`PowerI4`: a simple and accurate power spectrum estimator for cosmological applications

Emiliano Sefusatti, Martin Crocce

October 3, 2017

Contents

1 Introduction

`PowerI4` is an FFT-based code to measure the power spectrum of a distribution of particles in a box implementing up to 4^{th}-order grid interpolation and an interlacing technique to reduce the aliasing contribution arising from the sampling of the density field. The Fourier-space density, once corrected for mass assignment window effects can also be saved and used to compute higher-order correlation functions such as the bispectrum.

A quantitative evaluation of aliasing effects on the estimate of the Fourier-space density field as of the power spectrum, and their relation to the mass assignment scheme adopted, can be found in Sefusatti et al. (2016). We refer the reader to that paper for a presentation of several numerical tests performed on the `PowerI4` outputs and for comparisons with other codes. Please cite that reference if you use `PowerI4` in your research.

`PowerI4` is based on a previous version by Roman Scoccimarro which, in turn, used algorithms developed by Hugh Couchmann.

2 Aliasing and interlacing

This section provides a basic theoretical background for the algorithms implemented in the code. We refer to reader to Sefusatti et al. (2016) and references therein for further details and test results.

2.1 Definitions

We focus our attention on a continuous fields defined over a finite, cubic volume $V = L^3$ which we will assume equivalent to the case of infinite volume but periodic with period L along each of the three spatial directions (the extension to the general case of a generic box or different periods while trivial and allowed by the code, would require a significantly more cumbersome notation).

The code provides, in the first place, an estimate of the Fourier-space density contrast of a distribution of N_P particles of equal mass m, corresponding to a discrete realisation of a continuous density field $\rho(\mathbf{x})$, so that

$$\rho(\mathbf{x}) = \sum_{j=1}^{N_P} m\, \delta_D(\mathbf{x} - \mathbf{x}_j)\,, \tag{1}$$

with $\delta_D(\mathbf{x})$ being a Dirac delta function. The relative dimensionless density contrast is defined, in configuration space, as

$$\delta(\mathbf{x}) \equiv \frac{\rho(\mathbf{x})}{\bar{\rho}} - 1\,, \tag{2}$$

where $\bar{\rho}$ is the mean density in the volume V, so that

$$\delta(\mathbf{x}) = \frac{1}{\bar{n}_P} \sum_{j=1}^{N_P} \delta_D(\mathbf{x} - \mathbf{x}_j) - 1\,, \tag{3}$$

with $\bar{n}_P = N_P/V$ being the particle density.

The Fourier transform of the density contrast is defined, in our convention, as

$$\widetilde{\delta}(\mathbf{k}) \equiv \int_V \frac{d^3x}{(2\pi)^3}\, e^{-i\mathbf{k}\cdot\mathbf{x}}\, \delta(\mathbf{x})\,, \tag{4}$$

with the inverse given by the series

$$\delta(\mathbf{x}) \equiv k_f^3 \sum_{\mathbf{k}} e^{i\mathbf{k}\cdot\mathbf{x}}\, \widetilde{\delta}(\mathbf{k})\,, \tag{5}$$

since $\mathbf{k}$ is discretised in multiples of the fundamental frequency $k_f = 2\pi/L$ in each spatial direction. We will always explicitly show the argument of each function in addition to identify Fourier-space objects with a tilde.[1]

The power spectrum $P(k)$ is then defined as the two-point function of $\delta(\mathbf{k})$, that is

$$\langle \delta(\mathbf{k}_1)\, \delta(\mathbf{k}_2) \rangle \equiv \frac{\delta^K_{\mathbf{k}_{12}}}{k_f^3}\, P(k_1)\,, \tag{6}$$

with $\langle \ldots \rangle$ representing the ensemble average, $\mathbf{k}_{i_1,\ldots,i_n} \equiv \mathbf{k}_{i_1} + \ldots + \mathbf{k}_{i_n}$ and $\delta^K_{\mathbf{k}}$ being a Kronecker delta defined in such a way to be equal to 1 for a vanishing argument $\mathbf{k} = 0$, zero otherwise . Similarly, the bispectrum $B(k_1, k_2, k_3)$ is defined as

$$\langle \delta(\mathbf{k}_1)\, \delta(\mathbf{k}_2)\, \delta(\mathbf{k}_3) \rangle \equiv \frac{\delta^K_{\mathbf{k}_{123}}}{k_f^3}\, B(k_1, k_2, k_3)\,. \tag{7}$$

In this convention the continuous limit is recovered for $V \to \infty$ or $k_f \to 0$ with $\delta^K_{\mathbf{k}}/k_f^3 \to \delta_D(\mathbf{k})$, the latter being a Dirac delta function.

