_1)n_g(x_2)n_g(x_3) [A_{\ell}(x_{12}, x_{13}; k_1, k_2; R_0) + 5 \text{ perms.}]$$  \hspace{1cm} (4.10)$$

$$= 2 \int d\mathbf{x}_1 d\mathbf{x}_2 d\mathbf{x}_3 n_g(x_1)n_g(x_2)n_g(x_3) [2A_{\ell}(x_{12}, x_{13}; k_1, k_2; R_0) + A_{\ell}(x_{13}, x_{23}; k_1, k_2; R_0)]$$

$$= 4\overline{DDR}_{\ell I}(k_1, k_2; R_0) + 2\overline{DDR}_{\ell II}(k_1, k_2; R_0),$$

where we have separated out terms with different structures under permutation. Physically the two terms arise from whether the two $k$ vectors correspond to (a) one data-data side of the triangle and one data-random side or (b) two data-random sides. Both terms may be written in terms of a sum over galaxy pairs (due to the two $n_g$ fields) with an integral over the field of randoms; the first gives

$$\overline{DDR}_{\ell I}(k_1, k_2; R_0) = \sum_{i \neq j} \int d\mathbf{x}_i n_g(\mathbf{x}_i)A_{\ell}(\mathbf{x}_{ij}, \mathbf{x}_{ij}; k_1, k_2; R_0)$$  \hspace{1cm} (4.11)$$

$$= \frac{1}{2} n(-1)^\ell (2\ell + 1) \sum_{i \neq j} \int d\mathbf{x}_i j_r(k_1|x_{ij}|)j_r(k_2|x_{ij}|)W(|x_{ij}|; R_0)W(|x_{ij}|; R_0)L_\ell(\hat{x}_{ij} \cdot \hat{x}_{ij}) + (k_1 \leftrightarrow k_2)$$

$$= \frac{1}{2} n(-1)^\ell (2\ell + 1) \sum_{i \neq j} j_r(k_1|x_{ij}|)W(|x_{ij}|; R_0)E_{\ell}^I(x_i, x_j; k_2; R_0) + (k_1 \leftrightarrow k_2).$$

This is simply a count of all pairs in the survey separated by distances up to $R_0$, weighted by a kernel $E^I \ell$. This function may be rewritten as

$$E_{\ell}^I(x_i, x_j; k_2; R_0) = \int d\mathbf{x}_3 j_r(k_2|x_{ij}|)W(|x_{ij}|; R_0)L_\ell(\hat{x}_{ij} \cdot \hat{x}_{ij})$$  \hspace{1cm} (4.12)$$

$$= \int d\mathbf{r} j_r(k_2|\mathbf{r}|)W(|\mathbf{r}|; R_0)L_\ell(\hat{x}_{ij} \cdot \hat{x}_{ij})$$

$$= 4\pi \delta^K_{R_0} \int r^2 dr j_r(k_2r)W(r; R_0) = \delta^K_{R_0} \overline{W}(k_2R_0).$$

MNRAS 000, 1–32 (2020)
where we have substituted \( \mathbf{r} = \mathbf{x}_3 \equiv \mathbf{x}_i - \mathbf{x}_1 \) in the second line (possible by periodicity) and integrated over angle in the third, noting that this is simply the spherical Fourier transform of \( \mathcal{W}(\mathbf{r}; R_0) \). Thus
\[
\mathcal{D}D \mathcal{R}^{I}_f(k_1, k_2; R_0) = \frac{1}{2} \pi^2 K_0^2 \mathcal{W}(k R_0) \sum_{i \neq j} \delta(k_1 | \mathbf{x}_i - \mathbf{x}_j |) \mathcal{W}(\mathbf{x}_i; R_0) + (k_1 \leftrightarrow k_2).
\]

(4.13)

Similar treatment is possible for \( \mathcal{D}D \mathcal{R}^{II}_f \), giving
\[
\mathcal{D}D \mathcal{R}^{II}_f(k_1, k_2; R_0) = \sum_{i \neq j} \int d\mathbf{x}_3 n_r(\mathbf{x}_3) A_f(\mathbf{x}_3; \mathbf{x}_3; k_1, k_2; R_0)
\]
\[
= n(-1)^f (2f + 1) \sum_{i \neq j} \int d\mathbf{x}_3 j_r(k_1 | \mathbf{x}_3 |) j_r(k_2 | \mathbf{x}_3 |) \mathcal{W}(\mathbf{x}_3; R_0) \mathcal{W}(\mathbf{x}_3; R_0) L_r(\mathbf{x}_3; \hat{\mathbf{x}}_3)
\]
\[
= n(-1)^f (2f + 1) \sum_{i \neq j} \mathcal{E}^{II}_f(\mathbf{x}_i; \mathbf{x}_j; k_1, k_2; R_0).
\]

(4.14)

The form of this is somewhat different to \( \mathcal{D}D \mathcal{R}^{I}_f \). In particular, we note that the sum over tracer particles \( i \) and \( j \) is not restricted to \( |\mathbf{x}_i - \mathbf{x}_j| \) by a pair-separation window \( W \), though, due to the triangle inequality and the constraints that \( |\mathbf{x}_i - \mathbf{x}_j| \leq R_0 \) and \( |\mathbf{x}_j - \mathbf{x}_i| \leq R_0 \), \( \mathcal{E}^{II}_f = 0 \) for \( |\mathbf{x}_i - \mathbf{x}_j| \geq 2R_0 \). Computation reduces to a weighted sum over all pairs up to a radius \( 2R_0 \). It is possible to substantially simplify the kernel \( \mathcal{E}^{II}_f \) into a function depending only on the separation between \( i \) and \( j \), the derivation of which is discussed in Appendices A & B. Since the pair-separation function used in this paper (Eq. 2.5) is simply an approximation, it becomes exact in the limit of infinite randoms, and we find the term to be well approximated with a bispectrum in -space.

In practice, the \( \mathcal{D}D \mathcal{R}^{II}_f \) term is extremely laborious to compute. Since the pair-count computation scales as \( R_0^3 \), we must sample eight times as many pairs for this term compared to the other bispectrum contributions. Furthermore, the \( \mathcal{E}^{II}_f \) function needs to be pre-computed for each of the \( m_{\text{bins}}(m_{\text{bins}} + 1)/2 \) combination of \( m_{\text{bins}} \) radial and \( n_r \) Legendre bins and interpolated, which significantly degrades the computational efficiency, especially when the number of radial bins is large. For this reason, we will adopt a different method to compute \( \mathcal{D}D \mathcal{R}^{II}_f \), making use of a random catalog.

First, we rewrite the term in integral form, separating out the \( i = j \) term and using the expansion formula for Legendre polynomials, as for the \( \mathcal{D}D \mathcal{I} \) counts:
\[
\mathcal{D}D \mathcal{R}^{II}_f(k_1, k_2; R_0) = (-1)^f (2f + 1) \sum_{i \neq j} \int d\mathbf{x}_3 n_r(\mathbf{x}_3) j_r(k_1 | \mathbf{x}_3 |) j_r(k_2 | \mathbf{x}_3 |) \mathcal{W}(\mathbf{x}_3; R_0) \mathcal{W}(\mathbf{x}_3; R_0) L_r(\mathbf{x}_3; \hat{\mathbf{x}}_3)
\]
\[
= 4\pi (-1)^f \sum_{i \neq j} \int d\mathbf{x}_3 n_r(\mathbf{x}_3) \left[ \sum_i j_r(k_1 | \mathbf{x}_3 |) \mathcal{W}(\mathbf{x}_3; R_0) Y_{\ell m}(\mathbf{x}_3) \right] \left[ \sum_j j_r(k_2 | \mathbf{x}_3 |) \mathcal{W}(\mathbf{x}_3; R_0) Y_{\ell m}^*(\mathbf{x}_3) \right] - (-1)^f (2f + 1) \sum_{i \neq j} \int d\mathbf{x}_3 n_r(\mathbf{x}_3) \left[ \sum_i j_r(k_1 | \mathbf{x}_3 |) j_r(k_2 | \mathbf{x}_3 |) \right] \mathcal{W}^2(\mathbf{x}_3; R_0).
\]

(4.15)

We now assume that we have a random catalog with \( N_{\text{rand}} \) particles positioned at \( \{ \mathbf{r}_i \} \) (which are drawn from a uniform distribution encompassing the simulation volume). It is generally advisable to use more randoms than tracer particles, thus we set \( N_{\text{rand}} = f_{\text{rand}} N_{\text{data}} \) for \( N_{\text{data}} \) tracer particles and \( f_{\text{rand}} \gg 1 \). With this simplification, the integral may be written as a summation:
\[
\mathcal{D}D \mathcal{R}^{II}_f(k_1, k_2; R_0) \approx (-1)^f (2f + 1) \frac{1}{f_{\text{rand}}} \sum_i \left[ \frac{4\pi}{2f + 1} \sum_{m=-\ell}^{\ell} A_{\ell m}(\mathbf{x}_r; k_1, R_0) A_{\ell m}^*(\mathbf{x}_r; k_2, R_0) - C_{\ell}(\mathbf{x}_r; k_1, k_2; R_0) \right],
\]

(4.16)

where \( A_{\ell m} \) and \( C_{\ell} \) are the functions defined in Eq. 4.9 and we divide by the ratio of randoms to data-points; \( f_{\text{rand}} \). In this form the \( \mathcal{D}D \mathcal{R}^{II}_f \) term can be computed analogously to the \( \mathcal{D}D \mathcal{I} \) term, except that we compute the \( A_{\ell m} \) and \( C_{\ell} \) coefficients from the position of each random-particle, rather than each tracer-particle position. Whilst we note that this is strictly only an approximation, it becomes exact in the limit of infinite randoms, and we find the term to be well approximated with \( f_{\text{rand}} \) of order a few. Other choices of window function \( \mathcal{W}(\mathbf{x}; R_0) \) may lead to more tractable forms for the kernel \( \mathcal{E}^{II}_f \) and hence \( \mathcal{D}D \mathcal{R}^{II}_f \), especially if one deals only with unbiased estimators.

### 4.4 k-space Binning

One consideration remains; the effects of finite \( k \) bins. As in PE20 and analogous to the power spectrum (Eq. 3.5), we consider a bispectrum in \( k \)-space bins \( a, b \) as
\[
B_{\ell}^{ab}(R_0) = \frac{4\pi}{v_a v_b} \int k_1^2 dk_1 k_2^2 dk_2 \Theta^a(k_1) \Theta^b(k_2) B_{\ell}(k_1, k_2; R_0),
\]

(4.17)
where \( v_x \) is the volume of \( k\)-bin \( x \). Practically, the binning simply modifies the integration kernels used in the pair-count summations (since we add \( k\)-space binning directly). In particular \( \overline{RRR}_\ell \) and \( \overline{DDD}_\ell \) become

\[
\overline{RRR}_\ell^{ab}(R_0) = 6v_n^3 \overline{W}^{ab}_\ell(R_0) \overline{W}^{ab}_\ell(R_0) \overline{W}^{ab}_\ell(R_0)K
\]

\[
\overline{DDD}_\ell^{ab}(R_0) = 6(-1)^f(2f + 1) \sum_i \left\{ \frac{4\pi}{2f + 1} \sum_{m=-f}^{f} A^{ab}_{\ell m}(x_i; R_0) A^{ab}_{\ell m}(x_i; R_0) - C^{ab}_\ell(x_i; R_0) \right\}
\]

where the superscripts \((a, b)\) indicate the relevant \( k\)-space bin \( a, b \), following the definitions of Eqs. 3.7 & 3.8 and

\[
A^{ab}_{\ell m}(x_i; R_0) = \sum_{j \neq i} j^a_i \ell (|x_i - x_j|) W(|x_i - x_j|; R_0) Y^b_{\ell m}(x_j)
\]

\[
C^{ab}_\ell(x_i; R_0) = \sum_{j \neq i} j^b_i \ell (|x_i - x_j|) Y^a_{\ell m}(x_j) W^2(|x_i - x_j|; R_0).
\]

For \( \overline{D\overline{D}\overline{R}} \) we obtain

\[
\overline{D\overline{D}\overline{R}}_{\ell}^{I, ab}(R_0) = \frac{1}{2} \sum_i \left\{ \frac{4\pi}{2f + 1} \sum_{m=-f}^{f} A^{ab}_{\ell m}(x_i; R_0) A^{ab}_{\ell m}(x_i; R_0) - C^{ab}_\ell(x_i; R_0) \right\}
\]

\[
\overline{D\overline{D}\overline{R}}_{\ell}^{II, ab}(R_0) = (-1)^f(2f + 1) \sum_i \left\{ \frac{4\pi}{2f + 1} \sum_{m=-f}^{f} A^{ab}_{\ell m}(x_i; R_0) A^{ab}_{\ell m}(x_i; R_0) - C^{ab}_\ell(x_i; R_0) \right\}
\]

\[
-\frac{1}{2} \sum_i \left\{ \frac{4\pi}{2f + 1} \sum_{m=-f}^{f} A^{ab}_{\ell m}(x_i; R_0) A^{ab}_{\ell m}(x_i; R_0) - C^{ab}_\ell(x_i; R_0) \right\}
\]

\[
-\frac{1}{2} \sum_{i,j} \left\{ \frac{4\pi}{2f + 1} \sum_{m=-f}^{f} A^{ab}_{\ell m}(x_i; R_0) A^{ab}_{\ell m}(x_i; R_0) - C^{ab}_\ell(x_i; R_0) \right\}
\]

4.5 Summary

We conclude with a summary. The small-scale windowed bispectrum estimator may be written as a combination of pair counts over particles (or galaxies) in the simulation box as

\[
\overline{P}_\ell^{ab}(R_0) = \frac{1}{6v_n^3} \left\{ \overline{D\overline{D}\overline{R}}_{\ell}^{I, ab}(R_0) - 3\overline{D\overline{D}\overline{R}}_{\ell}^{II, ab}(R_0) + 2\delta^K_{R_0} \overline{W}^{ab}(R_0) \overline{W}^{ab}(R_0) \right\}
\]

\[
\overline{P}_\ell^{ab}(R_0) = \frac{1}{6v_n^3} \left\{ \overline{D\overline{D}\overline{R}}_{\ell}^{I, ab}(R_0) - 3\overline{D\overline{D}\overline{R}}_{\ell}^{II, ab}(R_0) + 2\delta^K_{R_0} \overline{W}^{ab}(R_0) \overline{W}^{ab}(R_0) \right\}
\]

\[
\overline{P}_\ell^{ab}(R_0) = \frac{1}{6v_n^3} \left\{ \overline{D\overline{D}\overline{R}}_{\ell}^{I, ab}(R_0) - 3\overline{D\overline{D}\overline{R}}_{\ell}^{II, ab}(R_0) + 2\delta^K_{R_0} \overline{W}^{ab}(R_0) \overline{W}^{ab}(R_0) \right\}
\]

where we sum over all pairs of particles \( i, j \) in the box with separations up to distance \( R_0 \) and denote the random particles by the index \( r \). Note that, due to the analytic random integrals and the \( \overline{D\overline{D}\overline{R}} \) simplifications, our full estimator has complexity \( \mathcal{O}(N^2 R_0^3) \), far superior to the \( \mathcal{O}(N^2 R_0^6) \) scaling claimed in PE20 (for fixed volume and \( R_0 \), this is simply \( \mathcal{O}(N^2) \)). Our power spectrum and bispectrum estimators thus require similar computational time, and we note that this is analogous to the \( \mathcal{O}(N^2) \) 3PCF estimators of Sleipnir & Eisenstein (2015).

5 RELATING WINDOWED AND TRUE SPECTRA

To provide physical interpretation of our spectra, we must understand the impact of the pair-separation window function \( W(r; R_0) \). One application of this is the comparison of data to models; although the windowed spectrum matches its unwindowed form on very small scales, we may wish to compute a window-convolved theoretical model that works on all scales. For simplicity we will work with unbinned spectra, though similar conclusions hold in the binned case.

