Alma Mater Studiorum - Università di Bologna

School of Science
Department of Physics and Astronomy
Master Degree Programme in Astrophysics and Cosmology

Enabling cosmological analysis at BAO scale with a new matter anisotropic three-point correlation function emulator

Graduation Thesis

October 14, 2024

Presented by:
Kristers Nagainis

Supervisor:
Prof. Michele Ennio Maria Moresco

Co-supervisor:
Dott. Massimo Guidi

Dott.ssa Sofia Contarini

Academic year 2023-2024
Graduation date III

Abstract

Since the early 21st century, modern cosmology has been revolutionized by the discovery of the accelerated expansion of the universe [56, 53]. It is believed that dark energy is responsible for this acceleration, even though the nature of this mysterious component is still unknown, posing one of the most pressing open questions in cosmology. Furthermore, it was discovered that the luminous gravitational component is only a small fraction of the total one. The majority is dark matter, which also revolves around numerous unanswered questions [3]. Multiple methods have been developed to analyze the dark components of the Universe by different observables in the last twenty years [37]. Even though the cosmological parameters have been estimated with impressive accuracy [3], the nature of dark energy and dark matter still remains unknown. Despite that, the improvements over the different observational probes have led to disagreements in the acquired values, where the most notable is the Hubble tension [2].

Large-scale structures (LSS) have become one of the most important observational probes in cosmology. Analyzing their evolution and clustering, fundamental accuracy has been achieved for the determination of cosmological parameters [74]. This has been reached thanks to spectroscopic Stage III surveys probing the large-scale distribution of galaxies. Future data analyses for Stage IV surveys, such as Euclid [44] and the Dark Energy Spectroscopic Instrument (DESI; [1]) will be playing even more relevant contribution. These new surveys will bring an impressive amount of data, which allows moving on from the well-established method of constraining cosmological parameters by the Baryonic Acoustic Oscillations (BAO) with lower order statistics [54] to higher order methods. One such is the three-point correlation function (3PCF) – or bispectrum, in Fourier space – which offers multiple improvements over the two-point statistics: breaking the degeneracies between cosmological parameters, providing a better description of the non-Gaussian density field, and serving as an additional tool to constrain the cosmological parameters.

Modeling and measuring the bispectrum is easier than 3PCF from a computational point of view, and for this reason, there has been more activity in this field up to now [38]. On the other hand, the 3PCF offers several advantages with respect to the bispectrum, such as dealing better with the survey footprint, which is relevant for complicated surveys. Only recently, the gap between the bispectrum and 3PCF is being bridged from a theoretical point of view [30, 75, 28, 55], however, the computation of theoretical models remains a serious computational challenge. To extract cosmological information, different approaches have been explored, such as template fitting [46] or

full-shape analysis. Moreover, to be viable in cosmological analysis, the latter needs the development of efficient numerical methods, such as emulators, to significantly speed up its calculations. For these reasons, so far, few attempts have been made to exploit the cosmological contents present in the BAO feature in the three-point correlation function [50, 68], and only one pioneering work has started working in this direction using novel methods to further improve the characterization of higher-order statistics: developing an anisotropic description of the 3PCF [71].

The aim of this Thesis is to provide a robust method to explore the importance of the BAO features in the 3PCF, extensively studying its appearance and significance in the isotropic and anisotropic components, and use it to provide forecasts on the accuracy of cosmological parameters that can be derived from such analysis. Moreover, to fully explore the cosmological information contained in the anisotropic 3PCF, we go beyond this template-fitting approach by developing, validating, and testing a machine learning emulator for the matter anisotropic 3PCF.

A thorough analysis is performed to estimate the signal-to-noise ratio (SNR) of the anisotropic 3PCF, to derive which triangle configurations and multipoles provide the largest signal. It is found that the cosmological information is mostly contained in the isosceles and squeezed triangles. In addition, a new metric for the detectability of the BAO feature in the anisotropic 3PCF is developed, allowing us to assess the configurations that maximize this information. In this case, we confirm that squeezed triangle configurations are the best ones, in particular with sides 60, 80, 100, 120, or 140 h^{-1}Mpc, depending on the multipole. The highest detectability was at the 100 h^{-1}Mpc. For the isosceles triangles, the highest signal was around the configuration $r_{12} = r_{13} = 100\ h^{-1}$Mpc.

A step forward is made by developing a machine learning-based emulator for the matter anisotropic 3PCF, significantly reducing computational costs while maintaining high accuracy. The emulator is constructed for the redshift space 3PCF in TripoSH basis [70], where it is decomposed into eight different multipoles - four for the isotropic and four for the anisotropic multipoles of 3PCF. The emulator is 10 million times faster than the regular computation while maintaining sub-percent accuracy.

Based on these results, Fisher forecasts were generated for a Stage IV spectroscopic survey with a volume of 43 h^{-3}Gpc^3 - corresponding to the Euclid DR3 full light cone volume [44] - , using the developed 3PCF emulator for the Ω_m, h, and A_s cosmological parameters. In addition, the analysis for each multipole was performed. It was discovered that the multipole $\ell_1 = 3, \ell_2 = 3, L = 0$ has the highest constraining power, but, more importantly, the combination of anisotropic and isotropic multipoles gives better constraints than only the isotropic component. The improvement of constraining power for the anisotropic and isotropic over only isotropic multipoles is around 40%. Instead, if we apply the condition of $r_{min} > 40\text{h}^{-1}$Mpc, then the improvement is only around 8%. These forecasts were performed with a large volume of $43h^{-3}$Gpc^3 and a theoretical covariance, thus the errors are underestimated, but the relative comparison maintains true.

This thesis is organized as follows:

- In Chapter 1, we introduce the ΛCDM model, the origin of BAO and the cosmological parameters, that describe the Universe. Further on, the observational effects, such as RSD and Alcock-Paczyński effect are analyzed, which is followed by open scientific questions in cosmology. Then, the aim of the thesis is portrayed.

- In Chapter 2, the clustering of matter and galaxies will be discussed. First, a brief overview of the 2PCF and the power spectrum will be presented, followed by an in-depth discussion of the 3PCF and the redshift space 3PCF. Additionally, various decomposition methods for the 3PCF will be introduced.
- Chapter 3 focuses on emulators and their construction, starting with an emulator for the power spectrum, including both no-wiggle and only-wiggle versions. The chapter will then introduce the 3PCF emulator, discussing its accuracy and the modifications made to the Python library *CosmoPower* to enhance its functionality.
- In Chapter 4, we delve deeper into the analysis of the 3PCF, examining the signal-to-noise ratio (SNR) of the 3PCF and the detectability of BAO. This analysis will provide insights into the utility of the 3PCF for cosmological studies.
- Chapter 5 presents Fisher matrix forecasts using the newly developed 3PCF emulator. This section includes a detailed analysis of each 3PCF multipole, evaluating their constraining power and identifying potential limitations.
- Finally, in Chapter 6, we draw conclusions based on the results obtained throughout the thesis and propose directions for future research.

Contents

List of Figures

List of Tables

1

Introduction

1.1 The ΛCDM Universe

There have been many ideas and descriptions of our universe throughout time. In the 20th century, we developed the ΛCDM model to describe our universe, which has been very successful and is considered to be the best description of the Universe we currently have. The name itself contains the cornerstones of this model:

- Cosmological constant Λ: this is the parameter in the model that describes the dominant component of today's universe: dark energy. It is also responsible for the accelerated expansion that we are experiencing in the world that we are living in.

- Cold dark matter: this is the second-largest component of the universe, which is dark matter. It has many probes to prove its existence, but it has never been directly observed due to its non-interactive nature. Nevertheless, from today's structure of the Universe and its evolution being *bottom-up* it must be "cold", therefore massive.

- Baryonic matter: The last component is baryonic matter itself, which is understood the most out of the previously mentioned components, as it is visible and can be directly measured.

All these components, together also with the radiation component, contribute to the evolution of the Universe in one way or another. To describe the overall evolution of the universe, we can use Friedmann's equations. They are derived from Einstein's field equations of General Relativity under the assumption of a homogeneous and isotropic universe, described by the Friedmann-Lemaître-Robertson-Walker (FLRW) metric.

Assuming the FLRW metric and a perfect fluid energy-momentum tensor, Einstein's field equations simplify the Friedmann equations, which govern the expansion dynamics of the universe. The first Friedmann equation relates the expansion of the universe or the Hubble parameter with the energy content of the universe:

$$H^2 \equiv \frac{\dot{a}}{a} = \frac{8\pi G}{3}\rho - \frac{kc^2}{a^2} + \frac{\Lambda c^2}{3}, \tag{1.1}$$

where H is the Hubble parameter, $\dot{a}$ is the scale factor of the universe and $\dot{a}$ is the time derivative of it, ρ is the total energy density of the universe, k is the curvature constant, where the value -1 is for a hyperbolic, 0 is for flat and $+1$ is for closed universe, and finally Λ is the cosmological constant.

The energy density refers to the amount of energy contained in a given volume of space, which contributes to the dynamics of the universe's expansion. The total energy density consists of all the previously mentioned components of the universe, thus it would be defined as $\rho = \rho_r + \rho_m + \rho_\Lambda$. It is the sum of the radiation, matter, and dark energy components accordingly. Each of the component evolves differently with the expansion of the universe. For example, $\rho_r \propto a^{-4}$, $\rho_m \propto a^{-3}$ and $\rho_\Lambda \propto const.$

Not only the energy density is a significant component for the expansion of the universe, but it also dictates the geometry of our Universe. We can define the critical density as:

$$\rho_{crit} = \frac{3H_0^2}{8\pi G}. \tag{1.2}$$

If our Universe's energy density value is exactly as the critical energy density, then the geometry of the universe will be Euclidean. Any deviation from it changes the geometry — a higher energy density would be attributed to a spherical or closed universe, but a smaller one would be for a hyperbolic universe.

With ρ_{crit} in hand, we can also define the density parameter as:

$$\Omega_s = \frac{\rho_s}{\rho_{crit}}, \tag{1.3}$$

where the subscript s is for any species or component of the universe. This is useful, as it allows depicting all the energy densities with the same units. Thus, if we would sum all the density parameter components together, according to the ΛCDM model, we would get the value of 1, which means that we live in an Euclidean Universe.

By introducing the new definitions of the energy density parameter, equation 1.1 can be reformulated to help analyze the expansion history of the universe based on its various components:

$$\frac{H(t)^2}{H_0^2} = \sum_{s=r,m,\nu,DE} \Omega_s a(t)^{-3(1+\omega)}, \tag{1.4}$$

where the parameter ω denotes the equation of state, which is different for each of the components. For perfect fluids, it ranges from $-1/3$ to $+1$, and the value for dark energy is considered to be outside of this range. This equation is also assuming Euclidean geometry, as it does not include a parameter for the curvature of the universe.

The second Friedmann's equation describes the acceleration of the universe, and it is expressed as:

$$\frac{\ddot{a}}{a} = -\frac{4\pi G}{3}(\rho + \frac{3p}{c^2}) + \frac{\Lambda}{3}. \tag{1.5}$$

It can be seen that the acceleration $\ddot{a}$ is dependent on the equation of state for the considered component.

The equations 1.4 and 1.5 are crucial in understanding the evolution of the universe, as they show the importance of precisely determining the cosmological components. Only from that, we can understand most of the history of the Universe in terms of its scale, which in turn is important for the growth of structures.

The ΛCDM is a very successful model, and it has not changed for the last 25 years. It has many reasons for its success, and one of such is its simplicity. It can be described by only 6 cosmological parameters - density parameters $\Omega_b h^2$ and $\Omega_{DM} h^2$, dimensionless Hubble parameter h, the ratio of the sound horizon to the angular diameter distance to the last scattering surface θ_*, scalar amplitude of the primordial density perturbations A_s and the slope of the initial power spectrum n_s [62].

Another key reason for its success is that these parameters have been precisely measured through multiple independent experiments, with the *Planck* satellite providing the most notable data on the CMB [3]. Beyond this, the ΛCDM model has been remarkably successful in predicting the large-scale structure of the Universe, such as galaxy clustering, the CMB anisotropies, and the accelerated expansion due to dark energy, confirmed through supernova observations. Its predictive power spans across a wide range of probes, including Baryonic acoustic oscillations (BAO) and structure formation, further solidifying it as the best model we currently have.

1.2 Early Universe and the Baryonic Acoustic Oscillations

If we simply extrapolate the evolution of the Universe back in time from Friedmann's equations, we are faced with multiple problems, such as the monopole, horizon, and flatness problem [14]. In short, it predicts that we should have had as many magnetic monopoles as baryons, the Universe should not have been homogeneous and isotropic, and the flatness of our Universe could have only been achieved by extreme fine-tuning of the initial conditions of the Universe. In order to address and solve these issues, the inflationary paradigm was introduced [33]. Even though there are many theories of the possible inflationary descriptions, all of them result in a random density field for the initial density fluctuations [78]. Many of these theories [49] and observations [5] [4] predict random Gaussian field, produced by Gaussian quantum fluctuations in the scalar field. At the same time, there is also the possibility of a non-Gaussian density field emerging from the inflationary epoch [81, 9], which has not yet been confirmed or denied from modern observations [16].

Thus, from inflation, we have radiation and matter that is not perfectly homogeneous, but with random overdensities and underdensities throughout the volume of the Universe. At the same time, the baryonic matter was ionized, due to the very hot environment, therefore the electrons were not bounded by the nuclei. The charged particles interact with the photons, making radiation and baryonic matter coupled together, called photon-baryon fluid.

The gravity tries to compress the overdensities, while radiation pressure acts against it. This interplay between gravity and radiation pressure creates sound waves, that can propagate through the fluid. It is important to note that dark matter did not behave in the same manner as the photon-baryon fluid, because it does not interact via the electromagnetic force, thus it was not affected by the radiation pressure.

As the baryonic density fluctuations are present at all scales, each of them evolves differently in time. To be exact, the frequency of oscillation is equal to the wavenumber times the speed of sound [36]. All the scales oscillate until the time of recombination — the time when baryons become neutral. When baryons and radiation are not bound to act as a fluid anymore, the photons become free. At this moment all the oscillations "freeze", and going further on, the Universe evolves with this imprint on the density fluctuations.

The dark matter, even though it did not oscillate during this period, due to gravitational interaction it followed the imprint of the baryonic oscillations. The same happened for baryons, which followed the large overdensities of dark matter at smaller scales [39]. In the late epochs of the universe, both baryons and DM have caught up to each other, and now exhibit nearly equal profiles of their scale of overdensities.

1.3 Observational probes

1.3.1 Baryonic acoustic oscillations

The previously mentioned imprints from the baryons in the distribution of galaxies are called the Baryonic Acoustic Oscillations (BAO). They appear at a specific scale - 150 Mpc. This is due to the fact that the initial overdensities propagated as a spherical sound wave until the time of recombination. The speed of sound was three times smaller than the speed of light due to the extreme environment, and taking into account the given time and speed, the BAO were frozen at 150 Mpc and persisted throughout time. A visualization can be seen in Fig. 1.1, where it is depicted how the BAO features are imprinted randomly all through the Universe.

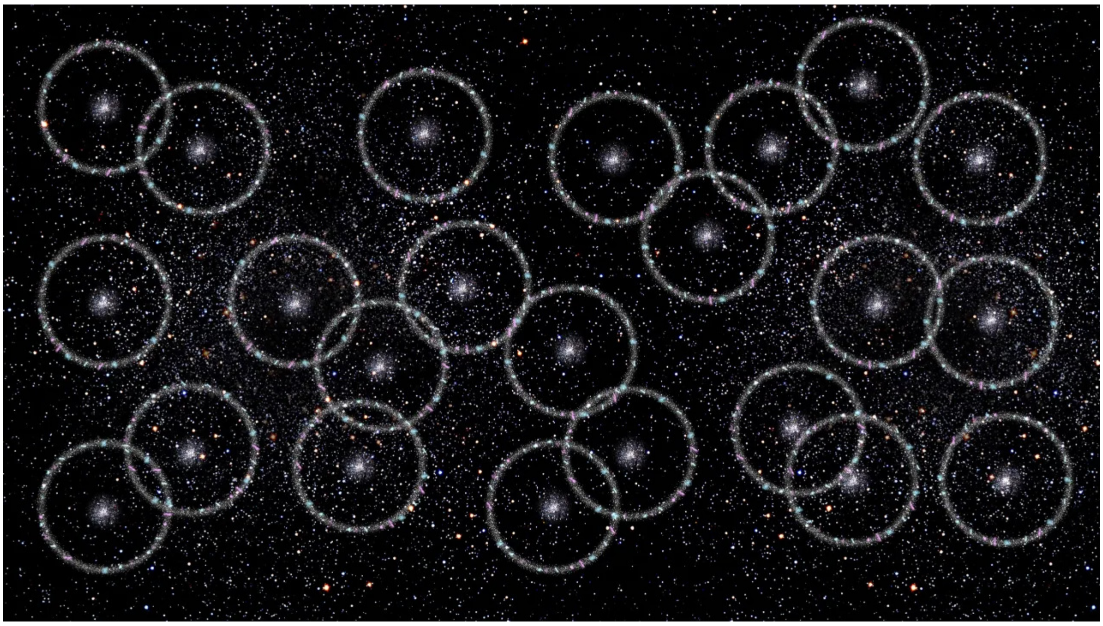
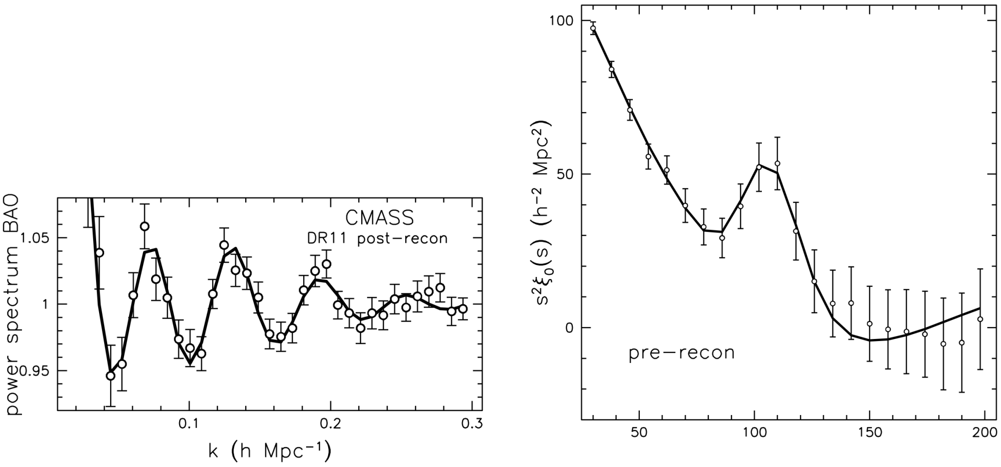


Figure 1.1. Illustration of the Baryonic Acoustic Oscillations in the late universe. Image credit: Gabriela Secara, Perimeter Institute.