[1] In this convention for the Fourier transform, the Dirac delta in real space and the Kronecker delta in Fourier space are given, respectively by the following representations

$$\delta_D(\mathbf{x}) = \frac{1}{V} \sum_{\mathbf{k}} e^{i\,\mathbf{k}\cdot\mathbf{x}} \quad \text{and} \quad \delta^K_{\mathbf{k}} = \frac{1}{V} \int d^3x\, e^{-i\,\mathbf{k}\cdot\mathbf{x}}\,.$$

2.2 FFT grid

Any FFT-based power spectrum estimator requires the sampling of the configuration-space density field on a regular grid, thereby introducing a spatial resolution responsible for aliasing spurious contributions to the Fourier-space density.

We will denote the linear size of the configuration-space (and Fourier-space) grid as N. The code allows to specify a different value for N for each spatial coordinate. However, as mentioned above, we will assume, to keep the notation of these notes simple, the case $N_x = N_y = N_z = N$.

The spatial resolution of the grid will be given by $H \equiv L/N$. In Fourier-space this resolution is represented by a Nyquist frequency defined as $k_{\rm Nyq} \equiv k_f\, N/2 = \pi/H$.

2.3 Direct summation

The code provides the option of computing the Fourier-space density by direct summation, thereby avoiding sampling in configuration space and related aliasing contribution. The evaluation is performed for all wavenumbers present on the grid of linear size N given in input. It should be clear that this is a *highly time-consuming option* since it scales as $N_P \times N^3$. The overdensity in this case is computed as

$$\widetilde{\delta}(\mathbf{k}) = \frac{1}{k_f^3}\frac{1}{N_P}\sum_{p=1}^{N_{\rm P}} e^{-i\,\mathbf{k}\cdot\mathbf{x}_p}\,. \tag{8}$$

In practice, for each position $\mathbf{x}_p$, a loop is perfomred over all values of the wavenumber $\mathbf{k}$ present on the grid.

2.4 Density interpolation on a grid

With the exception of the direct summation case, in the standard FFT-based estimator, the configuration-space density is first evaluated on a grid of linear size N. The density in each grid point $\mathbf{x}_G$ is computed by summing over the contributions provided by each particle weighted as a function of the distance according to the mass assignment scheme chosen. The resulting interpolated density is therefore the convolution of the desired density $\delta(\mathbf{x})$ with the weight function $W(\mathbf{x})$,

$$\delta_W(\mathbf{x}) = \int d^3y\,\delta(\mathbf{y})\,W^{(o)}(\mathbf{x}-\mathbf{y})\,. \tag{9}$$

The sampling in real space given by eq. (11) can be mathematically described in terms of the sampling function $Ш(\mathbf{x})$ (see, *e.g.* Hockney and Eastwood, 1981). This is defined as

$$Ш(\mathbf{x}) = \sum_{\mathbf{n}} \delta_D(\mathbf{x}-\mathbf{n})\,, \tag{10}$$

with $\mathbf{n} = \{n_x, n_y, n_z\}$ an integer vector. The sampled density, convolved with the window, can then be written as

$$\delta_G(\mathbf{x}) = Ш(\mathbf{x}/H)\delta_W(\mathbf{x}) \tag{11}$$

The function $\delta_G(\mathbf{x})$ is different from zero only for $\mathbf{x} = \mathbf{n}\,H$ and equals $\delta_W(\mathbf{x})$ at those points when integrated over.

The Fourier transform of $\delta_G(\mathbf{x})$ is obtain applying the convolution theorem so that

$$\widetilde{\delta}_G(\mathbf{k}) = k_f^3 \sum_{\mathbf{k}'} \widetilde{Ш}(\mathbf{k}')\,\widetilde{\delta}_W(\mathbf{k}-\mathbf{k}')\,, \tag{12}$$

where $\widetilde{\delta}_W(\mathbf{k})$ is the Fourier Transform of the convolved, continuous density field defined in eq. (9). $\widetilde{\delta}_G(\mathbf{k})$ now accounts for the discreteness of the grid assignment.