For the power spectrum, recall the initial definition as the Fourier transform of the windowed 2PCF;

\[
P(k; R_0) = \int dr \xi(r) W(r; R_0) e^{-i k \cdot r}.
\]

Since this is simply the Fourier transform of a product of functions, it can be rewritten as a convolution of the true power \( P(k) \) with the Fourier transform of the window function \( \overline{W}(k; R_0) \), i.e.

\[
P(k; R_0) = \left[ P \ast \overline{W} \right](k; R_0) \equiv \int \frac{dp}{(2\pi)^3} P(p) \overline{W}(k - p; R_0)
\]

(PE20, Eq. 3.4). To form the multipoles of the windowed power, it is more convenient to start with Eq. 5.1, and note that, since \( W \) is an isotropic function, the multipoles can be related to the multipoles of \( \xi \) as

\[
P_\ell(k; R_0) = 4\pi(-i)^f \int r^2 dr \xi_\ell(r) W(r; R_0) j_\ell(k r)
\]
multipoles are thus given by \( \xi_\ell(r) W(r; R_0) \). To obtain a windowed theory prediction, we simply insert a theoretical 2PCF function into the above equation, which can be computed efficiently via Hankel transforms.

In a similar vein, the windowed bispectrum can be written as a convolution of the true bispectrum \( B \):

\[
B(k_1, k_2; R_0) = \int \frac{d^3 p_1 d^3 p_2}{(2\pi)^6} B(p_1, p_2) \tilde{W}(k_1 - p_1; R_0) \tilde{W}(k_2 - p_2; R_0)
\]

(PE20, Eq. 7.14), or in terms of the 3PCF multipoles, \( \zeta_\ell \):

\[
B(k_1, k_2; R_0) = \int d\mathbf{x}_1 d\mathbf{x}_2 \zeta(\mathbf{x}_1, \mathbf{x}_2) W(\mathbf{x}_1; R_0) W(\mathbf{x}_2; R_0) e^{-i k_1 \cdot \mathbf{x}_1} e^{-i k_2 \cdot \mathbf{x}_2}
\]

(PE20, Eq. 7.14), or in terms of the 3PCF multipoles, \( \zeta_\ell \):

\[
B(k_1, k_2; R_0) = \sum_{\ell_1, \ell_2} (2\ell + 1)(2L_1 + 1)(2L_2 + 1) f^{L_1+L_2} \int d\mathbf{x}_1 d\mathbf{x}_2 \zeta(\mathbf{x}_1, \mathbf{x}_2) W(\mathbf{x}_1; R_0) W(\mathbf{x}_2; R_0)
\]

\[
\times j_{\ell_1}(k_1 x_1) j_{\ell_2}(k_2 x_2) L_{\ell_1} (\hat{\mathbf{x}}_1 \cdot \hat{\mathbf{x}}_2) L_{\ell_2} (\hat{\mathbf{k}}_1 \cdot \hat{\mathbf{k}}_2) L_{\ell_1} (\hat{\mathbf{k}}_1 \cdot \hat{\mathbf{k}}_2)
\]

\[
= (4\pi)^2 \sum_{\ell} (-1)^\ell (2\ell + 1) \int x_1^2 dx_1 x_2^2 dx_2 \zeta(\mathbf{x}_1, \mathbf{x}_2) j_{\ell}(k_1 x_1) j_{\ell}(k_2 x_2) W(\mathbf{x}_1; R_0) W(\mathbf{x}_2; R_0) L_{\ell}(\hat{\mathbf{k}}_1 \cdot \hat{\mathbf{k}}_2),
\]

using the plane wave expansion (Arfken et al. 2013, Eq. 16.63), spherical harmonic addition (NIST DLMF, Eq. 14.30.9) and noting that \( \int d\mathbf{x} L_\ell(\hat{\mathbf{x}} \cdot \hat{\mathbf{y}}) L_\ell(\hat{\mathbf{z}} \cdot \hat{\mathbf{y}}) = 4\pi \delta_{\ell \ell} L_\ell(\hat{\mathbf{y}} \cdot \hat{\mathbf{y}})/(2\ell + 1) \) via orthogonality (NIST DLMF, Eq. 14.17.6). The bispectrum multipoles are thus given by

\[
B_\ell(k_1, k_2; R_0) = (2\ell + 1) \int \frac{d\Omega_{k_1} d\Omega_{k_2}}{4\pi} B(k_1, k_2; R_0) L_\ell(\hat{\mathbf{k}}_1 \cdot \hat{\mathbf{k}}_2)
\]

(6.5)

\[
= (4\pi)^2 (-1)^\ell \int x_1^2 dx_1 x_2^2 dx_2 \zeta(\mathbf{x}_1, \mathbf{x}_2) j_{\ell}(k_1 x_1) j_{\ell}(k_2 x_2) W(\mathbf{x}_1; R_0) W(\mathbf{x}_2; R_0),
\]

which are simply the Fourier space multipoles of a function with 3PCF \( \zeta(\mathbf{x}_1, \mathbf{x}_2) W(\mathbf{x}_1; R_0) W(\mathbf{x}_2; R_0) \). Following this technique we may thus compare our windowed configuration-space power spectrum and bispectrum estimates to arbitrary theoretical models.

6 POWER SPECTRUM COVARIANCES

Given the above estimators for small-scale spectra, it is instructive to consider their theoretical covariance. This is similar in form to the treatment of Sugiyama et al. (2019a), though we additionally include a treatment of Legendre multipoles and the window function. In general, covariance is produced by three effects;

(i) **Intrinsic covariance**: the covariance of the underlying density field, which contains both Gaussian and non-Gaussian terms, modified by the window function \( W(r; R_0) \). This is the covariance one would obtain for an infinite sample of tracer particles (i.e. the covariance of Eq. 5.1). It is also the covariance of the matter power spectrum, subject to a double convolution with \( W \).

(ii) **Poisson covariance**: the covariance arising from the finite number of particles sampled. We expect this to dominate for small samples of particles (low \( n \)) and at large \( k \), where the intrinsic covariance is small.

(iii) **Super-sample covariance**: the covariance induced by modes larger than the simulation volume, which modulate the background density of the box (Takada & Hu 2013). This can be modeled by considering the power spectrum ‘response’ to long wavelength modes (e.g., Li et al. 2018). Since this is not usually included in cosmological simulations (though see Li et al. 2014 for an example) and its modeling is specific to the field in question (e.g., it differs for biased and unbiased tracers), it will be ignored in this work. It is however an important source of covariance for cosmological surveys.

For simulations, each of the above terms can be derived in terms of the power spectrum in a tractable format. In real surveys however, this is often far more difficult, since the non-trivial survey geometries can have a non-negligible impact, and require careful consideration. In practice this can be done by assuming a separable covariance (Wadekar & Scoccimarro 2019) or in terms of stochastic integrals over the survey selection function (O’Connell et al. 2016; O’Connell & Eisenstein 2019; Philcox & Eisenstein 2019; Philcox et al. 2020b). We do not consider these complexities here.

We begin by discussing the covariance of the power spectrum in full, considering both intrinsic and Poissonian contributions. To see how the latter terms appear, it is instructive to begin by rewriting the power spectrum estimator (Eq. 3.1, before
Note that this is simply a continuous form of the covariance presented in PE20.

\[
\langle \hat{n}_D(x) \rangle = \langle n_D(x) \rangle = n
\]

\[
\langle \hat{n}_D(x)\hat{n}_D(y) \rangle = \langle n_D(x)n_D(y) \rangle + \langle n_D(x) \rangle \delta_D(x-y)
\]

\[
= n^2 \left[ 1 + \xi(x-y) \right] + n \delta_D(x-y),
\]

where we denote the underlying density field as \( n_D \) (without a hat), and note that the Dirac delta \( \delta_D \) arises from the discrete nature of the field. Inserting the above into Eq. 6.1 confirms that \( \langle \hat{P}(k; R_0) \rangle = P(k; R_0) \) (as in Sec. 5), with the \( x_1 \neq x_2 \) restriction removing the Poissonian shot noise term.

Given the above form, the covariance may be written as

\[
\text{cov} \left( \hat{P}(k; R_0), \hat{P}(k' ; R_0) \right) = \langle \hat{P}(k; R_0)\hat{P}(k' ; R_0) \rangle - \langle \hat{P}(k; R_0) \rangle \langle \hat{P}(k' ; R_0) \rangle
\]

\[
= \frac{1}{V^2} \int_{x_1 \neq x_2} \int_{x_3 \neq x_4} d^3x_1 d^3x_2 d^3x_3 d^3x_4 K(x_1 - x_2; k, R_0)K(x_3 - x_4; k', R_0)
\]

\[
\times \left\{ \left[ \langle \hat{n}_D(x_1)\hat{n}_D(x_2) \rangle - \langle n_D(x_1) \rangle \delta_D(x_1-x_2) \right] \left[ \langle \hat{n}_D(x_3)\hat{n}_D(x_4) \rangle - \langle n_D(x_3) \rangle \delta_D(x_3-x_4) \right] - \left[ \langle \hat{n}_D(x_1) \rangle \delta_D(x_2-x_3) \right] \left[ \langle \hat{n}_D(x_3) \rangle \delta_D(x_4-x_2) \right] \right\}.
\]

The term in curly brackets may be written

\[
\{ \ldots \} = \left( \langle \delta_D(x_1)\delta_D(x_2) \rangle - \langle \delta_D(x_1) \rangle \delta_D(x_2-x_3) \right) \langle \hat{n}_D(x_3)\hat{n}_D(x_4) \rangle
\]

\[
= \left( \langle n_D(x_1)n_D(x_2) \rangle \langle n_D(x_3)n_D(x_4) \rangle - \langle n_D(x_1)n_D(x_3) \rangle \langle n_D(x_2)n_D(x_4) \rangle \right) + \langle \delta_D(x_1-x_3) \rangle \langle n_D(x_2)n_D(x_4) \rangle
\]

\[
+ 2\langle \delta_D(x_1-x_3) \rangle \langle n_D(x_2) \rangle \langle n_D(x_4) \rangle,
\]

where we have performed Poisson averaging, subject to the \( x_1 \neq x_2, x_1 \neq x_3 \) conditions. We have additionally grouped terms symmetric under \( x_1 \leftrightarrow x_2 \) and \( x_3 \leftrightarrow x_4 \) since the \( K \) kernels are symmetric under these permutations (except for a phase inversion, which will vanish upon Legendre multipole binning). Note that we have decomposed the covariance into terms depending on four, three and two copies of the density field due to the Poissonian expansion. To proceed, we note that \( n_D(x) = n(1 + \delta(x)) \) and take statistical averages, introducing the \( n \)-point correlation functions (again imposing symmetry constraints);

\[
\{ \ldots \} = n^4 \left[ 4\zeta(x_1-x_3) + 4\zeta(x_1-x_2,x_3) + 2\zeta(x_1-x_3)\zeta(x_2-x_4) + \zeta^{(4)}(x_1,x_2,x_3,x_4) \right]
\]

\[
+ 4n^3 \delta_D(x_1-x_3) [1 + \zeta(x_1-x_2) + \zeta(x_1-x_3) + \zeta(x_2-x_3) + \zeta(x_1,x_2,x_3)]
\]

\[
+ 2n^2 \delta_D(x_1-x_3) \delta_D(x_2-x_4) [1 + \zeta(x_1-x_2)],
\]

where \( \zeta \) and \( \zeta^{(4)} \) are the 3PCF and 4PCF respectively, whose arguments must sum to zero. We have additionally used Wick’s theorem to write the connected four-point correlator as \( \langle \delta(x_1)\delta(x_2)\delta(x_3)\delta(x_4) \rangle_c = \zeta(x_1-x_2)\zeta(x_3-x_4) + 2 \) perms.

---

5 Note that this is simply a continuous form of the covariance presented in PE20.
We can now insert the above into the covariance (Eq. 6.3), giving a sum of four-, three- and two-point terms;

\[
\text{cov} \left( \hat{P}(k; R_0), \hat{P}(k'; R_0) \right) = 4C(k, k'; R_0) + 3C(k, k'; R_0) + 2C(k, k'; R_0) \tag{6.6}
\]

\[
4C(k, k'; R_0) = \frac{1}{V^2} \int dx_1 dx_2 dx_3 dx_4 K(x_1 - x_2; k, R_0)K(x_3 - x_4; k', R_0) \times \left[ 4\xi(x_1 - x_3) + 4\xi(x_2, x_3) + 2\xi(x_1 - x_3)\xi(x_2 - x_4) + \xi(x_1, x_2, x_3) \right]
\]

\[
3C(k, k'; R_0) = \frac{4}{nV^2} \int dx_1 dx_2 dx_3 K(x_1 - x_2; k, R_0)K(x_3 - x_1; k', R_0) \times \left[ 1 + \xi(x_1 - x_2) + \xi(x_1 - x_3) + \xi(x_2 - x_3) + \xi(x_1, x_2, x_3) \right]
\]

\[
2C(k, k'; R_0) = \frac{2}{n^2V^2} \int dx_1 dx_2 dx_3 K(x_1 - x_2; k, R_0)K(x_3 - x_2; k', R_0) [1 + \xi(x_1 - x_2)]
\]

where we have dropped the \( x_1 \neq x_2 \) and \( x_3 \neq x_4 \) which are no longer necessary since the fields are now continuous thanks to the Poisson averaging. Here, the separation of the two types of covariance is clear: the four-point term is independent of the number density of tracers and is thus an intrinsic covariance, whilst the three- and two-point terms scale as \( n^{-1} \) and \( n^{-2} \) respectively and are a Poisson covariance. In the limit of an infinitely sampled density field, these terms go to zero. We now proceed to evaluate the individual terms.

### 6.1 Intrinsic Covariance

For the four-point intrinsic covariance, a number of simplifications are possible. First, we note that the terms involving a single 2PCF and 3PCF are identically zero for a periodic survey. To see this for the 2PCF term, we rewrite the covariance using variables \( x = x_1 - x_2, y = x_3 - x_4 \) and \( z = x_1 - x_3 \) (invoking translation invariance);

\[
4C(k, k'; R_0) \supset \frac{4}{V^2} \int dx_2 dx_3 dx_4 dx_5 K(x; k, R_0)K(y, k', R_0)\xi(z)
\]

\[
= \frac{4}{V} \left[ \int dxK(x; k, R_0) \right] \left[ \int dyK(y; k', R_0) \right] \left[ \int dz\xi(z) \right].
\]

Since the 2PCF is defined as the over-random probability for two galaxies to be separated by \( r \), \( \int dz\xi(z) = 0 \), so this term does not contribute to the covariance. A similar line of reasoning applies to the 3PCF term (setting \( x = x_1 - x_3, y = x_1 - x_3, z = x_1 - x_4 \));

\[
4C(k, k'; R_0) \supset \frac{4}{V^2} \int dx_1 dx_2 dx_3 dx_4 dx_5 K(x; k, R_0)K(z; k', R_0)\xi(x, y)
\]

\[
= \frac{4}{V} \left[ \int dxK(x; k, R_0) \right] \left[ \int dxK(x', k'; R_0) \right] \left[ \int dy\xi(x, y) \right] = 0,
\]

since the integral of the 3PCF over one of its arguments is zero.\(^9\) We thus obtain

\[
4C(k, k'; R_0) = 4C_G(k, k'; R_0) + 4C_NG(k, k'; R_0)
\]

\[
4C_G(k, k'; R_0) = \frac{2}{V^2} \int dx_1 dx_2 dx_3 dx_4 W(x_1 - x_2; R_0)W(x_3 - x_4; R_0)e^{-ik(x_1-x_2)}e^{-ik'(x_3-x_4)}\xi(x_1 - x_3)\xi(x_2 - x_4)
\]

\[
4C_NG(k, k'; R_0) = \frac{1}{V^2} \int dx_1 dx_2 dx_3 dx_4 W(x_1 - x_2; R_0)W(x_3 - x_4; R_0)e^{-ik(x_1-x_2)}e^{-ik'(x_3-x_4)}\xi(x_1, x_2, x_3, x_4)
\]

separating out the Gaussian and non-Gaussian components and inserting the definition of the kernel \( K \).