BAO can be directly observed from the distribution of galaxies in the Universe, where this information is typically compressed in the form of the two-point correlation function (2PCF) in configuration space, or in its analog in Fourier space, the power spectrum (both will be described more extensively in Sect. 2.2). Baryonic acoustic oscillations appear as wiggles in the power spectrum of galaxies, and in a distinctive peak in the 2PCF, as shown in Fig. 1.2.

Each cosmological parameter has different effects on the initial conditions of the density perturbations or the evolution of clustering. We will narrow it down only to the parameters that are used in this work further on - Ω_m, Ω_b, h, A_s, n_s -, with the assumption of a flat Universe, therefore $\Omega_{tot} = 1$. One of the most impactful parameters is the Ω_m, which includes the contribution of both dark and baryonic matter. From the assumption of flatness, it implies a balance between the contribution to the total energy density of dark matter and dark energy. Hence, the larger the relative amount of Ω_m, the smaller Ω_{DE} is. Dark energy does not play a significant role in the early Universe, as the dominant energy density is due to radiation or matter. However, its influence on the geometry and evolution of spacetime grows over time, becoming increasingly significant as the universe expands.

The early Universe is greatly influenced by the variation of Ω_m as well. The fact that



(a)

(b)

Figure 1.2. BAO in N-point statistics: **(a)** Power spectrum of CMASS DR11 galaxies, normalized by no BAO power spectrum. The black line represents the best model fit. Image taken from [7]. **(b)** Two-point correlation function of the same dataset. Image taken from [7].

we have larger Ω_m, means that the matter domination era will start sooner. In turn, this means that the recombination and the CMB will have happened earlier, thus also the BAO peak would be on shorter scales. Furthermore, the larger the Ω_m, the bigger the gravitational wells in the primordial density fluctuations. This would result in a larger amplitude of these density waves, which would increase the growth of structures. Then, if the baryonic component is fixed and only the dark matter is increased, it actually reduces the amplitude of the oscillations, due to the driving effect from the pressure exerted by photons. This greatly influences the amplitude and position of the BAO.

Moreover, Ω_b plays a crucial role in the formation and evolution of cosmic structures. During the oscillatory behavior of the baryons, the increase of Ω_b would result in larger baryon loading. This means that during the compression of baryons, they would manage to compress even more due to the additional gravitational effect. Instead, the rarefaction is not influenced. This can be understood by imagining a spring with a point-like distribution of baryons attached at the end. If they are released from the same height, they will also come back to the same height in a perfect world. This is true even if the mass distribution would be heavier, causing it to extend lower due to the additional mass. The same behavior is also for the oscillations in the gravitational wells. Thus, the baryonic loading changes the compressed peaks in the BAO, therefore only the amplitude is affected. This would result in peaks that have higher contrast between each other.

In addition to these, there are the cosmological parameters that directly describe the primordial fluctuations, namely A_s and n_s. A_s is the parameter for the amplitude of the primordial power spectrum, thus, by increasing this amplitude, the clustering is increased across all scales. Therefore, the BAO would also be more prominent. It is important to note that the positions of the BAO would be left unaffected.

On the other hand, the scalar spectral index n_s affects different scales differently. It describes how the density fluctuations vary with scale. If its value is unity, then the variations are the same on all scales. Otherwise, a lower tilt would result in less clustering for the small scales and more for the large scales. From our current measurements, it seems not to be exactly scale-invariant, as the value is found to be $n_s = 0.96$ [3].

Lastly, we have the dimensionless Hubble parameter h. As discussed in the equation 1.4, the expansion of the Universe is affected by every component in the Universe. Logically, the larger h is, the faster the expansion, the bigger the opposing force for the clustering, thus it would result in less clustering for the galaxies. Furthermore, the Hubble parameter is a crucial ingredient in the definition of distances in cosmology. Therefore, each time we will report the distance or scale, it will be inversely proportional to the Hubble parameter.

As discussed above, the value of every cosmological parameter has a great impact on the evolution of our Universe. By obtaining the exact values of these parameters, we can confirm or disregard different theories of our Universe and be closer to understanding the place we live in. There are many ways to obtain the values of the cosmological parameters from observations, and among the most prominent ones are the aforementioned N-point statistics, such as the power spectrum or two-point correlation function. However, these theories are typically based on matter, but we can only observe luminous objects, such as galaxies, which are biased tracers of the underlying distribution of matter.

1.3.2 Biased tracers

To obtain any constraints on cosmological parameters from N-point correlation functions (described at length in section 2.2), either observations or simulations are needed. On the one hand, simulations have the advantage that the statistical tools can be directly applied to matter, thus, including dark matter. The disadvantage is the fact that simulations are never perfect, as they are based on several assumptions about the universe. On the other hand, it is also possible to analyze real observations, such as the ones obtained from large spectroscopic surveys: the Sloan Digital Sky Survey (*SDSS*) [82], 2dF Galaxy Redshift Survey (*2dFGRS*) [17], Dark Energy Survey (*DES*) [15], Baryon Oscillation Spectroscopic Survey (*BOSS*) [21], Euclid [44]. Different surveys focus on different targets, as well as different redshift bins.

When considering these surveys, the most obvious tracers of matter are galaxies. At first glance it seems to be a straightforward target - they are luminous objects, and they can be observed up to 300 million years after the Big Bang, so they cover a large portion of the timescale of the Universe. Galaxies form in dark matter halos, and they represent the best tracers to track the distribution of matter. There are several caveats to this, the first being non-linearity in the evolution of structures. Even if the primordial density field was Gaussian, galaxies interacted and evolved, and now non-linearity must be included in the theories, which complicates the analysis.

The most simple relation to link the distribution of galaxies and matter is the following:

$$\delta_g \boldsymbol{x} = b\delta_m(\boldsymbol{x}), \tag{1.6}$$

where b is the linear galaxy bias parameter. Essentially, if b is smaller or larger than one, it means that galaxies cluster less or more than dark matter.

This is only the linear term for the galaxy bias, but there can also be higher-order bias parameters, which account for the non-linear relationship between galaxy and matter overdensities. It can be expressed as:

$$\delta_g \boldsymbol{x} = b\delta_m(\boldsymbol{x}) + \frac{b_2}{2}\delta_m^2(\boldsymbol{x}) + \frac{b_3}{6}\delta_m^3(\boldsymbol{x}) + ..., \tag{1.7}$$

where the b_2 and b_3 are the non-linear bias parameters. Additionally, there is the stochastic bias, which adjusts for the fact that the relationship between the δ_g and δ_m is not perfectly deterministic. It can be expressed as $\delta_g \boldsymbol{x} = b\delta_m(\boldsymbol{x}) + \epsilon$.

If we include primordial non-Gaussianity, then we can have additional bias terms, that can include scale-dependent bias or non-local bias. These are the biases that are from the imprint of the initial conditions of the Universe [13]. Non-Gaussianity can also appear from the non-linear gravitational effects from later evolution, but the biases can be disentangled from both sources of non-Gaussianity [8]. As it can be seen, there can be several bias terms for the observed overdensities, that are usually determined from the data. Hence, for a precise analysis, they must be added as free parameters to the model.

Last but not least, the surveys do not measure the actual distances of the galaxies, but rather the redshift. This directly impacts our observations by introducing additional complexities, such as the Alcock-Paczyński (AP) and Redshift-Space Distortion (RSD) effects, which must be accounted for in the observational analysis.

1.3.3 Redshift-space distortion

The RSD is an important effect, that produces additional clustering of the galaxies and changes the actual shape of the N-correlation functions or their counterpart in the Fourier space. It can be explained by Fig. 1.3, where the effect has been separated into two different scales - the large and the small one, therefore representing the linear and the non-linear effect. The linear effect squeezes the density contours, making them more elliptical. The non-linear effect is more local, as it is also on smaller scales, and it gives large displacements along the line of sight, thus giving the name "Finger of God".

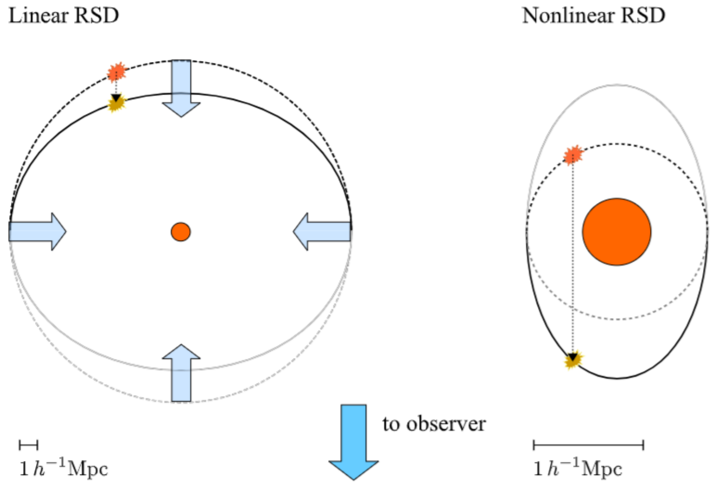


Figure 1.3. Visualization of the RSD effects in linear and non-linear scale from [23]. On the left side is the linear scale, where the real, circular contour of constant density, represented by the dashed line, is deformed into a squished ellipse, represented by the solid black line. The arrow shows the velocity flow, and the overdensity is shown as the extended shape on the contours. On the right side is the non-linear scale, therefore small scale, where the displacement of the overdensity is larger. This effect is also known as the "Finger of god".

To describe it mathematically, as represented in [23], we have to define how the observed location differs from the real one. The observed coordinate $\boldsymbol{x}_{obs}(z,\theta,\phi)$ can be expressed as follows:

$$\boldsymbol{x}_{obs}(z,\theta,\phi) = \boldsymbol{\chi}_{fid}(z)\hat{\boldsymbol{n}}(\theta,\phi), \tag{1.8}$$

where θ and ϕ are the positional angles on the projected celestial sphere defining the location of the object, $\hat{\boldsymbol{n}}$ is the unit vector depending on the θ and ϕ and $\boldsymbol{\chi}_{fid}(z)$ is the fiducial comoving distance, defined as $\boldsymbol{\chi}_{fid}(z) = \int_0^z \frac{dz'}{H_{fid}(z')}$. This means that it depends on two variables that we do not know well - redshift, as it also depends on the peculiar velocities, and the Hubble parameter, which is also not precisely known and is one of the parameters that we intend to constrain.

For comoving distance we can quantify the fact that we do not know the real cosmology as follows:

$$\chi_{fid}(z) = \chi(z) + \delta\chi(z), \tag{1.9}$$

where the real cosmology is $\chi(z)$ and $\delta\chi(z)$ is the departure from it. Similarly, we can also quantify the effect of peculiar velocities on the redshift as:

$$1 + z = \frac{1}{a_{em}}[1 + u_{||}], \quad \text{where } u_{||} = \boldsymbol{u_g} \cdot \boldsymbol{\hat{n}}, \tag{1.10}$$

where u_g is the galaxy peculiar velocity in the linear order, which quantifies the Doppler shift. Here we also assume that galaxy and matter peculiar velocities follow the relation $u_g \approx u_m$, which is a valid assumption on large scales. This is a useful approximation, as now we do not have to use velocity bias.

It is possible to estimate the errors for the observed coordinate in equation 1.8. Error for peculiar velocity:

$$\Delta\boldsymbol{x_{RSD}} = \frac{\partial \boldsymbol{x}}{\partial u_{||}} u_{||}\boldsymbol{\hat{n}} = \frac{1}{aH} u_{||}\boldsymbol{\hat{n}}, \tag{1.11}$$

where it can be seen that this effect only acts in the line of sight direction. If $u_{||} > 0$, then the distance appears larger due to additional Doppler shift, and the opposite is true for $u_{||} < 0$. The final deviation from the true coordinate can be expressed by combining the equation 1.11 and 1.9, acquiring:

$$\boldsymbol{x_{obs}} = \boldsymbol{x} + \left(\delta\chi(z) + \frac{u_{||}(\boldsymbol{x})}{aH}\right)\boldsymbol{\hat{n}}. \tag{1.12}$$

This equation expresses the implications of our lack of knowledge about the peculiar velocities and cosmology. With the obtained relations and further calculation of the Jacobian [23], it is now possible to see how RSD affects the observed overdensities.

$$1 + \delta_{g,\mathrm{obs}}(\boldsymbol{x}_{\mathrm{obs}}) = \bar{J}\left[1 + \delta_g(\boldsymbol{x}[\boldsymbol{x}_{\mathrm{obs}}]) - \frac{1}{aH}\frac{\partial}{\partial x_{||}} u_{||}(\boldsymbol{x}[\boldsymbol{x}_{\mathrm{obs}}])\right], \tag{1.13}$$

where $\bar{J}$ is the Jacobian defined at fixed, mean redshift, as usually surveys are looking at a narrow slice of redshifts, therefore all the parameters of the Jacobian depended on the average redshift bin $\bar{z}$. If the equation 1.13 would be used in the calculation of the N-point statistics, then it would affect the shape of the two- or three-point correlation functions. We can go further, and relate the δ_g to matter density δ_m. This approximation is valid, as usually surveys have small angular scales. If not, then distant observer approximation cannot be used. Using all this, in the end, we acquire the following relation:

$$\delta_{g,RSD}(\boldsymbol{k}) = \delta_m(\boldsymbol{k}) \left[b_1 + \mu_k^2 f \right], \tag{1.14}$$

where f is the growth rate defined as $f = \frac{dlnD}{dlna}$ and μ_k is the cosine between the line of sight and the wavevector $\boldsymbol{k}$. The equation 1.14 again states that in RSD we observe larger overdensities, due to the additional factor of $f\mu_k^2$. In addition, larger overdensities are also observed if the wavevector is along the line of sight. Equally, there will be no RSD effects if the $\boldsymbol{k}$ is perpendicular to LOS.

By measuring either the power spectrum or bispectrum, it is possible to disentangle the b_1 and f values from each other. For power spectrum there is the problem that the matter power spectrum is not known, therefore we can only get the constraints on $b_1\sigma_8$ and $f\sigma_8$, where σ_8 is the amplitude of density fluctuations in the universe on scales of 8 Mpc. If the bispectrum is used, then also the σ_8 can be disentangled.

1.3.4 Alcock-Paczyński effect

Alcock-Paczyński (AP) effect is due to the assumption of an incorrect cosmology in our measurements, which is inevitable as our current knowledge of the universe is not perfect. For this reason, an uncertainty is introduced in the derivation of the comoving distance. This effect is visualized in Fig. 1.4.

To quantify the AP effect, we will separate the transverse coordinates x^1, x^2 and longitudinal x^3. We will again use the distant observer approximation, and choose the reference point of our coordinate system as $\boldsymbol{x}_{obs} = 0$ and $z = \bar{z}$, which is the mean value of the observed redshift slice. First, the real and observed transverse coordinates can be expressed as:

$$(x^1, x^2) = \chi(z)(\theta^1, \theta^2), (x^1_{obs}, x^2_{obs}) = \chi_{fid}(z)(\theta^1, \theta^2), \tag{1.15}$$

where (θ^1, θ^2) are the observed galaxy position in the celestial sphere. Inserting the equation 1.9 in the real coordinate equation 1.15, we obtain:

$$(x^1, x^2) = \left[1 - \frac{\delta\chi(z)}{\chi_{fid}(z)} \right] (x^1_{obs}, x^2_{obs}). \tag{1.16}$$

This gives us the relation between the real and observed transverse coordinates due to the AP effect. The line of sight coordinate x^3 is different, as it only depends on the

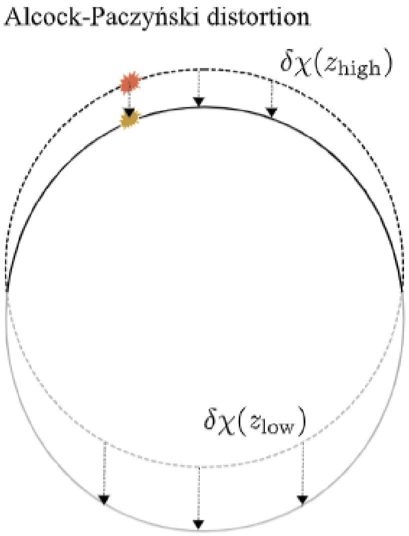

Alcock-Paczyński distortion
$\delta\chi(z_{\rm high})$
$\delta\chi(z_{\rm low})$

Figure 1.4. Visualization of the AP effect from [23]. The true constant density contour, represented by the dashed line, is displaced by the factor $\chi(z)$ to the solid line density contour, which is the observed one. This produces an elliptic shape because the displacement is dependent on redshift, thus galaxies closer would be affected more by the assumption of the wrong cosmology.

redshift, and we have assumed that $x^3 = 0$. Therefore:

$$x^3_{obs} = \chi_{fid}(z) - \chi_{fid}(\bar{z}) \backsimeq \frac{1}{H_{fid}(\bar{z})(z-\bar{z})}, \tag{1.17}$$

where the expression on the left was obtained from the linear order Taylor expansion. Now expressing the real coordinate x^3 with similar manipulation as in for the transverse coordinates, we obtain:

$$x^3(z) = \left(1 - \frac{\delta H(\bar{z})}{H_{fid}(\bar{z})}\right) x^3_{obs}. \tag{1.18}$$

Comparing the equation 1.18 and 1.16, it is clear that they are different from each other, which explains why we obtain elliptical distortion from the AP effect.

To summarize, the coordinates can be expressed as follows:

$$\boldsymbol{x}(\boldsymbol{x}_{\rm obs}) = \Big((1-\alpha_\perp)x^1_{obs}, (1-\alpha_\perp)x^2_{obs}, (1-\alpha_\parallel)x^3_{obs}\Big) \tag{1.19}$$

, where $\alpha_\perp$ and $\alpha_\parallel$ is defined as:

$$\alpha_\perp = \left.\frac{\delta\chi}{\chi_{\rm fid}}\right|_{\bar{z}}, \quad \alpha_\parallel = \left.\frac{\delta H}{H_{\rm fid}}\right|_{\bar{z}}. \tag{1.20}$$

Similarly to RSD, also AP's influence on the overdensities can be included. They can be implemented as a redefinition of the real and observed wavevector, which in the end yields the combined observed overdensities:

$$\delta_{g,\mathrm{obs}}(\boldsymbol{k}_{\mathrm{obs}}) = \left[b_1 + f\mu_k^2\right]\delta_m(\boldsymbol{k})\Big|_{\boldsymbol{k}=\left([1+\alpha_\perp]k^1_{\mathrm{obs}},\,[1+\alpha_\perp]k^2_{\mathrm{obs}},\,[1+\alpha_\parallel]k^3_{\mathrm{obs}}\right)}. \tag{1.21}$$

The equation 1.21 states that even if there would be no RSD, which we can induce by saying that $f = 0$, we would still observe the AP effect in our power spectrum or bispectrum. It causes anisotropies in the observed spectrum, and both of these effects - AP and RSD - can be disentangled from the observations.

