Prepared for submission to JCAP

Reconstruction Code

Martin White[a,b]

[a]Department of Physics, University of California, Berkeley, CA
[b]Lawrence Berkeley Laboratory, 1 Cyclotron Road, Berkeley, CA

E-mail: mjwhite@lbl.gov

Abstract. We present a (C++) code to perform reconstruction on a galaxy catalog, to sharpen the baryon acoustic oscillation feature for measuring distances. The code implements the standard, lowest-order algorithm presented in [1] with periodic boundary conditions using a multigrid relaxation technique with a full multigrid V-cycle based on damped Jacobi iteration. Inputs are a catalog of objects and two random catalogs which serve to specify the selection function and act as the "shifted" fields. Output are 'shifted' versions of the object and second random catalogs.

Keywords: cosmological parameters from LSS – baryon acoustic oscillations – galaxy clustering

Contents

1 Introduction

Sound waves traveling in the early Universe imprint a characteristic scale in the large-scale clustering of objects which can be used as a standard ruler to probe the expansion history of the Universe. These so-called baryon acoustic oscillations are partially damped by non-linear evolution, and techniques to remove this damping are now routinely applied in the analysis of survey data and assumed in forecasts of future performance. This process of 'reconstruction' comes in several flavors, but the most widely used is that of [1] (which itself draws on earlier work going back decades). In this note we present a new implementation of this 'standard' reconstruction code which uses an efficient multigrid solver to compute the displacements along which objects are moved in the process of reconstruction.

There has been an extensive study of this 'standard' reconstruction procedure in the literature, from both the analytic [2–5] and numerical [6–12] perspectives. These works discuss the robustness of the algorithm, its performance and several issues of implementation. A particularly detailed discussion in given in [7, 9, 12]. The only new piece of the current code is the use of a multigrid method to solve for the displacement, $\vec{\Psi}$, given the density field. Earlier codes used Fourier transform methods or linear algebra libraries. Multigrid techniques provide an efficient alternative which can easily handle non-parallel lines of sight, require no external libraries and allow very efficient parallelism with multiple threads.

2 Procedure

The inputs are three files (containing RA, DEC, redshift and possibly a weight) for the objects and two random catalogs, a linear bias factor (b), a growth parameter (f) and a Gaussian filtering scale (R_f). The first random catalog is used to define the selection function of the survey (and hence to compute the density field). This should be large. The second random catalog is used to generate the 'shifted' field which is needed for computation of the post-reconstruction clustering signal. The files are read and angular coordinates are converted to 3D Cartesian coordinates using a ΛCDM distance-redshift relation. The default value is $\Omega_m = 0.3$ (set in `recon.cpp`) and distances are computed in h^{-1}Mpc so that no value of h needs to be specified. The distance calculation includes only matter and a cosmological constant, there is no contribution from radiation or massive neutrinos. The outputs are the Cartesian coordinates of the 'shifted' object and random catalogs, with the shift vectors computed using lowest order Lagrangian perturbation theory with the assumed (linear) bias, b, and rate-of-growth parameter, f.

The code first estimates densities by depositing the objects and the first set of randoms onto a regular Cartesian mesh using CIC interpolation in a box which encloses the random

catalog entirely with a 25 per cent margin (for current and future surveys this is likely sufficient). The grid size is defined by the variable `Ng` in `global.h`. It should be set such that the grid spacing is slightly smaller than the smoothing length [it would not be difficult to modify the code to do this automatically, but currently it is assumed that the physical set-up will not change frequently]. In principle any value of `Ng` can be used but the code is more efficient if the prime factors are few and low numbers. Powers of two or three times powers of two work well. Significantly different choices may require retuning of the parameters in the multigrid algorithm.

A grid of $\delta = \rho_{\rm obj}/\rho_{\rm ran} - 1$ is formed and regions with $\rho_{\rm ran} = 0$ are set to $\delta = 0$ ensuring padding at the mean density. At this stage the density constrast is divided by the (supplied) large-scale bias, b, to convert from object fluctuations to mass fluctuations. Note: as with most situations involving division by a gridded density field, it is important that there be enough randoms that the density field on the grid is well defined! To help mitigate numerical errors associated with this problem, any grid point with too few randoms is assigned the mean density. The density contrast is then Gaussian smoothed by brute-force[1] convolution with a kernel $\exp[-(x/R_f)^2/2]$.