We proceed by changing variables and integrating over one volume once (for the Gaussian part);

\[
4C_G(k, k'; R_0) = \frac{1}{V^2} \int dr dr' dx dx' \left[ \xi(x - r')\xi(x - r) + \xi(x - r)\xi(x + r) + \xi(x - r')\xi(x + r) \right] W(r; R_0)W(r'; R_0)e^{-ikr}e^{-ik'r}
\]

\[
= \frac{1}{V} \int dr dr' \left[ |\xi(r')|^2 + |\xi(r)|^2 + |\xi(r + r')|^2 \right] W(r; R_0)W(r'; R_0)e^{-ikr}e^{-ik'r}
\]

\[
4C_NG(k, k'; R_0) = \frac{1}{V^2} \int dr dr' dx dx' \xi(x, x', x + r + r') W(r; R_0)W(r'; R_0)e^{-ikr}e^{-ik'r}
\]

where we have used the convolution operator \( \ast \) on the product of two 2PCFs. Note that we have separated the two 2PCF terms previously assumed to be symmetric; this will allow us to robustly show the (previously assumed) symmetry properties for \( \xi(x) \) and \( \xi(x, y) \).
even multipoles. We now write all quantities in terms of their Fourier-space counterparts and apply the convolution theorem:

\begin{equation}
4C_G(k, k'; R_0) = \frac{1}{V} \int d^3 p \int d^3 p_2 \rho(p) \rho(p_2) \mathcal{W}(p_2; R_0) \mathcal{W}(p; R_0) \left[ \delta(p - (r + r')) + e^{i \phi(p_2 + k)} e^{i \phi(p - k)} r \right] (6.11)
\end{equation}

\[
4C_{NG}(k, k'; R_0) = \frac{1}{V} \int d^3 p d^3 p_2 d^3 p_3 d^3 p_4 2^2 (p) \mathcal{W}(p_2; R_0) \mathcal{W}(p_3; R_0) \mathcal{W}(p_4; R_0) \left[ \delta(p - (r + r + r')) + e^{i \phi(p_2 + p_3 + p_4)} e^{i \phi(p - k)} r \right]
\end{equation}

where \( \mathcal{W} \) is the Fourier transform of \( \mathcal{W} \) and \( \mathcal{W} \) is the (fixed) line-of-sight vector. Considering the first part of the Gaussian covariance, which have a similar form (e.g., Scoccimarro et al. 1999b; Bloomfield et al. 2020),

\begin{equation}
\text{cov} (P_T(k; R_0), P_T(k'; R_0)) \equiv 4C_{G}^{\ell T}(k, k'; R_0) + 4C_{NG}^{\ell T}(k, k'; R_0)
\end{equation}

\begin{equation}
= (2\ell + 1)(2\ell' + 1) \int \frac{d^3 \Omega_k}{4\pi} \int \frac{d^3 \Omega_{k'}}{4\pi} L_\ell(\hat{k} \cdot \hat{n}) L_{\ell'}(\hat{k}' \cdot \hat{n}) L_\ell(\hat{k}) L_{\ell'}(\hat{k}') L_{\ell''}(\hat{n}) \cdot L_{\ell''}(\hat{n}')(6.16)
\end{equation}

This is simplified with the relation

\begin{equation}
\int \frac{d^3 \Omega_k}{4\pi} e^{-i k \cdot r} L_\ell(\hat{k} \cdot \hat{n}) = \sum_L (2L + 1)(-i)^L J_L(k r) \int \frac{d^3 \Omega_k}{4\pi} L_L(\hat{k} \cdot \hat{r}) L_{\ell L}(\hat{k} \cdot \hat{n}) (6.15)
\end{equation}

via the plane wave expansion and Legendre polynomial orthogonality. This implies

\begin{equation}
4C_{G}^{\ell T}(k, k'; R_0) = \frac{(2\ell + 1)(2\ell' + 1)}{V} (\ell' + \ell) \int \frac{d^3 p}{(2\pi)^3} \mathcal{P}(p) \int d^3 r d^3 r' \mathcal{W}(r; R_0) \mathcal{W}(r'; R_0) e^{i \phi(p - r')} \mathcal{W}(r' \cdot \hat{k}') L_{\ell}(\hat{r}' \cdot \hat{n}') L_{\ell'}(\hat{r}' \cdot \hat{n'}) (6.16)
\end{equation}

setting \( k = k' \). Following similar logic, we can compute the second Gaussian term (with \( k' \to -k' \)) which is identical except for \( i^{\ell'} \to -i^{\ell'} \).

We may proceed by performing the angular integrals over \( r \) and \( r' \) (noting that \( \mathcal{W} \) is isotropic), which have a similar form to the above:

\begin{equation}
\int d^3 \Omega_r e^{i \phi r} L_{\ell L}(\hat{r} \cdot \hat{n}) = 4\pi i^{\ell L} j_L(\ell L) L_{\ell L}(\hat{r} \cdot \hat{n}), (6.17)
\end{equation}

leaving us with

\begin{equation}
4C_{G}^{\ell T}(k, k'; R_0) = \frac{(2\ell + 1)(2\ell' + 1)}{V} \left[ (-1)^{\ell'} + 1 \right] \int \frac{d^3 p}{(2\pi)^3} \mathcal{P}(p) \omega_{\ell}(p; k, R_0) \omega_{\ell}(p; k', R_0) L_{\ell}(\hat{p} \cdot \hat{n}) L_{\ell'}(\hat{p} \cdot \hat{n}) (6.18)
\end{equation}

defining the \( \omega_{\ell} \) function:

\begin{equation}
\omega_{\ell}(p; k, R_0) = 4\pi \int_0^{R_0} r^2 j_L(\ell k r) j_L(\ell r) \mathcal{W}(r; R_0), (6.19)
\end{equation}

noting that this is sharply peaked at \( p \sim k \) for large \( R_0 \). Note that this recovers the aforementioned symmetry for even \( \ell' \) (or by symmetry \( \ell \)).

\[10\] Due to the form of the window function, \( \omega_{\ell} \) is analytic, and may be computed by inserting the definition of \( \mathcal{W}(r; R_0) \) then integrating over the resulting polynomial-weighted pair of spherical Bessel functions using the indefinite integral results given in Bloomfield et al. (2017).
The next step is to integrate over the angular part of \( \mathbf{p} \), though this requires knowledge of the angular dependence of \( P^2(\mathbf{p}) \). To do this, we use the fact that the multipoles of \( P^2 \) are related to those of \( P \) via

\[
\left[ P^2 \right]_L(p) = \int \frac{d\Omega_p}{4\pi} L_L(\hat{\mathbf{p}} \cdot \hat{\mathbf{n}}) P^2(p) = (2L + 1) \sum_{l_1l_2} \left( \begin{array}{ccc} \ell_1 & \ell_2 & L \\ 0 & 0 & 0 \end{array} \right) P_{\ell_1}(p)P_{\ell_2}(p),
\]

where the term in parentheses is a Wigner 3j symbol (NIST DLMF, Eq. 34.2.4), assuming \( (\ell_1, \ell_2, L) \) to obey triangle conditions, using the result

\[
\int d\Omega_p L_{l_1}(\hat{\mathbf{p}} \cdot \hat{\mathbf{n}}) L_{l_2}(\hat{\mathbf{p}} \cdot \hat{\mathbf{n}}) L_L(\hat{\mathbf{p}} \cdot \hat{\mathbf{n}}) = 4\pi \left( \begin{array}{ccc} \ell_1 & \ell_2 & L \\ 0 & 0 & 0 \end{array} \right)^2,
\]

via the Gaunt integral (NIST DLMF, Eq. 34.3.21). Inserting this relation and integrating over \( \mathbf{p} \) (again via Eq. 6.21) gives the final result

\[
4C_{\ell\ell'}^G(k, k'; R_0) = (2\ell + 1)(2\ell' + 1) V \int \frac{d^2p}{2\pi^2} \omega_{\ell}(p; k, R_0) \omega_{\ell'}(p; k', R_0) Q_{\ell\ell'}(p) 
\]

(assuming \( \ell, \ell' \) to be even), where the symmetric function \( Q_{\ell\ell'} \) is defined in terms of the multipoles of \( P(k) \) by

\[
Q_{\ell\ell'}(p) = \sum_{l_1l_2} (2L + 1) \left( \begin{array}{ccc} \ell & \ell' & L \\ 0 & 0 & 0 \end{array} \right)^2 P_{\ell_1}(p)P_{\ell_2}(p).
\]

Whilst this may seem complex, the various symmetries required for a non-zero 3j symbol mean that the expression is relatively manageable at small \( \ell, \ell' \), with important values including

\[
Q_{00}(p) = P_0^2(p) + \frac{1}{5} P_2^2(p) + \ldots 
\]

\[
Q_{02}(p) = \frac{2}{5} P_0^2(p)P_2(p) + \frac{4}{35} P_2^4(p) + \ldots 
\]

\[
Q_{22}(p) = \frac{1}{5} P_0^2(p) + \frac{4}{35} P_0^2(p)P_2(p) + \frac{3}{35} P_2^4(p) + \ldots 
\]

ignoring terms above the quadrupole for brevity. If we assume an isotropic field, such that \( P_{\ell'} = 0 \) for \( \ell > 0 \), we obtain

\[
Q_{\ell\ell'}(k) = (2\ell + 1)^{-1} \delta_{\ell\ell'}^k P_{0}(k). 
\]

Taking the \( R_0 \to \infty \) limit (i.e. replacing \( \omega_{\ell} \) with Eq. A5) yields

\[
\lim_{R_0 \to \infty} 4C_{\ell\ell'}^G(k, k'; R_0) = \frac{(2\ell + 1)(2\ell' + 1)}{V} Q_{\ell\ell'}(k) \times \frac{2\pi^2}{k^2} \delta_D(k - k') 
\]

\[
= \left[ 2\ell'^k \frac{2\ell + 1}{V} P_0^2(k') + \ldots \right] \times \frac{2\pi^2}{k^2} \delta_D(k - k'),
\]

reproducing familiar results (e.g., Sugiyama et al. 2019a; Li et al. 2019; Wadefalk & Scoccimarro 2019), with the final line obtained by assuming the power to be dominated by the monopole \( P_0 \).

For the non-Gaussian part, a similar derivation is possible, again expressing the \( \tilde{W} \) functions in terms of their inverse Fourier transforms and integrating over the Legendre polynomials. Following some algebra, we obtain the result

\[
4C_{\ell\ell'}^{NG}(k, k'; R_0) = \frac{(2\ell + 1)(2\ell' + 1)}{V} (-1)^{\ell + \ell'} \int \frac{d^2p_1}{2\pi^2} \frac{d^2p_2}{2\pi^2} \tilde{T}_{\ell\ell'}(p_1, p_2) \omega_{\ell}(p_1; k, R_0) \omega_{\ell'}(p_2; k', R_0),
\]

where \( \tilde{T}_{\ell\ell'} \) are the multipole moments of a collapsed triaspectrum, defined as

\[
\tilde{T}_{\ell\ell'}(p_1, p_2) = \int \frac{d\Omega_{p_1}}{4\pi} \frac{d\Omega_{p_2}}{4\pi} \tilde{T}(p_1 - \mathbf{p}, p_2 - \mathbf{p}) L_{\ell}(\mathbf{p} \cdot \hat{\mathbf{n}}) L_{\ell'}(\mathbf{p} \cdot \hat{\mathbf{n}}).
\]

As \( R_0 \to \infty \), this has the limit

\[
\lim_{R_0 \to \infty} 4C_{\ell\ell'}^{NG}(k, k'; R_0) = \frac{(2\ell + 1)(2\ell' + 1)}{V} (-1)^{\ell + \ell'} \tilde{T}_{\ell\ell'}(k, k'),
\]

again agreeing with the standard form.

### 6.2 Poisson Covariance

As previously mentioned, the four-point term is not the only contributor to the covariance; the limited number of tracer particles give two- and three-point terms, which vanish in the limit of infinite \( n \). We start with the three-point term of Eq. 6.6, relabelling variables by translation invariance:

\[
3C(k', k'; R_0) = \frac{4}{nV^2} \int d\mathbf{x}_1 d\mathbf{x}_2 d\mathbf{y} K(x; \mathbf{k}, R_0) K(y; \mathbf{k}', R_0) [1 + \xi(x) + \xi(y) + \xi(x - y) + \xi'(x, y)].
\]
Several terms are simplified by noting
\[
\int dx K(x; k, R_0) = \int dx e^{-ik \cdot x} W(x; R_0) = \tilde{W}(k; R_0)
\]
\[
\int dx K(x; k, R_0) \zeta(x) = \int dx e^{-ik \cdot x} W(x; R_0) \zeta(x) = P(k; R_0)
\]
\[
\int dx dy K(x; k, R_0) K(y; k', R_0) \zeta(x, y) = \int dx dy e^{-ik \cdot x} e^{-ik' \cdot y} W(x; R_0) W(y; R_0) \zeta(x, y) = B(k, k'; R_0)
\]
thus
\[
3C(k, k'; R_0) = \frac{4}{nV} \left[ \tilde{W}(k; R_0)\tilde{W}(k'; R_0) + \tilde{W}(k; R_0)P(k'; R_0) + \tilde{W}(k'; R_0)P(k; R_0) + B(k, k'; R_0) \right] (6.31)
\]
We expect the bispectrum term to be subdominant to the power spectrum terms, thus this can usually be ignored, though we keep it in the below for completeness. For the first four terms, it is straightforward to extract the Legendre multipoles.\(^{11}\)
\[
3C_{\ell\ell'}(k, k'; R_0) \supset \frac{4}{nV} \left[ \tilde{W}(k; R_0)\tilde{W}(k'; R_0)\delta^{K}_{\ell\ell'}R^{K}_{0} + \tilde{W}(k; R_0)P_{\ell}(k'; R_0)\delta^{K}_{\ell\ell'}R^{K}_{0} + \tilde{W}(k'; R_0)P_{\ell}(k; R_0)\delta^{K}_{\ell\ell'}R^{K}_{0} + B_{\ell}(k, k'; R_0)\delta^{K}_{\ell\ell'}R^{K}_{0} \right] (6.32)
\]
The remaining three-point term may be rewritten by expressing the 2PCF in Fourier space;
\[
3C_{\ell\ell'}(k, k'; R_0) \supset \frac{4}{nV} (2\ell + 1)(2\ell' + 1) \int \frac{dp}{(2\pi)^3} P(p) \int dx dy W(x; R_0) W(y; R_0) \int \frac{d\Omega_k d\Omega_{k'}}{(4\pi)^2} L_{\ell}(\hat{k} \cdot \hat{\eta}) L_{\ell'}(\hat{k'} \cdot \hat{\eta}) e^{-i(k-p) \cdot x} e^{-i(k'+p) \cdot y} (6.34)
\]
Next we use the result that
\[
\int dx W(x; R_0) \int \frac{d\Omega_k}{4\pi} L_{\ell}(\hat{k} \cdot \hat{\eta}) e^{-i(k-p) \cdot x} = \int dx W(x; R_0) e^{ip \cdot x} \sum_L (\frac{(L-\ell)}{2})^2 L_{\ell}(\hat{k} \cdot \hat{\eta}) L_{\ell}(\hat{\eta} \cdot \hat{x}) (6.35)
\]
\[
= \int dx W(x; R_0) \int \frac{d\Omega_k}{4\pi} L_{\ell}(\hat{k} \cdot \hat{\eta}) L_{\ell}(\hat{\eta} \cdot \hat{x}) (6.36)
\]
utilizing Legendre polynomial completeness and the plane wave expansion, giving
\[
3C_{\ell\ell'}(k, k'; R_0) \supset \frac{4}{nV} (2\ell + 1)(2\ell' + 1)(-1)^\ell \int \frac{dp}{(2\pi)^3} P(p) \omega_{\ell}(p; k, R_0) \omega_{\ell'}(p; k', R_0) L_{\ell}(\hat{p} \cdot \hat{\eta}) L_{\ell'}(\hat{p'} \cdot \hat{\eta}) (6.37)
\]
\[
= \frac{4}{nV} (2\ell + 1)(2\ell' + 1)(-1)^\ell \int \frac{dp^2}{2\pi^2} \sum_L \frac{p_{\ell,\ell'}^2}{2} P_{\ell}(p) \omega_{\ell}(p; k, R_0) \omega_{\ell'}(p; k', R_0) L_{\ell}(\hat{p} \cdot \hat{\eta}) L_{\ell'}(\hat{p'} \cdot \hat{\eta}) (6.38)
\]
where we have expressed \(P(p)\) in terms of its multipoles and used the result of Eq. 6.21 to evaluate the integral over three Legendre polynomials. Note that, for large \(R_0\), the \(\omega\) functions will be sharply peaked, enforcing \(p \approx k \approx k'\). In the limit of \(R_0 \to \infty\), we obtain
\[
\lim_{R_0 \to \infty} 3C_{\ell\ell'}(k, k'; R_0) = \frac{4}{nV} (2\ell + 1)(2\ell' + 1)(-1)^\ell \frac{2\pi^2}{k^2} \delta_D (k - k') \sum_L \left( \begin{array}{c} L \\ 0 \end{array} \right) \left( \begin{array}{c} \ell \\ 0 \end{array} \right) \left( \begin{array}{c} \ell' \\ 0 \end{array} \right) ^2 P_L(k) + \frac{4}{nV} \delta^{K}_{\ell\ell'}B_{\ell}(k, k'; R_0), (6.39)
\]
noting that the other three-point terms only contribute to the zero-lag covariance \((k = 0 \text{ and/or } k' = 0)\) which has been ignored.

