Astronomy & Astrophysics manuscript no. output
April 16, 2025

Measuring the angular homogeneity scale in the era of large galaxy surveys

P. Fanha[⋆1,2], A. Da Silva[3,4], J. Fonseca[1,2], J.P. Mimoso[3,4], Z. Sakr [5,6,7]

(Affiliations can be found after the references)

April 16, 2025

ABSTRACT

An abstract will appear here...

Key words. select up to 6 key words from the list given in https://www.aanda.org/for-authors/latex-issues/information-files#pop, e.g.: Methods: statistical – Surveys – Cosmology: observations – large-scale structure of Universe – cosmological parameters

1. Introduction

One of the foundational assumptions of modern cosmology is the Cosmological Principle, the idea that the Universe is statistically homogeneous and isotropic on sufficiently large scales. This principle underpins the standard ΛCDM model and plays a pivotal role in the interpretation of a wide range of cosmological observations. While the statistical isotropy of the Universe has been generally confirmed, most notably through the analysis of the Cosmic Microwave Background (CMB), direct tests of spatial homogeneity remain more subtle and challenging; see [1] and references therein.

To probe homogeneity in the large-scale distribution of matter, statistical tools derived from galaxy clustering are commonly used. Among these, the fractal dimension D_2 has emerged as a powerful indicator of the scale at which the Universe transitions from a clustered to a homogeneous distribution. When combined with the concept of the angular homogeneity scale θ_H — the angular scale beyond which galaxy correlations become consistent with homogeneity — these measures offer a direct observational avenue to test the scale of homogeneity across redshifts and assess the validity of the Cosmological Principle.

Recent studies, such as those of [2], have demonstrated how D_2 and θ_H can be robustly estimated using the two-point angular correlation function (2PACF) using the Landy-Szalay (LS) estimator. These analyses have laid the groundwork for using angular clustering statistics not only to test the transition to homogeneity, but also to explore how this transition may evolve with redshift. More recently, [3] have shown that such measurements can further constrain cosmological parameters, including dark energy properties, thereby bridging the gap between statistical geometry and fundamental physics.

In this work, we expand on this methodology by comparing multiple approaches to compute D_2 and θ_H from the same underlying 2PACF, all within the framework of the Landy-Szalay estimator. We used realistic numerical simulations of galaxy surveys to test the robustness and consistency of these homogeneity diagnostics across different redshift bins and under varying systematic effects. Our aim is to quantify how methodological choices affect the estimation of D_2 and θ_H, and, in turn, to determine with confidence how we can assert the emergence of homogeneity in observational data.

This paper is structured as follows. In Section 2, we review the theoretical background of cosmological fractal dimension analysis and the adopted data set. Section 3 describes the data processing steps of our methods to estimate the homogeneity scale, θ_H, and its uncertainty, σ_{θ_H}. In Section 4, we present the results of our different computational approaches and conclude with a discussion of the implications of our findings in Section 5.

2. Analysis background

2.1. Theoretical framework

The concept of fractal dimension, D_2, is, in general, defined as a scalar quantity that quantifies the complexity of a pattern as a ratio of the change in detail to the change in scale, given some domain-specific definition of 'detail'. In this work, we consider the projected distribution of galaxies in the sky and, as such, D_2 is typically defined as a measure of the change in the observed surface number count of galaxies N in a spherical cap of changing angular radius θ:

$$D_2(\theta) = \frac{d\ln N(<\theta)}{d\ln\theta}. \tag{1}$$

The observed number counts N can be rewritten as a function of the scaled number counts, $\mathcal{N} = N/\bar{N}$, where $\bar{N} = \bar{n}A(\theta)$ is the average surface number count in a spherical cap of unit radius and solid angle $A(\theta) = 2\pi(1-\cos\theta)$ and $\bar{n}$ is the surface number density (i.e. per unit solid angle).

This leads to the following equation for D_2:

$$D_2(\theta) = \frac{d\ln \mathcal{N}(<\theta)}{d\ln\theta} + \frac{\theta\sin\theta}{1-\cos\theta}. \tag{2}$$

The scaled number counts $\mathcal{N}(<\theta)$ can be obtained analytically from the 2PACF as such:

$$\mathcal{N}(<\theta) = 1 + \frac{1}{1-\cos\theta}\int_0^\theta \omega(\theta')\sin\theta'\,d\theta' \tag{3}$$

and thus, if we can determine the 2PACF, we can also obtain the fractal dimension D_2.