We solve for the Zeldovich [13] displacements, $\vec{\Psi}$, using a standard multigrid technique employing a full multigrid V-cycle with damped Jacobi iteration (see below). Both the data and the randoms are then moved by $-\vec{\Psi}$ to form the so-called 'displaced' and 'shifted' fields. We choose to move both the objects and the randoms by an additional factor of $f\hat{r}\hat{r}\cdot\vec{\Psi}$ (i.e. in the line-of-sight direction). An alternative, and common, choice is to enhance the shift in the line-of-sight direction only for the objects, and not the randoms (see [5] for further discussion and comparison). The Cartesian coordinates, with h^{-1}Mpc units, for the shifted and displaced fields (along with any weights) are then written to file.

Currently all of the I/O is done with ascii flat files. This can be time consuming, especially for large catalogs. A more efficient implementation would read and write binary files of some form.

3 Solving for $\vec{\Psi}$

We need to extract the displacement field, $\vec{\Psi}$, from the observed, redshift-space density field. In this section we present the formalism for an unbiased tracer of the density field – for biased tracers the source, δ, and the factor f both[2] need to be divided by b.

Within the Zeldovich approximation [13]

$$\vec{\nabla}\cdot\vec{\Psi} + f\vec{\nabla}\cdot\left(\hat{r}\left\{\hat{r}\cdot\vec{\Psi}\right\}\right) = -\delta \tag{3.1}$$

To first order $\vec{\Psi}$ is irrotational, so we can write $\vec{\Psi} = \nabla\phi$ and hence

$$\nabla^2\phi + f\vec{\nabla}\cdot\left(\left[\hat{r}\cdot\vec{\nabla}\phi\right]\hat{r}\right) = -\delta \tag{3.2}$$

To simplify the equations slightly note that we can rewrite the term linear in f as

$$\vec{\nabla}\cdot\left(\left[\hat{r}\cdot\vec{\nabla}\phi\right]\hat{r}\right) = \sum_{ij}\hat{r}_i\hat{r}_j\left(\partial_i\partial_j\phi\right) + \frac{2}{r}\hat{r}_j\partial_j\phi \tag{3.3}$$

[1] Aside from the ascii I/O, this is the slowest step in the code – for this reason we have also added an option to smooth the density field using FFTs if you have the FFTW library (v3).

[2] Note, in the literature this change of $f \to \beta = f/b$ is sometimes misprinted. The easiest way to derive the need for this is to consider a plane-parallel situation with $\hat{r} = \hat{z}$ and linear theory redshift space distortions in k-space: $\delta_k \to (b + f\mu_k^2)\delta_k$ and look at Eq. (3.2).

Discretizing the second derivatives to second order[3] we have, e.g.

$$\partial_x^2\phi \approx \frac{\phi_{i-1,j,k} - 2\phi_{i,j,k} + \phi_{i+1,j,k}}{h^2} \tag{3.4}$$

where h is the mesh size. We shall write this second-order accurate difference as D_x^2. This leads to a 3D stencil for ∇^2 with six 1's along the Cartesian directions and -6 in the center.

If we approximate the radial direction appearing in Eq. (3.2) as that pointing to the central mesh point (i.e. again neglect the variation of the line-of-sight direction in the derivative) then the term proportional to f in the above becomes simply

$$\sum_{i,j=1}^{3} \hat{r}_i\hat{r}_j D_i D_j\phi + \frac{2}{r}\sum_{j=1}^{3} \hat{r}_j D_j\phi \tag{3.5}$$

Consider the second derivative term. The $i = j$ components of the central difference all involved $\phi_{i,j,k}$ while the others will not. Since $\hat{r}\cdot\hat{r} = 1$ the coefficient of the central, $\phi_{i,j,k}$ term is thus replaced by $-6 \to -6(1 + f/3)$ while the coefficients of the terms offset by ± 1 become slightly more complex. If we use a central difference for the 1st derivative, it is also independent of $\phi_{i,j,k}$.

Continuing to work to 2nd order the central differences appear as, e.g.