Furthermore, $\alpha_\perp$ and $\alpha_\parallel$ can be determined from BAO observations. The peak of BAO is well known, as it lies around $150Mpc$. Not only does it give information about the cosmological parameters on its own, but also through AP effects. In particular, as the true position of the peak is known, it is possible to calibrate the observed peak, thus giving the values of $\chi(\bar{z})$ and $H(\bar{z})$.

To condense, RSD and AP effects are not a liability, but they give additional constraints on cosmological parameters. They change the shape and amplitude of the measured correlation function, such that it is more difficult to appropriately model them. Nevertheless, from RSD there is the possibility to obtain estimates of f and b_1, but from AP - $\chi(\bar{z})$ and $H(\bar{z})$.

1.4 Open scientific questions

While significant progress has been made in understanding the early universe, many aspects remain unknown. Observational effects such as RSD and the AP portray the limitations of our current knowledge and show the complexities present in cosmological studies. These effects are not the only ones affected by our lack of knowledge and are only a fraction of the gap in our understanding of various domains of cosmology. Among the most notable open questions is the nature of the initial quantum fluctuations following cosmic inflation - specifically, whether these primordial fluctuations were Gaussian or exhibited non-Gaussian characteristics.

The characteristics of primordial fluctuations are not only crucial to understand the evolution of structure in the universe, but also to understand better the inflation. As of now, there are many models of inflation [47], and none have yet been confirmed or denied, as it is impossible to have direct observations of it. Therefore, we are left to work with the aftermath of the inflation, measuring different cosmological parameters, that are indicating of one model or another. For example, the primordial amplitude A_s and the spectral index n_s give direct information about the inflation, as well as the curvature, growth factor, and other parameters. These can give valuable information, that enables us to understand better the physical properties of inflation.

Another crucial open question in cosmology concerns the nature of dark energy. It is still uncertain even if it exists, as its effects might be just the modifications of the general relativity in largest scales [63]. This is due to the fact that there are multiple unsolved problems regarding dark energy. One is denoted as the biggest problem in physics - the cosmological constant problem - as it has the largest difference between the theory and observations, having the difference up to the order of 120. From observations, we have found that the energy density of the cosmological constant must be 10^{-47} GeV [19], but the quantum field theory predicts 10^{74} GeV, if we assume that it originated from the vacuum energy density. This results in extreme *fine-tuning problem*.

There is also the *coincidence problem* because in today's epoch, the dark energy and dark matter energy density parameters have values with a similar order of magnitude: 0.7 and 0.3 respectively. This would again require considerable fine-tuning in the early days of the universe, as then the ratio of both density parameters was very different.

Apart from the dark energy, there is also dark matter, and here we do not know what exactly it consists of. It has been widely accepted as the largest component of matter, which is responsible for the galaxy rotation curves, where the velocity stays constant over larger radii, as well the large masses of the galaxies and also the observations of CMB and theory of structure formation, which requires the presence of dark matter [11]. Nevertheless, we still do not know the mass of the DM particle, its physical properties, or even the absolute certainty of its existence.

This brings us to the Hubble tension, which also exhibits the discrepancies between the early and late universe observations. From the CMB observations [3], therefore the early universe, the value of the Hubble constant is 67.7 km/s Mpc^{-1}. On the other hand, from the Cepheids, Supernovae Type Ia, and other late-time observations it is 73.2 km/sMpc^{-1} [61]. This discrepancy is with 4σ to 6σ certainty [22]. The late observations directly measured the Hubble constant, but the early measurements had to base it on the ΛCDM model. This brings another issue with the overall accepted model.

Modern theories and observations of cosmology have found the best model - ΛCMD -, which can explain most of our universe. However, there are still many problems with it, which mostly revolve around inflation, dark matter, and dark energy. It is evident that we have still not found the correct model to describe our universe, and we must seek the answers to the previously laid out questions, to be closer to truth.

1.5 Aim of the thesis

There are many questions to answer about our Universe - what is the nature of dark energy and dark matter, why do we get different values for the Hubble parameter, what was the evolutionary path of large-scale structures, and many more. One of the best ways to seek answers to these questions is by analyzing Large-Scale Structures through the clustering of galaxies and matter. If we can understand how structures form throughout the evolution of the Universe, we can unveil more about inflation, dark energy, dark matter, and also other domains of cosmology. Understanding the physics of structure clustering allows us to understand the totality of the physical laws governing the whole universe, as well as its evolution.

So far, one of the most widely used methods to extract cosmological information from the galaxy clustering of the LSS is the N-point statistics. In particular, the two-point correlation function (2PCF), or its counterpart in Fourier space, the power spectrum, describes how matter or galaxy pairs are distributed relative to each other in the Universe. This has been extensively studied and applied, but the next order - three-point correlation function (3PCF) or bispectrum -, which describes the distribution of triplets, has yet to realize its potential.

3PCF has several advantages over the 2PCF. Firstly, the statistical properties of a Gaussian field are fully described by the two-point statistics. Thus, if there is a non-Gaussianity in the galaxy distribution, which can arise from the primordial initial conditions, the non-linear evolution of matter in the late Universe, or the biased relation between matter perturbations and luminous tracers, we need higher-order statistics. Hence, it can be used to model galaxy bias parameters beyond the linear order, as well as break degeneracies between cosmological parameters (in comparison with the 2PCF), for example, the linear bias b_1, the growth factor f and σ_8 [32, 76]. Lastly, it can be used as an additional tool to analyze the structure clustering and constrain cosmological parameters, therefore increasing the accuracy of these analyses. Despite offering the same information content, the 3PCF also has a crucial advantage over its Fourier counterpart - bispectrum -, as the 3PCF does not suffer from mode coupling and better deals with the survey footprint.

Despite its advantages, the computation of the 3PCF is significantly more complex and computationally demanding than the 2PCF or even the bispectrum. The computational cost becomes a large barrier, as one model can take up to 48 CPU hours, therefore it can take years to use it for the constraints of cosmological parameters. Moreover, the anisotropic component of the 3PCF has not been analyzed thoroughly, thus the SNR and BAO feature appearance within it is still poorly explored, as well as the anisotropic contribution for accuracy in the cosmological parameter constraining is yet to be understood.

To analyze these fundamental questions, we use the modeling of the anisotropic 3PCF as proposed in [70]. It enables us to analyze separately both the multipoles of the isotropic and the anisotropic components of the 3PCF, efficiently unraveling the information content within them. Nevertheless, the computational time of this model remains significant.

This is where the development of a 3PCF emulator becomes crucial. Emulators are machine learning-based models designed to predict the results of computationally expensive simulations or models with high accuracy and in a short amount of time. By training an emulator on pre-computed model data, it becomes possible to predict the 3PCF efficiently for a wide range of cosmological parameters, dramatically reducing the computational cost.

The usage of an emulator allows to perform different kinds of analysis. It can serve as a tool to analyze the impact of cosmology on the shape of the 3PCF, or to forecast the accuracy of cosmological parameter constraints.

To summarize these critical issues, which must be addressed, the aim of this Thesis is:

- Developing an emulator for the redshift space 3PCF;
- Using the developed models for the anisotropic 3PCF, assess and analyze the SNR of 3PCF and investigate the BAO signal and detectability;
- Using the emulator, produce forecasts for the cosmological parameters from the 3PCF and evaluate the impact of different triangle configurations on the constraining power.

2

Matter and galaxy clustering in the Universe

Understanding the clustering of large-scale structures of the Universe is one of the key tasks in modern cosmology. To depict its complexity, many aspects need to be taken into consideration: the governing physical laws and physical processes involved, the shape of the Universe, the components in the Universe and their nature, as well as the initial conditions of the hot Big Bang.

To understand the framework of clustering, perturbation theory (PT) [10, 23, 14] must be applied. The foundation of it lies within the gravitational instability and the gravitational amplification of the primordial density fluctuations. Then these theories can be verified or refuted by observations of the large-scale structures, which are best described by different statistical tools.

2.1 Different approaches to clustering analysis

To compare our observations of the LSS with the Standard perturbation theory or Lagrangian perturbation theory [10], it is necessary to develop the appropriate statistical tools. Statistics is needed for several reasons [10]: firstly, as discussed before, the large-scale structures have evolved from primordial density perturbations. Nowadays it is impossible to observe them, therefore we do not have a direct probe of the initial conditions. Secondly, the timescales over which the overdensities evolve are orders of magnitude larger than our possibility to observe them individually, therefore we cannot follow the evolution path of a single object.

We model the observable Universe as a stochastic realization of a probabilistic ensemble, meaning it is just one of many possible outcomes that could have emerged. Any predictions about this realization must therefore rely on the statistical properties of the primordial perturbations. By applying statistical tools, we can better map the evolution of large-scale structures and compare these maps to our theoretical expectations, ultimately refining our understanding of the Universe's formation and evolution.

There are various statistical tools available in the market of cosmological analysis.

For example, Minkowski functionals [41] quantify the shape and structure of spatial patterns through geometric and topological measures, such as volume, surface area, curvature, and are particularly useful for capturing non-Gaussian features.

The J function [41] provides a scale-dependent measure of clustering by comparing the probability of finding a neighboring point within a given distance to that expected under complete spatial randomness, revealing clustering patterns and voids.

Wavelet phase harmonics (WPH) [6] is a newer method that captures phase correlations from wavelet transforms across multiple scales, offering a way to characterize non-Gaussian features in primordial density fluctuations.

With many other available methods, the most popular and widely used are the N-point correlation functions.

2.2 N-point correlation functions

Correlation functions are the standard method to describe the clustering of matter and galaxies. They quantify the clustering properties for a set of points that are spatially distributed in space. They can be distributed in a three-dimensional space, but also projected in two-dimensional space, for example, on the surface of a celestial sphere, which can still give valuable information. Correlation functions can be calculated either between galaxies, dark matter halos, or matter distribution. Even though galaxies are the easiest to observe due to their luminous nature, it is harder to develop the appropriate theories for it. However, matter cannot be probed directly, as a large portion of it is dark matter, which cannot be directly observed. Therefore, galaxies are used as tracers for matter evolution, but that comes with a cost, as it requires including more parameters, such as galaxy bias parameter, as well as other assumptions and uncertainties.

For now, we will explore a general overview of how to construct correlation functions [14]. Let us assume an average density of galaxies n_{gal}. The probability $\delta^2 P_2$ to find a galaxy in a volume δV_1 and another galaxy in δV_2 at a distance r_{12} is represented by the equation:

$$\delta^2 P_2 = n_{gal}^2 [1 + \xi(r_{12})] \delta V_1 \delta V_2 \tag{2.1}$$

where the correlation function $\xi(r_{12})$ represents the excess probability, over a uniform random distribution, of finding another galaxy at distance r_{12}, considering that the volumes δV were taken randomly within a representative volume. This means that if $\xi(r_{12}) = 0$, the galaxies in the representative volume are scattered randomly. If it is positive, then they tend to cluster at that distance, but if it is negative — galaxies avoid each other.

Following the equation 2.1, we can construct the mean number of galaxies in distance r such as:

$$\langle N\rangle_r = \frac{4}{3}\pi n_{gal} r^3 + 4\pi n_v \int_0^r \xi(r'_{12}) r'_{12} dr'_{12} \tag{2.2}$$

where the second term shows the excess over a uniform random distribution.

In the case of a two-dimension catalog, we have to opt for the angular correlation function, which can be defined as the following equation:

$$\delta^2 P_2 = n_\Omega^2[1 + w(\theta_{12})]\delta\Omega_1\delta\Omega_2 \tag{2.3}$$

Where $w(\theta_{12})$ is the angular two-point correlation function, n_Ω is the average number of galaxies per unit solid angle and $\delta\Omega_1$ and $\delta\Omega_2$ is the solid angle separated by an angle θ_{12}.

Similarly, we can extract correlation functions for $N > 2$. The difference now is that they will depend on the lower-order correlation functions. For example, the three-point correlation function is defined in the equation:

$$\delta^3 P_3 = n_{gal}^3[1 + \xi(r_{12}) + \xi(r_{13}) + \xi(r_{23}) + \zeta(r_{12}, r_{13}, r_{23})]\delta V_1\delta V_2\delta V_3, \tag{2.4}$$

where $\zeta(r_{12}, r_{13}, r_{23})$ is the connected three-point correlation function, as it does not depend on the two-point correlation functions. The angular three-point correlation function can be defined in the same way as before, shown in the equation:

$$\delta^3 P_3 = n_{gal}^3\left[1 + w(\boldsymbol{\theta}_{12}) + w(\boldsymbol{\theta}_{23}) + w(\boldsymbol{\theta}_{31}) + z(\boldsymbol{\theta}_{12}, \boldsymbol{\theta}_{13}, \boldsymbol{\theta}_{23})\right]\delta\Omega_1\delta\Omega_2\delta\Omega_3, \tag{2.5}$$

where $z(\boldsymbol{\theta}_{12}, \boldsymbol{\theta}_{23}, \boldsymbol{\theta}_{31})$ is the connected three-point angular correlation function, which represents the excess probability of finding the galaxies in the solid angle elements $\delta\Omega_1\delta\Omega_2\delta\Omega_3$ separated by angles $\theta_{12}, \boldsymbol{\theta}_{13}, \boldsymbol{\theta}_{23}$. Similarly, this analysis can be expanded to any order correlation function.

2.3 Two-point correlation function and power spectrum

Previously, we defined the two-point correlation function (2PCF) from the physical standpoint of probabilities. It can also be defined as the joint average of the average density at two locations, as seen in equation 2.6:

$$\xi(r) = \langle \delta(\boldsymbol{x})\delta(\boldsymbol{x}+\boldsymbol{r})\rangle. \tag{2.6}$$

Due to the statistical homogeneity and isotropy, the 2PCF depends only on the $r = |\boldsymbol{r}|$. It can be seen that the 2PCF measures the correlation of density contrast δ at different positions in the universe. Writing the density contrast $\delta(\boldsymbol{x})$ in configuration space, we get the following:

$$\delta(\boldsymbol{x}) = \int d^3k\, \delta(\boldsymbol{k}) e^{i\boldsymbol{k}\cdot\boldsymbol{x}}, \tag{2.7}$$

where we can define $\delta(\boldsymbol{k})$, the density contrast in Fourier space, in the same manner, but with an additional prefactor:

$$\delta(\boldsymbol{k}) = \frac{1}{(2\pi)^3}\int d^3x\, \delta(\boldsymbol{x}) e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}. \tag{2.8}$$

Now we can see the correlator in the Fourier space, using the equation 2.8:

$$\langle \delta(\boldsymbol{k})\delta(\boldsymbol{k'})\rangle = \frac{1}{(2\pi)^6}\int d^3x d^3r \langle \delta(\boldsymbol{x})\delta(\boldsymbol{x}+\boldsymbol{r})\rangle e^{-i\cdot((\boldsymbol{k}+\boldsymbol{k'})\cdot\boldsymbol{x}+(\boldsymbol{k'}\cdot\boldsymbol{r}))}, \tag{2.9}$$

where the three-dimensional Dirac delta is defined as the following:

$$\delta_D^{(3)}(\boldsymbol{k}+\boldsymbol{k'}) = \frac{1}{(2\pi)^3}\int d^3x e^{-i(\boldsymbol{k}+\boldsymbol{k'})\cdot\boldsymbol{x}} \tag{2.10}$$

In the equation 2.9, we have the 2PCF definition in the integral, as well as the three-dimensional Dirac delta. By inserting these equations, we get:

$$\langle \delta(\boldsymbol{k})\delta(\boldsymbol{k'})\rangle = \frac{1}{(2\pi)^3}\delta_D^3(\boldsymbol{k}+\boldsymbol{k'})\int d^3r \xi(r) e^{i\boldsymbol{k}\cdot\boldsymbol{r}} \tag{2.11}$$

From equation 2.11 emerges the definition of power spectrum:

$$P(k) = \frac{1}{(2\pi)^3}\int d^3r \xi(r) e^{-i\boldsymbol{k}\cdot\boldsymbol{r}} \tag{2.12}$$

and similarly also the relation of 2PCF and power spectrum:

$$\xi(r) = \int d^3k P(k) e^{i\boldsymbol{k}\cdot\boldsymbol{r}} \tag{2.13}$$

This shows that the power spectrum and 2PCF are two sides of the same coin — the former is represented in the Fourier space, while the latter is in configuration or real space.

2.4 Three-point correlation function and bispectrum

The 2PCF or power spectrum is completely sufficient to capture the statistical properties of a Gaussian density field. Therefore, if we assume that our initial density perturbations are Gaussian, we do not need any higher-order statistics.

Wick's theorem explains just that, indicating that any higher-order correlation function can be decomposed into a sum of products of the two-point correlation functions. The fact that we have a Gaussian field, means that all the odd correlation functions will be zero, therefore:

$$\langle \delta(\boldsymbol{k}_1) \cdots \delta(\boldsymbol{k}_{2p+1}) \rangle = 0, \tag{2.14}$$

But if it is even, then it can be decomposed in the sum of 2PCF as the following:

$$\langle \delta(\boldsymbol{k}_1) \cdots \delta(\boldsymbol{k}_{2p}) \rangle = \sum_{\text{all pair associations}} \prod_{p \text{ pairs } (i,j)} \langle \delta(\boldsymbol{k}_i)\delta(\boldsymbol{k}_j) \rangle, \tag{2.15}$$

As good as it sounds, we are still in need of higher-order correlation functions. The main reason is that even if the primordial field would be Gaussian, the evolution of these perturbations would turn it into non-Gaussian from the non-linear effects and biasing. Furthermore, different inflationary models predict a non-Gaussian primordial field. For this reason, we shall develop the basis for the lowest-order correlation function, which can describe the non-Gaussianity — three-point correlation function (3PCF).