⋆ e-mail: up201908149@edu.fc.up.pt

The fractal dimension, D_2, provides us with a way to define a scale of transition to homogeneity: given any distribution of galaxies that becomes homogeneous at sufficiently large scales in the projected space, we get:

$$\frac{d\ln \mathcal{N}(<\theta)}{d\ln\theta} \to 0 \ , \quad D_2(\theta) \to D_2^H(\theta) = \frac{\theta\sin\theta}{1-\cos\theta}. \tag{4}$$

In principle, homogeneity is reached when D_2 approaches D_2^H asymptotically. In practice, and also due to observational constraints, a common definition of the angular homogeneity scale, θ_H, is the lowest angular separation at which D_2 crosses the 1% deviation from $D_2^H(\theta)$. This is the definition that we use in this work.

The quantities defined above are calculated from galaxy surveys using numerical estimators. The Landy-Szalay estimators [should I specify why we're using the LS estimator?] for the 2PACF and scaled number counts are:

$$\omega^{LS}(\theta_i) = \frac{DD(\theta_i) - 2DR(\theta_i) + RR(\theta_i)}{RR(\theta_i)} \tag{5}$$

$$\mathcal{N}(<\theta_i) = 1 + \frac{\sum_{\theta_j=0}^{\theta_i}\left[DD(\theta_j) - 2DR(\theta_j) + RR(\theta_j)\right]}{\sum_{\theta_j=0}^{\theta_i} RR(\theta_j)} \tag{6}$$

where DD, DR and RR are the number of pairs of galaxies in the angular separation bin θ_i in the data catalogue (DD), between the data and the randoms catalogue (DR), and in the randoms catalogue (RR). All three of these quantities are normalized by the total number of pairs in the corresponding catalogue(s): for DD (d) and RR (r), the normalization factor is $N_{d,r}\,(N_{d,r}-1)\,/2$, whereas for DR it is $N_d \times N_r$.

D_2 is then obtained from the numerical scaled number counts with Eq. (2).

2.2. Simulation data set

In this work, we use galaxy mock simulations derived from the MICE Grand Challenge (MICE-GC) N-body simulation, designed to model the large-scale structure of the Universe. These simulations provide estimates of galaxy luminosities tailored for wide-field surveys such as the Dark Energy Survey (DES) [4] and the Euclid mission [5]. The mock galaxy catalogue is constructed using a hybrid approach that combines a Halo Occupation Distribution (HOD) model with Halo Abundance Matching (HAM) to populate Friends-of-Friends (FoF) dark matter halos identified in the MICE-GC simulation. The cosmological parameters adopted for the simulation are $\Omega_m = 0.25$, $\sigma_8 = 0.8$, $n_s = 0.95$, $\Omega_b = 0.044$, $\Omega_\Lambda = 0.75$, and $h = 0.7$.

The full version 2 of the MICE mock galaxy catalogue [6–8] comprises 499,609,997 galaxies and is publicly available through the CosmoHub portal[1] [9]. For our analysis, we use a smaller subset of the catalogue, covering ADS: check the area 600 square degrees from a sky region where DES i-band magnitudes are complete down to $i = 24$, and Euclid H-band magnitudes are complete down to H ~23.0, up to $z \sim 1.4$. AdS: Are we selecting a sky area set by dec > 30 or dec < 30 AND 30 < ra < 60? if not, we need to revise the previous paragraph for DES. Please include a table with details about the redshift bins, the average redshift, the number of galaxies, and the number of random points in each bin.

[1] https://cosmohub.pic.es

3. θ_H estimation methods

Several approaches were tested to estimate θ_H from $D_2(\theta)$ at each redshift bin. The main concerns when using the estimated D_2 curves are:

- The 2PACF is sensitive to the baryonic acoustic oscillations (BAO), and by definition, D_2 will be as well, and more strongly so.
- D_2 is especially sensitive to clustering under- and over-densities [as shown in [10]... (ALFALFA paper added)].

In addition, we are looking for cosmological model-independent methods to estimate θ_H and the associated uncertainty, without making any prior assumptions on the behaviour of D_2 or on the distribution of galaxies.

For these reasons, we present two different methods of obtaining D_2 from the data.

1. *d2*: Estimation of D_2 directly from the averaged D_2 LS estimator over the N_r random catalogues used;
2. *wtheta*: Estimation of D_2 using Eq. (3) from a fit to the averaged 2PACF over the N_r random catalogues used.