$$\partial_x\partial_y\phi \approx \frac{\phi_{i+1,j+1,k} - \phi_{i+1,j-1,k} - \phi_{i-1,j+1,k} + \phi_{i-1,j-1,k}}{4h^2} \tag{3.6}$$

and

$$\partial_x\phi \approx \frac{\phi_{i+1,j,k} - \phi_{i-1,j,k}}{2h} \tag{3.7}$$

which do not include $\phi_{i,j,k}$. Eq. (3.5), times h^2, then becomes

$$\begin{aligned}
&- 2\phi_{i,j,k} \\
&+ \hat{r}_x^2\left(\phi_{i-1,j,k} + \phi_{i+1,j,k}\right) + \hat{r}_y^2\left(\phi_{i,j-1,k} + \phi_{i,j+1,k}\right) + \hat{r}_z^2\left(\phi_{i,j,k-1} + \phi_{i,j,k+1}\right) \\
&+ \hat{r}_x\hat{r}_y\left(\phi_{i-1,j-1,k} + \phi_{i+1,j+1,k} - \phi_{i+1,j-1,k} - \phi_{i-1,j+1,k}\right)/2 \\
&+ \hat{r}_x\hat{r}_z\left(\phi_{i-1,j,k-1} + \phi_{i+1,j,k+1} - \phi_{i+1,j,k-1} - \phi_{i-1,j,k+1}\right)/2 \\
&+ \hat{r}_y\hat{r}_z\left(\phi_{i,j-1,k-1} + \phi_{i,j+1,k+1} - \phi_{i,j+1,k-1} - \phi_{i,j-1,k+1}\right)/2 \\
&+ \theta\hat{r}_x\left(\phi_{i+1,j,k} - \phi_{i-1,j,k}\right) + \theta\hat{r}_y\left(\phi_{i,j+1,k} - \phi_{i,j-1,k}\right) + \theta\hat{r}_z\left(\phi_{i,j,k+1} - \phi_{i,j,k-1}\right)
\end{aligned} \tag{3.8}$$

where we have written $\theta = h/r$ and in practice inclusion of these θ-terms makes little difference to most problems. Note, if we take the line-of-sight direction to be fixed throughout the grid, e.g. $\hat{r} \simeq \hat{z}$ and take $\theta = 0$, then $\hat{r}_x = \hat{r}_y = 0$ and $\hat{r}_z = 1$ so the sum of the ∇^2 and f terms reduces to increasing the D_z^2 central difference by a factor of $1 + f$ while leaving all the others unchanged, as expected.

[3] We use a central difference scheme, which gives second-order accurate approximations. These formulae are most easily derived by Taylor series expansion.

Having discretized the equations, we can now rewrite them in a form conducive to relaxation

$$\begin{aligned}(6+2f)\,\phi_{i,j,k} &= h^2\delta_{i,j,k} \\ &+ (1+f\hat{r}_x^2)\,(\phi_{i-1,j,k}+\phi_{i+1,j,k}) \\ &+ (1+f\hat{r}_y^2)\,(\phi_{i,j-1,k}+\phi_{i,j+1,k}) \\ &+ (1+f\hat{r}_z^2)\,(\phi_{i,j,k-1}+\phi_{i,j,k+1}) \\ &+ f\theta\hat{r}_x\,(\phi_{i+1,j,k}-\phi_{i-1,j,k}) + f\theta\hat{r}_y\,(\phi_{i,j+1,k}-\phi_{i,j-1,k}) + f\theta\hat{r}_z\,(\phi_{i,j,k+1}-\phi_{i,j,k-1}) \\ &+ f\hat{r}_x\hat{r}_y\,(\phi_{i-1,j-1,k}+\phi_{i+1,j+1,k}-\phi_{i+1,j-1,k}-\phi_{i-1,j+1,k})\,/2 \\ &+ f\hat{r}_x\hat{r}_z\,(\phi_{i-1,j,k-1}+\phi_{i+1,j,k+1}-\phi_{i+1,j,k-1}-\phi_{i-1,j,k+1})\,/2 \\ &+ f\hat{r}_y\hat{r}_z\,(\phi_{i,j-1,k-1}+\phi_{i,j+1,k+1}-\phi_{i,j+1,k-1}-\phi_{i,j-1,k+1})\,/2 \end{aligned} \tag{3.9}$$

We solve this system using a very standard multigrid technique, using a full multigrid with a V-cycle based upon damped Jacobi iteration. We implemented the algorithm using the simplest V-cycle. The number of V-cycle calls is defined by the `Niter` parameter in the routine `fmg` in `multigrid.cpp` and the number of Jacobi steps per level is set by the `Nit` parameter in `jacobi` in `multigrid.cpp`. Intermesh transfers are done using a standard stencil, with full-weighting for the restriction operator. The code is `OpenMP` parallel, assuming shared memory.

Having obtained ϕ on the grid we compute $\vec{\Psi}$ using the second-order, central difference approximation as above, e.g.

$$\Psi_x = \nabla_x\phi \approx \frac{\phi_{i+1,j,k}-\phi_{i-1,j,k}}{2h} \tag{3.10}$$

and similarly for ∇_y and ∇_z. The shift vectors at the positions of the objects and randoms are then obtained from CIC interpolation and the positions moved by $\vec{r} \to \vec{r} - \mathbf{R}\vec{\Psi}$ where $R_{ij} = \delta_{ij} + f\hat{r}_i\hat{r}_j$ is a redshift-space operator.