We notice now that the FT of the sampling function is a sampling function in k-space. Since we are dealing with periodic functions in position space the Delta functions are replaced by Kronecker deltas as

$$\widetilde{Ш}(\mathbf{k}) = \frac{1}{k_f^3}\sum_{\mathbf{m}} \delta^K_{\mathbf{k}-2\pi\,\mathbf{m}/H}\,. \tag{13}$$

Plugging eq. (13) in eq. (12) we obtain

$$\widetilde{\delta}_G(\mathbf{k}) = \sum_{\mathbf{m}} \widetilde{\delta}_W\left(\mathbf{k} - \frac{2\pi}{H}\mathbf{m}\right)\,, \tag{14}$$

where all terms with $\mathbf{m} \neq 0$ in the sum represent the aliasing contribution to the estimate $\widetilde{\delta}_G(\mathbf{k})$ of the density $\widetilde{\delta}_W(\mathbf{k})$, which, we should keep in mind, includes the effect of the interpolation function $W(\mathbf{x})$. Such effects can be partially removed, at least from the leading contribution in the sum above, simply dividing by the FT of the window function itself, $W(\mathbf{k})$.

2.5 Interlacing

The interlacing technique employs two sampling of density field $\delta_W(\mathbf{x})$. In addition to the sampling in eq. (11), denoted from now on as

$$\delta_{G1}(\mathbf{x}) = \text{Ш}(\mathbf{x}/H)\delta_W(\mathbf{x})\,, \tag{15}$$

we will consider a second grid given by

$$\delta_{G2}(\mathbf{x}) = \text{Ш}\left(\frac{\mathbf{x}}{H} + \frac{1}{2}\mathbf{v}\right)\delta_W(\mathbf{x})\,, \tag{16}$$

where the vector $\mathbf{v} = \{1, 1, 1\}$ provides a shift of $H/2$ in each spatial direction. Taking the Fourier Transform we get, as above,

$$\widetilde{\delta}_{G1}(\mathbf{k}) = \sum_{\mathbf{m}} \widetilde{\delta}_W\left(\mathbf{k} - \frac{2\pi}{H}\mathbf{m}\right)\,, \tag{17}$$

while for the second grid we have that, using the shift theorem,

$$FT\left[\text{Ш}\left(\frac{\mathbf{x}}{H} + \frac{1}{2}\mathbf{v}\right)\right] = \widetilde{\text{Ш}}(\mathbf{k})\, e^{i\,\mathbf{k}\cdot\mathbf{v}\,H/2}\,, \tag{18}$$

leading to

$$\begin{aligned}\widetilde{\delta}_{G2}(\mathbf{k}) &= k_f^3 \sum_{\mathbf{k}'} \text{Ш}(\mathbf{k}')\, e^{i\,\mathbf{k}\cdot\mathbf{v}\,H/2}\, \widetilde{\delta}_W(\mathbf{k} - \mathbf{k}') \\ &= \sum_{\mathbf{m}} e^{i\pi\,\mathbf{m}\cdot\mathbf{v}}\, \widetilde{\delta}_W\left(\mathbf{k} - \frac{2\pi}{H}\mathbf{m}\right) \\ &= \sum_{\mathbf{m}} (-1)^{m_x+m_y+m_z}\, \widetilde{\delta}_W\left(\mathbf{k} - \frac{2\pi}{H}\mathbf{m}\right)\,. \end{aligned} \tag{19}$$

Taking the mean of the two samples in Fourier-space, we get

$$\begin{aligned}\widetilde{\delta}_G(\mathbf{k}) &= \frac{1}{2}\left[\widetilde{\delta}_{G1}(\mathbf{k}) + \widetilde{\delta}_{G2}(\mathbf{k})\right] \\ &= \sum_{\mathbf{m}} \frac{1}{2}\left[1 + (-1)^{m_x+m_y+m_z}\right] \widetilde{\delta}_W\left(\mathbf{k} - \frac{2\pi}{H}\mathbf{m}\right)\,. \end{aligned} \tag{20}$$

In this way, for any odd value of the sum $m_x + m_y + m_z$, the corresponding aliasing contribution is removed. In particular this happens for the largest contributions occurring for $m_x + m_y + m_z = 1$. We can single-out the $\mathbf{m} = 0$ mode as

$$\widetilde{\delta}_G(\mathbf{k}) = \widetilde{\delta}_W(\mathbf{k}) + \sum_{\mathbf{m}\neq 0} \frac{1}{2}\left[1 + (-1)^{m_x+m_y+m_z}\right] \widetilde{\delta}_W\left(\mathbf{k} - \frac{2\pi}{H}\mathbf{m}\right)\,. \tag{21}$$

and write the final density correcting for the mass assignment window taking into account the FT of the convolution in eq. (9) as