Computing the two-point term proceeds similarly, starting from
\[
3C(k, k'; R_0) = \frac{4}{n^2V^2} \int dx dx' K(x; k, R_0) K(x'; k', R_0) [1 + \xi(x)], (6.40)
\]
thus
\[
2C_{\ell\ell'}(k, k'; R_0) = \frac{4}{n^2V^2} (2\ell + 1)(2\ell' + 1) \int dx \int \frac{d\Omega_k d\Omega_{k'}}{4\pi^2} e^{-i(k+k') \cdot x} W^2(x; R_0) [1 + \xi(x)] L_{\ell}(\hat{k} \cdot \hat{\eta}) L_{\ell'}(\hat{k'} \cdot \hat{\eta}) (6.41)
\]
\[
= \frac{4}{n^2V^2} (2\ell + 1)(2\ell' + 1)(-i)^{\ell+\ell'} \int x^2 dx W^2(x; R_0) \xi_{\ell}(k x) \xi_{\ell'}(k' x) \int \frac{d\Omega_k d\Omega_{k'}}{4\pi^2} L_{\ell}(\hat{k} \cdot \hat{\eta}) L_{\ell'}(\hat{k'} \cdot \hat{\eta}) [1 + \xi(x)], (6.42)
\]
11 Only the bispectrum term is non-trivial; this may be computed by expanding \(B(k, k'; R_0) = \sum_L B_L(k, k'; R_0) L_L(\hat{k} \cdot \hat{k'})\) and using spherical harmonic theorems.
using Eq. 6.17. The integral over \( \Omega_k \) can be performed via Legendre polynomial completeness (for the first term) and using Eq. 6.21 for the second, expressing \( \xi(x) \) in terms of its multipoles \( \xi_\ell(x) \). This yields

\[
2C_{\ell\ell'}(k, k'; R_0) = \frac{4}{n^2 V} (2\ell + 1)(2\ell' + 1)(-i)^{\ell + \ell'} 4\pi \int x^2 dx W^2(x; R_0) j_{\ell}(kx) j_{\ell'}(k'x) \left[ \frac{s_{\ell\ell'}}{2\ell + 1} + \sum_L \left( \begin{array}{c} \ell' \\ 0 \\ 0 \end{array} \right) \xi_L(x) \right].
\] (6.39)

Whilst the \( R_0 \to \infty \) limit of this expression is not informative, the unbinned version is more so;

\[
\lim_{R_0 \to \infty} 2C(k, k'; R_0) = \frac{4}{n^2 V} \left[ (2\pi)^2 \delta_D(k + k') + P(k + k') \right],
\] (6.40)

consisting of a constant term on the \( k + k' = 0 \) diagonal and a (subdominant) off-diagonal term.

### 6.3 \( k \)-space Binning and Summary

For proper comparison with data, we ought to consider the covariance of the binned power spectra, \( P^b_{\ell}(R_0) \), which is related to the standard covariance via

\[
\text{cov} \left( P^a_{\ell}(R_0), P^b_{\ell}(R_0) \right)_{\text{intrinsic}} = 4 \left[ C^G_{\ell\ell'}(R_0) + C^C_{\ell\ell'}(R_0) \right]
\]

\[
= 2 \left( \frac{2\ell + 1)(2\ell' + 1)}{V} \right) \int \frac{d^2 p}{2\pi} \omega^a_{\ell}(p; R_0) \omega^b_{\ell}(p; R_0) Q_{\ell\ell'}(p).
\] (6.41)

Since \( k \) and \( k' \) only enter the covariance through the \( \omega_k \) functions, introducing binning is equivalent to replacing \( \omega_k(p; R_0) \) with \( \omega^b_k(p; R_0) \) (and similarly for \( k' \)), where \( \omega^b_k \) is defined as

\[
\omega^b_k(p; R_0) = 4\pi \int_0^{R_0} r^2 dr j_{\ell}(pr) j_{\ell'}(r) W(r; R_0).
\] (6.42)

For the window function of Eq. 2.5, this is analytic (and computable in terms of Bessel function recursion relations, as in Bloomfield et al. 2017), yet complex. The final intrinsic covariance is hence

\[
\text{cov} \left( P^a_{\ell}(R_0), P^b_{\ell}(R_0) \right)_{\text{intrinsic}} = 4 \left[ C^G_{\ell\ell'}(R_0) + C^C_{\ell\ell'}(R_0) \right]
\]

\[
= 2 \left( \frac{2\ell + 1)(2\ell' + 1)}{V} \right) \int \frac{d^2 p}{2\pi} \omega^a_{\ell}(p; R_0) \omega^b_{\ell}(p; R_0) Q_{\ell\ell'}(p).
\] (6.43)

In the limit of \( R_0 \to \infty \), we obtain

\[
\lim_{R_0 \to \infty} C^G_{\ell\ell'}(R_0) = 2 \left( \frac{2\ell + 1)(2\ell' + 1)}{V} \right) \int \frac{d^2 p}{2\pi} \omega^a_{\ell}(p; R_0) \omega^b_{\ell}(p; R_0) Q_{\ell\ell'}(p).
\] (6.44)

where \( Q_{\ell\ell'}^{ab}(p) \) and \( T_{\ell\ell'}^{ab} \) are the binned forms of \( Q_{\ell\ell'}(k) \) and \( T_{\ell\ell'}(k, k') \) respectively. The approximate form of the Gaussian covariance is derived assuming the power to be dominated by the monopole. This matches standard results (e.g., Scoccolmaro et al. 1999b; Sugiyama et al. 2019a; Wadekar & Scoccolmaro 2019; Li et al. 2019), noting that \( Vr_\alpha/(2\pi)^2 \) is equal to the number of \( k \)-space modes in the bin.

For the Poisson covariance, computation is similar, yielding

\[
\text{cov} \left( P^a_{\ell}(R_0), P^b_{\ell}(R_0) \right)_{\text{Poisson}} = 3 C^G_{\ell\ell'}(R_0) + 2 C^C_{\ell\ell'}(R_0)
\]

\[
= 3 \left[ C^G_{\ell\ell'}(R_0) + C^C_{\ell\ell'}(R_0) \right]
\]

\[
= \frac{4}{n^2 V} \left[ 2\ell + 1)(2\ell' + 1)(-i)^{\ell + \ell'} 4\pi \int x^2 dx W^2(x; R_0) j_{\ell}(kx) j_{\ell'}(k'x) \left[ \frac{s_{\ell\ell'}}{2\ell + 1} + \sum_L \left( \begin{array}{c} \ell' \\ 0 \\ 0 \end{array} \right) \xi_L(x) \right].
\] (6.45)
In the limit of $R_0 \to \infty$, and ignoring zero-lag terms, we obtain
\begin{align}
\lim_{R_0 \to \infty} 3C_{\ell \ell'}^{ab}(R_0) &= \frac{4}{nV} \frac{(2\pi)^3}{v_a} \delta_{ab}^{k} (2\ell + 1)(2\ell' + 1)(-1)^{\ell} \sum_{L} \left( L 0 \ell 0 \ell' \right) \frac{P^a_L + 4}{nV} \Delta_{\ell \ell'}^{ab} \Delta_{\ell \ell'}^{ab} \\
\lim_{R_0 \to \infty} 2C_{\ell \ell'}^{ab}(R_0) &= \frac{4}{n^2V} \frac{(2\pi)^3}{v_a} \delta_{ab}^{k} \Delta_{\ell \ell'}^{ab} (2\ell + 1)(-1)^{\ell} \\
&\quad + \frac{4}{n^2V}(2\ell + 1)(2\ell' + 1)(-1)^{\ell + \ell'} 4\pi \int x^2 dx j^2_\ell(x) j^2_{\ell'}(x) \sum_{L} \left( L 0 \ell 0 \ell' \right) 2 \xi_L(x).
\end{align}

This completes the covariance computation.

### 7 CROSS-COVARIANCE OF THE POWER SPECTRUM AND BISPECTRUM

Before considering the covariance of the bispectrum itself, we give brief results pertaining to the cross-covariance of the power spectrum multipoles $P_\ell(k)$ and the isotropic bispectrum multipoles $B_\ell(k_1, k_2)$. Due to the greater mathematical complexities of this compared to the power spectrum covariance, we will consider only Gaussian terms (both intrinsic and Poissonian) and work in the $R_0 \to \infty$ limit (additionally assuming infinite randoms). The principal effect of finite $R_0$ is to add off-diagonal contributions (as well as to modify the low-$k$ covariances), but we expect it to be small for wide $k$-bins and moderate $R_0$. We further avoid complexity by ignoring $k$-space binning in this section.

In the $R_0 \to \infty$ limit, our covariances will be identical to those obtained from conventional analyses. For this reason, we take our starting point to be the cross-covariance presented in Sugiyama et al. (2019a, Sec. 2.4), which include contributions from shot-noise but not the projection onto the Legendre multipole basis. In our notation, and dropping all non-Gaussian results, we take our starting point to be the cross-covariance presented in Sugiyama et al. (2019a, Sec. 2.4), which include contributions from shot-noise but not the projection onto the Legendre multipole basis. In our notation, and dropping all non-Gaussian terms, we can write
\[\text{cov} \left( \hat{P}(k), \hat{B}(k_1, k_2) \right) = \frac{2}{V} (2\pi)^3 \delta_D(k + k_1) P^{(N)}(k_1) B^{(N)}(k_1, k_2, k_3) + 2 \text{cyc.} \] (7.1)

defining $k_3 = -k_1 - k_2$, with cyclic interchanges performed over $(k_1', k_2', k_3')$. This uses the asymmetric definitions
\[P^{(N)}(k) = P(k) + \frac{1}{n} \]
\[B^{(N)}(k_1, k_2, k_3) = \frac{1}{n} (P(k_2) + P(k_3)) \]
(Sugiyama et al. 2019a), where we have again dropped all non-Gaussian terms. In this case, we note that there is no intrinsic Gaussian covariance, and further, that Eq. 7.1 mixes together terms of different orders in $n^{-1}$ for brevity.

Projecting onto multipoles, we obtain
\[\text{cov} \left( \hat{P}_\ell(k), \hat{B}_\ell(k_1, k_2) \right) = (2\ell + 1)(2\ell' + 1) \int_{\Omega_{k_1} \Omega_{k_2}} L(\hat{k} \cdot \hat{n}) L(\hat{k}_1 \cdot \hat{n}_1) L(\hat{k}_2 \cdot \hat{n}_2) \text{cov} \left( \hat{P}_\ell(k), \hat{B}_\ell(k_1, k_2) \right), \]
(7.3)
for (fixed) line-of-sight vector $\hat{n}$, writing $\int_{\Omega} = \int d\Omega_{\hat{n}}$ for brevity. Inserting Eq. 7.1, we find two terms with different structures under cyclic permutation;
\[\text{cov} \left( \hat{P}_\ell(k), \hat{B}_\ell(k_1, k_2) \right) = C_{\ell \ell'}^{A}(k_1, k_2) + C_{\ell \ell'}^{B}(k_1, k_2) \]
(7.4)

where we have introduced $k_3$ via a Dirac function, and denoted $\delta^{(k_3)}_D \equiv 2\pi^2 / (k_3 k_3) \times \delta_D(k_3 - k)$, i.e. the radial part of the Dirac function. To simplify this we can expand the spectra in terms of their multipoles, i.e. $P(k_1) \equiv \sum_{L} P_L(k) L(\hat{k} \cdot \hat{n})$ and compute the integrals over Legendre polynomials analytically. Whilst the full calculation is somewhat lengthy, it yields the following results
\[C_{\ell \ell'}^{A}(k_1, k_2) \]
\[2(2\ell + 1)(2\ell' + 1) \]
\[\left( \begin{array}{ccc} L & L' & 1 \ 
0 & 0 & 0 \ \end{array} \right) 2 \frac{P^{(N)}(k)}{P_L(k)} P_L(k_1) + (k_1 \leftrightarrow k_2) \]
\[2(2\ell + 1)(2\ell' + 1) \]
\[\left( \begin{array}{ccc} L & L' & 1 \ 
0 & 0 & 0 \ \end{array} \right) 2 \frac{P^{(N)}(k)}{P_L(k)} P_L(k_2) + (k_1 \leftrightarrow k_2) \]
\[C_{\ell \ell'}^{B}(k_1, k_2) \]
\[2(2\ell + 1)(2\ell' + 1) \]
\[\left( \begin{array}{ccc} L & L' & 1 \ 
0 & 0 & 0 \ \end{array} \right) 2 \frac{P^{(N)}(k)}{P_L(k)} P_L(k_1) + (k_1 \leftrightarrow k_2) \]
\[2(2\ell + 1)(2\ell' + 1) \]
\[\left( \begin{array}{ccc} L & L' & 1 \ 
0 & 0 & 0 \ \end{array} \right) 2 \frac{P^{(N)}(k)}{P_L(k)} P_L(k_2) + (k_1 \leftrightarrow k_2) \]
\[\]
using $\int d\mathbf{k} \, P(\mathbf{k}) e^{i \mathbf{k} \cdot \mathbf{x}} \equiv \xi(\mathbf{x})$ and defining

$$T_{\ell_1 \ell_2 \ell_3}(k_1, k_2) = 4\pi \int x^2 dx \, j_{\ell_1}(k_1 x) j_{\ell_2}(k_2 x) \xi_{\ell_3}(x).$$

(7.6)

In the simple case of a $1/r^2$ 2PCF, this may be evaluated using Gradsteyn et al. (2007, Eqs. 6.574.1-3). In the general case, Eq. 7.6 can be evaluated via the prescriptions of Assassi et al. (2017) or Slepian et al. (2019).