We expect that the second method be much less sensitive to the survey region chosen for analysis, and therefore, allow us to obtain more reliable estimates of θ_H, assuming that the function fitted to the angular correlation function does not capture the small oscillations in the curve.

We adopted a data-driven modeling framework, using symbolic regression (SR; via `PySR`[2], a Python wrapper for Julia's `SymbolicRegression.jl` package) to find candidate functions that fit $\omega(\theta)$ for each redshift bin, allowing these functions to be composed only of the basic operators $+$, $-$, $\times$, $/$, $\hat{}$ along with the exponential function $\exp(x)$, setting a maximum complexity of 15, as well as applying interval arithmetic to require that any fit function be real and well-defined in the entire interval from 0 deg to the maximum angular separation considered. The reason for these constraints is that a) the fit function should not capture small-scale oscillations as much as possible, b) increasing complexity (number of terms in the expression) slows down the symbolic regression search process and does not seem to provide any benefit, and c) Eq. (3) requires integrating $\omega(\theta)\sin\theta$ from the lower limit of 0 deg.

The best-fit expression is selected by minimizing a Mahalanobis χ^2 loss term, including uncertainties in the data points obtained by methods that we shall discuss further down in this paper, in Sections 3.1 and 3.2.

We then use `SymPy`[3], a Python package for symbolic computation, to manipulate the output expressions from `PySR`.

From the best-fit function, we obtain curves for D_2 in general by numerical integration using the Clenshaw-Curtis quadrature implemented in Python's `scipy` package, since we found that it was rarely the case that the integral admitted a closed-form antiderivative.

The same approach to fitting was applied to the *d2* method of obtaining D_2 from the data, using the same constraints, except in this case we obtain directly an expression for D_2. One peculiarity of this method is that we fitted a function to the left member of:

$$D_2(\theta) - 0.99D_2^H(\theta) = 0 \tag{7}$$

rather than directly to the estimated $D_2(\theta)$, since in this case determining θ_H for a single D_2 curve can be reduced to root finding.

[2] https://github.com/MilesCranmer/PySR
[3] https://www.sympy.org/

We chose to use a bisection algorithm for simplicity and guaranteed convergence.

For the *wtheta* method, we can calculate $\frac{d\ln\mathcal{N}(<\theta)}{d\ln\theta}$ directly, and then find the root of:

$$\frac{d\ln\mathcal{N}(<\theta)}{d\ln\theta} + 0.01D_2^H(\theta) = 0 \tag{8}$$

which is equivalent to Eq. (7).

In general, the final estimate of θ_H presented will depend on which uncertainty estimation method is used, as explained in Section 3.2.

3.1. Estimation of uncertainties in data

In order to quantify the error on our estimates of the transition scales θ_H, two routes exist: external error estimation methods, for instance, making mock survey regions resembling our data catalogue of interest; and internal error estimation methods, applying jackknife/bootstrap resampling techniques to the data. External error estimation methods suffer from the problem of being cosmological model-dependent, and thus cannot give unbiased results. Therefore, our only recourse is to attempt to estimate the uncertainty associated to $\omega(\theta)$ and $D_2(\theta)$ from the data itself.

Since we were already using `TreeCorr`[4], a Python package for computing correlation functions, to numerically estimate $\omega(\theta)$ and $D_2(\theta)$, and because `TreeCorr` already implements different covariance estimation methods, including jackknife and bootstrap, we used this package to obtain covariance estimates for our quantities of interest.

Both of these techniques require dividing our survey region of interest into subregions, and these regions must be large enough to be uncorrelated and statistically represent the entire survey. Thus, there is an upper limit on the number of regions in which we can subdivide our survey, which depends on the correlation length of the distribution of galaxies and on the size of the survey. On the other hand, we also want to maximize the number of subregions we have if that leads to a better estimate of the covariance. We attempted to choose a number of subregions that contain at least twice the correlation length of the catalogue while validating that the choice of patches does not significantly affect our estimates [should I add here the plots for the subdivided survey? for guaranteeing that the number of patches doesn't matter - it will have to be later? though I could still show EVR + whatever plots], considering the performance implications of using a larger number of patches.

Although we make the decision of the number of subregions (or 'patches', as `TreeCorr` calls them), we let `TreeCorr` use its default k-means clustering algorithm to subdivide the survey space into roughly equal-area patches. In its documentation [5], `TreeCorr` shows that it has a particularly efficient implementation, especially if you let it use its own information obtained from its tree-based pair-counting algorithm (corresponding to setting the option kmeans_init = 'tree' [how do I style this text?]).