The definition of 3PCF is the following:

$$\zeta(\boldsymbol{r_1}, \boldsymbol{r_2}) = \langle \delta(\boldsymbol{x_1})\delta(\boldsymbol{x_2})\delta(\boldsymbol{x_3}) \rangle, \tag{2.16}$$

where $\boldsymbol{r_1} = \boldsymbol{x_1} - \boldsymbol{x_2}$ and $\boldsymbol{r_2} = \boldsymbol{x_2} - \boldsymbol{x_3}$, which are the relative coordinates of the chosen points. By using the equation 2.8 again, we obtain the definition of bispectrum:

$$\langle \delta(\boldsymbol{k_1})\delta(\boldsymbol{k_2})\delta(\boldsymbol{k_3}) \rangle = \delta_D(\boldsymbol{k_1} + \boldsymbol{k_2} + \boldsymbol{k_3})(2\pi)^3 B(\boldsymbol{k_1}, \boldsymbol{k_2}), \tag{2.17}$$

where the statistical homogeneity is represented by the triangle condition $\boldsymbol{k_1} + \boldsymbol{k_2} + \boldsymbol{k_3} = 0$.

2.5 Estimators

If we want to obtain the N-point correlation functions from observations or simulations, we must use estimators. One of the first estimators has been brought forward by Peebles

[52], where, for example, for 2PCF the estimator is of the form:

$$\hat{\xi} = \frac{DD}{RR} - 1, \tag{2.18}$$

where D is a point from the dataset, but R is randomly generated in the same volume as the observations. The "hat" ˆ for ξ indicates an estimator. A similar form can also be used for the three-point statistics: $\hat{\xi} = \frac{DDD-RRR}{RRR} + 2$.

There are several aspects to consider for estimators - the observations are made in a finite volume with a finite number of sampling cells, as well as with a certain geometry of the survey. Furthermore, edge corrections and shot noise or discreteness effect, which is due to the finite number of points used to describe a smooth random field, must be dealt with. There is also the instrumentation and observational biases, which include the limits of the telescope itself, as well as star contamination and dust extinction. Additionally, dynamical bias affects the quantities of interest, for example, for 3D surveys it is the peculiar velocity. Also, we usually conduct surveys on visible matter, which does not necessarily follow the same properties as matter [10]. It is also important what kind of objects are observed, because some of them can vary with redshift.

It is useful to find an estimator with minimal variance [10]. For the 2PCF, an improved estimator was proposed by Landy and Szalay [43] of the form:

$$\xi = \frac{DD - 2DR + RR}{RR}, \tag{2.19}$$

which will be further denoted as LS estimator. LS estimator has the variance proportional to $\frac{1}{N^2}$, where N is the number of points in the survey, thus reaching the second-order, while other estimators have the proportionality of $\frac{1}{N}$.

The equation 2.18 can be expressed in terms of overdensities as $\hat{\xi} = \langle(1+\delta_1)(1+\delta_2)\rangle_s = \langle\delta_1 + \delta_2 + \delta_1\delta_2\rangle$. The linear terms here indicate additional terms for the variance. On the other hand, the LS estimator can be written as:

$$\hat{\xi} = \frac{(D_1 - R_1)(D_2 - R_2)}{R_1 R_2} = \langle\delta_1\delta_2\rangle, \tag{2.20}$$

which indicates the lowest possible variance, assuming the previously mentioned additional terms are positive. The 3PCF with minimal variance is expressed as:

$$\hat{\zeta} = \frac{(D_1 - R_1)(D_2 - R_2)(D_3 - R_3)}{R_1 R_2 R_3} = \langle\delta_1\delta_2\delta_3\rangle, \tag{2.21}$$

The generalization of the equation 2.20 to higher order statistics has been proposed by [73] with the following equation:

$$\hat{f}_N = \frac{(D_1 - R_1)(D_2 - R_2)...(D_i - R_i)}{R_1 R_2 ... R_i}, \tag{2.22}$$

where i is the order and $\hat{f}_N$ is the N-point correlation function. This ensures variance as $\langle \delta_1 \delta_2 ... \delta_i \rangle$, which ensures no extra terms. It can be further expressed as the generalized LS:

$$\hat{f}_N = \frac{1}{S} \sum_i \binom{N}{i} (-1)^{N-i} \left(\frac{D}{\bar{n}_g}\right)^i \left(\frac{R}{\bar{n}_r}\right)^{N-i}, \tag{2.23}$$

where normalization number $S \equiv \int \Theta(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N)\, \mathrm{d}^D \boldsymbol{x}_1 \ldots \mathrm{d}^D$.

2.6 Multipole expansion

The three-point correlation function or bispectrum are complicated objects to analyze. Due to the difficulties in developing theories for them, the consensus in the last decade has been to pick specific triangle configurations to investigate. They do not represent the whole picture, therefore new methods must be developed.

In general, the 3PCF has 9 parameters that define it — three coordinates in the three-dimensional space. For the modeling, as discussed before, it can be reduced to 3 parameters, using the transverse invariance due to the homogenous property of the Universe and rotational invariance due to the isotropy. But in redshift space it is different, as the isotropy is no longer valid, therefore we have 6 parameters. It can be reduced to 5 by having rotational invariance around the line of sight. Nevertheless, the increase in the parameter space makes the modeling and computations more difficult.

The natural direction of investigation is the decomposition into multipoles. It gives several benefits over the analysis of separate triangle configurations. Firstly, it has physical reasoning, as it describes better the rotational symmetries, and at the same time the anisotropies induced by the RSD. Expanding allows one to separate different 3PCF components and analyze them separately, as well as summing them up to get the full 3PCF. Here a logical pushback would be — to get the full 3PCF, an infinite amount of multipoles would have to be re-summed. It turns out that, even though there are different formalisms with different decompositions, in all of them a finite number of multipoles contain the significant information of the 3PCF, usually no more than 10, as 3PCF does not have a complex angular structure. This means that by decomposing in the first few multipoles, it is possible to capture nearly the full scope of 3PCF.

In terms of observations, the greatest benefit over the usage of the regular estimators is the reduction of computational costs. By decomposing the 3PCF, it is possible to reform the counting of the triplets, where one of the examples is further explored in section 2.7, exploiting radial binning. There it reduces the computation costs from $\mathcal{O}(N^3)$ to $\mathcal{O}(N^2)$. All in all, decomposition has even more benefits, which will explored in the further sections.

Nonetheless, one of the first efforts was by [60, 72], who, with the parametrization of the triangle side lengths k_1, k_2 and angle between them θ, thus the model in real space, performed multipole expansion for the bispectrum as:

$$B(k_1, k_2, \theta) = \sum_l B_l(k_1, k_2) P_l(cos\theta) \frac{2l+1}{4\pi}, \tag{2.24}$$

where $P_l(cos\theta)$ is the Legendre polynomial of lth order and B_l can be obtained through integration as $B_l = 2\pi \int BP_l d(cos\theta)$. This is a useful expression, as it gives an inherent representation of rotational symmetry and the possibility to represent all triangle configurations. It also gives room to build an estimator that can constrain the bias parameter by exploiting ratios of multipoles.

Further work has been done with this decomposition to include the modeling of the RSD, as well as to model the 3PCF [66, 65]. The following is the first model for the configuration space from the perturbation model. The formalism will depend on the already established model of the monopole of bispectrum in RSD [60]:

$$\begin{aligned} B_s(k_1, k_2, x) =& b_1^3 P(k_1) P(k_2) \Big[\tilde{F}_2(k_1, k_2; x) D_{\mathrm{SQ1}}(\beta, x) \\ &+ \tilde{G}_2(k_1, k_2; x) D_{\mathrm{SQ2}}(\beta, k_1, k_2; x) \\ &+ D_{\mathrm{NLB}}(\beta, \gamma; x) + D_{\mathrm{FOG}}(\beta, k_1, k_2; x) \\ &+ \text{cyc.} \Big], \end{aligned} \tag{2.25}$$

where k_1, k_2, x are the wavevectors of two triangle sides and $x = \hat{k}_1 \cdot \hat{k}_2$, b_1 is the linear bias and $P(k)$ is the linear power spectrum, $\tilde{F}_2$ and $\tilde{G}$ is the second order density field and velocity field kernels, D is the growth rate, $\beta = \frac{f}{b_1}$ and $\gamma = \frac{2b_2}{b_1}$, where f is the growth rate and b_2 is non-linear bias. The subscripts: *SQ1* and *SQ2* are linear and non-linear squashing, while *NLB* and *FOG* are the non-linear bias and fingers of God terms.

We can now exploit the equation 2.24, but before that, we must separate the bispectrum in terms that involve $\frac{1}{k_3^2}$ where $k^3 = |\boldsymbol{k_1} + \boldsymbol{k_2}|$ and the terms absent from it. Then, from the equation 2.25, which is the fully averaged bispectrum, we can separately estimate the expansion for each of the terms in the square brackets, which are expressed in detail in [66]. After summing all the separate terms, the bispectrum needs to be transformed to configuration space. It can be done with the inverse Fourier transform as:

$$\zeta_\ell(r_1, r_2) = (-1)^\ell \int \frac{k_1^2 k_2^2 dk_1 dk_2}{(2\pi^2)^2} B_{s,\ell}(k_1, k_2) j_\ell(k_1 r_1) j_\ell(k_2 r_2), \tag{2.26}$$

where the subscript "s" denotes redshift space and j_ℓ are Bessel functions.

2.7 Spherical Harmonics Decomposition

It is not an easy task to tackle the full 5D redshift space anisotropic 3PCF. To capture the anisotropies, a new method was developed, which now uses bipolar spherical harmonics [67]. To start with, it was chosen to expand in both triangle sides as:

$$\hat{\zeta}(r_1, r_2; \mathbf{x}) = \frac{4\pi}{\sqrt{(2l+1)(2l'+1)}} \times \sum_{lm} \sum_{l'm'} \hat{\zeta}_{ll'}^{mm'}(r_1, r_2; \mathbf{x}) Y_{lm}(\hat{r}_1) Y^*_{l'm'}(\hat{r}_2). \tag{2.27}$$

where $\mathbf{x}$ is the vertex of the triangle where the opening angle is defined. It can also be defined as the line of sight and z axis: $\hat{x} = \hat{n} = z$. Here it is clear, that we are dealing with an anisotropic case, as $l \neq l'$ as in [66]. Due to the properties of the RSD, it is rotationally symmetric around the line of sight, therefore the 3PCF can be azimuthally averaged:

$$\hat{\zeta}_{\mathrm{azi}}(r_1, r_2; \mathbf{x}) = \int_0^{2\pi} \mathrm{d}\phi \, \hat{\zeta}(r_1, r_2; \mathbf{x}), \tag{2.28}$$

where the subscript "azi" denotes the azimuthal averaging and ϕ is the azimuthal angle. With further manipulation, as shown in [67], we acquire that the average 3PCF scales with the Kronecker delta $\delta^K_{mm'}$. This shows that only the condition for the spin as $m = m'$ is viable in equation 2.27 for the averaged 3PCF.

This is a powerful tool that allows us to already analyze the anisotropic 3PCF. To use it on real data, a faster algorithm has been developed than $\mathcal{O}(N^3)$, which scales as $N(nV_{R_{max}})$ instead [65]. Here N is the number of objects, n is their number density, and $V_{R_{max}}$ is the volume of a chosen R_{max} that the statistics are calculated at for each object.

The algorithm is described in Fig. 2.1. Firstly, the points are gathered around the primary point and radially binned. The radial binning for the density field is defined as:

$$\bar{\delta}(r_i; \hat{r}_i; \mathbf{x}) = \int r^2 \, \mathrm{d}r \, \Phi(r; r_i) \delta(\mathbf{x} + \mathbf{r}), \tag{2.29}$$

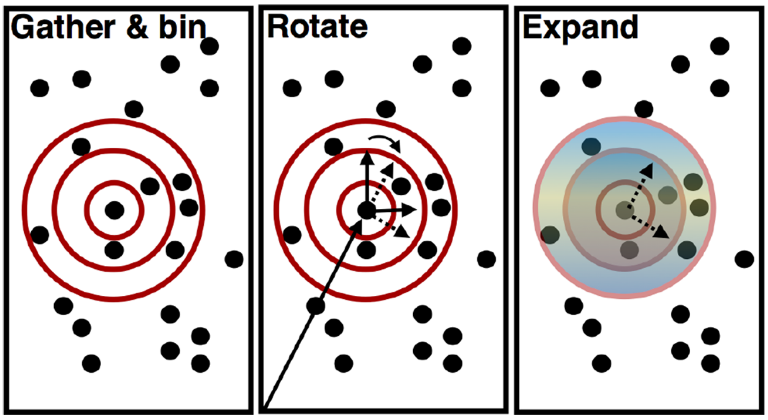


Figure 2.1. Algorithmic visualization, taken from [67], for the estimator of expanded 3PCF. In the first part "Gather & bin" an object is selected, galaxies are gathered in the $V_{R_{max}}$ range, and then they are radially binned. Then in the "rotate" part, the coordinate system is rotated, in order for the primary object (the one in the center) to have an aligned line of sight and z. In the last step "expand", in each of the radial bins the angular dependence of the density field is expanded. The color denotes the different values of a_{lm}.

where $\Phi(r; r_i)$ is the binning function. Here the primary galaxy is at point $\mathbf{x}$, but the secondary galaxies are denoted as r are inside the bins r_i. Incorporating the equation 2.29 in the multipole expansion of the 3PCF, we get:

$$\hat{\zeta}^m_{ll'}(r_1, r_2; \mathbf{x}) = \delta(\mathbf{x}) \int \mathrm{d}\Omega_1 \mathrm{d}\Omega_2\, \bar{\delta}(r_1; \hat{r}_1; \mathbf{x}) \bar{\delta}(r_2; \hat{r}_2; \mathbf{x}) Y_{lm}(\hat{r}_1) Y^*_{l'm}(\hat{r}_2). \tag{2.30}$$

From the definition of the spherical harmonic coefficients a_{lm}, it is possible to obtain:

$$\hat{\zeta}^m_{ll'}(r_1, r_2; \mathbf{x}) = \delta(\mathbf{x}) \left[a_{lm}(r_1; \mathbf{x}) a^*_{l'm}(r_2; \mathbf{x}) + a^*_{lm}(r_1; \mathbf{x}) a_{l'm}(r_2; \mathbf{x})\right], \tag{2.31}$$

To calculate the a_{lm} coefficient for a primary galaxy at each radial bin, it scales as $nV_{R_{max}}$. And as it must be done for each galaxy, the final scaling of the algorithm is as stated before: $N(nV_{R_{max}})$.

2.8 Tripolar spherical decomposition

The problem with the previously mentioned methods is that all of them are spherically averaging over the line of sight. This means that all the methods are only measuring the monopole. Furthermore, as seen in equation 2.31, the spin parameter $m \neq 0$, therefore they can always vary under rotation. To deal with this, we will describe a new method, which is also the basis of this thesis, based on the work of Sugiyama et al. [70, 71].

These issues can be resolved by a new method called TripoSH, where the bispectrum is decomposed into three spherical harmonics. It is defined as follows:

$$B(\mathbf{k_1},\mathbf{k_2},\hat{n}) = \sum_{JM_J}\sum_{\ell_1\ell_2L\ell_{12}} B^{JM_J}_{\ell_1\ell_2L\ell_{12}}(k_1,k_2)S^{JM_J}_{\ell_1\ell_2L}(\hat{k_1},\hat{k_2},\hat{n}), \tag{2.32}$$

where the TripoSH basis $S^{JM_J}_{\ell_1\ell_2L}(\hat{k_1},\hat{k_2},\hat{n})$ is defined as:

$$S^{JM_J}_{\ell_1\ell_2L}(\hat{k_1},\hat{k_2},\hat{n}) = \sum_{m_1m_2m_{12}M} C^{\ell_{12}m_{12}}_{\ell_1m_1;\ell_2m_2}C^{JM_J}_{\ell_{12}m_{12};LM} \times y^{m_1}_{\ell_1}(\hat{k_1})y^{m_2}_{\ell_2}(\hat{k_2})y^{M}_{L}(\hat{n}), \tag{2.33}$$

and the TripoSH coefficients are defined as:

$$B^{JM_J}_{\ell_1\ell_2L}(k_1,k_2) = \sum_{m_1m_2m_{12}M} C^{\ell_{12}m_{12}}_{\ell_1m_1;\ell_2m_2}C^{JM_J}_{\ell_{12}m_{12};LM} \times B^{m_1m_2M}_{\ell_1\ell_2L}(k_1,k_2), \tag{2.34}$$

and the Clebsch–Gordan coefficients are formulated as:

$$C^{\ell_3m_3}_{\ell_1m_1;\ell_2m_2} = (-1)^{\ell_1-\ell_2+m_3}\sqrt{2\ell_3+1}\begin{pmatrix}\ell_1 & \ell_2 & \ell_3\\ m_1 & m_2 & -m_3\end{pmatrix}. \tag{2.35}$$

From the isotropic property of the Universe, we can make further simplifications, that allow us to constrain the total angular momentum to zero, thus $J=0$ and $M_J=0$. Including the parity symmetry of the Universe, we have the final expression for the multipoles of the bispectrum:

$$B_{\ell_1\ell_2L}(k_1,k_2) = H_{\ell_1,\ell_2,L}\sum_{m_1,m_2,M}\begin{pmatrix}\ell_1 & \ell_2 & L\\ m_1 & m_2 & M\end{pmatrix}B^{m_1m_2M}_{\ell_1\ell_2L}(k_1,k_2), \tag{2.36}$$

where $H_{\ell_1,\ell_2,L} = \begin{pmatrix}\ell_1 & \ell_2 & L\\ 0 & 0 & 0\end{pmatrix}$ selects components for which $\ell_1+\ell_2+L$ is even. This simplifies equation 2.32 by substituting it with equation 2.36. The same can be done for the 3PCF — if the isotropy and parity symmetry is assumed, the multipoles of 3PCF are defined as:

$$\zeta_{\ell_1\ell_2L}(r_1,r_2) = H_{\ell_1,\ell_2,L}\sum_{m_1,m_2,M}\begin{pmatrix}\ell_1 & \ell_2 & L\\ m_1 & m_2 & M\end{pmatrix}\zeta^{m_1m_2M}_{\ell_1\ell_2L}(r_1,r_2). \tag{2.37}$$

The bispectrum and 3PCF can be related to each other by Hankel transformation.

From 3PCF to bispectrum it is expressed as:

$$B_{\ell_1\ell_2L}(k_1,k_2) = (-i)^{\ell_1+\ell_2}(4\pi)^2 \int \mathrm{d}r_1\, r_1^2 \int \mathrm{d}r_2\, r_2^2 \times j_{\ell_1}(k_1r_1)j_{\ell_2}(k_2r_2)\zeta_{\ell_1\ell_2L}(r_1,r_2), \tag{2.38}$$

and from bispectrum to 3PCF as:

$$\zeta_{\ell_1\ell_2L}(r_1,r_2) = i^{\ell_1+\ell_2} \int \frac{\mathrm{d}k_1k_1^2}{2\pi^2} \int \frac{\mathrm{d}k_2k_2^2}{2\pi^2} \times j_{\ell_1}(r_1k_1)j_{\ell_2}(r_2k_2)B_{\ell_1\ell_2L}(k_1,k_2), \tag{2.39}$$

where the j_ℓ are the Bessel functions.