8 IDEALIZED BISPECTRUM COVARIANCE

We now present a brief discussion of the bispectrum auto-covariance of the bispectrum, leading on from the preceding sections. Since the mathematics of this section is significantly more involved than the above, we will assume Gaussianity and the $R_0 \rightarrow \infty$ limit (as for the cross-spectrum) and additionally work in real-space (which is equivalent to assuming that the power spectra are dominated by their monopole contributions). We defer consideration of the full covariance including all the above effects to future work. Starting from the results of Sugiyama et al. (2019a, Sec. 2.5), we obtain

$$\text{cov}\left(\hat{B}(k_1, k_2), \hat{B}(\mathbf{k}_1', \mathbf{k}_2')\right) = C_{PPP}^{\mathbf{k}_1, \mathbf{k}_2, \mathbf{k}_1', \mathbf{k}_2'} + C_{BB}^{\mathbf{k}_1, \mathbf{k}_2, \mathbf{k}_1', \mathbf{k}_2'} + C_{PT}^{\mathbf{k}_1, \mathbf{k}_2, \mathbf{k}_1', \mathbf{k}_2'}$$

(8.1)

with permutations performed over $\{\mathbf{k}_1', \mathbf{k}_2', \mathbf{k}_3'\}$ as before. This uses the additional (symmetrized) definition

$$T^{\mathbf{k}_1, \mathbf{k}_2, \mathbf{k}_1', \mathbf{k}_2'} = \frac{1}{2\pi^2} \left[ P(|\mathbf{k}_1 + \mathbf{k}_2'|) + P(|\mathbf{k}_1 + \mathbf{k}_2'|) + P(|\mathbf{k}_2 + \mathbf{k}_1'|) + P(|\mathbf{k}_2 + \mathbf{k}_1'|) \right].$$

(8.2)

The multipole covariance follows from

$$\text{cov}\left(\hat{B}_{\ell}(k_1, k_2), \hat{B}_{\ell'}(\mathbf{k}_1', \mathbf{k}_2')\right) = (2\ell + 1)(2\ell' + 1) \int \Omega_{\ell_1} \Omega_{\ell_2} \Omega_{\ell_1'} \Omega_{\ell_2'} L_{\ell}(\hat{\mathbf{k}}_1 \cdot \hat{\mathbf{k}}_2) L_{\ell'}(\hat{\mathbf{k}}_1' \cdot \hat{\mathbf{k}}_2') \text{cov}\left(\hat{B}(k_1, k_2), \hat{B}(\mathbf{k}_1', \mathbf{k}_2')\right).$$

(8.3)

We now consider the individual terms in this expansion.

8.1 PPP Term

The computation of the first term of the multipole covariance proceeds similarly to that of the 3PCF discussed in Slepian & Eisenstein (2015, Sec. 6). We begin by explicitly including the momentum conserving Dirac deltas $\delta_D(k_1 + k_2 + k_3) + \delta_D(k_1' + k_2' + k_3')$ in the six-point covariance

$$C_{PPP}^{\ell}(k_1, k_2, k_1', k_2') = \frac{1}{2V} \int \Omega_1 \Omega_2 \Omega_3 \Omega_4 \Omega_5 \Omega_6 \int p^{(N)}(k_1) p^{(N)}(k_2) p^{(N)}(k_3) L_{\ell}(\hat{\mathbf{k}}_1 \cdot \hat{\mathbf{k}}_2) L_{\ell'}(\hat{\mathbf{k}}_1' \cdot \hat{\mathbf{k}}_2') (2\pi)^3 \delta_D(k_1 + k_2 + k_3)$$

$$\times \left[ (2\pi)^3 \delta_D(k_1' + k_3') (2\pi)^3 \delta_D(k_2 + k_3') (2\pi)^3 \delta_D(k_3 + k_3') + 5 \text{ perms.} \right].$$

(8.4)

We now separate this into the sum of two components with different structures under permutation and apply the Dirac deltas;

$$C_{PPP}^{\ell}(k_1, k_2, k_1', k_2') = C_{PPP, A}^{\ell}(k_1, k_2, k_1', k_2') + C_{PPP, B}^{\ell}(k_1, k_2, k_1', k_2')$$

(8.5)

where

$$C_{PPP, A}^{\ell}(k_1, k_2, k_1', k_2') = \frac{1}{2V} \int \Omega_1 \Omega_2 \Omega_3 \Omega_4 \Omega_5 \Omega_6 \int p^{(N)}(k_1) p^{(N)}(k_2) p^{(N)}(k_3) L_{\ell}(\hat{\mathbf{k}}_1 \cdot \hat{\mathbf{k}}_2) L_{\ell'}(\hat{\mathbf{k}}_1' \cdot \hat{\mathbf{k}}_2') (2\pi)^3 \delta_D(k_1 + k_2 + k_3)$$

$$\times \left[ L_{\ell}(\hat{\mathbf{k}}_1 \cdot \hat{\mathbf{k}}_2) \delta_{k_1, k_1'} \delta_{k_2, k_2'} + 3 \text{ sym.} \right].$$

$$C_{PPP, B}^{\ell}(k_1, k_2, k_1', k_2') = \frac{1}{V} \int \Omega_1 \Omega_2 \Omega_3 \Omega_4 \Omega_5 \Omega_6 \int p^{(N)}(k_1) p^{(N)}(k_2) L_{\ell}(\hat{\mathbf{k}}_1 \cdot \hat{\mathbf{k}}_2)$$

$$\times \left[ (2\pi)^3 \delta_D(k_1 + k_2 + k_3) p^{(N)}(k_3) L_{\ell'}(\hat{\mathbf{k}}_1' \cdot \hat{\mathbf{k}}_2' \delta_{k_1, -k_1'} \delta_{k_2, -k_2'} + 3 \text{ sym.} \right].$$
where we define

\[ \frac{\delta^K_{\ell\ell'}}{2 \ell + 1} \]

where \( \Gamma \) using Slepian & Eisenstein (2015, Eq. 59) where the parentheses indicate a 3j symbol and

\[ \Omega \]

For the first intrinsic covariance component, it is useful to switch the order of integral over \( k_1 \) and \( k_2 \) via Fubini’s theorem (separating out the 1/n term in \( p^{(N)}(k_3) \)) and note that

\[ \int \frac{k_3^2 dk_3}{2 \pi^2} j_0(k_3 x) P(k_3) = \xi(x) \]

since this is merely a spherical Fourier transform. Simplification thus yields

\[ C^{PPP,A}_{\ell\ell'}(k_1, k_2, k'_1, k'_2) = \frac{(2 \ell + 1)(2 \ell' + 1)}{2V} p^{(N)}(k_1) p^{(N)}(k_2) \left[ \delta^{k_1 k'_1}_{D} \delta^{k_2 k'_2}_{D} + 3 \text{ sym} \right] \]

\[ \times \sum_L (-1)^{\ell+\ell'} (2L+1) \left( \begin{array}{ccc} L & \ell & \ell' \\ 0 & 0 & 0 \end{array} \right)^2 S_{LL}(k_1, k_2) + \frac{1}{n} \frac{\delta^K_{\ell\ell'}}{2 \ell + 1} \]

\[ C^{PPP,B}_{\ell\ell'}(k_1, k_2, k'_1, k'_2) = \frac{(2 \ell + 1)(2 \ell' + 1)}{V} p^{(N)}(k_1) p^{(N)}(k_2) \sum_L (-1)^{\ell+\ell'+L} (2L+1) \left( \begin{array}{ccc} L & \ell & \ell' \\ 0 & 0 & 0 \end{array} \right)^2 \]

\[ \times \frac{p^{(N)}(k'_1) p^{(N)}(k'_2) R_{LLL}(k_1, k_2, k'_1, k'_2) + 3 \text{ sym}}{2 \ell + 1} \]

where we define

\[ S_{\ell\ell'}(k_1, k_2) = 4\pi \int x^2 dx j_{\ell}(k_1 x) j_{\ell'}(k_2 x) \xi(x). \]

which is a special case of Eq. 7.6. Note that the covariance expressions given above are relatively simple to compute since each depends only on one numerical integral.

### 8.2 BB Term

Calculation of the BB term proceeds similarly, starting with the form

\[ C^{BB}_{\ell\ell'}(k_1, k_2, k'_1, k'_2) = \frac{(2 \ell + 1)(2 \ell' + 1)}{V} \int_{k_1 k_2} \int_{k'_1 k'_2} L_{LL}(k_1 \cdot k_2) L_{LL}(k'_1 \cdot k'_2) (2\pi)^3 \delta_D(k_1 + k_2 + k_3) \delta_D(k'_1 + k'_2 + k'_3) \]

\[ \times \left[ B^{(N)}(k_1, k_2, k_3) B^{(N)}(k'_1, k'_2, k'_3) (2\pi)^3 \delta_D(k_1 - k'_1) + 8 \text{ perms} \right]. \]
Splitting into three terms with different permutative structures and inserting the definition of $B^{(N)}$ gives:

\[
\begin{align*}
\frac{C^{BB,A}_{\ell\ell}(k_1, k_2, \ell', k_2')}{(2\ell + 1)(2\ell' + 1)} &= \frac{1}{n^2 V} \int_{\Omega_{k_1} \Omega_{k_2} \Omega_{\ell_2} \Omega_{\ell_2'}} L_\ell(\hat{k}_1 \cdot \hat{k}_2) L_{\ell'}(\hat{k}_1' \cdot \hat{k}_2')(2\pi)^3 \delta_D(k_1 + k_2 - k_1' - k_2') \quad (8.13) \\
\times \left[(P(k_1) + P(k_2))(P(k_1') + P(k_2'))\right] \\
\frac{C^{BB,B}_{\ell\ell}(k_1, k_2, \ell', k_2')}{(2\ell + 1)(2\ell' + 1)} &= \frac{1}{n^2 V} \int_{k_1 k_2} \int_{\Omega_{k_1} \Omega_{k_2} \Omega_{\ell_2} \Omega_{\ell_2'}} L_\ell(\hat{k}_1 \cdot \hat{k}_2) L_{\ell'}(\hat{k}_1 \cdot \hat{k}_2')(2\pi)^3 \delta_D(k_1 + k_2 + k_2' - k_1') \\
\times \left[(P(k_2) + P(k_3))(P(k_2') + P(k_3'))\right] \delta^{k_1 k_2'}_D + 3 \text{ perms.} \\
\frac{C^{BB,C}_{\ell\ell}(k_1, k_2, \ell', k_2')}{(2\ell + 1)(2\ell' + 1)} &= \frac{1}{n^2 V} \int_{k_1 k_2} \int_{\Omega_{k_1} \Omega_{k_2} \Omega_{\ell_2} \Omega_{\ell_2'}} L_\ell(\hat{k}_1 \cdot \hat{k}_2) L_{\ell'}(\hat{k}_1 \cdot \hat{k}_2')(2\pi)^3 \delta_D(k_1 + k_2 + k_2' + k_1') \\
\times \left[(P(k_1) + P(k_2))(P(k_1') + P(k_2'))\right] + 3 \text{ perms.}
\end{align*}
\]

Following a lengthy calculation, similar in form to the above, we obtain

\[
\begin{align*}
\frac{C^{BB,A}_{\ell\ell}(k_1, k_2, k_3, k_4)}{(2\ell + 1)(2\ell' + 1)} &= \frac{1}{n^2 V} \left[(P(k_1) + P(k_2))(P(k_1') + P(k_2'))\right] (-1)^{\ell + \ell'} R_{\ell\ell'}(k_1, k_2, k_1', k_2') \quad (8.14) \\
\frac{C^{BB,B}_{\ell\ell}(k_1, k_2, k_3, k_4)}{(2\ell + 1)(2\ell' + 1)} &= \frac{1}{n^2 V} \left[-(-1)^{\ell'} S_{\ell\ell'}(k_1, k_2) + P(k_2)\delta_{00}^K\right] \left[(-1)^{\ell'} S_{\ell\ell'}(k_1, k_2') + P(k_2')\delta_{00}^{K'}\right] \delta^{k_1 k_2'}_D + 3 \text{ perms.} \\
\frac{C^{BB,C}_{\ell\ell}(k_1, k_2, k_3, k_4)}{(2\ell + 1)(2\ell' + 1)} &= \left[(P(k_1) + P(k_2))\right] \left[-(-1)^{\ell + \ell'} R_{\ell\ell'}(k_1, k_2, k_1', k_2') S_{\ell\ell'}(k_1, k_1') + \delta_{00}^{K K'} P(k_1')\right] + 3 \text{ perms.}
\end{align*}
\]

where we define

\[
R_{\ell_1 \ell_2 \ell_3 \ell_4}(k_1, k_2, k_3, k_4) = 4\pi \int x^2 dx j_{\ell_1}(k_1 x) j_{\ell_2}(k_2 x) j_{\ell_3}(k_3 x) j_{\ell_4}(k_4 x).
\]

This can be computed analytically, following the prescription of Mehrem (2009) and Fabrikant (2013).

### 8.3 PT Term

We finally turn to the PT term of the bispectrum covariance. This has the form

\[
C^{PT}_{\ell\ell'}(k_1, k_2, k_1', k_2') = \frac{(2\ell + 1)(2\ell' + 1)}{V} \int_{k_1 k_2} \int_{\Omega_{k_1} \Omega_{k_2} \Omega_{\ell_2} \Omega_{\ell_2'}} L_\ell(\hat{k}_1 \cdot \hat{k}_2) L_{\ell'}(\hat{k}_1' \cdot \hat{k}_2')(2\pi)^3 \delta_D(k_1 + k_2 + k_3) \delta_D(k_1' + k_2' + k_3') \quad (8.16)
\]

Here, we find three terms with different structures:

\[
\begin{align*}
\frac{C^{PT,A}_{\ell\ell'}(k_1, k_2, k_1', k_2')}{(2\ell + 1)(2\ell' + 1)} &= \frac{1}{2n^2 V} \int_{k_1 k_2} \int_{\Omega_{k_1} \Omega_{k_2} \Omega_{\ell_2} \Omega_{\ell_2'}} L_\ell(\hat{k}_1 \cdot \hat{k}_2) L_{\ell'}(\hat{k}_1 \cdot \hat{k}_2')(\hat{k}_1 \cdot \hat{k}_2)(2\pi)^6 \delta_D(k_1 + k_2 + k_3) \delta_D(k_1' + k_2' + k_3' - k_1) \quad (8.17) \\
\times \left[(P(k_2) + P(k_3))(P(k_2) + P(k_3) + P(k_2') + P(k_3'))\right] + 3 \text{ perms.} \\
\frac{C^{PT,B}_{\ell\ell'}(k_1, k_2, k_1', k_2')}{(2\ell + 1)(2\ell' + 1)} &= \frac{1}{2n^2 V} \int_{k_1 k_2} \int_{\Omega_{k_1} \Omega_{k_2} \Omega_{\ell_2} \Omega_{\ell_2'}} L_\ell(\hat{k}_1 \cdot \hat{k}_2) L_{\ell'}(\hat{k}_1' \cdot \hat{k}_2')(2\pi)^6 \delta_D(k_1 + k_2 + k_3) \delta_D(k_1' + k_2' + k_3' - k_1) \\
\times \left[p^{(N)}(k_1) + p^{(N)}(k_2 + k_3) + p^{(N)}(k_3 + k_1') + p^{(N)}(k_3 + k_1') + p^{(N)}(k_3 + k_1') + p^{(N)}(k_3 + k_1')\right] + 1 \text{ perm.} + 1 \text{ sym.} \\
\frac{C^{PT,C}_{\ell\ell'}(k_1, k_2, k_1', k_2')}{(2\ell + 1)(2\ell' + 1)} &= \frac{1}{2n^2 V} \int_{k_1 k_2} \int_{\Omega_{k_1} \Omega_{k_2} \Omega_{\ell_2} \Omega_{\ell_2'}} L_\ell(\hat{k}_1 \cdot \hat{k}_2) L_{\ell'}(\hat{k}_1' \cdot \hat{k}_2')(2\pi)^6 \delta_D(k_1 + k_2 + k_3) \delta_D(k_1' + k_2' + k_3' - k_3) \\
\times \left[p^{(N)}(k_3) + p^{(N)}(k_1 + k_3') + p^{(N)}(k_1 + k_3') + p^{(N)}(k_1 + k_3') + p^{(N)}(k_1 + k_3') + p^{(N)}(k_1 + k_3')\right].
\end{align*}
\]
These may be simplified in a similar way to before, writing the power spectra in terms of correlation functions and computing the angular integrals. Following a similar calculation, this yields

$$\frac{C^{PT,A}_{2l'}}{(2l+1)(2l'+1)} = \frac{1}{n^2V}\delta_{DD}^k p_n(k_1) \times \left[ \frac{\ell'^{\ell'}}{2\ell + 1} C_{\ell'}(k_2, k'_2) + \sum_{L}(\ell'^{\ell'} + L)^2/(2L + 1) \begin{pmatrix} \ell' & 0 \\ 0 & 0 \end{pmatrix} L^2 \right] + 3 \text{perms.}$$

(8.18)

$$\frac{C^{PT,B}_{2l'}}{(2l+1)(2l'+1)} = \frac{1}{n^2V}\delta_{DD}^k p_n(k_1) \times \left[ (-1)^{\ell} C_{\ell'}(k_2, k'_2) \sum_{L}(\ell'^{\ell'} + L)^2/(2L + 1) \begin{pmatrix} \ell' & 0 \\ 0 & 0 \end{pmatrix} L^2 R_{\ell L}(k_1, k'_1, k'_2) + (k'_1 \leftrightarrow k'_2) \right] + 1 \text{perm.} + 1 \text{sym.}$$

$$\frac{C^{PT,C}_{2l'}}{(2l+1)(2l'+1)} = \frac{1}{n^2V} \sum_{LL'PP'QQ} (2L + 1)(2L' + 1)(2P + 1)(2P' + 1)(2Q + 1)(-1)^{\ell + L'}(-1)^{P + Q - L - \ell'} \times \begin{pmatrix} \ell' & L' \\ 0 & 0 \end{pmatrix} \begin{pmatrix} L' & P' \\ 0 & 0 \end{pmatrix} \begin{pmatrix} P' & Q' \\ 0 & 0 \end{pmatrix} C_{LL'}(k_1, k_2, k'_1, k'_2) S_{LL'}(k_1, k'_1, k'_2) S_{LL'}(k_1, k_2) + 3 \text{sym.}$$

where we additionally define

$$S_{\ell_L\ell_L}(k_1, k_2) = 4\pi \int x^2 dx j_{\ell_L}(k_1 x) j_{\ell_L}(k_2 x) j_{\ell_L}(k_3 x) \xi(x).$$

(8.19)

As for Eq. 7.6, this can be computed via the prescription of Slepian et al. (2019) (or Fabrikant 2013 for a $1/r^2$ 2PCF).