`TreeCorr` then computes a covariance estimate using bootstrap as follows:

1. From the full N_{patch} patches, select N_{patch} patches with replacement.
2. Calculate the total correlation function or D_2 that would have been computed from these patches.
3. The auto-correlations are included at the selected repetition for the bootstrap samples, meaning that if a patch is repeated, its auto-correlation is included that many times. Cross-correlations between patches are included only if the two patches are different.
4. Repeat the steps above a total of $N_{bootstrap}$ times to build up a large set of resampled correlation functions, $\{\omega_n\}$.
5. The covariance estimate is then calculated as the sample variance of these resampled results:

$$C_{ij} = \frac{1}{N_{\rm bootstrap}-1}\sum_n \left[\omega_n(\theta_i) - \bar{\omega}(\theta_i)\right]\left[\omega_n\left(\theta_j\right) - \bar{\omega}\left(\theta_j\right)\right] \tag{9}$$

The number of bootstrap resamplings, $N_{bootstrap}$, used in this analysis was the default, 500, which we confirmed to be an adequate number by comparing results to jackknife covariance estimates.

Jackknife resampling is computed by treecorr as follows:

1. From the full N_{patch} patches, remove one patch at random.
2. Calculate the correlation function or D_2 that would have been computed from the $N_{patch} - 1$ patches as:

$$C_{ij} = \frac{N_{\rm patch}-1}{N_{\rm patch}}\sum_n \left[\omega_n(\theta_i) - \bar{\omega}(\theta_i)\right]\left[\omega_n\left(\theta_j\right) - \bar{\omega}\left(\theta_j\right)\right] \tag{10}$$

In this case, the number of resamplings equals the number of subregions. It is also required that the number of angular separation bins (the number of data points in the $\omega(\theta)$ or $D_2(\theta)$ estimates) match or exceed the number of resamplings, otherwise the covariance matrix is not guaranteed to be non-singular. [are you sure that's not the case with bootstrap too?]

In both of these methods, there is some ambiguity in the way that the correlation function (or the derived quantity D_2) is calculated. Both of these quantities are calculated as a function of pairs of galaxies, and there are multiple ways to count pairs: one can consider only pairs where both galaxies lie within selected patches in the iteration, or alternatively consider pairs where at least one galaxy lies within selected patches and attribute a different weight to cross-pairs (pairs where one of the galaxies is not in the selected region). treecorr implements different weighting schemes. [MP22 (paper mentioned in treecorr)] shows that selecting the 'geom' weighting scheme for bootstrap and 'match' scheme for jackknife provides significantly more accurate estimates of the covariance matrix than counting only pairs inside selected patches (the weighting scheme names correspond to the names given by treecorr), and so that is the choice we made.

If all conditions above are met, we expect [I have studies somewhere around here that suggest this... should cite them.] the covariance estimates between bootstrap and jackknife to be similar, even with different number of patches or number of angular separation bins.

One particular issue is worth noting here: whereas we found covariance matrices for $\omega(\theta)$ to be non-singular, for $D_2(\theta)$ we had to recourse to diagonal loading to use these covariance matrices for error estimates. We assume this difference is due to the higher correlation between angular separation bins in D_2 than in the 2PACF. It is possible that increasing the number of patches would solve this, but, setting aside the additional consideration of the exponential increase in computation time, with a small survey area, more patches would make results unreliable. The regularization applied to the D_2 covariance estimates adds a really small constant to the diagonal values, which essentially lead

[4] https://github.com/rmjarvis/TreeCorr
[5] https://rmjarvis.github.io/TreeCorr/_build/html/cov.html#cross-patch-weights

the really small eigenvalues to stop causing numerical instabilities, while not affecting noticeably the principal components of the matrix. [point out that plots showing this will be presented later in results? or present them now?].

The final covariance matrices used in the fitting process as described in the next section, both for the $\omega(\theta)$ and for the $D_2(\theta)$ quantities, are the average of the covariance matrices obtained for each $\omega^i(\theta)$ or $D_2^i(\theta)$ using a different random catalogue.

3.2. Estimation of uncertainties in SR fits and θ_H

Throughout the SR search process, PySR is set to automatically optimize all constants in the candidate functions, such that any given candidate function selected will already be the best fit given the general expression of the function. Nonetheless, it does not have any parameters other than θ and it does not produce any useful error estimates to propagate uncertainties to θ_H. Hence, we developed/implemented and tested multiple different methods to estimate uncertainties in SR fits.