It is not necessary to calculate every possible combination of multipoles, as there are four rules that constrain this choice [70]. Firstly, the property of the bispectrum $B_{\ell_1,\ell_2,L}(k_1,k_2) = B_{\ell_2,\ell_1,L}(k_1,k_2)$ allows us to choose $\ell_1 > \ell_2$. Secondly, the axial symmetry around the LOS allows us only to pick $L =$ even. This can be done because of the assumed plane-parallel approximation, which gives axially symmetric anisotropies in RSD. Thirdly, because rotationally symmetric assumption, we get $|\ell_1-\ell_2| \leqslant L \leqslant |\ell_1+\ell_2|$. Lastly, due to the parity symmetry mentioned before, we get that $\ell_1 + \ell_2 =$ even.

These four constraints give the important property of the bispectrum multipoles: $B_{\ell_1\ell_2L} = B^*_{\ell_1\ell_2L}$. They also inform us what choices of multipoles we have. In this paper we will look at 8 multipoles, the first four being the isotropic case, where $L = 0$, and the other four being the anisotropic case $L = 2$, which are purely induced by the RSD and AP effects. This is due to the fact that we have assumed an isotropic and homogeneous universe. As a reminder, the expansions depicted by ℓ_1 and ℓ_2 are for r_1 and r_2 accordingly. Here the interesting multipoles are $L = 2, \ell_1 = 2, \ell_2 = 0$ and $L = 2, \ell_1 = 3, \ell_2 = 1$, as they are asymmetric in their nature, as $\ell_1 \neq \ell_2$. The combinations have been summarized in Table 2.1:

L	(ℓ_1, ℓ_2)			
$L = 0$	$(0,0)$	$(1,1)$	$(2,2)$	$(3,3)$
$L = 2$	$(2,0)$	$(1,1)$	$(3,1)$	$(2,2)$

Table 2.1. From the physical constraints of the Universe and the choice from the work authors to expand up to $L = 2$ the first 4 combinations, these are the possible multipole configurations.

2.8.1 The choice of coordinate system

It is also important what kind of coordinate system is chosen, as seen already in the previous section 2.6. The physical properties are unchanged from the change of coordinate system, but, nevertheless, it does change the equations. It is also useful to carefully pick the coordinate system, in order to relate the results of previously developed theories with certain coordinate choices. We have the choice to adopt the same formalism as [60], where $\hat{k_1} = z$ or the recent one of [67], where $\hat{n} = z$. These systems are visualized in Fig. 2.2.

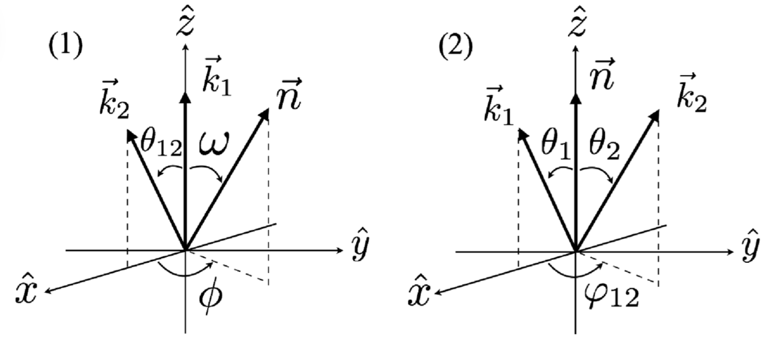


Figure 2.2. Choice of coordinate system visualization by [70]. The left part stands on the choice of $\hat{k_1} = \hat{z}$. The right coordinate system is based on $\hat{n} = \hat{z}$.

In this work, the coordinate system choice $\hat{k_1} = \hat{z}$ is adopted. The following coordinate system transformation must be done:

$$\begin{aligned}
\mathbf{k}_1 &= \{0, 0, k_1\}, \\
\mathbf{k}_2 &= \{k_2 \sin\theta_{12}, 0, k_2 \cos\theta_{12}\}, \\
\hat{\mathbf{n}} &= \{\sin\omega\cos\phi, \sin\omega\sin\phi, \cos\omega\}.
\end{aligned} \tag{2.40}$$

This means that also the bispectrum has changed the parameters that it depends on, which is expressed in the equation 2.40. Not only that, also the spherical harmonics are different, therefore the equation 2.36 changes to:

$$\begin{aligned}
B_{\ell_1\ell_2 L}(k_1, k_2) = N_{\ell_1\ell_2 L} H_{\ell_1\ell_2 L} &\int \frac{\mathrm{d}\cos\omega\,\mathrm{d}\phi}{4\pi} \int \frac{\mathrm{d}\cos\theta_{12}}{2} \\
&\times \left[\sum_M \begin{pmatrix} \ell_1 & \ell_2 & L \\ 0 & -M & M \end{pmatrix} y_{\ell_2}^{-M*}(\theta_{12}, 0) y_L^{M*}(\omega, \phi) \right] \\
&\times B(k_1, k_2, \theta_{12}, \omega, \phi),
\end{aligned} \tag{2.41}$$

where $N_{\ell_1\ell_2 L}$ is the normalization factor.

2.8.2 Template model

Having covered the decomposition formalism, it is now possible to construct the template model of the bispectrum, from which we can obtain the template 3PCF. This allows us to construct a theoretical prediction for the galaxy or matter 3PCF in redshift space, given a certain cosmology. Furthermore, there has always been the problem of using 3PCF models in Markov-Chain Monte Carlo (MCMC) algorithms to constrain cosmological parameters, as the computational costs are too expensive. The computational costs of the following model and its decomposition are still significant, but they do allow computing a large data set of different cosmological models to build an accurate emulator, which will be further discussed in chapter 3.

This model is based on the Lagrangian theory [10, 71]. Based on that, we will now include the effects of RSD for the displacement vector [71] as:

$$\Psi(\mathbf{q}) = \Psi_{\text{real}}(\mathbf{q}) + \frac{\dot{\Psi}_{\text{real}}(\mathbf{q}) \cdot \hat{\mathbf{n}}}{H}\hat{\mathbf{n}}, \tag{2.42}$$

where the subscript "real" indicates real space and $\dot{\Psi}$ is the derivative in time of the displacement vector. Regarding the galaxy density fluctuations in Lagrangian formalism, they can be expressed as:

$$\delta(\mathbf{x}) = \int d^3q(1 + \delta_{bias}(\mathbf{q}))\delta_D(\mathbf{q} - \mathbf{x} - \boldsymbol{\Psi}) - 1, \tag{2.43}$$

where $\delta_{bias}(\mathbf{q})$ represents the density fluctuations of the biased objects at their initial positions, or more simply, the initial distribution of these objects, which in our case are the galaxies. Here the RSD effects come through the $\boldsymbol{\Psi}$ parameter. And the parameter δ_{bias} can be expanded to second order as:

$$\delta_{bias}(\mathbf{q}) = b_1^L\delta_{lin}(\mathbf{q}) + \frac{1}{2}b_2^L\left[\delta_{lin}(\mathbf{q})\right]^2 + b_K^L\left(\frac{\delta_{q_i}\delta_{q_j}}{\delta_q} - \frac{1}{3}\delta_{ij}\right)^2\delta_{lin}^2(\mathbf{q}), \tag{2.44}$$

where i and j represent the spatial coordinates, and δ_{ij} is the Kronecker delta. They are also analytically related to the Euler bias parameters. These bias parameters will be regarded as free parameters in the following template model.

First, we need to see if the infra-red (IR) flows affect the N-point statistics. These corrections are due to the large-scale initial density perturbations, that can affect the further evolution of the overall structures. We can define the IR flow by separating it into two components — the ones that are defined at the origin and the ones that are not:

$$\mathbf{\Psi}(\mathbf{q}) = \mathbf{\Psi}(\mathbf{q} \neq 0) + \bar{\mathbf{\Psi}}, \tag{2.45}$$

where $\bar{\mathbf{\Psi}} = \mathbf{\Psi}(\mathbf{q} = 0)$ is the IR flow.

Now we can apply the Γ-expansion, which allows extracting information from the N-point statistics, using BAO. It is a method based on the statistical properties of the observables, therefore in the scope of standard perturbation theory (SPT) — density and velocity field. The first-order term of expansion of the density field is the Gaussian distribution, but the higher order can explain non-Gaussianity. The most important property of this expansion is that it allows us to separate the integrals that involve mode coupling from those that do not. This distinction provides valuable insights into the distribution of BAO information; specifically, modes that are not coupled tend to carry more information than those that are coupled.

In the SPT formalism, we can expand the density field as:

$$\tilde{\delta}(\mathbf{k}) = \sum_{n=1}^{\infty} \prod_{i=1}^{n} \int \frac{d^3 p_i}{(2\pi)^3} \tilde{\delta}_{\mathrm{lin}}(\mathbf{p}_i)(2\pi)^3 \delta_D(\mathbf{k} - \mathbf{p}_{1n}) Z^{[n]}(\mathbf{p}_1, \ldots, \mathbf{p}_n), \tag{2.46}$$

where $\mathbf{p}_{1n} = \mathbf{p}_1 + .. + \mathbf{p}_n$, $Z^{[1](\mathbf{k})}$ is the Kaiser factor, but $Z^{[n\geqslant 2](\mathbf{k})}$ are the non-linear kernel functions, including the RSD term.

We can use the same Γ-expansion for the bispectrum. It can be split into coupled and non-coupled terms:

$$\begin{aligned} B(k_1, k_2, k_3) = & \left(B_{GG}(k_1, k_2) P_{\mathrm{lin}}(k_1) P_{\mathrm{lin}}(k_2) + 2 \text{ cyc.}\right) \\ & + \left(B_{GM}(k_1, k_2) P_{\mathrm{lin}}(k_1) + 2 \text{ cyc.}\right) \\ & + \left(B_{MG}(k_1, k_2) P_{\mathrm{lin}}(k_2) + 2 \text{ cyc.}\right) \\ & + \left(B_{MM}(k_1, k_2) + 2 \text{ cyc.}\right) \\ & + \left(B_{MMM}(k_1, k_2) + 2 \text{ cyc.}\right), \end{aligned} \tag{2.47}$$

where each bispectrum term has been visualized and explained by Fig. 2.3. The components of each bispectrum term are further shown in [71].

As mentioned before, with infra-red (IR) flows, large wavelength perturbations can have effects on the overall structure of the universe. But it can be shown that the IR effect can be canceled out of the final expression of the bispectrum, if all the components of the equation 2.47 are summed together, when treated appropriately.

If we include the IR flow in the density fluctuations and proceed with calculations as in the 3.4 section of [71], we will eventually come across the damping factor $\mathcal{D}(\mathbf{k})$,

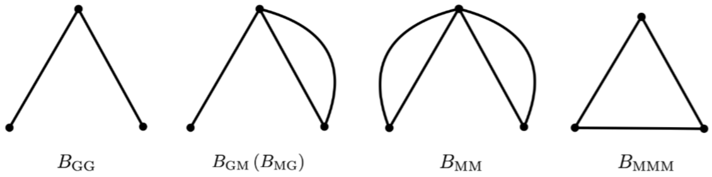


Figure 2.3. Visualization of the bispectrum Γ-expansion terms by [71]. The straight lines represent integrals with the linear power spectrum, which does not have mode coupling. The curved lines, on the other hand, show the coupled terms, which have integrals with an infinite number of integration dimensions.

which is due to IR flows. It is expressed as follows:

$$\mathcal{D}(\mathbf{k}) = exp\left(-\frac{k^2(1+2f\mu^2+f^2\mu^2)}{2}\sigma^2_{dd}\right), \tag{2.48}$$

where the $\sigma^2_{dd} = \frac{1}{3}\int \frac{dp}{2\pi^2}P_{lin}(p)$ is the dispersion of the linear displacement vector. Each term in the equation 2.47 can be expressed as the product of damping factors and the tree-level solution of the bispectrum. For example, the non-coupled term can be expressed as:

$$B_{GG}(k_1,k_2)P_{\mathrm{lin}}(k_1)P_{\mathrm{lin}}(k_2) = \mathcal{D}(k_1)\mathcal{D}(k_2)\mathcal{D}(k_{12})B_{\mathrm{tree}}(k_1,k_2), \tag{2.49}$$

where $B_{tree}(\mathbf{k}_1,\mathbf{k}_2) = 2Z^{[1]}(\mathbf{k}_1)Z^{[1]}(\mathbf{k}_2)Z^{[2]}(\mathbf{k}_1,\mathbf{k}_2)P_{lin}(k_1)P_{lin}(k_2)$ is the leading order solution for the bispectrum. When each of these newly expressed terms is summed up together, the IR flow contribution cancels out, and only the tree-level solution is left. Therefore, the non-linear contributions have been dealt with.

Based on the IR cancellation formalism, we can now expand these results to build a template model for the power spectrum, which then can be applied to the bispectrum. As discussed before, the mode-coupled terms have negligible information on BAO. We can use this together with the empirical results of [25], denoting the mode-coupled power spectrum P_{MC} as a power spectrum with no BAO features, which is defined as P_{nw}, where "nw" means no wiggle (as BAO features in the power spectrum looks like wiggles). Therefore, the template power spectrum is the following:

$$P^{(\mathrm{temp})}(k) = G^2(k)P_{\mathrm{lin}}(k) + P_{\mathrm{MC}}(k) \rightarrow \left[Z^{[1]}(k)\right]^2\left[D^2(k)P_w(k) + P_{\mathrm{nw}}(k)\right], \tag{2.50}$$

where the P_w is the only wiggle power spectrum, which can be simply obtained as the subtraction from the full and no wiggle power spectrum.

Similarly, it can be done for the bispectrum. In this case, whenever we have the mode-coupled term, we exchange it with $P_{lin}(k) \to P_{nw}(k)$, but for the non-coupled, we opt for $P_{lin}(k) \to P_w(k)$. Therefore, the final galaxy bispectrum template in RSD looks like this:

$$\begin{aligned} B^{(\text{temp})}(k_1, k_2) = 2\, Z^{[1]}(k_1) Z^{[1]}(k_2) Z^{[2]}(k_1, k_2) \Big\{ & \mathcal{D}(k_1)\mathcal{D}(k_2)\mathcal{D}(k_{12}) P_w(k_1) P_w(k_2) \\ & + \mathcal{D}^2(k_1) P_w(k_1) P_{\text{nw}}(k_2) + \mathcal{D}^2(k_2) P_{\text{nw}}(k_1) P_w(k_2) \\ & + P_{\text{nw}}(k_1) P_{\text{nw}}(k_2) \Big\} + 2\,\text{cyc.} \end{aligned} \tag{2.51}$$

This is the model that goes into the equation 2.32, and it is also implemented in the Python package called *Mod3l* [28]. From the equation 2.51 we can also obtain the 3PCF, by leveraging the 2D Hankel transform, using the 2D FFT.

2.8.3 Calculation of the three-point correlation function multipoles

To calculate the multipoles of the 3PCF, which is described by equation 2.39, we must calculate two complicated integrals. They include Bessel functions, which are highly oscillating and are complicating the calculation further. Therefore, the calculations require a very well-sampled grid. The standard quadrature method scales as $\mathcal{O}(N_k^2 N_r^2)$, where N_k are the number of sampling points in the Fourier space, which is $N_k \gg N$, while N_r are the desired sampling in the real space [27]. The alternative method is to use a two-dimensional method called Fast Fourier transform for logarithmically spaced points (2D FFT-Log), which is considerably faster. It does suffer from ringing and aliasing problems, which are the reactions to steep changes in the function and periodic folding of frequencies [34]. This can be dealt with by carefully choosing the appropriate k_{min} and k_{max}, and by applying padding.

To disentangle this method, we first describe the Fast Fourier transform. The Fourier Transform is a mathematical technique that transforms a function or dataset from its original domain, usually configuration space, into the frequency domain or the Fourier space or vice versa. In our case, it will be done from the Fourier space to the configuration space. It can reduce the computation costs from $\mathcal{O}(N_{\text{particle}}^2)$ to $\mathcal{O}(N_{\text{grid}}^2 log N_{\text{grid}})$, where N_{grid}^2 is the total sampling for the grid for the bispectrum multipoles [28]. It does so by using the Cooley-Tukey FFT Algorithm [18], which exploits the symmetries of the discrete Fourier transform (the direct way of computation). In a nutshell, if we have a

1D integral like $\int_a^b f(k)j(kr)$, then using FFT we can express $f(k) = \sum_m c_m e^{z_m}$ as a sum, where c_m are the Fourier coefficients and z_m is generally a complex number. This allows to have faster computation, while still maintaining good accuracy.

As in the equation 2.39 we have two Bessel functions, the 1D FFT approach will not suffice, and a 2D FFT must be used. To simply the further calculations, we will express the dimensionless bispectrum multipole as follows [31]:

$$\Delta_{\ell_1\ell_2 L}(k_1, k_2) = \frac{k_1^3 k_2^3}{(2\pi^2)^2} B_{\ell_1\ell_2 L}(k_1, k_2). \tag{2.52}$$

This can now be decomposed as the following expression:

$$\Delta_{\ell_1\ell_2 L}(\mathbf{k}_p, \mathbf{k}_q) = \frac{1}{N^2} \sum_{m=-\frac{N}{2}}^{\frac{N}{2}} \sum_{n=-\frac{N}{2}}^{\frac{N}{2}} \tilde{c}_{\ell_1\ell_2 L,mn} k_0^{-in} k_0^{-im} k_p^{\nu_1+i\eta_m} k_q^{\nu_2+i\eta_n}, \tag{2.53}$$

where N is the number of elements in the k array, $\tilde{c}$ is the Fourier coefficient, ν are the bias parameters, but $\eta_i = \frac{2\pi i}{N\Delta_{lnk}}$, where Δ_{lnk} is the logarithmic spacing for the k array. Using the equation 2.53 and inserting it into equation, 2.39, and also including that real data is bin averaged, we obtain:

$$\begin{aligned}\zeta_{\ell_1,\ell_2,L}(\bar{r}_i, \bar{r}_j) = \frac{\pi \bar{r}_{i,\min}^{2-\nu_1} \bar{r}_{j,\min}^{2-\nu_2}}{16AN^2} &\sum_{m,n=-N/2}^{N/2} c_{\ell_1\ell_2 L,mn} k_0^{-i(\eta_m+\eta_n)} r_{12}^{-i\eta_m} r_{13}^{-i\eta_m} \\ &\times g_\ell(\nu_1 + i\eta_m) g_\ell(\nu_2 + i\eta_n) s(D - \nu_1 - i\eta_m, n) s(D - \nu_2 - i\eta_m, n),\end{aligned} \tag{2.54}$$

where $s(D, n) = \frac{n-1}{D}$ and $A = \bar{r}_{i,min}^2 \bar{r}_{j,min}^2 [s(2, n)]^2$.This is the averaged expression for the 2D FFT-Log, which allows going from bispectrum multipoles to 3PCF multipoles. As seen, one of the parameters is the grid spacing, which is a very important parameter for an accurate estimate.