References

[1] D. J. Eisenstein, H.-J. Seo, E. Sirko, and D. N. Spergel, *Improving Cosmological Distance Measurements by Reconstruction of the Baryon Acoustic Peak*, *ApJ* **664** (Aug., 2007) 675–679, [astro-ph/0604362].

[2] N. Padmanabhan, M. White, and J. D. Cohn, *Reconstructing baryon oscillations: A Lagrangian theory perspective*, *PRD* **79** (Mar., 2009) 063523, [arXiv:0812.2905].

[3] Y. Noh, M. White, and N. Padmanabhan, *Reconstructing baryon oscillations*, *PRD* **80** (Dec., 2009) 123501, [arXiv:0909.1802].

[4] S. Tassev and M. Zaldarriaga, *Towards an optimal reconstruction of baryon oscillations*, *JCAP* **10** (Oct., 2012) 6, [arXiv:1203.6066].

[5] M. White, *Reconstruction within the Zeldovich approximation*, *ArXiv e-prints* (Apr., 2015) [arXiv:1504.0367].

[6] H.-J. Seo, J. Eckel, D. J. Eisenstein, K. Mehta, M. Metchnik, N. Padmanabhan, P. Pinto, R. Takahashi, M. White, and X. Xu, *High-precision Predictions for the Acoustic Scale in the Nonlinear Regime*, *ApJ* **720** (Sept., 2010) 1650–1667, [arXiv:0910.5005].

[7] N. Padmanabhan, X. Xu, D. J. Eisenstein, R. Scalzo, A. J. Cuesta, K. T. Mehta, and E. Kazin, *A 2 per cent distance to $z = 0.35$ by reconstructing baryon acoustic oscillations - I. Methods and application to the Sloan Digital Sky Survey*, *MNRAS* **427** (Dec., 2012) 2132–2145, [arXiv:1202.0090].

[8] X. Xu, A. J. Cuesta, N. Padmanabhan, D. J. Eisenstein, and C. K. McBride, *Measuring D_A and H at z=0.35 from the SDSS DR7 LRGs using baryon acoustic oscillations*, *MNRAS* **431** (May, 2013) 2834–2860, [arXiv:1206.6732].

[9] A. Burden, W. J. Percival, M. Manera, A. J. Cuesta, M. Vargas Magana, and S. Ho, *Efficient reconstruction of linear baryon acoustic oscillations in galaxy surveys*, *MNRAS* **445** (Dec., 2014) 3152–3168, [arXiv:1408.1348].

[10] R. Tojeiro, A. J. Ross, A. Burden, L. Samushia, M. Manera, W. J. Percival, F. Beutler, J. Brinkmann, J. R. Brownstein, A. J. Cuesta, K. Dawson, D. J. Eisenstein, S. Ho, C. Howlett, C. K. McBride, F. Montesano, M. D. Olmstead, J. K. Parejko, B. Reid, A. G. Sánchez, D. J. Schlegel, D. P. Schneider, J. L. Tinker, M. V. Magaña, and M. White, *The clustering of galaxies in the SDSS-III Baryon Oscillation Spectroscopic Survey: galaxy clustering measurements in the low-redshift sample of Data Release 11*, *MNRAS* **440** (May, 2014) 2222–2237, [arXiv:1401.1768].

[11] M. Vargas-Magaña, S. Ho, X. Xu, A. G. Sánchez, R. O'Connell, D. J. Eisenstein, A. J. Cuesta, W. J. Percival, A. J. Ross, E. Aubourg, J. R. Brownstein, S. Escoffier, D. Kirkby, M. Manera, D. P. Schneider, J. L. Tinker, and B. A. Weaver, *The clustering of Galaxies in the SDSS-III Baryon Oscillation Spectroscopic Survey: potential systematics in fitting of baryon acoustic feature*, *MNRAS* **445** (Nov., 2014) 2–28.

[12] A. Burden, W. J. Percival, and C. Howlett, *Reconstruction in Fourier space*, *ArXiv e-prints* (Apr., 2015) [arXiv:1504.0259].

[13] Y. B. Zel'dovich, *Gravitational instability: An approximate theory for large density perturbations.*, *A&A* **5** (Mar., 1970) 84–89.