9 PRACTICAL APPLICATION

We now turn to a practical application of the power spectrum algorithm (Sec. 3) and bispectrum algorithm (Sec. 4) to simulated data. Both of these have been incorporated into the HIPSTER C++ package, which is highly optimized and allows for fast estimation of small-scale spectra, by exhaustive weighted pair counting up to some maximum radius $R_0$. Previously the package included only an algorithm for power spectra in arbitrary geometries; we augment it with the efficient periodic-box power spectrum and bispectrum algorithms developed in this work. The code is publicly available online, with extensive documentation. In HIPSTER, the spherical harmonics needed for the bispectrum estimator are computed efficiently by working in Cartesian space, as in Slepian & Eisenstein (2015). The limiting step of our algorithm is in repeated calculation of the Bessel function weights $j_{\ell}^m$, we expect runtime could be substantially reduced by making effective use of GPUs. Note that, via simple modifications to HIPSTER, one may also compute isotropic bispectra in arbitrary geometries, as well as anisotropic bispectra.

9.1 Discussion of Algorithm Scalings and Hyperparameters

Before diving into numerical results, we present a brief discussion of the leading dependencies and scalings of our configuration-space spectral code HIPSTER. Denoting the computation time by $T$, we have the following:

- $T \propto N n$ for $N$ tracer particles with number density $n$, or $T \propto N^2$ at fixed volume. This occurs since both the power spectrum and bispectrum estimators can be written as a count over all pairs of particles up to some maximum radius, the number of which scales as $N n$. In particular, note that the computation times for the power spectrum and bispectrum are comparable, unlike conventional methods.
- $T \propto R_0^3$ for pair-count truncation scale $R_0$. The number of particles within $R_0$ from a given primary is proportional to $n R_0^3$, explaining this scaling.
- $T \sim n_k n_f$ (power spectrum) or $T \sim n_k^2 n_f$ (bispectrum) for $n_k$ $k$-bins and $n_f$ Legendre bins. For each pair of particles we must compute their contribution to every possible $k$ and $\ell$ bin. Note that this is not an exact scaling, since many of the computational processes (e.g., choosing particles) are independent of the binning. We further stress that the speed is sensitive only the number of $k$-bins used, and not their magnitudes.
- $T \propto (1 + f_{\text{rand}})$ for the bispectrum, when using a random catalog $f_{\text{rand}}$ times larger than the data. This is because the number density $n_{\text{rand}}$ of randoms satisfies $n_{\text{rand}} = n f_{\text{rand}}$ and we must count all data-data and random-data pairs (but not the random-random pairs since these are computed analytically). We find $f_{\text{rand}} \sim 3$ to be sufficient in practice.

HIPSTER.readthedocs.io.

MNRAS 000, 1–32 (2020)
• $T \propto N_{\text{cpu}}^{-1}$ when computed on $N_{\text{cpu}}$ CPU cores. Since the algorithm is a pair count it can be trivially parallelized.

A key hyperparameter of the code is the truncation scale $R_0$, which strongly influences the runtime. Although a larger $R_0$ implies a count over more pairs of particles, choosing an optimal $R_0$ is not as simple as minimizing the variance of the estimators at fixed runtime (which is the way in which the number of random particles used in 2PCF and 3PCF estimators are optimized), since lowering $R_0$ reduces only the number of large separation pairs counted, which have minimal impact on high-$k$ estimates. As shown in Sec. 5 and PE20, the primary effect of $W(r; R_0)$ is to convolve the true spectra with a window function; this causes the slight distortions at low-$k$, both their means and errors, though one might note minor deviations in the smallest bins, and that the computation time can be reduced with smaller $R_0$. For $R_0 = 100h^{-1}\text{Mpc}$ ($50h^{-1}\text{Mpc}$), $k \gtrsim 0.25h\text{Mpc}^{-1}$ ($0.5h\text{Mpc}^{-1}$) can be measured robustly using HIPSTER, without consideration of the convolution window. Using $R_0 \sim k_{\text{min}}^{-1}$ gives $T \sim k_{\text{min}}^{-3}$, making this method optimal for high-$k$ spectral measurements. This further sets the optimal $k$-binning, requiring $\Delta k \gtrsim 3/R_0$ to minimize the covariance between bins. In the below, we principally use linear bins with $\Delta k = 0.05$, satisfying this condition, reducing the correlation between nearby bins. Note that the above argument assumes that we do not perform window-convolution of the theory model; this would partially ameliorate these distortions, a full Fisher analysis is needed to assess the impact on cosmological observables due to the truncation.

For conventional Fourier-transform estimators, the principal scaling is with the size of the discrete particle grid, $N_{\text{grid}}$, with the complexity scaling as $O(N_{\text{grid}} \log N_{\text{grid}})$ thanks to the efficient FFT algorithm. In practice, $N_{\text{grid}}$ is set by the Nyquist frequency $k_{\text{Nyq}} = \pi N_{\text{grid}}/L$ of the length $L$ box; we require $k \lesssim k_{\text{Nyq}}/2$ to avoid the affects of aliasing. $N_{\text{grid}}$ thus scales linearly with the maximum $k$-vector used, giving $T \sim k_{\text{max}} \log k_{\text{max}}$, opposite to HIPSTER. Assuming the most computationally intensive segment of a spectral algorithm to be performing the FFTs, power spectrum computation is roughly independent of the number of bins (unlike the configuration-space estimators), though the bispectrum estimators, which are proportional to $n_f^3$ as discussed below, are not superior to HIPSTER in this regard. Further benefits of our estimators are that they do not suffer from shot-noise, and they require very little memory, since only the sample of particles has to be held and not large FFT grids.

Given that our estimators have a strong scaling with the number of particles, $N$, not seen in FFT-based approaches, it is often useful to subsample the data by using only a random fraction $1/f_{\text{sub}}$ of the full catalog of tracer particles. Whilst this is not necessary for the computation of halo spectra, it is of great importance for matter spectra, where $n$ is large. This does not affect the intrinsic covariance (which is independent of $n$), but boosts the two- and three-point Poisson terms. This is discussed further in Sec. 9.3.

### 9.2 Simulated Power Spectra

#### 9.2.1 Comparison of Methods

To determine whether our power spectrum estimator is accurate, we compute the $z = 0$ power spectrum of 100 N-body simulations, taken from the Quijote project (Villaescusa-Navarro et al. 2019); a suite of over 40,000 simulations run using the GADGET-III TreePM + SPH code (Springel 2005). Here, we use only a subset of the fiducial cosmology simulations, each of which have boxsize $L = 1h^{-1}\text{Gpc}$, and contain $512^3$ cold dark matter (CDM) particles evolving started from $z = 127$, with initial conditions generated from second-order Lagrangian perturbation theory (2LPT). For the majority of this section, we focus on halo power spectra, though we note the algorithms are equally applicable to matter power spectra. Halo catalogs are computed with the friends-of-friends algorithm (Davis et al. 1985) with a linking length of $b = 0.2$. Furthermore, we add redshift-space-distortions using the velocity of the simulated halos in combination with the Hubble expansion parameter.

For each simulation, power spectra are estimated (a) via HIPSTER as detailed above, and (b) via the nbodykit code (Hand et al. 2018) which computes spectra in the conventional method via FFTs. For both codes, this is performed for $\ell \in (0.2, 4)$ over $k$-bins with fixed width $\Delta k = 0.05h\text{Mpc}^{-1}$ for $k \in [0.2]h\text{Mpc}^{-1}$. In the FFT-based method, spectra are computed with triangle-shaped-cell interpolation using grid-sizes of $N_{\text{grid}} \in (512, 1024)$, giving Nyquist frequencies of $k_{\text{Nyq}} = \pi N_{\text{grid}}/L \in (1.6, 3.2)h\text{Mpc}^{-1}$. We subtract the shot-noise contribution $P_{\text{shot}} = n^{-1}$ from the spectrum monopole, where $n$ is the number density of tracers $\sim 4 \times 10^{-4} h^3 \text{Mpc}^{-3}$. For HIPSTER, we use truncation scales $R_0 \in (50, 100)h^{-1}\text{Mpc}$, and do not need to remove shot-noise. Considering computation time, for a full galaxy catalog containing $\sim 4 \times 10^5$ particles with this choice of $k$-bins and Legendre multipoles, HIPSTER with $R_0 = 100h^{-1}\text{Mpc}$ truncation requires $\sim 60$ core-minutes to compute each spectrum, whilst the analogous computation with nbodykit needs $\sim 20$ core-minutes at $N_{\text{grid}} = 1024$. We note however that HIPSTER can compute spectra up to much higher $k$ with no additional cost (recalling that the computation time scales with the number of bins), and that the computation time can be reduced with smaller $R_0$ or by subsampling the data.

The corresponding spectra are shown in Fig. 1, and we immediately note that the two codes appear highly consistent in both their means and errors, though one might note minor deviations in the smallest $k$-bins, and slight differences in the error bars from the largest $k$-bins. As discussed in Sec. 5, the measured configuration-space power spectrum is a convolution of the true spectrum with a window function; this causes the slight distortions at low-$k$ (see also PE20, Fig. 2). At large scales, where $k$ approaches the Nyquist frequency of the box, we do not expect FFT-based estimates to be accurate.
To investigate these effects in detail, in Fig. 3 we plot the ratio of HIPSTER and nbodykit power spectra as a function of the pair-count truncation scale $R_0$ and grid-size $N_{\text{grid}}$. Considering the average over 100 mocks, we find sub-percent agreement between the two methods for $k > 0.5 h \text{Mpc}^{-1}$ ($k > 0.25 h \text{Mpc}^{-1}$) for $R_0 = 50 h^{-1} \text{Mpc}$ ($100 h^{-1} \text{Mpc}$), indicating that our estimators are unbiased. At smaller $k$, we observe oscillatory behavior in the average ratio which can be attributed to the inclusion of a pair-count window function in HIPSTER. This effect is clearly amplified at larger $\ell$ and smaller $R_0$, as noted in PE20, but negligible at large $k$. For the smaller value of $N_{\text{grid}}$, there is a clear divergence near the Nyquist frequency ($\sim 1.6h \text{Mpc}^{-1}$), representing the fundamental limit of FFT-based approaches. Whilst this is not observed for the $N_{\text{grid}} = 1024$ sample, we expect deviations to start appearing near $k \approx 2h \text{Mpc}^{-1}$ based on the $N_{\text{grid}} = 512$ data. An initial glance at the monopole ratio near $k = 1.5h \text{Mpc}^{-1}$ would suggest that HIPSTER is inaccurate there; in fact this is caused by the monopole power crossing zero around this wavenumber (and hence the ratio being between two small stochastic quantities), and is not found in the quadrupole.

Notably, we observe significant stochastic variations in the HIPSTER-to-FFT ratio across the datasets. If the two estimators were measuring the same quantity from identical data we would not anticipate such variation; however, this is not expected in our scenario since, due to the finite truncation radius HIPSTER only ‘sees’ a subset of all particle pairs, and thus will exhibit different noise properties. The fluctuations in this ratio are amplified for larger $\ell$ and smaller $R_0$, which follows by noting that (a) the higher multipole measurements carry an intrinsically larger variance (which scales as $2\ell + 1$) and (b) modified pair-counts with larger $R_0$ ‘see’ more of the survey, and thus are expected to be less variable. The full covariance of the measurements is discussed below.

9.2.2 Joining Estimating $P_\ell(k)$ on All Scales

Whilst we have shown configuration-space estimators to be accurate and unbiased estimators of the galaxy power spectra on small-scales, we do not anticipate that they will replace Fourier-transform methods, primarily due to the large $R_0$ that would be needed for unbiased measurements of the low-$k$ power. However, given that FFT-based methods are optimal at small $k$,
Figure 3. Ratio of power spectra computed from the configuration-space HIPSTER and FFT-based nbodykit codes, for the monopole (upper) and quadrupole (lower) power. Left and right panels show the ratio computed using HIPSTER pair count truncation widths of $R_0 = 50h^{-1}\text{Mpc}$ and $100h^{-1}\text{Mpc}$ respectively. For each panel, we plot the ratio obtained using 512$^3$ (green) and 1024$^3$ (red) FFT grids showing the mean and standard error in the mean across 100 mocks. The blue curves show the individual ratios from the 100 mocks.

These spectra are shown in full in Fig. 1, and we note large amplitude oscillations at low $k$ due to the pair-separation window, though this is negligible at large $R_0$ and $k$.

and configuration-space estimators perform best at large $k$, a natural approach would be combine the two techniques, allowing accurate spectra to computed across a wide range of wavenumbers, without the problems of aliasing or shot-noise (from the FFTs) and convolution-distortions (from HIPSTER). Further, this allows us to use small $N_{\text{grid}}$ and small $R_0$, giving an efficient computation both in terms of time and memory allocation.

To demonstrate this, we measure the power spectrum multipoles, $P_\ell(k)$, from a single Quijote simulation, this time using the full distribution of 512$^3$ matter particles in redshift-space at $z = 0$. This is done for a set of 150 $k$-bins in the range [0, 10] $h^{-1}\text{Mpc}$, using 100 linearly spaced bins for $k < 1 h^{-1}\text{Mpc}$ and 50 logarithmically spaced bins for larger $k$. For $k < 1 h^{-1}\text{Mpc}$, we compute the spectra via FFTs using nbodykit with $N_{\text{grid}} = 512$ (which Fig. 3 has shown to be percent-level accurate in this regime) and for $k > 1 h^{-1}\text{Mpc}$, HIPSTER is used, truncating at $R_0 = 30 h^{-1}\text{Mpc}$ for speed, which will have negligible impact on the high-$k$ power. We further subsample the HIPSTER particle catalog by 100 times to keep computation times manageable.\(^{13}\)

Fig. 2 shows the resulting spectra; there is no discernable difference between the FFT spectra at $k < 1 h^{-1}\text{Mpc}$ and the HIPSTER spectra at $k > 1 h^{-1}\text{Mpc}$, giving an efficiently computed spectrum across a broad region in $k$-space. However, whilst the measurements themselves are consistent, there is a slight change in the variances (which are computed using the formulae of Sec. 6 for the FFT and HIPSTER sections separately) across $k = 1 h^{-1}\text{Mpc}$. This is primarily attributed to the switch from linear to logarithmic bins across $k = 1 h^{-1}\text{Mpc}$, as well as the addition of subsampling in HIPSTER, which boosts the Poissonian noise. If one had used a binning scheme that varied smoothly across the boundary and a smaller subsampling ratio, we would not expect to observe this effect.