To use the 2D FFT-Log algorithm, we needed to define the grid over which the computations will be conducted. This will dictate the accuracy of our computation. The grid is separated into the configuration variables $(k_1, k_2, \cos\theta)$ and orientation variables (μ, ϕ). The orientation variables are dealt by the algorithm itself, but k_1 and k_2 are logarithmically in the interval $[k_{min}, k_{max}] \in \mathbb{R}^+$, while $\cos\theta$ is defined in the interval $[-1, 1]$, which will be used as nodes for the Gauss-Legendre formula, which in turn is necessary for the calculation in the equation 2.41.

The k_1, k_2 must be logarithmically spaced, because we have to cover a wide range of values, going from 10^{-4} to 10^1. This means that linearly it would be 10^5 points, which would lead to expensive computational cost. Therefore, logarithmic spacing greatly reduces these costs. Finally, the grid is defined as the Cartesian product between k_1, k_2, and $\cos\theta$. Instead, the spherical surface, defined by the orientation variables, is partitioned into curvilinear quadrilaterals.

To decompose the bispectrum into its harmonic components, the calculation is performed at each grid point across every partition of the spherical surface. The algorithm requires selecting specific multipole configurations, which ultimately enables the extraction of the individual multipole contributions to the bispectrum.

2.8.4 Decomposition of the model

Even though the FFT algorithm is significantly faster than the alternative methods, such as DFT, but applying it to the equation 2.51 is still a lot of computational cost. For example, to calculate a single multipole for the 3PCF, we have to go through various steps. First, we have to compute N_x integrals for a single triangle configuration bispectrum multipole $B_{\ell_1\ell_2 L}$ due to the angular part between the triangle sides. Then we have to use the 2D-FFT algorithm, to obtain the multipole for 3PCF, scaling as $\mathcal{O}(N_{\rm grid}^2 log N_{\rm grid})$. Furthermore, we need to repeat it for every triangle configuration, scaling as N_r^2. Finally, all of this has to be repeated for every multipole. In the end, depending on the chosen grid, it could be attributed to more than a trillion integrals.

To isolate the fitting parameters, that the 3PCF depends on, and speed up the calculations for the model, the fitting parameters are linearized. To do so, the following steps have been implemented:

- Fixing the shape of the linear power spectrum P_{lin} and no-wiggle P_{nw}.
- Expressing the damping factor as $\mathcal{D}(k) = \exp\left(-\frac{k^2(1-\mu^2)\Sigma_\perp^2 + k^2\mu^2\Sigma_\parallel^2}{4}\right)$, where the $\Sigma_\parallel$ and $\Sigma_\perp$ are the smoothing parameters, that can be also fixed.

These two steps allow the template redshift space 3PCF to depend only on 5 parameters if the AP effect is excluded:

$$(b_1\sigma_8, f\sigma_8, \sigma_8, b_2\sigma_8^2, b_K 2\sigma_8^2). \tag{2.55}$$

This enables us to express the template bispectrum model as a linear combination of the mentioned parameters as coefficients:

$$Z_1(k_1)Z_1(k_2)Z_2(k_1,k_2)\sigma_8^4 = \sum_{p=1}^{14} X^{(p)} B^{(p)}(k_1,k_2), \tag{2.56}$$

where the $B^{(p)}$ can be pre-calculated, but $X^{(p)}$ not. The detailed derivations of each component can be seen in [71]. Further on, we can replace $P_{lin} \to \frac{\sigma_8^2 P_{lin,fid}}{\sigma_{8,fid}^2}$ and $P_{nw} \to \frac{\sigma_8^2 P_{lin,fid}}{\sigma_{8,fid}^2}$, where the fiducial values can be also pre-computed. Now by putting the equation 2.56 in the 2.51, the only thing that cannot be pre-computed is the $X^{(p)}$, if AP effects are not taken into account.

To introduce the AP effects, the previous decomposition of pre-calculated parameters would not work anymore, as there is an additional angular component to the LOS due to the anisotropic nature of the AP effect. This means that the triple integral for the 3PCF must be recalculated every time to see the AP effect.

To deal with this, a new parametrization is introduced, differentiating from the subsection 1.3.4: $\alpha^3 = \alpha_\perp^2 \alpha_\parallel$ called isotropic dilation and $(1+\varepsilon)^3 = \frac{\alpha_\parallel}{\alpha_\perp}$ called anisotropic warping. Then, by applying those to define the true wavenumbers and then expressing the bispectrum multipoles, we get:

$$B^{(\mathrm{AP})}_{\ell_1\ell_2\ell}(k_1, k_2) = \frac{4\pi h^2_{\ell_1\ell_2\ell}}{\alpha^6} \int \frac{d^2\hat{k}_1}{4\pi} \int \frac{d^2\hat{k}_2}{4\pi} \int \frac{d^2\hat{n}}{4\pi}\, S^*_{\ell_1\ell_2\ell}(\hat{k}_1, \hat{k}_2, \hat{n}) B(k_{\mathrm{true},1}, k_{\mathrm{true},2}), \tag{2.57}$$

where $\mathbf{k}_{true}$ is the true wavevector. Then, by expanding the true wavevector as:

$$k_{true} = k'(1 + \Delta k), \tag{2.58}$$

where k' and Δk is:

$$\begin{aligned} k' &= k\left(\frac{1+\varepsilon}{\alpha}\right)\left[1 + \frac{1}{3}\left((1+\varepsilon)^{-6} - 1\right)\right]^{\frac{1}{2}}, \\ \Delta k &= \left\{1 + \frac{\frac{2}{3}\mathcal{L}_2(\hat{k}\cdot\hat{n})\left((1+\varepsilon)^{-6} - 1\right)}{\left[1 + \frac{1}{3}\left((1+\varepsilon)^{-6} - 1\right)\right]}\right\}^{\frac{1}{2}} - 1. \end{aligned} \tag{2.59}$$

2.9 Relation with lower order expansions

Even though decomposing the 3PCF in triple spherical harmonics represents a powerful tool on its own, it is useful to relate it with previously developed approaches for consistency. For example, the first idea was to expand in only one spherical harmonics [60, 72], which is expressed by equation 2.24. In more general form, including also ω and ϕ due to RSD, which describe the orientation of the triangle, bispectrum decomposition can be written as:

$$B(k_1, k_2, \hat{n}, \omega, \phi) = \sum_{LM} B_{ML}(k_1, k_2, \hat{n}) Y_L^M(\omega, \phi), \tag{2.60}$$

.

This equation can be related to TripoSH decomposition by:

$$B_{\ell_1 \ell_2 L}(k_1, k_2) = \frac{N_{\ell_1 \ell_2 L} H_{\ell_1 \ell_2 L}}{\sqrt{(4\pi)(2L+1)}} \int \frac{d\cos\theta_{12}}{2} \left[\sum_M \begin{pmatrix} \ell_1 & \ell_2 & L \\ 0 & -M & M \end{pmatrix} y_{\ell_2}^{-M*}(\cos\theta_{12}, 0) \right] \times B_{LM}(k_1, k_2, \theta_{12}). \tag{2.61}$$

Then we can also look at the latest development of decomposing in two spherical harmonics [67], which also has a different coordinate system. As mentioned before, the TripoSH does not depend on the coordinate system, so we are free to change it. In this case, we will choose the LOS to the z axis as $\hat{n} = z$, which is represented on the right side of Fig. 2.2. The new coordinate system looks like this:

$$\begin{aligned} \mathbf{k}_1 &= \{k_1 \sin\theta_1,\ 0,\ k_1 \cos\theta_1\}, \\ \mathbf{k}_2 &= \{k_2 \sin\theta_2 \cos\varphi_{12},\ k_2 \sin\theta_2 \sin\varphi_{12},\ k_2 \cos\theta_2\}, \\ \hat{\mathbf{n}} &= \{0,\ 0,\ 1\}. \end{aligned} \tag{2.62}$$

The double spherical decomposition has been already described by the equation 2.31. They both can be related through this equation:

$$B_{\ell_1 \ell_2 L}(k_1, k_2) = (2L+1) H_{\ell_1 \ell_2 L} \sum_m \begin{pmatrix} \ell_1 & \ell_2 & L \\ m & -m & 0 \end{pmatrix} (-1)^m B_{\ell_1 \ell_2}^m(k_1, k_2). \tag{2.63}$$

This equation can also be transformed to configuration space, therefore obtaining the 3PCF, which has an identical form:

$$\zeta_{\ell_1 \ell_2 L}(r_1, r_2) = (2L+1) H_{\ell_1 \ell_2 L} \sum_m \begin{pmatrix} \ell_1 & \ell_2 & L \\ m & -m & 0 \end{pmatrix} (-1)^m \zeta_{\ell_1 \ell_2}^m(r_1, r_2). \tag{2.64}$$

2.9.1 Interpretation of 3PCF

Before emulating the 3PCF, it is important to fully understand TripoSH decomposition. There are different ways to approach it, but firstly it is important to understand what is the 3PCF in essence, and how the BAO appear in it. To begin with, we can compare different triangle configurations separately, by fixing two sides of the triangle r_{12} and

r_{13}, and varying the angle between the sides θ. In this way we can try to find separately, what kind of triangle configurations give the highest BAO signal. For now, it is a good assumption that they must appear if one or more sides cross the BAO peak. This type of analysis has been done before [29], and even BAO detection have been made [50], which can be seen in the Fig. 2.4.

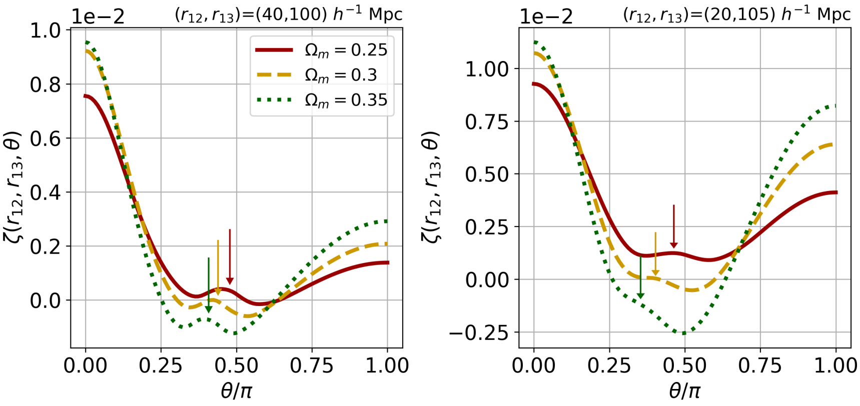


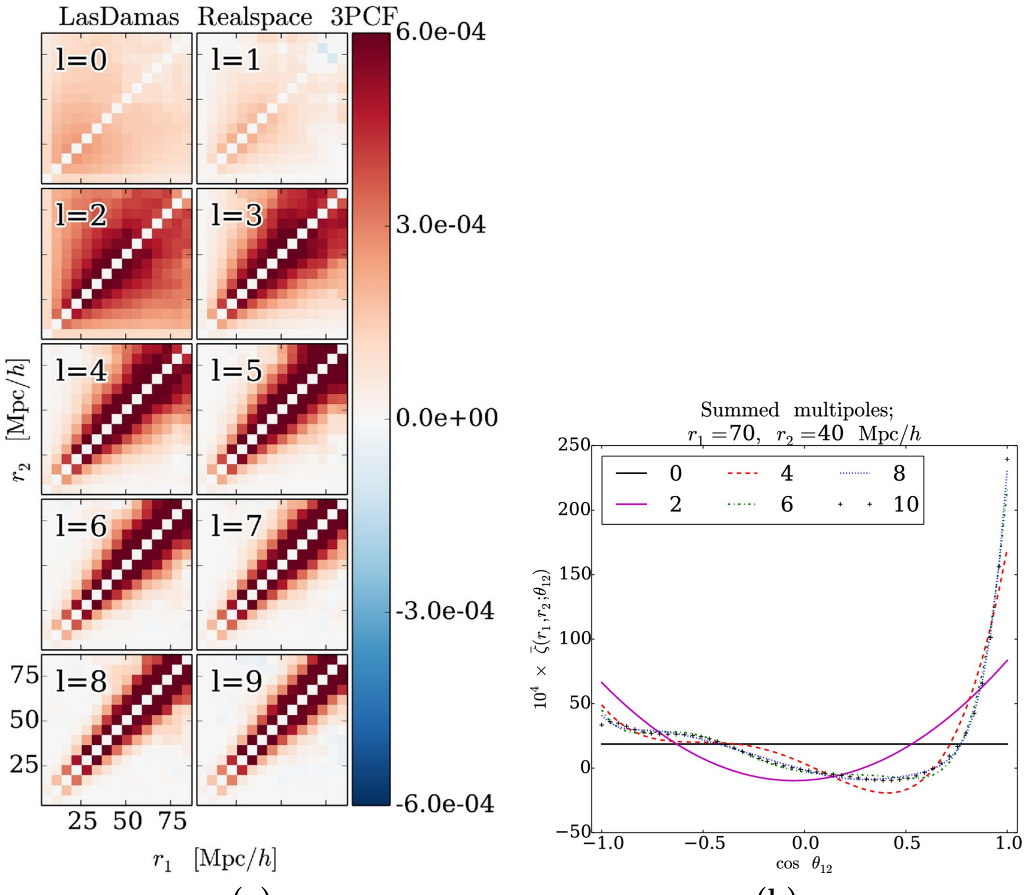
Figure 2.4. BAO detection in the connected 3PCF ζ (taken from [50]). The different colors represent the variability in the cosmological parameter Ω_m, but each column is for a different triangle configuration.

From this figure it can be noticed, that both of the 3PCF take a U-shape formed, thus giving higher 3PCF values for the elongated triangles. This is because cosmic structures tend to move along density field gradients under the influence of non-linear gravitational instabilities [29]. But the BAO signal wouldn't be present in all the configurations, but only in those, that cross the BAO scale of around 100 Mpc/h. For example, in the case of the $(r_{12}, r_{13}) = (20, 105)$ Mpc/h we have one side that is directly BAO scale, and then the other one is considerably smaller. Thus, the third side will range from 85 Mpc/h to 125 Mpc/h, thus it will have an impact from BAO through almost all values of θ.

Taking the other case in the Fig. 2.4, which is $r_{12}, r_{13} = (40, 100)$ Mpc/h, it can be seen that the peak is sharper. This is because now the third side ranges from 60 to 140, thus BAO is concentrated in the range $0.3 \leq \theta \geq 0.5$. This is important, as it shows how the squeezed triangles have a higher signal, as it is more concentrated.

The next step would be to decompose the 3PCF. Also, that can be done in a multitude of ways, but the first and foremost is to perform spherical harmonic decomposition. We have an option to perform the decomposition on any of the three parameters that the 3PCF depends on. If we only decompose with respect to one of the parameters, the most sensible would be the angular one, θ. This has been done in the section 2.6.

With this in mind, on the left side of Fig. 2.5 we can see how this decomposition looks. In this particular case, the elements on the diagonal were excluded due to the fact that perturbation theory is not accurate for the squeezed triangles. Then, on the right side is the main idea of the decomposition. In general, in order to reconstruct the full 3PCF, we would need to re-sum an infinite number of terms in the expansion. However, due to the weak angular dependence of the 3PCF, we only need a finite number of terms. On top of that, we can reach convergence even when the expansion includes fewer than ten terms. This is very powerful, as it not only allows us to analyze separately the angular dependence of the 3PCF, but also greatly speeds up the calculations.



(a)

(b)

Figure 2.5. Spherical decomposition of the angular dependence for the 3PCF, taken from [65]: **(a)** Applied to the LasDamas real space mock catalogs, where the diagonal is excluded due to the dominant nature of the squeezed triangles, which are not well described in the perturbation theory. **(b)** The re-summed 3PCF for different order of expansion ℓ for the LasDamas real space mock catalogues.

The next option is to decompose in both the triangle sides r_1 and r_2 [67], which also allows probing the anisotropic behavior of 3PCF due to RSD. The visualization of it can be seen in Fig. 2.6. This is even more powerful than the method by equation 2.24, as now we can probe the RSD effects. Nevertheless, BipoSH has its own shortcomings, such as all the expansion factors include $m \neq 0$, therefore they are variable under rotation [70]. Also, it only probes the monopole term of the 3PCF.

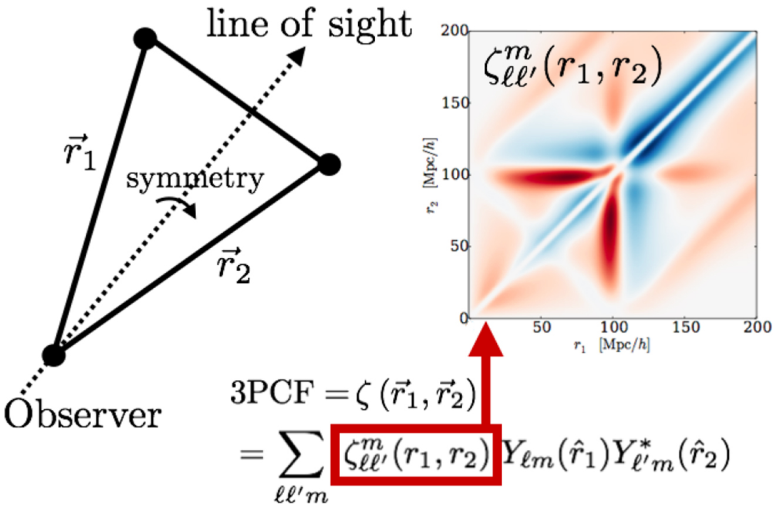

line of sight
$\vec{r}_1$
symmetry
$\vec{r}_2$
Observer
$\zeta^m_{\ell\ell'}(r_1, r_2)$
200
150
100
50
0
r_2 [Mpc/h]
50
100
150
200
r_1 [Mpc/h]
$3\text{PCF} = \zeta(\vec{r}_1, \vec{r}_2)$
$= \sum_{\ell\ell' m} \boxed{\zeta^m_{\ell\ell'}(r_1, r_2)} Y_{\ell m}(\hat{r}_1) Y^*_{\ell' m}(\hat{r}_2)$

Figure 2.6. Illustration for 3PCF spherically decomposed in r_1 and r_2, taken from [67]. The left drawing depicts the parameters in the equation, as well as the assumed symmetry along the line of sight. The equation and the red box show the basis of the spherical decomposition. The plot on the right is a suggestive figure to show how the 3PCF multipoles might look.