$$\begin{aligned}\widetilde{\delta}_{out}(\mathbf{k}) &= \frac{\widetilde{\delta}_W(\mathbf{k})}{\widetilde{W}(\mathbf{k})} + \frac{1}{\widetilde{W}(\mathbf{k})}\sum_{\mathbf{m}\neq 0}\frac{1}{2}\left[1+(-1)^{m_x+m_y+m_z}\right]\widetilde{\delta}_W\left(\mathbf{k}-\frac{2\pi}{H}\mathbf{m}\right) \\ &= \widetilde{\delta}(\mathbf{k}) + \frac{1}{\widetilde{W}(\mathbf{k})}\sum_{\mathbf{m}\neq 0}\frac{1}{2}\left[1+(-1)^{m_x+m_y+m_z}\right]\widetilde{\delta}_W\left(\mathbf{k}-\frac{2\pi}{H}\mathbf{m}\right). \qquad (22)\end{aligned}$$

where the sum on the r.h.s. represent the residual aliasing contribution.

We notice that the second sample of eq. (16) could be alternatively defined as

$$\delta_{G2}(\mathbf{x}) = \text{Ш}\left(\frac{\mathbf{x}}{H}\right)\delta_W\left(\mathbf{x}-\frac{H}{2}\mathbf{v}\right), \qquad (23)$$

describing a sampling on the same points but of a density field displaced by $H\mathbf{v}/2$. This correspond to the actual implementation in the code. In this case the Fourier transform will be given by

$$\begin{aligned}\widetilde{\delta}_{G2}(\mathbf{k}) &= k_f^3\sum_{\mathbf{k}'}\text{Ш}(\mathbf{k}')\,e^{i\,(\mathbf{k}-\mathbf{k}')\cdot\mathbf{v}\,H/2}\widetilde{\delta}_W(\mathbf{k}-\mathbf{k}') \\ &= e^{i\,\mathbf{k}\cdot\mathbf{v}\,H/2}\sum_{\mathbf{m}} e^{i\pi\,\mathbf{m}\cdot\mathbf{v}}\,\widetilde{\delta}_W\left(\mathbf{k}-\frac{2\pi}{H}\mathbf{m}\right) \\ &= e^{i\,\mathbf{k}\cdot\mathbf{v}\,H/2}\sum_{\mathbf{m}}(-1)^{m_x+m_y+m_z}\,\widetilde{\delta}_W(\mathbf{k}-\frac{2\pi}{H}\mathbf{m}). \qquad (24)\end{aligned}$$

The desired mean of the two samples in this case is obtained with the weighted sum

$$\widetilde{\delta}_G(\mathbf{k}) = \frac{1}{2}\left[\widetilde{\delta}_{G1}(\mathbf{k}) + e^{-i\,\mathbf{k}\cdot\mathbf{v}H/2}\,\widetilde{\delta}_{G2}(\mathbf{k})\right], \qquad (25)$$

that includes the proper correction for the phase of $\widetilde{\delta}_{G2}(\mathbf{k})$.

2.6 The power spectrum estimator

Given the best estimate of the Fourier-space density field $\widetilde{\delta}(\mathbf{k})$ the power spectrum estimator is defined, from eq. (6) as

$$\hat{P}(k) \equiv \frac{k_f^3}{N_k}\sum_{\mathbf{q}\in k}\widetilde{\delta}(\mathbf{q})\,\widetilde{\delta}(-\mathbf{q}) \qquad (26)$$

where the sum runs over all wavenumbers $\mathbf{q}$ such that $k-\Delta k/2 \leq q < k+\Delta k/2$, Δk representing the bin size. The normalisation factor N_k gives the number of modes in the shell

$$N_k \equiv \sum_{\mathbf{q}\in k} \qquad (27)$$

and can be approximated for large k by the integral

$$\begin{aligned}N_k &\simeq \frac{1}{k_f^3}\int_{k-\Delta k/2}^{k+\Delta k/2} dq\,q^2 d\Omega \\ &= 4\pi\,\frac{k^2\,\Delta k}{k_f^3} + \mathcal{O}(\Delta k^3) \qquad (28)\end{aligned}$$

3 The algorithms

In this section, we will describe some details of the algorithms implementing the various passages leading to the power spectrum estimation starting from the catalog of the particle positions.