With the above choice of binning and hyperparameters, both codes take a few core-hours to compute their regions of the

\(^{13}\) From Sec. 6, we note that this will affect only the covariance of the shot-noise-induced 2- and 3-point terms. Since the number of particles is large, these terms are inherently small, thus this subsampling does not have a huge impact on the measured covariances.
Recall that subsampling by \( f_{\text{sub}} \) corresponds to using a random subset of \( 1/f_{\text{sub}} \) of the dataset, and thus decreasing the number density by a factor \( f_{\text{sub}} \).
9.4 Simulated Bispectra

9.4.1 Comparison of Methods

To test the validity of our new bispectrum algorithm, HIPSTER, we follow a similar procedure to the above, computing the isotropic bispectra from both conventional FFT-based approaches and HIPSTER. Since the computational effort involved in computing bispectra usually significantly exceeds that of power spectra, we consider only a single redshift-space halo catalog here, containing \(\sim 4 \times 10^5\) particles. The HIPSTER bispectra are computed using the algorithm discussed in Sec. 4, with a total of 40 linearly spaced \(k\)-bins up to \(k_{\text{max}} = 0.8 h^{-1}\text{Mpc}^{-1}\). We use the first seven Legendre multipoles \(\ell = 0 \rightarrow 5\) to \(\ell = 6\), giving a total of 5740 non-trivial bispectrum bins in \((k_1, k_2, \ell)\)-space. In terms of hyperparameters, we adopt a ratio of randoms-to-galaxies of \(f_{\text{rand}} = 3\) (found to be sufficient in initial testing) and truncate particle pair counts at \(R_0 = 100 h^{-1}\text{Mpc}\) which requires \sim 20 core-hours to compute with no data subsampling. (An analogous computation with \(R_0 = 50 h^{-1}\text{Mpc}\) takes only \sim 3 core-hours on the same machine.)

For the conventional FFT-based calculation, we use the routines provided in the Pylians code (Villaescusa-Navarro 2018), which is based on Watsonkinson et al. (2017), involving loops of forward and reverse FFTs. We refer the reader to Schusatti (2005), Jeong (2010) and Watsonkinson et al. (2017) for a full discussion of this procedure. FFT-based algorithms conventionally compute \(B(k_1, k_2, k_3)\) (where \(k_1, k_2, k_3\) satisfy triangle inequalities) or \(B(k_1, k_2, \mu_{12})\), where \(\mu_{12} = \vec{k}_1 \cdot \vec{k}_2\), whilst our estimators require \(B_\ell(k_1, k_2)\) (to make use of spherical harmonic decompositions and to aid visualization); conversion between the two is non-trivial, but possible via the integral

\[
B_\ell(k_1, k_2) = \frac{2\ell + 1}{2} \int_{-1}^{1} \frac{d\mu_{12}}{2\pi} B(k_1, k_2, \mu_{12}) L_\ell(\mu_{12}) = \frac{2\ell + 1}{2} \sum_{i=0}^{N} B(k_1, k_2, \mu_i) L_\ell(\mu_i) w_i.
\]

(9.1)

Here, we have approximated the integral via Gauss-Legendre quadrature, which defines a set of \(N + 1\) values of \(\mu_i \in [-1, 1]\) and corresponding weights \(w_i\) from properties of Legendre polynomials and their derivatives. Here, we use tenth order quadrature which gives highly accurate results, requiring only 11 \(\mu\)-bins to be computed for each \(k_1, k_2\) pair. (Note that each \(k_1, k_2\) pair can be computed independently, allowing for efficient parallelization.) As with the power spectrum, the FFT-based bispectrum estimator naturally picks up a significant shot-noise term, which, in periodic simulations, has the expected form

\[
B(k_1, k_2, k_3) = \frac{1}{n} [P(k_1) + P(k_2) + P(k_3)] + \frac{1}{n^2},
\]

(9.2)

for a number with simulation density \(n\) (e.g. Chan & Blot 2017). Here, \(P(k)\) are the one-dimensional power spectra which are computed alongside the bispectra in the Pylians code. We note that this term is not present in configuration-space approaches. Due to the large number of FFTs that must be performed to compute the bispectrum in all combinations of bins, the process requires significant computation time; \sim 200 core-hours using \(N_{\text{rand}} = 512\) FFTs. For a given \(k_1, k_2\) pair, computation of the \(B_\ell(k_1, k_2)\) multipoles will involve a sum over triangles with \(k_3 \in |[k_1 - k_2], k_1 + k_2|\), thus to avoid the Nyquist limit, we require \(k_{\text{max}} < k_{\text{Nyq}}/2\), justifying our choice of \(k_{\text{max}}\).

Given that HIPSTER contains different assumptions to a standard FFT-based approach due to the window function (which leads to \(k\)-space convolutions and smoother results), it is useful to compare the results to an independent method that also implements a window function. To this end, we first compute the 3PCF multipoles \(\xi^{ij}(r_1, r_2)\) via the \(O(N^2)\) algorithm of Slepian & Eisenstein (2015), then estimate the bispectrum via a numerical Fourier-space integral with the window function inserted. This gives the matrix product:

\[
B_\ell^{3\text{PCF}}(k_1, k_2) \equiv (4\pi)^2(-1)^\ell \sum_i \left[ x_i^2 \Delta x_i W(x_i; R_0) \right] \sum_j \left[ x_j^2 \Delta x_j W(x_j; R_0) \right] \xi^{ij}_\ell,
\]

(9.3)

where \(\xi^{ij}_\ell\) is the 3PCF estimate in bins \(i, j\) centered at \(x_i, x_j\) and the summations are over all bins with \(x_i \leq R_0\). For simplicity we do not apply \(k\)-space binning (assuming \(B_{\ell}^{ab} \approx B_\ell(k_a, k_b)\) where \(k_2\) is the central value of bin \(x_i\)). Practically, we compute the 3PCF \(RRR\) term analytically and mask the \(i = j\) diagonal of \(\xi^{ij}_\ell\), since this has additional shot-noise contributions in the algorithm. Via this method, the bispectrum is computed in \sim 10 core-hours. Since the estimator contains a 3PCF discretized into \(1 h^{-1}\text{Mpc}\) bins, we expect that the resulting spectrum will be inaccurate for \(k \gtrsim 1 h^{-1}\text{Mpc}\) due to aliasing. This may be thought of as a discretized limit of HIPSTER; in Eq. 9.3 we compute the bispectrum by counting pairs of particles in given bins, before performing the Fourier integral, whilst in HIPSTER we perform the Fourier integral directly, effectively using infinite configuration-space bins. Whilst the 3PCF estimator asymptotically equals the HIPSTER algorithm in the limit of infinite 3PCF bins, this would require infinite memory, thus it is advantageous to perform the Fourier integral directly, as done by HIPSTER.

Fig. 5 compares the bispectra computed from the three methods. Firstly, we note good agreement between the spectra computed with HIPSTER and via numerical integration of the 3PCF across the range of \(k\)-bins and multipoles \(\ell\) tested, implying that the configuration-space algorithm has been implemented successfully. Both methods include the same double convolution by the pair-separation window function (of width \(R_0 = 100 h^{-1}\text{Mpc}\)), as in Eq. 5.4. A more meaningful comparison is thus between the top and bottom plots of Fig. 5, comparing the configuration-space and FFT-based approaches. We make two main observations: (1) the \(k\) and \(\ell\) dependencies of the bispectrum are comparable between the two approaches, especially at
The first indicates that HIPSTER is measuring the bispectrum correctly; though at small $k$, we expect small deviations from the convolution window (which are explored further in the subsequent section) and at large $k$ (roughly $k_1 + k_2 \gtrsim k_{NYQ}/2 = 0.8 \text{ h Mpc}^{-1}$, FFT aliasing starts to have an impact. Secondly, the different noise properties can be explained by noting that (a) the window convolution gives some smoothing between adjacent bins (primarily on scales $\Delta k \approx 3/R_0$) that will reduce stochasticity and (b) configuration-space estimators are simply sums over smooth functions (with random arguments), and thus expected to be smooth. Averaging over a number of simulations would significantly reduce the sampling noise, yet this is computationally expensive for FFT-type estimators. Based on the above, we conclude that the HIPSTER bispectrum estimator gives unbiased estimates of the true power spectra, surpassing conventional methods at moderate $k$-scales and above.

9.4.2 Measuring the Bispectrum up to large $k$

With this in hand, we proceed to evaluate the bispectrum up to large $k$ using our configuration-space algorithm, implemented in HIPSTER. For this, we use 100 halo catalogs from the same simulations as before, this time using a mass cut of $M_{\text{min}} = 3.1 \times 10^{13} \text{ M}_\odot$ to represent a more realistic galaxy sample. Since the isotropic bispectrum considered in this paper is insensitive to redshift-space distortions, we work in real-space for this test, though this will not affect the results. In the top panel of Fig. 6, the bispectrum is plotted using 60 $k$-bins up to $k = 3 \text{ h Mpc}^{-1}$ adopting a truncation radius of $R_0 = 100 \text{ h}^{-1} \text{ Mpc}$. This computation takes ~5 core-hours on a modern machine and we note that an analogous computation using FFTs would require $N_{\text{grid}} \gtrsim 3800$ (setting $|k_1 + k_2| < k_{NYQ}/2$ to avoid aliasing) and ~5000 core-hours, which is computationally infeasible to run on a large number of simulations.

In the lower panel of Fig. 6, we plot the difference in the bispectra obtained from $R_0 = 50 \text{ h}^{-1} \text{ Mpc}$ and $100 \text{ h}^{-1} \text{ Mpc}$. We first note that this is everywhere small compared to the bispectrum, indicating that accurate (and ~8 times faster) measurements can be obtained using the smaller radius. The first clear difference between the spectra is seen in the horizontal and vertical ‘stripes’ at low $k$, which can be attributed to the window function convolution causing oscillations in the bispectra around the $R_0 \rightarrow \infty$ limit. (Due to the sharply-peaked nature of the function in $k$-space, this is negligible for if all wavenumbers are greater than ~0.38 $\text{ Mpc}^{-1}$). Further, we observe a clear diagonal bias in the multipoles; this arises from the same effects, since the diagonal $k_1 \approx k_2$ terms necessarily have contributions from $(k_1, k_2, k_3)$, i.e. modes with small $k_3$. If the bispectra were plotted in the full $(k_1, k_2, k_3)$-plane we would not observe this behavior away from the origin. Furthermore, we note increasing stochastic variations between the bispectra at large $k$. This can be explained by noting that (a) the two bispectra ‘see’ a slightly different...
density field due to the different choices of $R_0$, and (b), the fractional error in the bispectrum scales as $(2\ell + 1)/(k_1 k_2)$. In the normalization factor used in Fig. 6, we thus expect increasing statistical error with $k$.

10 SUMMARY AND OUTLOOK

In this work, we have developed and implemented $O(N^2)$ algorithms to compute the multipoles of the power spectrum and isotropic bispectrum that are optimized for cosmological simulations and highly efficient on small scales. This builds upon the work of Philcox & Eisenstein (2020), which introduced the technique of computing spectra via configuration-space weighted multipoles computed by counting all particles up to $R_0 = 100 h^{-1}\text{Mpc}$ (top) and the difference between the bispectra at $R_0 = 100 h^{-1}\text{Mpc}$ and $R_0 = 50 h^{-1}\text{Mpc}$ (bottom, multiplied by 20 for visibility). This uses the average bispectrum across 100 halo catalogs in real-space, each containing $\sim 10^5$ halos at $z = 0$, subject to the selection function $M_{\text{min}} > 3.1 \times 10^{13} h^{-1} M_\odot$. All spectra are computed using the HIPSTER code on a 20-core machine, with a run-time of $\sim 1$ and 5 core-hours for $R_0 = 50 h^{-1}\text{Mpc}$ and $100 h^{-1}\text{Mpc}$ respectively. Note that we normalize by an additional factor of $(2\ell + 1)$ compared to Fig. 5. The differences between the bispectra at the two truncation radii are well-understood, and discussed in the text.

The main benefits of our estimators, which are publicly available in the HIPSTER package, are as follows:

- **Speed**: Unlike Fourier-transform methods, the runtime of the configuration-space estimators decreases with the modulus of the wavenumber considered, since a smaller truncation scale $R_0$ can be used. Furthermore, a bispectrum measurement requires only two $O(N^2)$ pair counts, rather than repeated Fourier transforms. Accurate galaxy power spectrum and bispectrum measurements are possible down to small scales ($k > 1 h \text{Mpc}^{-1}$) in minutes using HIPSTER, and the runtime can be easily reduced by subsampling the data.
- **Memory**: On small scales, Fourier-transform methods require a fine grid, whose memory scales as $N_{\text{grid}}^3$, thus small-scale spectral computation must be performed on high-memory nodes. (As an example, consider $N_{\text{grid}} = 1024$. At single precision, this requires almost 9 GB of memory per Fourier-space grid.) Since HIPSTER carries only the initial particles and arrays containing the output spectra, its memory usage is significantly less, and does not depend on scale.
- **Aliasing**: Since the estimators do not require gridding, the corresponding spectra do not suffer from aliasing. There is hence no notion of a Nyquist frequency, and the spectra can be computed up to arbitrarily large $k$.
- **Accuracy**: HIPSTER has been rigorously tested and found to be in sub-percent agreement with FFT-based methods. At low $k$, we observe effects from the truncation of the pair-counts at $R_0$, but these are negligible for $k R_0 \gtrsim 3$.
- **Shot-noise**: Since we work in configuration-space, we can easily exclude self-counts from the estimators. This avoids having to remove shot-noise in post-processing which is non-trivial for the bispectrum.

Whilst HIPSTER is excellent for measuring small-scale spectra, it is suboptimal on the largest scales since a great number of memory per Fourier-space grid.) Since HIPSTER carries only the initial particles and arrays containing the output spectra, its memory usage is significantly less, and does not depend on scale.
of particle pairs must be counted. By combining FFT-based estimators with HIPSTER we can measure spectra efficiently on all scales, using both a small \( N \) grid and low truncation radius \( R_0 \). This combined method requires orders of magnitude less computational resources than using either method alone. An additional application is to scale-free simulations, since HIPSTER is not limited by the fundamental frequency of the box, thus we are free to choose any desired \( k \)-space binning.

A number of extensions may be possible. An easily achievable one is the computation of the anisotropic bispectrum, which includes the effects of redshift-space distortions (Scoccimarro 2015; Slepian & Eisenstein 2018; Sugiyama et al. 2019b). Whilst this increases the dimensionality of the space by two (as we must parametrize the orientation of the redshift-space triangle with respect to the line-of-sight), it can be computed using a very similar algorithm requiring negligible additional computation time. In essence, this corresponds to replacing the sum over spherical harmonics in the bispectrum estimators with a product of two harmonics of different order, following Slepian & Eisenstein (2018). Furthermore, whilst we have focused on periodic simulations in this work, it is possible to compute the bispectrum in \( O(N^2) \) time for arbitrary survey geometries, which simply requires abandonment of the analytic random integrals discussed above and inclusion of sets of random particles, as done for the power spectrum in Philcox & Eisenstein (2020). We ought also to consider the bispectrum covariance in greater detail (including redshift-space effects, non-Gaussianity and the pair-separation window function), which, whilst algebraically difficult, is not intractable. We defer such considerations to future work.