The final option, and the one that is used in this work, is the combination of the equation 2.24 and Fig. 2.6, thus the decomposition in all three parameters of the 3PCF. It can be calculated by exploiting the TripoSH decomposition shown in equation 2.32, where the template bispectrum is used, expressed in equation 2.51. Then it can be transformed to configuration space by 2D-FFT with equation 2.54, where to speed up the calculation even more, the decomposition of the model is used, described in equation 2.56 and AP effects included by equation 2.59.

The matter redshift space 3PCF in TripoSH basis expanded into 8 multipoles, is seen in Fig. 2.7. To summarize, the 3PCF depends on the length of triangle sides $\hat{r}_{12}$ and $\hat{r}_{13}$, as well as the line of sight $\hat{n}$. In TripoSH each of these dependencies are decomposed, which can be seen by the ℓ_1, ℓ_2, and L above each of the plots accordingly.

For example, $L = 0$ is the first expansion for the line of sight. Therefore, it represents the isotropic signal from clustering, and they can be seen in the first row of Fig. 2.7. If

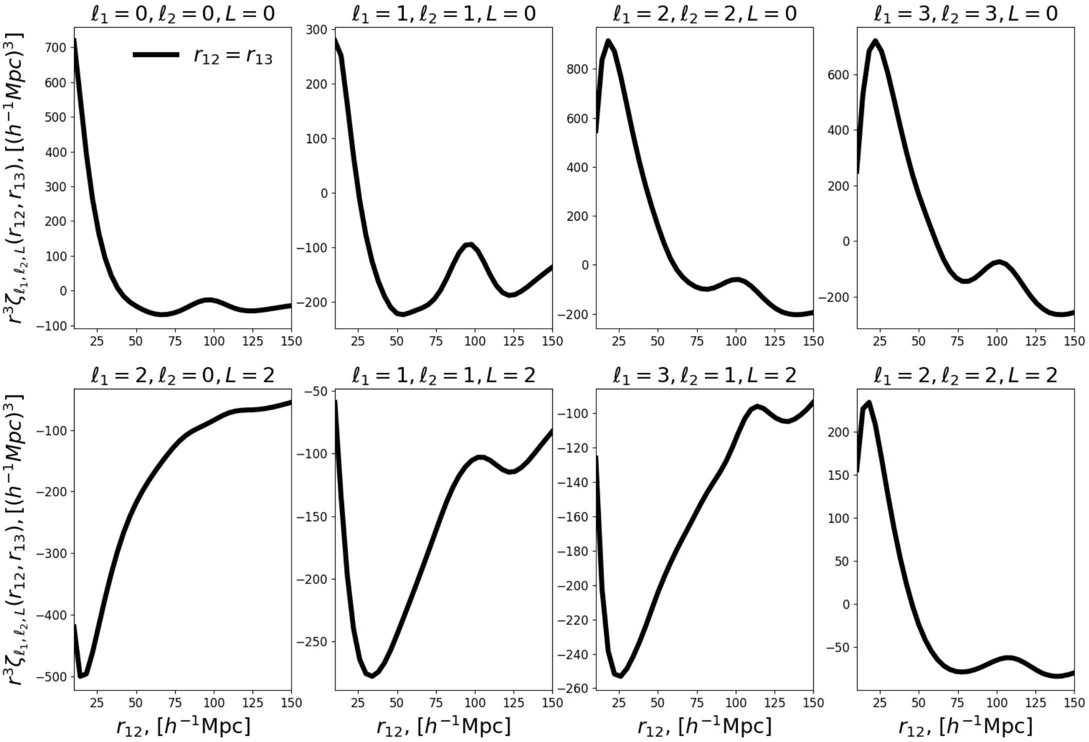


Figure 2.7. 3PCF in TripoSH basis with 8 expanded multipoles, where only the triangles that comply with the condition $r_{13} = r_{12}$ are displayed. The cosmological parameters were chosen according to *Planck18* results [3].

we expand beyond, we will then have the anisotropic signal coming from the RSD and AP effects, which can be seen in the second row.

Then we also have the expansion for r_{12} and r_{13}, which are four in each row. If we summed the whole first row, therefore all the expansion for the triangle sides, we would acquire the isotropic or the monopole term, which has been studied extensively in previous studies [66]. The quadrupole or the anisotropic term is on the second row, denoted with $L = 2$. It is important to stress that in the case of non-Gaussianity, there would be no signal. However, due to the RSD and AP effects, we observe a signal there, although the amplitude is much lower.

The Fig. 2.7 shows the 3PCF specifically for isosceles triangles, satisfying the condition $r_{12} = r_{13}$. If we wanted to display all the triangles, then it should have been displayed as a grid, which is harder to comprehend, but will be nevertheless analyzed further on in the section 4.2. Going back to our example, at first glance, it might seem that some of the multipoles are almost identical, like *220* and *330*, as well as *112* and *312*. It is important to notice that there are clear amplitude differences. Other than that, there are slight differences in the peak location, shape, and even influence from different cosmologies. This is to stress that each of the multipoles does contain independent information, and it is one of the goals of the thesis to understand which of these multipoles is the most informative.

2.10 Covariance matrix

Up until this point, we have covered how to deal with 3PCF from observations with the framework of estimators in the section 2.5, but also laid down the theoretical framework for the models themselves in the equation 2.51. To have a complete analysis, we also need to develop a way to estimate the errors for the 3PCF, which will allow us to perform SNR and Fisher analysis further on.

Errors can be estimated with the covariance matrix. The errors for each of the element in the matrix, which in our case are different triangle configurations, are on the diagonal. It is due to the fact that it represents how the parameter varies by itself. The elements outside the diagonal show how different triangles in 3PCF vary with respect to each other.

Thus, if we want to attribute errors for our 3PCF function, we can suffice only with the diagonal elements. But if we want to go further and constrain the cosmological parameters, optimally, we require the full covariance matrix.

Before delving into the theoretical aspects of how to compute the covariance matrix, there are a few key properties worth mentioning. It is always symmetric, as the variance between two elements does not change by changing the order of elements. Furthermore, the diagonal is always non-negative, as it represents the variance around the mean value, which by definition is non-negative. This is not true for the off-diagonal elements, as the negative values just represent anti-covariance between the two elements.

From the covariance matrix, we can also compute the correlation matrix, which is a normalized covariance matrix:

$$Cor(X_i, X_j) = \frac{Cov(X_i, X_j)}{\sigma_i \sigma_j}, \tag{2.65}$$

where the X_i and X_j in this work are the different triangle configurations, and the σ_i and σ_j are the corresponding variances of these elements. This gives us a dimensionless matrix, which is not scale-dependent anymore. Therefore, it ranges from 1 to -1. This allows visualizing the correlations between different triangle configurations better, giving a new perspective of the containing information. By definition all the values on the diagonal will be 1, therefore it is useful only for the analysis of the off-diagonal correlation.

We will work in the Fourier space for simplicity reasons, and only at the end we will transform our calculations to the configuration space. We will also assume a Gaussian random field, which induces the constraint $\langle\hat{\zeta}\rangle = 0$. The definition of the covariance matrix is as follows:

$$Cov \equiv \langle \hat{\zeta}(\mathbf{r}_1, \mathbf{r_2})\hat{\zeta}(\mathbf{r'}_1, \mathbf{r'}_2)\rangle. \tag{2.66}$$

To compute it, we will use again the discrete Fourier transform, as explained in the subsection 2.8.3. Thus, in the redshift space we again define the 3PCF for clarity:

$$\hat{\zeta}(\mathbf{r}_1, \mathbf{r_2}) = \int \frac{d^3s}{V}\delta_{RSD}(\mathbf{s})\delta_{RSD}(\mathbf{s}+\mathbf{r}_1)\delta_{RSD}(\mathbf{s}+\mathbf{r}_2), \tag{2.67}$$

where δ_{RSD} is the density field in redshift space, where further on the subscript RSD will be discarded for simplicity. Based on the previous literature computations [65, 75, 71], we can obtain the final form of the Covariance, represented as:

$$\begin{aligned}
\mathrm{Cov}_{\ell_1\ell_2 m\ell_1'\ell_2'm'}(r_1, r_2; r_1', r_2') =& \frac{(4\pi)^{3/2}}{V}(-1)^{m+m'}(-1)^{\ell_1+\ell_2+\ell_1'+\ell_2'} \\
&\times \int r^2\, dr \sum_{\ell_q\ell_p\ell_k} \frac{1}{\sqrt{(2\ell_q+1)(2\ell_p+1)(2\ell_k+1)}} \\
&\times \sum_{J_1J_2J_3} D_{\ell_1\ell_2J_1}C_{\ell_1\ell_2J_2J_3}\begin{pmatrix} J_1 & J_2 & J_3 \\ 0 & 0 & 0 \end{pmatrix} \\
&\times \Bigg\{ \xi_{\ell_k}(r)\Big[w_1 f^{\ell_q}_{J_1\ell_1\ell_1'}(r; r_1, r_1')f^{\ell_p}_{J_2\ell_2\ell_2'}(r; r_2, r_2') \\
&\quad + w_2 f^{\ell_q}_{J_1\ell_1\ell_2'}(r; r_1, r_1')f^{\ell_p}_{J_2\ell_2\ell_1'}(r; r_2, r_2')\Big] + \begin{pmatrix} J_1 & J_2 & J_3 \\ S_1 & S_2 & S_3 \end{pmatrix} \\
&\times \Bigg[f^{\ell_q}_{J_1\ell_1}(r; r_1)\Big(w_3 f^{\ell_p}_{J_2\ell_2\ell_2'}(r; r_2, r_2')f^{\ell_k}_{J_3\ell_1'}(r; r_1')\delta^K_{S_1-m,S_3-m'} \\
&\quad + w_4 f^{\ell_p}_{J_2\ell_2\ell_1'}(r; r_2, r_1')f^{\ell_k}_{J_3\ell_2'}(r; r_2')\delta^K_{S_1-m,S_3m'}\Big) \\
&\quad + f^{\ell_p}_{J_2\ell_2}(r; r_2)\Big(w_5 f^{\ell_q}_{J_1\ell_1\ell_2'}(r; r_1, r_2')f^{\ell_k}_{J_3\ell_1'}(r; r_1')\delta^K_{S_2m,S_3-m'} \\
&\quad + w_6 f^{\ell_q}_{J_1\ell_1\ell_1'}(r; r_1, r_1')f^{\ell_k}_{J_3\ell_2'}(r; r_2')\delta^K_{S_2m,S_3m'}\Big)\Bigg]\Bigg\}.
\end{aligned} \tag{2.68}$$

Here the w_i are the weights that depend on the total angular momentum and the spin m and m'. Then the $f^{\ell}_{H\ell}$ are the f-tensors from the 1D or 2D integral transforms, that represent the wavevector magnitude from the projected covariance. Here we have obtained the important equation, where the covariance is used for the BipoSH decomposition, which includes the spin moments m and m'. To apply it to the TripoSH basis, we need to transform this covariance.

This can be done by a 2D linear change of coordinates between the two covariances, shown in:

$$\begin{aligned} \mathrm{Cov}_{\ell_1\ell_2 L}^{\ell_1'\ell_2' L'} &= (2L+1)(2L'+1) H_{\ell_1\ell_2 L} H_{\ell_1'\ell_2' L'} \sum_{m,m'} (-1)^{m+m'} \begin{pmatrix} \ell_1 & \ell_2 & L \\ m & -m & 0 \end{pmatrix} \\ &\times \begin{pmatrix} \ell_1' & \ell_2' & L' \\ m' & -m' & 0 \end{pmatrix} \mathrm{Cov}_{\ell_1\ell_2 m}^{\ell_1'\ell_2' m'} . \end{aligned} \tag{2.69}$$

It is important to stress that we have assumed a Gaussian random field, which yields several simplifications in our calculations. Although very useful, it does provide a biased covariance matrix, as we discard the non-Gaussianity, which will be present in low-redshift observations and theories. This has non-negligible effects, particularly for the small scales. These effects can be estimated from the comparison between the numerical covariance calculations, which have the base of non-Gaussianty included in the simulations, and the theoretical. It has been shown in literature [75] that the most affected are the small scales, where the Gaussian covariance gives underestimates of the errors, as the theoretical ones are smaller than the ones obtained from the numerical covariance matrix. As the diagonal of the covariance is used for the estimation of the errors, it will both affect the analysis further in Chapter 4 and 5, where the SNR analysis and forecasts of the parameters will be performed.

3

Development of emulators for clustering statistics

In the previous chapter, we have discovered the complexity of the underlying model of the 3PCF. We also estimated the massive computational costs required to compute a single model, described in depth in the subsection 2.8.3. If the 3PCF model is needed only for one cosmology, it is not a big issue to compute it. The problems arise when thousands of these models are needed for cosmological parameter constraining or any other analysis.

Therefore, emulators are the most popular choice to speed up these calculations. In this chapter I will describe what is an emulator, how it works and why is it needed in the case of 3PCF models, which will be explained in the sections 3.1 and 3.2.

After that I will walk through the steps of building three emulators for matter power spectrum, no wiggle power spectrum, and only wiggle power spectrum in the section 3.3. This is important for several reasons. Firstly, no wiggle and only wiggle power spectra are needed for the calculation of the 3PCF, as seen in the equation 2.51. Thus, it represents one approach to accelerate the 3PCF calculation, which will further come true as a negligible speed-up. The main reason is to understand the possible ranges of the cosmological parameters, develop strategies for building an emulator, explore the hyperparameter space, and find the most suitable neural network architecture. It is useful to start with the power spectrum, as several emulators have already been built for it [69, 42, 12]. This enables us to compare the existing models and their accuracy, as well as their hyperparameters, dataset length, and the choice for number of wavenumbers.

After that, I will present the results from the emulation of the 3PCF in the section 3.4. It will include the analysis of the neural network architecture on the basis of the power spectrum strategies. It also includes a section 3.4.3, where we have made a useful modification for the used Python library in the construction of the emulator. There we explain the reasoning behind it and also show the improvements from the modified version of this library.

3.1 Machine learning tools and applications in cosmology

Machine learning (ML) has become a crucial tool in modern research. Especially in astrophysics and cosmology, where we have more and more data, as well as increasingly more complex theories. ML can find patterns that otherwise can be missed. By doing so, it can also fasten the computation of complex theoretical models without losing significant information. There are many ways to apply machine learning in science and beyond, which will be briefly explored in the next section.

Machine learning — a subset of artificial intelligence (AI) that focuses on developing algorithms and statistical models that enable computers to perform specific tasks without explicit instructions. ML systems learn from data, identifying patterns and making predictions or decisions based on new input data.

There are different kinds of algorithms. They are divided into two large categories: supervised and unsupervised learning. Supervised learning is for labeled data. For example, if I want to make an ML model to recognize asteroids in observations, I would have to label the asteroids in my dataset. On the contrary, unsupervised learning is without any labeling of the data. Therefore, the algorithm would be free to find any patterns without any instructions.

In each case, there are various available algorithms that must be chosen according to the needs. For example, classification algorithms range from simple models like logistic regression, which provides binary predictions from a set of input variables, to more sophisticated algorithms, such as decision trees or even convolution neural networks, that can classify objects in an image.

In the case of this thesis, there is a need for a faster calculation of a difficult function, given cosmological parameters. This is a complex task, therefore it reduces the possible algorithms available. One of the options is Gaussian Process Regression (GPR). It is a non-parametric, probabilistic model used for regression tasks. It can also give an estimate of the uncertainty of those predictions. It is a powerful tool, but has high computational costs for large datasets and complex problems. The other option is neural networks, which can be used to build an emulator.

3.2 Neural networks and emulators

Neural networks represent a powerful class of models in the arsenal of statistical learning tools, particularly suited for the emulation of complex cosmological functions like the matter power spectrum or 3PCF. Unlike Gaussian processes, which inherently provide a measure of uncertainty but can struggle with very large datasets or highly nonlinear relationships, neural networks excel in handling large volumes of data and capturing difficult patterns through their deep learning architecture. These networks are structured similarly to our brain — as a cluster of neurons connected to each other.

In the case of neural networks, they have two inherent structural properties — the number of layers and the number of neurons in each layer. The basic structure can be seen in Fig. 3.1.

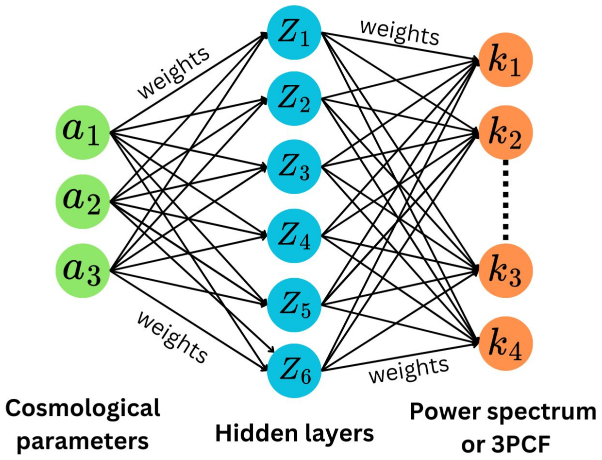


Figure 3.1. Visualization of the principles of the neural network structure. a_i are the input parameters, in this case, they are three. Hidden layers in this case are only one and consist of six neurons, denoted by their activation value Z_i. The output layer is the wanted function - either power spectrum or 3PCF. In the case of the power spectrum, the number of neurons corresponds to the number of wavenumbers for which we want the neural network to predict the power spectrum.

The input layer in our case is cosmological parameters. They can be any amount of parameters, but in this example, they are three. Each of them has a value a_i. The next part is the “dark forest” of neural networks — hidden layers. These are the neurons that find patterns and correlations in order to connect the input layers to the output ones. In this example, there is one hidden layer, which consists of six neurons. Each neuron is connected to all of the input parameters. Each connection or synapse has weights, for example, the connection from a_1 to Z_1 has weight w_{11}.

Z_1 is obtained by a weighted sum of the input layers, so $Z_1 = a_1 w_{11} + a_2 w_{12} + a_3 w_{13}$. That is not all, as then this weighted sum is passed to the activation function. It can come in different forms, which depend on the use case, but the idea is to introduce non-linearity in the algorithm and allow it to describe more complex functions. For example, one of the activation functions can be a sigmoid. It is great for binary classification problems, as it gives the layer value of 0 or 1.