3.1 Catalog input

The code inputs the particles' positions in Cartesian coordinates, $\mathbf{r}_j = \{r_x, r_y, r_z\}_j$, with $p = 1, \ldots, N_P$, N_P being the total number of particles. Given the box linear size L, the positions are normalised so that for each coordinate $(i = 1, 2, 3)$ we have

$$x_i = \min\left[1 - \epsilon,\ \frac{\mathrm{mod}(r_i, L)}{L}\right] \in [0, 1) \tag{29}$$

with $\epsilon = 1 - 10^{-7}$ (the exclusion of the value 1 is enforced). To avoid the original coordinate r_i exceeding the limits $[0, L)$, its modulus is considered.

3.2 Mass assignment schemes

The sampling of the density is evaluated on a grid of linear size N. Grid points can be labeled as $\mathbf{x}_G(\mathbf{n}) = \mathbf{n}\,H$ in terms of the integer vector $\mathbf{n} = \{n_x, n_y, n_z\}$ and $n_i = 0, \ldots, N-1$, while $H = L/N$ is the linear grid spacing.

The density $\delta_G(\mathbf{x})$ defined in the previous section, is computed in each grid point by summing over the contributions provided by each particle as

$$\delta^G_{\mathbf{n}} = \frac{1}{N_P}\sum_{p=1}^{N_P} W^{(o)}(\mathbf{x}_G - \mathbf{x}_p) \tag{30}$$

where $W^{(o)}(\mathbf{x})$ represents the three-dimensional weight function for the mass assignment scheme of order o. This is given by

$$W^{(o)}(\mathbf{x}_G - \mathbf{x}_p) = W^{(o)}_{\mathrm{1D}}\left(n_x - \frac{x_p}{H}\right) W^{(o)}_{\mathrm{1D}}\left(n_y - \frac{y_p}{H}\right) W^{(o)}_{\mathrm{1D}}\left(n_z - \frac{z_p}{H}\right) \tag{31}$$

with the contribution $W^{(o)}_{\mathrm{1D}}(s)$ for each direction depending on the mass assignment scheme selected as

- Nearest Grid Point (NGP)

$$W^{(1)}_{\mathrm{1D}}(s) = \begin{cases} 1 & \text{for } |s| < \frac{1}{2} \\ 0 & \text{otherwise} \end{cases} \tag{32}$$

- Cloud-In-Cell (CIC)

$$W^{(2)}_{\mathrm{1D}}(s) = \begin{cases} 1 - |s| & \text{for } |s| < \frac{1}{2} \\ 0 & \text{otherwise} \end{cases} \tag{33}$$

- Triangular Shaped Cloud (TSC)

$$W^{(3)}_{\mathrm{1D}}(s) = \begin{cases} \frac{3}{4} - s^2 & \text{for } |s| < \frac{1}{2} \\ \frac{1}{2}\left(\frac{3}{2} - |s|\right)^2 & \text{for } \frac{1}{2} \le |s| < \frac{3}{2} \\ 0 & \text{otherwise} \end{cases} \tag{34}$$

- Piecewise Cubic Spline (PCS)

$$W^{(4)}_{\mathrm{1D}}(s) = \begin{cases} 4 - 6\,s^2 + 3\,|s|^3 & \text{for } 0 \le |s| < 1 \\ (2 - |s|)^3 & \text{for } 1 \le |s| < 2 \\ 0 & \text{otherwise} \end{cases} \tag{35}$$

The code reads the each particle coordinates and assigns the weights on the grid points where the contribution is non-zero. These are o^3 points for a scheme of order o. For all such schemes the sum of all contributions for each particle is equal to 1.

3.3 Standard FFT without interlacing

Given the configuration space, interpolated density field $\delta^G_{\mathbf{n}}$ of eq. (30), a Fast Fourier Transform (FFT) is performed taking advantage of the FFTW libraries (Frigo and Johnson, 2012). If the interlacing option is not selected, this simply amounts to a double precision real-to-complex transform invoked as dfftw_plan_dft_r2c_3d. This will provide the Fourier-space density

$$\widetilde{\delta}^G_{\mathbf{m}} = \frac{1}{k_f^3} \sum_{\mathbf{n}} \delta^G_{\mathbf{n}} \, e^{2\pi i \, \mathbf{m}\cdot\mathbf{n}/N} \,, \tag{36}$$

where the integer vector $\mathbf{m}$ labels the wavenumbers $\mathbf{k} = k_f \, \mathbf{m}$.