REFERENCES

Alcock C., Paczynski B., 1979, Nature, 281, 358
Anderson L., et al., 2014, MNRAS, 441, 24
Arfken G., Weber H., Harris F., 2013
Assassi V., Simonović M., Zaldarriaga M., 2017, J. Cosmology Astropart. Phys., 2017, 054
Beutler F., et al., 2017, MNRAS, 464, 3409
Bianchi D., Gil-Marín H., Ruggeri R., Percival W. J., 2015, MNRAS, 453, L11
Blake C., et al., 2011, MNRAS, 415, 2876
Bloomfield J. K., Face S. H. P., Moss Z., 2017, arXiv e-prints, p. arXiv:1703.06428
Chan K. C., Blot L., 2017, Phys. Rev. D, 96, 023528
D’Amico G., Gleyzes J., Kokron N., Markovic D., Senatore L., Zhang P., Beutler F., Gil-Marín H., 2019, arXiv e-prints, p. arXiv:1909.05271
Davis M., Efstathiou G., Frenk C. S., White S. D. M., 1985, ApJ, 292, 371
DeRose J., et al., 2019, ApJ, 875, 69
Eisenstein D. J., et al., 2005, ApJ, 633, 560
Fabrikant V., 2013, Quarterly of Applied Mathematics, 71
Feldman H. A., Kaiser N., Peacock J. A., 1994, ApJ, 426, 23
Ferguson J. R., Regan D. M., Sheddard E. P. S., 2012, Phys. Rev. D, 86, 063511
Fonseca de la Bella L., Regan D., Seedy D., Hotchkiss S., 2017, J. Cosmology Astropart. Phys., 2017, 039
Fonseca de la Bella L., Regan D., Seedy D., Parkinson D., 2018, arXiv e-prints, p. arXiv:1805.12394
Garrison L. H., Eisenstein D. J., Ferrer D., Tinker J. L., Pinto P. A., Weinberg D. H., 2018, ApJS, 236, 43
Gil-Marín H., Noreña J., Verde L., Percival W. J., Wagner C., Manera M., Schneider D. P., 2015a, MNRAS, 451, 539
Gil-Marín H., et al., 2015b, MNRAS, 452, 1914
Gil-Marín H., et al., 2016, MNRAS, 460, 4210
Gradshteyn I. S., Ryzhik I. M., Jeffrey A., Zwillinger D., 2007, Table of Integrals, Series, and Products
Hamana T., et al., 2020, PASJ, 72, 16
Hand N., Li Y., Slepian Z., Seljak U., 2017, J. Cosmology Astropart. Phys., 2017, 002
Hand N., Feng Y., Beutler F., Li Y., Modí C., Seljak U., Slepian Z., 2018, AJ, 156, 160
Hikage C., et al., 2019, PASJ, 71, 43
Hung J., Ferguson J. R., Sheddard E. P. S., 2019, arXiv e-prints, p. arXiv:1902.01830
Ivanov M. M., Simonović M., Zaldarriaga M., 2019a, arXiv e-prints, p. arXiv:1909.05277
Ivanov M. M., Simonović M., Zaldarriaga M., 2019b, arXiv e-prints, p. arXiv:1912.08208
Jeong D., 2010, PhD thesis, University of Texas at Austin
Jing Y. P., Börner G., 2001, MNRAS, 325, 1389
Kaiser N., 1987, MNRAS, 227, 1
Landy S. D., Szalay A. S., 1993, ApJ, 412, 64
Li Y., Hu W., Takada M., 2014, Phys. Rev. D, 89, 083519
Li Z., Jing Y. P., Zhang P., Cheng D., 2016, ApJ, 833, 287
Li Y., Schmittfull M., Seljak U., 2018, J. Cosmology Astropart. Phys., 2018, 022
Li Y., Singh S., Yu B., Feng Y., Seljak U., 2019, J. Cosmology Astropart. Phys., 2019, 016
McAlpine S., et al., 2016, Astronomy and Computing, 15, 72
McCarty I. G., Schaye J., Bird S., Le Brun A. M. C., 2017, MNRAS, 465, 2936
Mehrem R., 2009, arXiv e-prints, p. arXiv:0909.0494
Mehrem R., Hohenegger A., 2010, Journal of Physics A Mathematical General, 43, 455204
Miyatake H., et al., 2015, ApJ, 806, 1
More S., Miyatake H., Mandelbaum R., Takada M., Spergel D. N., Brownstein J. R., Schneider D. P., 2015, ApJ, 806, 2
NIST DLMF

Efficient Power Spectrum and Bispectrum Estimators
ACKNOWLEDGEMENTS

OHEP thanks Daniel Eisenstein, Lehman Garrison, David Spergel, Francisco Villaescusa-Navarro and, in particular, Zachary Slepian for insightful conversations and feedback on the manuscript. OHEP acknowledges funding from the Herchel-Smith foundation.

APPENDIX A: AN ANALYTIC FORM FOR THE $E^II_\ell$ KERNEL

Here we consider the $E^II_\ell$ kernel defined in Eq. 4.14, which allows the bispectrum estimator (in particular the $\hat{D}\hat{D}\hat{R}^II_\ell$ term) to be written without any use of random particles. First, we write

$$E^II_\ell(x_i, x_j, k_1, k_2; R_0) = \int dx_3 J_\ell(k_1|x_3|)J_\ell(k_2|x_3|)W(x_3, R_0)W(x_3, R_0)\hat{L}_{\ell}(\hat{x}_3 \cdot \hat{x}_j)$$

$$= \frac{4\pi}{2\ell + 1} \int dx_3 \sum_{m=-\ell}^{\ell} J_\ell(k_1|x_3|)J_\ell(k_2|x_3|)W(x_3, R_0)W(x_3, R_0)Y^*_{\ell m}(\hat{x}_3)Y_{\ell m}(\hat{x}_j)$$

$$= \frac{4\pi}{2\ell + 1} \int dx_3 f_{\ell m}(x_i - x_3;k_1)f_{\ell m}(x_3 - x_j;k_2).$$

MNRAS 000, 1–32 (2020)
where we have used the spherical harmonic addition theorem in the second line (NIST DLMF, Eq. 14.30.9) and defined the functions \( f_{lm} \). We can express \( f_{lm} \) in terms of its Fourier counterpart;

\[
E^{ll}_l(x_i, x_j, k_1, k_2; R_0) = \frac{4\pi}{2l+1} \sum_{m=-l}^{l} \int d\Omega_3 \frac{dp_1}{(2\pi)^3} \frac{dp_2}{(2\pi)^3} \tilde{f}_{lm}(p_1; k_1) f_{lm}(p_2; k_2) e^{i(p_1 \cdot x_i) e^{i(p_2 \cdot x_j)}} \tag{A2}
\]

where we have integrated over \( x_3 \) and the resulting Dirac delta function. Note that this result may be similarly obtained using the convolution theorem. To proceed we require expressions for \( \tilde{f}_{lm} \):

\[
\tilde{f}_{lm}(p; k) = \int dr e^{i\mathbf{p} \cdot \mathbf{r}} j_l(k|\mathbf{r}|) W(|\mathbf{r}|; R_0) Y_{lm} (\hat{\mathbf{r}}) \tag{A3}
\]

\[
\tilde{f}_{lm}(p; k) = \sum_{L=0}^{\infty} \sum_{M=-L}^{L} 4\pi (-i)^L \int dr j_L(p|\mathbf{r}|) Y_{LM}^*(\hat{\mathbf{r}}) j_l(k|\mathbf{r}|) W(|\mathbf{r}|; R_0) Y_{LM} (\hat{\mathbf{r}})
\]

Note that \( \sum_{m} Y_{LM}(\hat{\mathbf{p}}) Y_{LM}^*(\mathbf{p}) = (2L+1)/(4\pi) \) and that this is simply an inverse Fourier transform of a spherical function. The resulting kernel depends only on the distance \( |x_j| \), and, if \( \omega_{ij}(p; k, R_0) \) is known, reduces to a one-dimensional integral (of a large number of) sets of Bessel functions and polynomials. Numerical integration is not too difficult, since \( \omega_\ell \) depend only on the \( k \)-bin and multipole used, of which there are a finite number. For speed, the kernel should be pre-computed for each set of \( k \)-bins and \( \ell \) for an array of \( |x_j| \) values and interpolated when needed. This is helped by the finite domain; \( E^{ll}_l(x_i, x_j) = 0 \) for all \( |x_i - x_j| \geq 2R_0 \). An alternative (but equivalent) expression for \( E^{ll}_l \) is given in Appendix B.

It is instructive to consider the special case of \( R_0 \to \infty \), practically corresponding to a pair-count over all pairs in the survey. Here \( W(r) \) is unity for all \( r \) and \( \omega_\ell \) simplifies to

\[
\lim_{R_0 \to \infty} \omega_{ij}(p; k, R_0) = 4\pi \int_0^{\infty} r^2 dr j_L(p|kr|) = \frac{2\pi^2}{kp} \delta_D(k - p), \tag{A5}
\]

by the closure relation for spherical Bessel functions.\(^{16}\) Inserting this into the expression for \( E^{ll}_l \) gives

\[
\lim_{R_0 \to \infty} E^{ll}_l(x_i - x_j, k_1, k_2) = \frac{p^2 dp}{2\pi^2} j_0(p|x_j|) \frac{2\pi^2}{k_1 k_2} \delta_D(k_1 - p) \delta_D(k_2 - p) = \frac{2\pi^2}{k_1 k_2} \delta_D(k_1 - k_2) j_0(k_1|x_j|). \tag{A6}
\]

Note that this (a) only contributes when \( k_1 \approx k_2 \) and (b) has no dependence on \( \ell \). For finite \( R_0 \), we thus expect the kernel to have little off-diagonal power and similar forms for each multipole moment (although we expect deviations as \( k \to 0 \), since this is where the pair-count truncation is most important).

When \( k \)-space binning is included, we obtain

\[
E^{ll}_{l,ab}(r; R_0) = \int \frac{p^2 dp}{2\pi^2} j_0(p) \omega^I_{l}(p; R_0) \omega^I_{l}(p; R_0), \tag{A7}
\]

where \( \omega^I_{l} \) is defined in Eq. (6.42), leading to

\[
\tilde{DDR}^{ll}_{l,ab}(R_0) = n(-1)^\ell (2\ell + 1) \sum_{i \neq j} E^{ll}_{l,ab}(x_i - x_j; R_0). \tag{A8}
\]

Using this approach, we remove all dependencies on a random catalog, though practically computation time is slow due to (a) compute a highly non-trivial analytic form for \( E^{ll}_l \) or (b) store and call a large number of interpolators to evaluate \( E^{ll}_l \).

\(^{16}\) Note that \( \delta_D(k - k')/k^2 \) is simply the radial part of \( \delta_D(k + k') \).
APPENDIX B: ALTERNATIVE FORM FOR $\mathcal{E}_f^{II}$

We here derive an alternative, but equivalent form for the second DDR kernel, $\mathcal{E}_f^{II}$. Starting from Eq. A4 and inserting the definition of $\omega_f$ (Eq. 6.19) we obtain

$$\mathcal{E}_f^{II}(\mathbf{x}_i - \mathbf{x}_j, k_1, k_2; R_0) = 8 \int_0^\infty p^2 dp \int_0^{R_0} r_1^2 dr_1 j_\ell(p r_1) j_\ell(k_1 r_1) W(r_1; R_0) \left[ \int_0^{R_0} r_2^2 dr_2 j_\ell(p r_2) j_\ell(k_2 r_2) W(r_2; R_0) \right].$$

(B1)

Note that this contains an integral over all $p$ of the product of three spherical Bessel functions, which can be rewritten using the relation

$$\int_0^\infty p^2 dp \int_0^{R_0} r_1^2 dr_1 j_\ell(p r_1) j_\ell(k_1 r_1) j_\ell(k_2 r_1) W(r_1; R_0) = \frac{\pi \beta(\Delta)}{4 |k_j| r_j^2} L_\ell(\Delta),$$

(B2)

for $\Delta = (r_1^2 + r_2^2 - |x_j|^2)/(2r_1r_2)$. Legendre polynomial $L_\ell$ and

$$\beta(\Delta) = \begin{cases} \frac{1}{2} & \Delta = \pm 1 \\ 1 & -1 < \Delta < 1 \\ 0 & \text{else} \end{cases}$$

(B3)

(Mehrem 2009; Mehrem & Hohenegger 2010; Fonseca de la Bella et al. 2017, a special case of Gradshteyn et al. 2007, Eq. 6.578.8). Inserting this relation gives

$$\mathcal{E}_f^{II}(\mathbf{x}_i - \mathbf{x}_j, k_1, k_2; R_0) = 2 \pi \int_0^{R_0} r_1 dr_1 \int_0^{R_0} r_2 dr_2 \frac{r_1 r_2}{|k_j|} \beta(\Delta) j_\ell(k_1 r_1) j_\ell(k_2 r_2) W(r_1; R_0) W(r_2; R_0) L_\ell(\Delta),$$

(B4)

simplifying the kernel to a two-dimensional integral, which must be evaluated numerically. Including $k$-space binning in bins $a$ and $b$, this becomes

$$\mathcal{E}_f^{II,ab}(\mathbf{x}_i - \mathbf{x}_j; R_0) = 2 \pi \int_0^{R_0} r_1 dr_1 \int_0^{R_0} r_2 dr_2 \frac{r_1 r_2}{|k_j|} \beta(\Delta) j_\ell^a(r_1) j_\ell^b(r_2) W(r_1; R_0) W(r_2; R_0) L_\ell(\Delta),$$

(B5)

with $j_\ell^a$ defined in Eq. 3.7. This can be simplified somewhat further by consideration of the function $\beta(\Delta)$, which is non-zero only for $|r_1 - r_2| \leq |x_j| \leq r_1 + r_2$, giving:

$$\mathcal{E}_f^{II,ab}(\mathbf{x}_i - \mathbf{x}_j; R_0) = \begin{cases} 2 \pi \int_0^{R_0} r_1 dr_1 \int_0^{r_1} r_2 dr_2 \frac{r_1 r_2}{|k_j|} j_\ell^a(r_1) j_\ell^b(r_2) W(r_1; R_0) W(r_2; R_0) L_\ell(\Delta) & \text{if } x_{ij} \leq R_0 \\ 2 \pi \int_0^{r_1} r_1 dr_1 \int_0^{r_2 - x_{ij}} r_2 dr_2 \frac{r_1 r_2}{|k_j|} j_\ell^a(r_1) j_\ell^b(r_2) W(r_1; R_0) W(r_2; R_0) L_\ell(\Delta) & \text{else.} \end{cases}$$

(B6)

APPENDIX C: RECURSIVE FORM FOR $D_f(U)$

Below, we derive an analytic form for the $D_f(u)$ functions appearing in Eq. 3.7. Starting from the definition of $D_f(u)$ as an indefinite integral,

$$D_f(u) \equiv \int u^2 du j_\ell(u)$$

(C1)

we note that $j_\ell(u)$ satisfies the Sturm-Liouville equation for integer $\ell$:

$$- \frac{d}{du} \left( u^2 j_\ell(u) \right) + (\ell + 1) j_\ell(u) = u^2 j_\ell(u).$$

(C2)

(NIST DLMF, Eq. 10.47.1) and hence

$$D_f(u) = -u^2 j_\ell'(u) + (\ell + 1) \int du j_\ell(u) = u^2 j_{\ell+1}(u) - \ell u j_\ell(u) + (\ell + 1) \int du j_\ell(u),$$

(C3)

neglecting an arbitrary constant of integration and inserting a recursion relation for the Bessel function derivative (NIST DLMF, Eq. 10.51.2). Next, we define

$$I_f(u) = \int du j_\ell(u) = \frac{1}{\ell} \int \left[ (\ell - 1) j_{\ell-2}(u) - (2\ell - 1) j_{\ell-1}(u) \right],$$

(C4)

using an additional recursion relation (Eq. 10.51.1 NIST DLMF) to substitute for $j_\ell(u)$. This yields the recursive definition

$$D_f(u) = u^2 j_{\ell+1}(u) - \ell u j_\ell(u) + (\ell + 1) I_f(u)$$

$$I_f(u) = (\ell - 1) I_{\ell-2}(u) - (2\ell - 1) I_{\ell-1}(u)$$

(C5)

for $\ell \geq 2$ with standard results $I_0(u) = \text{Si}(u)$, $I_1(u) = -j_0(u)$, where $\text{Si}(u)$ is the Sine integral.

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