- **Method 1 (path integral).**
  [cite path integral paper arXiv:1205.0364v2] describes a method that uses the definition of $1-\sigma$ confidence interval for a Gaussian likelihood functional of the residuals:

  $$\mathcal{L} \propto \exp\left(-\chi^2(f)/2\right) \tag{11}$$

  to estimate the $1-\sigma$ standard deviation, modelled as a second-order polynomial. $\chi^2(f)$ corresponds to the squared Mahalanobis distance for a SR fit function f.
  The standard deviation can then be propagated using the linear error propagation formula to estimate the uncertainty in θ_H:

  $$\sigma_{\theta_H} = \left[\frac{\sigma_y}{|df/d\theta|}\right]_{\theta=\theta_H} \tag{12}$$

  In this case, the final θ_H presented is the root found from the single best-fit D_2 curve.
  [should I give more info on this method? or just let reader refer to paper?]
- **Method 2 (Fisher).**
  In this method, we parametrize the candidate fit function $f(\theta)$ by transforming all constants into variables $a_0, a_1, ...$, such as in the example (note that basic operators selected in PySR, such as the exponential function e, are not replaced):

  $$f(\theta) = 1.4\theta^{0.2} + e^{2.83\theta} \rightarrow f(\theta; a_0, a_1, a_2) = a_0\theta^{a_1} + e^{a_2\theta} \tag{13}$$

  We can estimate an upper bound [is it still an upper bound when you have a first-order approximation?] for the covariance of these parameters around the optimum values via a first-order approximation to the Fisher information matrix.
  Let $\mathcal{I}(a_0, a_1, ...)$ be the Fisher information matrix:

  $$\mathcal{I}(a_0, a_1, ...) = -\mathbb{E}\left[\nabla^2 \log L(a_0, a_1, ...)\right] \tag{14}$$

  For a weighted least squares $\chi^2 = r^T C^{-1} r$, with $r = y_{|\theta} - f(\theta; a_0, a_1, ...)$ the residuals vector, the log-likelihood for normal residuals is:

  $$\log L(a_0, a_1, ...) = -\frac{1}{2}\chi^2(a_0, a_1, ...) + \text{const} \tag{15}$$

  from which we get:

  $$\nabla^2 \log L(a_0, a_1, ...) = -J^T C^{-1} J - \text{second order term} \tag{16}$$

  where J is the Jacobian matrix of the residuals.
  Near the optimum, if we assume that the second order term is small, we can discard it, and obtain an approximate version of the Fisher matrix that looks like:

  $$\mathcal{I}(a_0, a_1, ...) \approx J^T C^{-1} J \tag{17}$$

  whose inverse C_F provides an approximate lower bound on the covariance matrix.
  The uncertainty in the estimate of θ_H is then propagated using the linear error propagation formula, assuming $\theta_H = \theta_H(a_0, a_1, ...)$:

  $$\sigma_{\theta_H} = \sqrt{\left(\frac{\partial\theta_H}{\partial a_0}\right)^2 C_{F;a_0,a_0} + \left(\frac{\partial\theta_H}{\partial a_1}\right)^2 C_{F;a_1,a_1} + 2\frac{\partial\theta_H}{\partial a_0}\frac{\partial\theta_H}{\partial a_1}C_{F;a_0,a_1} + ...} \tag{18}$$

  calculating the derivatives of θ_H with the implicit function theorem:

  $$\frac{\partial\theta_H}{\partial a_i} = -\frac{\partial f/\partial a_i}{\partial f/\partial\theta}. \tag{19}$$

  As with method 1, we present the final θ_H as the root found from the single best-fit D_2 curve.
- **Method 3 (Monte Carlo).**
  We parametrize the SR fit function in the same way as in method 2, but use a nested sampler, in particular, the Python dynesty package [same old, same old - italics, footnote?], to estimate the posterior probability distributions of the fit parameters a_i. The decision to use a nested sampler as opposed to an MCMC algorithm was made because the former is more robust against and simpler to implement for high-dimensional or multi-modal problems. [I probably should explain that even though NC is geared towards evidence estimation, we can get good posterior estimates as well. - how?; also, that I can't control exactly what comes out of PySR which is also why this is more robust.]
  We use a Gaussian log-likelihood and $1-\sigma$ Gaussian prior distributions for each parameter a_i of the SR fit f. The sampled parameters and their corresponding importance weight, based on their likelihood and prior probability, can be used to estimate the distribution of values for θ_H by substituting each combination of sampled parameters in f and obtaining for each a single estimate of θ_H. The weighted median of these θ_H provide a final estimate for θ_H, while the 16% and 84% weighted percentiles of the distribution provide bounds for the associated uncertainty. [should I show plots to clarify?]