The density field $\delta_G(\mathbf{x})$ is the convolution of the real-space density $\delta(\mathbf{x})$ with the window function $W(\mathbf{x})$ and therefore needs to be corrected. This is done by the `fcomb` subroutine. In Fourier-space this is simply given by $\widetilde{\delta}(\mathbf{k}) = \widetilde{\delta}^G(\mathbf{k})/\widetilde{W}^{(o)}(\mathbf{k})$ where, for the mass assignment schemes considered

$$\begin{aligned} \widetilde{W}^{(o)}(\mathbf{k}) \equiv \widetilde{W}^{(o)}_{\mathbf{m}} &= \left[\frac{\sin(k_x \, H/2)}{k_x \, H/2}\right]^o \left[\frac{\sin(k_y \, H/2)}{k_y \, H/2}\right]^o \left[\frac{\sin(k_z \, H/2)}{k_z \, H/2}\right]^o \\ &= \left[\frac{\sin(m_x \, \pi/2)}{m_x \, \pi/2}\right]^o \left[\frac{\sin(m_y \, \pi/2)}{m_y \, \pi/2}\right]^o \left[\frac{\sin(m_z \, \pi/2)}{m_z \, \pi/2}\right]^o \,, \end{aligned} \tag{37}$$

The output density grid will then be

$$\widetilde{\delta}^{\rm out}_{\mathbf{m}} = \frac{\widetilde{\delta}^G_{\mathbf{m}}}{\widetilde{W}^{(o)}_{\mathbf{m}}} \,, \tag{38}$$

3.4 FFT and interlacing

If the interlacing option is selected the code is sampling the real-space density twice,

$$\delta^{G1}_{\mathbf{n}} = \frac{1}{N_P} \sum_{p=1}^{N_P} W^{(o)}(\mathbf{n}H - \mathbf{x}_p) \,, \tag{39}$$

$$\delta^{G2}_{\mathbf{n}} = \frac{1}{N_P} \sum_{p=1}^{N_P} W^{(o)}(\mathbf{n}H - \mathbf{x}_p - \mathbf{v}H/2) \,, \tag{40}$$

where the second grid samples $\delta_W(\mathbf{x})$ displacing the particle coordinates as

$$\mathbf{x}_p \quad \longrightarrow \quad \mathbf{x}_p + \left\{\frac{H}{2}, \frac{H}{2}, \frac{H}{2}\right\} \,, \tag{41}$$

implementing eq. (23). The two real-to-complex Fourier transforms are performed at once by means of a single complex-to-complex FT. A complex array is defined from the two real arrays $\delta^{G1}_{\mathbf{n}}$ and $\delta^{G2}_{\mathbf{n}}$ as

$$\delta^C_{\mathbf{n}} = \delta^{G1}_{\mathbf{n}} + i \, \delta^{G2}_{\mathbf{n}} \,, \tag{42}$$

and its FT is computed as a regular complex-to-complex 3D FFT invoked as dfftw_plan_dft_3d to get

$$\widetilde{\delta}^C_{\mathbf{n}} = \widetilde{\delta}^{G1}_{\mathbf{n}} + i \, \widetilde{\delta}^{G2}_{\mathbf{n}} \,. \tag{43}$$

In the `fcomb` subroutine, the final density field will then be obtained from the combination

$$\begin{aligned} \widetilde{\delta}^{\rm out}_{\mathbf{m}} &= \frac{1}{4\,\widetilde{W}^{(o)}_{\mathbf{m}}} \left[\left(1 - i e^{-\pi i \mathbf{m}\cdot\mathbf{v}/N_G}\right) \widetilde{\delta}^C_{\mathbf{m}} + \left(1 + i e^{-\pi i \mathbf{m}\cdot\mathbf{v}/N_G}\right) \left(\widetilde{\delta}^C_{-\mathbf{m}}\right)^*\right] \\ &= \frac{1}{4\,\widetilde{W}^{(o)}_{\mathbf{m}}} \left[\left(1 - i e^{-\pi i \mathbf{m}\cdot\mathbf{v}/N_G}\right) \left(\widetilde{\delta}^{G1}_{\mathbf{m}} + i \, \widetilde{\delta}^{G2}_{\mathbf{m}}\right) + \left(1 + i e^{-\pi i \mathbf{m}\cdot\mathbf{v}/N_G}\right) \left(\widetilde{\delta}^{G1}_{\mathbf{m}} - i \, \widetilde{\delta}^{G2}_{\mathbf{m}}\right)\right] \\ &= \frac{1}{2\,\widetilde{W}^{(o)}_{\mathbf{m}}} \left[\widetilde{\delta}^{G1}_{\mathbf{m}} + e^{-\pi i \mathbf{m}\cdot\mathbf{v}/N_G} \, \widetilde{\delta}^{G2}_{\mathbf{m}}\right] \,, \end{aligned} \tag{44}$$

corresponding to the expression in eq. (25) and including the mass assignment correction.