4. Results

5. Conclusions

Acknowledgements. This work was financed by Portuguese funds through FCT - Fundação para a Ciência e a Tecnologia in the framework of the following Research Projects: EXPL/FIS-AST/1368/2021 (AdS, AC, JPM), PTDC/FIS-AST/0054/2021 (NN, AdS, AC, JPM), UIDB/04434/2020, UIDP/04434/2020, and CERN/FIS-PAR/0037/2019 (NN, AdS, AC, JPM). This work was financed by FEDER – Fundo Europeu de Desenvolvimento Regional – funds through the COMPETE 2020 – Operational Programme for Competitiveness and Internationalisation (POCI) – and by Portuguese funds of FCT, under projects PTDC/FIS-AST/0054/2021 and UIDB/04434/2020 & UIDP/04434/2020. This work has made use of CosmoHub, developed by PIC (maintained by IFAE and CIEMAT) in collaboration with ICE-CSIC. CosmoHub received funding from the Spanish government (MCIN/AEI/10.13039/501100011033), the EU NextGeneration/PRTR (PRTR-C17.I1), and the Generalitat de Catalunya.

References

[1] Pavan kumar Aluri et al. "Is the observable Universe consistent with the cosmological principle?" In: *Classical and Quantum Gravity* 40.9, 094001 (May 2023), p. 094001. DOI: 10.1088/1361-6382/acbefc. arXiv: 2207.05765 [astro-ph.CO].

[2] R. S. Gonçalves et al. "Cosmic homogeneity: a spectroscopic and model-independent measurement". In: *MNRAS* 475.1 (Mar. 2018), pp. L20–L24. DOI: 10.1093/mnrasl/slx202. arXiv: 1710.02496 [astro-ph.CO].

[3] Xiaoyun Shao et al. "Cosmological constraints from angular homogeneity scale measurements". In: *European Physical Journal C* 85.3, 225 (Mar. 2025), p. 225. DOI: 10.1140/epjc/s10052-025-13987-4. arXiv: 2409.06009 [astro-ph.CO].

[4] T. M. C. Abbott et al. "The Dark Energy Survey Data Release 2". In: *ApJS* 255.2, 20 (Aug. 2021), p. 20. DOI: 10.3847/1538-4365/ac00b3. arXiv: 2101.05765 [astro-ph.IM].

[5] R. Euclid Collaboration: Scaramella, J. Amiaux, Y. Mellier, et al. "Euclid preparation. I. The Euclid Wide Survey". In: *A&A* 662, A112 (June 2022), A112. DOI: 10.1051/0004-6361/202141938. arXiv: 2108.01201 [astro-ph.CO].

[6] P. Fosalba et al. "The MICE grand challenge lightcone simulation - I. Dark matter clustering". In: *MNRAS* 448 (Apr. 2015), pp. 2987–3000. DOI: 10.1093/mnras/stv138. arXiv: 1312.1707 [astro-ph.CO].

[7] M. Crocce et al. "The MICE Grand Challenge lightcone simulation - II. Halo and galaxy catalogues". In: *MNRAS* 453 (Oct. 2015), pp. 1513–1530. DOI: 10.1093/mnras/stv1708. arXiv: 1312.2013 [astro-ph.CO].

[8] P. Fosalba et al. "The MICE Grand Challenge light-cone simulation - III. Galaxy lensing mocks from all-sky lensing maps". In: *MNRAS* 447 (Feb. 2015), pp. 1319–1332. DOI: 10.1093/mnras/stu2464. arXiv: 1312.2947 [astro-ph.CO].

[9] J. Carretero et al. "CosmoHub and SciPIC: Massive cosmological data analysis, distribution and generation using a Big Data platform". In: *Proceedings of the European Physical Society Conference on High Energy Physics. 5-12 July*. July 2017, 488, p. 488.

[10] Felipe Avila et al. "The scale of homogeneity in the local Universe with the ALFALFA catalogue". In: *JCAP* 2018.12, 041 (Dec. 2018), p. 041. DOI: 10.1088/1475-7516/2018/12/041. arXiv: 1806.04541 [astro-ph.CO].