4 Compilation and running

The code is organised in the following files:

- PowerI4.f90 : Main code, contains also the subroutines for density grid normalization and file outputs
- grid.f90 : specifies the FFT interpolated grid
- input_catalog.f90 : contains all the subroutines for reading different input catalog formats. You can add here a subroutine reading your own format.
- interpolation.f90 : contains all the interpolation subroutines (all different mass assignment schemes for assignment on single and double grids)
- parbox.f90 : contains global variables

PowerI4 requires the use of FFTW (www.fftw.org). Current version uses FFTW3. Make sure you download and compile FFTW and that you put the correct path to FFTW libraries in the Makefile (in the variables FFTW_LIBR and FFTW_INCL). You also need to include fftw3.f in the first line of PowerI4.f90 .

A Makefile to compile PowerI4 is provided. Detail your system configuration in the variable SYSTYPE. Running make should produce the executable power.e

We provide a parameters.ini file for a test run, and a set of particles to measure the spectrum from. They are both located under PowerI4_public/test/. To run do

```
./power.e < test/parameters.ini
```

This run should produce a power spectrum file ps_n0256_4i.out in PowerI4_public/test/ that should be similar to the file ps_n0256_4i.out in the same directory.

For further runs, set-up your own parameters.ini as described in the next section.

5 Inputs

The main contents of the parameters.ini file are shown in figure 1. In the following we provide detailed explanations for each entry.

Input file name The name of the input catalog file. In the case of GADGET snapshot files of the form snapshot.0, snapshot.1, ... only the global name (*i.e.* snapshot) should be provided.

Input file type The entry Input file type specifies the format of the input catalog. We provide two pre-built subroutines:

- **xyz-format**, Input file type = 0
  The subroutine reads an unformatted structure with x, y and z coordinate values in each row. This can be easily changed to ascii format.
- GADGET **format**, Input file type = 1
  We provide a subroutine to read comoving dark matter outputs from GADGET in Fortran binary format. In this case you only need to write the baseline filename. In addition the endianess might need to be specified in the open statement of the subroutine input_catalog_gadget in the module input_catalog.f90. This is done by simply setting convert='big_endian' or convert='little_endian'.
  For the GADGET format if you request to measure the power spectrum multipoles along one axis, PowerI4 will move the particles to redshift space along that same coordinate direction.

You can write add your own subroutine in the module input_catalog.f90.

Output file name The name of the output file with the power spectrum measurements.

```
# Input file name =
input_example_powmes.unformatted
# Input file type =
0
# Output file =
ps_n0256_4i.out
# FFT grid sizes =
256
256
256
# Interpolation order (integer 0 to 4) =
4
# Interlacing (true/false) =
true
# Box sides =
1.
1.
1.
# bin size in units of the fundamental frequency (linear binning)
1
# center of the first bin in units of the fundamental frequency
1
# measure multiples for anisotropic clustering (0,1,2,3, see below)
0
# output density file (see below)
0
# output density file name
density.out
```

Figure 1: Input parameters file `parameters.ini` in the default setting for the test run.

FFT grid sizes Three values for the size of the FFT grid in each Cartesian coordinate. Clearly this freedom might lead to different resolution in different directions.

Interpolation order This entry sets the mass assignment scheme and the order of the interpolation of the particles onto the FFT grid as follows:

- **Direct summation**, `Interpolation order = 0`
- **Nearest Grid Point (NGP)**, `Interpolation order = 1`
- **Cloud In Cell (CIC)**, `Interpolation order = 2`
- **Triangular Shaped Cloud (TSC)**, `Interpolation order = 3`
- **Piecewise Cubic Spline (PCS)**, `Interpolation order = 4`

Interlacing Two entries are possible: `true` and `false`. The memory requirement does not change (in this version of the code) since for the interlacing option a single complex array of size $\{N_x, N_y, N_z\}$ is allocated with the FFTW performed in place, while if interlacing is not chosen the code allocates a complex array of size $\{N_x/2+1, N_y, N_z\}$ and a real one of size $\{N_x, N_y, N_z\}$, with the FFTW performed out of place. The time required to interpolate the particles onto the array, however, doubles when interlacing is performed.

Box sides The sides of the box, $\{L_x, L_y, L_z\}$ in the units of choice. Rectangular boxes are allowed, leading to unequal resolution in Fourier-space in the different direction since the fundamental frequencies will be given by

$$k_{f,x} = \frac{2\pi}{L_x}, \quad k_{f,y} = \frac{2\pi}{L_y}, \quad k_{f,z} = \frac{2\pi}{L_z}\,.$$

We remember that the power spectrum has dimensions of $L_x L_y L_z$ and will therefore inherit the units provided for these entries. For `GADGET` catalogs the entries here are compared with the values provided by the input file header which are assumed to be given in kpc and the code stops if it finds different values. Please make sure that the `input_catalog_gadget` subroutine is reading properly your files.

Bin size in units of the fundamental frequency (linear binning) The code measures the power spectrum exclusively with a linear binning in the wavenumber k. The bin size is expressed in units of the fundamental frequency of the box $k_f = 2\pi/L$. In case of a box with different sizes in different direction the reference fundamental frequency is the smallest among the three, *i.e.* $k_f = \min(k_{f,x}, k_{f,y}, k_{f,z})$

Center of the first bin in units of the fundamental frequency This allows full control on the linear bins for your measurements. For example, choosing the default values of 1 as bin size and 1 as first bin center (both in units of k_f), the power spectrum will be measured in the following bins (always in units of k_f):

$$[0.5, 1, 5), \quad [1.5, 2, 5), \quad [2.5, 3.5), \ldots$$

Choosing instead 1 as bin size and 0.5 as first bin center we get

$$[0, 1), \quad [1, 2), \quad [2, 3), \ldots$$

(a legitimate choice that we, however, do not like as it will maximise the number of wavenumbers falling on the limits of the bins).

Power spectrum multipoles Provides the option of measuring the power spectrum quadrupole and hexadecapole in addition to the monopole and the Cartesian axis along which multipoles are defined.

- `measure multiples = 0`, only the monopole is measured
- `measure multiples = 1`, multipoles are measured along the x-axis
- `measure multiples = 2`, multipoles are measured along the y-axis
- `measure multiples = 3`, multipoles are measured along the z-axis

If multipoles are measured you have to make sure that the particle distribution is in redshift space in the same coordinate as the indicated axis for the multipole decomposition. The `GADGET`-format input does this automatically.

Output density Provides the output of the density grid either in configuration or Fourier-space:

- `output density = 0`, no density grid is output.
- `output density = 1`, Fourier-space density grid is output.
- `output density = -1`, Fourier-space density grid is output and the code stops immediately without measuring the power spectrum.

Output density file name The name of the file containing the density in output.

6 Outputs

There are two basic outputs of `PowerI4`: the Fourier-space density grid and the measured power spectrum.

Fourier-space density output The Fourier-space density grid is output as an unformatted Fortran file as follows:

```
open(unit=4,file=file_density,status='unknown',form='unformatted')
write(4) Nx,Ny,Nz,dble(Nptot),kFx,kFy,kFz,kNx
write(4) (((dcl(ix,iy,iz),ix=1,Nx/2+1),iy=1,Ny),iz=1,Nz)
close(4)
```

Power spectrum output The format of the output power spectrum file is given by a first line providing the total number of particles N_P and the expected shot-noise contribution assuming Poisson distribution, *i.e.* $(1/(2\pi\bar{n}_P)$, $\bar{n}_P$ being the mean particle density. The subsequent lines provide, in the different columns, the following quantities:

- `column 1`: center of the wavenumber k bin
- `column 2`: value of the wavenumber k *averaged* over the bin
- `column 3`: power spectrum monopole $P_0(k)$
- `column 4`: power spectrum quadrupole $P_2(k)$
- `column 5`: power spectrum hexadecapole $P_4(k)$
- `column 6`: number of independent modes in each bin, N_k in eq. (27).

If multipoles are not requested columns 4 and 5 will be absent. The output file `ps_n0256_4i.out` for the test catalog `input_example_powmes.unformatted` would therefore look like this:

```
2097152 1.9223412533521352E-009
0.6283185307E+01 0.8018238919E+01 0.9698655242E-04 9
0.1256637061E+02 0.1401654949E+02 0.3125523196E-04 31
0.1884955592E+02 0.1969250288E+02 0.2547871060E-04 49
...
```

References

M. Frigo and S. G. Johnson. FFTW: Fastest Fourier Transform in the West, January 2012. Astrophysics Source Code Library.

R. W. Hockney and J. W. Eastwood. *Computer Simulation Using Particles.* McGraw-Hill, 1981.

E. Sefusatti, M. Crocce, R. Scoccimarro, and H. M. P. Couchman. Accurate estimators of correlation functions in Fourier space. *Mon. Not. R. Astron. Soc.* , 460:3624–3636, August 2016. doi: 10.1093/mnras/stw1229.