The last thing is the bias of the neuron. This allows us to have more control over the activation of the neuron. For example, in the case of a sigmoid, the neuron would be activated, if the weighted sum is bigger than zero. The addition of bias allows us to tweak this threshold, which can activate the neuron. The final form for the activation value Z_1 for the neuron is:

$$Z_1 = f(a_1 w_{11} + a_2 w_{12} + a_3 w_{13} + b_1), \tag{3.1}$$

where f is the activation function and b_1 is the bias. The general expression for all of the neurons can be written as a product of matrices:

$$\sigma \left(\begin{bmatrix} w_{0,0} & w_{0,1} & \cdots & w_{0,n} \\ w_{1,0} & w_{1,1} & \cdots & w_{1,n} \\ w_{k,0} & w_{k,1} & \cdots & w_{k,n} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_3 \end{bmatrix} + \begin{bmatrix} b_0 \\ b_1 \\ b_n \end{bmatrix} \right) = Z \tag{3.2}$$

The equation 3.1 shows how the layer's activation number can be calculated from the input layer. Therefore, if there are multiple hidden layers, the needed calculations for each neuron get exponentially more. In our example alone, there are $3 \times 6 \times 4 = 84$ weights, if we assume that we have only 4 wavenumbers at which to predict power spectrum.

These weights and biases need to be adjusted from their initial guesses to have solid predictions. From the initial guess, we apply the cost function to our results, where it is possible to see how the prediction of the function deviates from the dataset function. It averages the cost over the dataset, and then we get a singular value, which indicates how well our model performs.

To minimize this cost, we use the method of gradient descent. We can apply a gradient to our cost function $\nabla C(w_i)$. In our simple example, it is already a function of 84 parameters. From this calculation the weights of our algorithm are iteratively adjusted, in order to find the minimum of our cost function.

This is done by using backpropagation. This is a method in which we start from the end of the algorithm, therefore the output layer. If we take one of the neurons from Fig. 3.1, for example, the k_1, from the cost function we know that the power spectrum value should be either higher or lower there, given a certain cosmology. Thus, we look at all the synapses connected to it, as well as the previous layer activations. There we can adjust the weights and biases to ensure that the value is getting closer to the true value. In the same manner, it keeps going layer by layer backward and repeats it for every output neuron. Then it is repeated for every power spectra in the data. There is a combination of “tweaks and nudges” for each weight and bias, and all of them are averaged over, to obtain the overall best-performing network. The data is separated into smaller batches for the computation of gradient descent, which is called stochastic gradient descent. This will be further explained later.

To summarize, training a neural network emulator involves backpropagation and gradient descent algorithms to minimize the error between the predicted and actual outputs from training datasets derived from simulations. The flexibility of neural networks allows them to adapt their internal parameters to fit complex, multidimensional data, making them exceptionally useful when the parameter space is vast and the relationships are nonlinear.

3.2.1 Designing the neural network

Even though the main idea of the neural network seems to be relatively simple, building a successful one involves multiple considerations that must be carefully addressed. To begin with, it is crucial to choose an appropriate task for it. It depends on the available resources and the complexity of the problem. For example, if we wanted to build a neural network for twenty cosmological parameters, it would require a very large dataset, in order to have a good sampling of it. So one problem is generating a sufficiently large dataset, but the other is training a model with a very large dataset, which can also take a considerable amount of time. Also, the parameters are important, because, if we have limited choice of them, we must pick the ones containing the largest amount of information. Therefore, it is important to understand the capabilities of our own resources, regarding the problem at hand.

Training Techniques

This brings us to the choice and generation of our data. Training an emulator involves careful *Data Preparation*: Generating a training set from a theoretical framework, which systematically explores the parameter space. It is crucial to have enough training data for the neural network to capture all the complexities and behavior of the function. For different training datasets a different model design is needed, but, if training data is not sufficient, even the most accurately chosen model design will not be accurate.

To ensure a well-spread parameter space, *Latin Hypercube sampling* (LHS) is used in this work. In LHS, the range of each input variable is divided into equally probable intervals. Then, for each variable, a value is randomly selected from each interval. The sampling is structured in such a way that all intervals are equally sampled, but each value is only used once. This process ensures that the sample is spread across the entire input space, making it the most representative of the possible combinations of input variables.

After that, some data might need some processing. It can be either normalizing or standardizing the input parameters to ensure they are on a similar scale, which helps in speeding up the convergence during training. In the case of the power spectrum, it is better to convert the power spectrum in logarithmic units, as then it has more evenly spread data, which is easier to process for the neural network.

Data Splitting: The dataset should be divided into training, validation, and test sets. The training set is used to fit the model, the validation set is used to tune the model's hyperparameters and prevent overfitting, and the test set is used to evaluate the model's performance after the training phase. A common split ratio is 70% training, 15% validation, and 15% test. In the case of the power spectrum, it was chosen to have 10% test and validation datasets, as the amount of data is large enough. In the case of 3PCF, this choice was analyzed separately. The test data is generated separately from the training set, to emphasize that the testing set was not taken as a fraction of the training set.

Hyperparameters in neural network

The architecture of a neural network must balance its complexity and performance to effectively model the function across the desired parameter space. This involves deciding all the hyperparameters of the neural network: the number of layers and nodes in a neural network, as well as batch size, learning rates, gradient accumulation steps, patience values, and the maximum epoch. Let us see how to choose each one of them:

- **Number of layers**: The number of layers in a neural network defines its depth, which influences its capacity to learn complex patterns. For functions such as the power spectrum and 3PCF, a deeper network might be necessary to capture the intricate relationships within the data. On the other hand, too many layers can induce longer computations and can lead to overfitting. A good start is 3 layers, and depending on the results it can be increased or decreased.

- **Number of neurons in each layer**: The number of neurons in each layer determines the network's width, affecting its ability to process information. A larger number of neurons can provide more computational power, as discussed about the number of operations needed to estimate the weights of each synapse, but also risks overfitting, especially with smaller datasets. A common practice is to pick the number of neurons as a power of two. Usually, a good start is a number between the number of input and output nodes, but it depends on each case individually, as it depends on the complexity of the functions and even on the available dataset.

- **Batch size**: Batch size refers to the number of training samples used to update the network's weights once, which goes back to the previously mentioned stochastic gradient descent method. A smaller batch size often leads to a noisier gradient, which is computationally faster and can help escape local minima, but may also cause the training process to be unstable. Larger batches provide a more accurate estimate of the gradient but consume more memory and computational resources.

- **Learning rates**: The learning rate governs the size of the steps the model takes in the direction of minimizing the loss function. It determines how quickly or slowly a model updates its weights in response to the estimated error during each iteration of training. A learning rate that is too high can cause the model to converge too quickly to a suboptimal solution or even diverge, skipping over the optimal solution entirely. Conversely, a learning rate that is too low can lead to a very slow convergence process, where the model might get stuck in a local minimum or require excessive training time to reach an acceptable level of accuracy.

- **Gradient Accumulation Steps**: Gradient accumulation is a technique used to effectively increase the batch size when limited memory is a constraint. In standard training, the model updates its weights after every batch of data is processed. However, when using gradient accumulation, instead of updating the model's weights after each batch, the gradients are accumulated over several batches before performing an update. This parameter is particularly useful when dealing with very large datasets or when using particularly memory-intensive models. The number of steps depends on the specific computational limitations.

- **Patience Values**: Patience values are used as an early stopping parameter, that allows to save computational resources, as well as prevent overfitting. It determines how many epochs should the algorithm keep training without any improvement in the cost function. Setting this parameter requires balancing between allowing adequate training time and preventing excessive training.

- **Maximum Epochs**: The maximum epochs parameter sets a hard limit on the number of times the learning algorithm will work through the entire training dataset. This prevents the model from training indefinitely and is often set based on previous tries. If the model converges reliably within a certain number of epochs, it is a good idea to set it slightly higher to ensure convergence.

- **Activation function**: The activation function is the one that determines the activation value of the neuron, taking as the input the weighted sum of all the previous layer activation values, weights, and its own bias. It gives the non-linear nature of the network, allowing to compute more complex problems with higher accuracy. There are numerous types, among the most popular being — sigmoid function, hyperbolic tangent (tanh), or Rectified Linear Unit (ReLU). Sometimes custom functions are made, in order to fit the needs of the user.

- **Optimizers**: Before, we only mentioned the gradient descent method as a possible optimizer. It turns out that there are many other options, and even better ones, to perform this task. But in general, optimizers are responsible for adjusting the model's weights to minimize the loss function during learning. They play a crucial role in determining how the model learns from data and how quickly it converges to a solution. Optimizers work by calculating the gradients of the loss function with respect to the model's parameters and then updating those parameters to reduce the loss. There are various types of optimizers, each with its own advantages and trade-offs.

  The most basic optimizer is Gradient Descent, which updates all weights in the network simultaneously after calculating the gradient of the cost function. A more practical option is Stochastic Gradient Descent (SGD), a variant of the basic gradient descent, which updates the model parameters more frequently, one mini-batch at a time. While this introduces more noise into the updates, it can also help the model escape local minima, making it a more robust choice in practice.

  RMSprop (Root Mean Square Propagation) is another powerful optimizer that divides the learning rate by a running average of recent gradient magnitudes. This approach helps stabilize the learning process, especially in situations with noisy or highly dynamic data. Adam (Adaptive Moment Estimation), one of the most widely used optimizers today, combines the strengths of AdaGrad and RMSprop. It adaptively adjusts the learning rate for each parameter based on estimates of the first and second moments of the gradients, making it particularly effective for handling sparse gradients and noisy data.

  Given its robustness and efficiency across a wide range of problems, Adam is the optimizer of choice for many modern neural networks, including the models used in this work. Each optimizer has its unique strengths and weaknesses, making it essential to select one that aligns with the specific characteristics of the data and the training objectives of the neural network.

- **Loss**: It is critical to evaluate how well the model is performing during training. The loss function quantifies the difference between the predicted outputs of the neural network and the actual target values from the training data. One of the most common loss functions is the Mean Squared Error (MSE), which is defined as:

  $$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2, \tag{3.3}$$

  where n is the number of data points, y_i is the true value, while the $\hat{y}_i$ is the predicted value. This function penalizes large deviations more heavily, which makes it sensitive to outliers in the data.

  The choice of the loss function has a direct impact on the model's learning process, as it conducts how the weights are adjusted. There are also different loss functions, such as *Mean absolute error*, and some more complicated ones, for example, *Log loss*, *Huber loss* or *Hinge loss*. In this work, MSE is used, which is also the most widely used in neural networks.

3.2.2 Emulators and Cosmopower

The goal of this work is to build an emulator for the 3PCF. But what is an emulator? An emulator is a machine learning-based model designed to replicate the behavior of a complex and computationally expensive model. Essentially, it is based on the training of a neural network, and from a certain input it predicts an accurate output, in our case power spectrum or 3PCF, with significantly lower computational costs. It is a great tool for very complex models that can take several minutes or even hours to produce a result, which can greatly limit its use in science. Emulators allow to acquire an output in less than a second, which enables more efficient exploration of complex models and aids in the interpretation of extensive cosmological data.

In this work, we used an existing library for building an emulator — *CosmoPower* [1] [69]. It uses machine learning to accelerate the Bayesian inference. Although it is made for the field of cosmology, it can also be generalized to other scientific fields. It works based on neural networks, and it is used mostly to constrain cosmological parameters. It already provides a few emulators, which are for CMB and matter power spectra, where

[1] The *CosmoPower* package can be found in: https://github.com/alessiospuriomancini/cosmopower

they cover a wide range of cosmological parameters.

CosmoPower provides two frameworks for the neural networks. One is the standard one, which was explained at the beginning of the chapter and is represented by Fig. 3.1, but the other option is neural networks with the addition of Principal component analysis (PCA). PCA transforms the data into a new coordinate system, where the first axis (principal component) captures the most variance in the data, the second axis captures the second most variance, and so on. By retaining only the first few principal components, you can reduce the dimensionality of the data while preserving most of the variability. It allows for speeding up calculations and sometimes making the models more effective, but in the scope of this work, it was not used, as there was no need for it.

It is worth noting that *Cosmopower* implements a custom activation function, in the form of:

$$f(\mathbf{x}) = \left(\boldsymbol{\gamma} + \left(1 + e^{-\boldsymbol{\beta}\odot\mathbf{x}}\right)^{-1} \odot (1 - \boldsymbol{\gamma})\right) \odot \mathbf{x}, \tag{3.4}$$

where $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$ are parameters that are optimized in the same way as the rest of the neural network parameters. This choice was made, as the authors of the code saw a slight increase in accuracy over the traditional activation functions, such as ReLU and tanh.

3.3 Emulating matter power spectrum

Even though the main goal of this Thesis is to produce an emulator for the 3PCF, there are several reasons to start with the emulation of the power spectrum. Firstly, one of the inputs for the calculation of the 3PCF is the power spectrum. Thus, building an emulator for the power spectrum makes the dataset generation for the 3PCF faster. It will be later seen why this is not a crucial step.

Secondly, there are no emulators for the redshift space 3PCF, but there are already many for the power spectrum. Hence, it is useful to first understand the intricacies of *CosmoPower*, parameter limits, neural network structures, and the structure of the pipeline, before delving into 3PCF. In this way, it is possible to compare the accuracy of the model with the literature and evaluate the success of the emulator by doing so.

To build an emulator, the following steps were conducted. Firstly, the choice of the cosmological parameters and their ranges were chosen for the training, as well as the number of data to be generated. Secondly, a deep dive into the choice of hyperparameters and the input data was analyzed to find the best-performing model. Thirdly, we performed analysis for any other reasons that could have affected the final accuracy. In this case, we found out that certain cosmological parameter constraints

Cosmological parameter	Minimal value	Maximal value
Ω_m	0.05	0.5
Ω_b	0.02	0.08
h	0.5	0.9
n_s	0.6	1.3

Table 3.1. Cosmological parameter ranges, that are used for the power spectrum emulator

must be implemented. Finally, we can compare our best models with the literature and the desired accuracy that we are seeking to obtain.

3.3.1 Choosing the parameters and their range

To emulate the matter power spectrum, the first step is to define what parameters will be emulated and what parameters will be fixed. As a base of parameters, the results from the mission *Planck* [3] will be taken. For the emulator, we chose to vary four parameters, in particular Ω_m, Ω_b, h, and n_{spec}. This choice was made based on previous literature on power spectrum emulators [69, 40], as well as choosing parameters, that affect the shape of the function, not only the amplitude [58]. Their values are depicted in Table 3.1.

To explore what are the effects on the matter power spectrum by each of these four parameters, the cosmology is fixed and only one of the parameters is changed within the previously mentioned ranges. The results can be seen in Fig. 3.2. It can be seen that the chosen wide range of parameter values has a great effect on the matter power spectrum. The most notable effect is seen by changing the Ω_m parameter. When the ratio of $\frac{\Omega_m}{\Omega_b}$ is close to unity, the BAO oscillations in the power spectrum play a dominant role in its shape.

3.3.2 Generating the training and testing dataset

When the parameters have been defined, the next step is to generate the training and testing dataset. To do so, the “CosmoBolognaLib” [48] library is used. The “CosmoBolognaLib” is a comprehensive software library developed and maintained by cosmologists at the University of Bologna. It provides a suite of C++ and Python tools specifically designed to support the statistical analysis and computations of cosmological data. “CosmoBolognaLib” includes functionalities for calculating matter power spectrum, two-point and three-point correlation functions, modeling the redshift-space distortions, and estimating the covariance matrices, among other tasks. This library combines

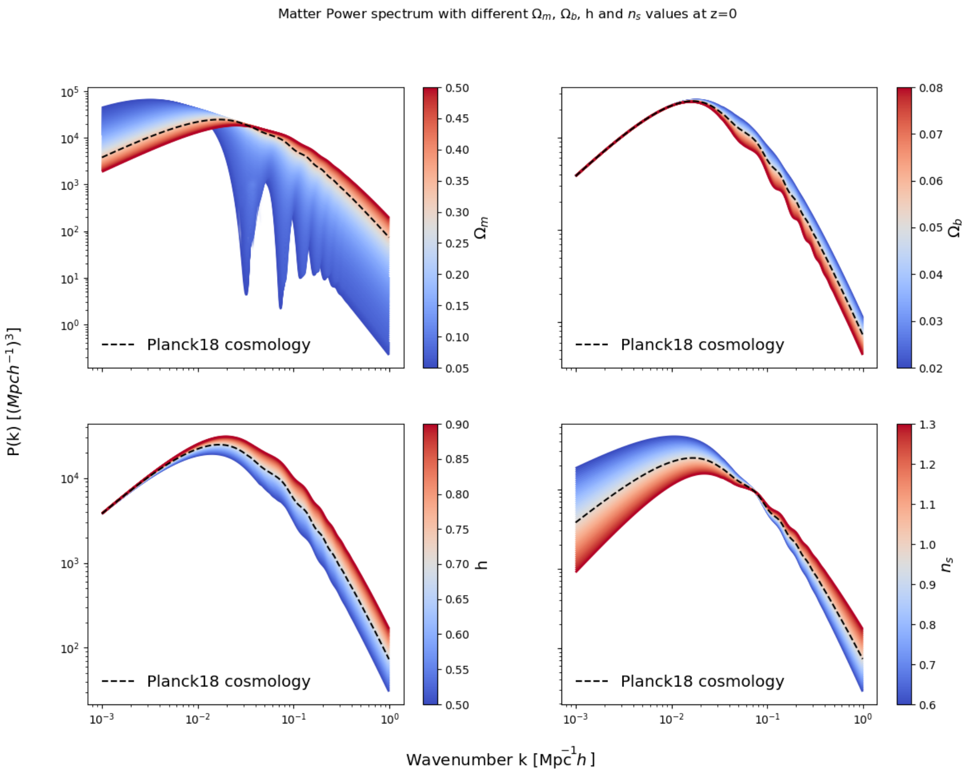


Figure 3.2. Cosmology effect on power spectrum; Matter power spectrum with parameters shown in Table 3.1, where the rest are fixed to the values from Planck results in 2018 [3]. *(Top right)* Varying matter density. Here for the lowest Ω_m also Ω_b was varied, as otherwise, the function diverges; *(Top left)* Varying baryon density; *(Bottom right)* Varying dimensionless Hubble parameter; *(Bottom left)* Varying spectral index.

robust, scientifically validated methods with an efficient computing framework, which can be freely used by the scientific community.

With “CosmoBolognaLib” the matter power spectrum was calculated for the training and testing dataset. In particular, the “Code for Anisotropies in the Microwave Background” (CAMB) [45] was used for this purpose. “CAMB” is a well-established cosmological software package used primarily to calculate cosmic microwave background (CMB) anisotropies and matter power spectra. It is based on the CMBFAST code and is used by most cosmologists studying the early Universe. The software computes the evolution of perturbations in the photon-baryon fluid and provides detailed predictions of the CMB anisotropies, as well as the matter power spectrum at both linear and non-linear scales. “CAMB” can be run as a standalone application or integrated with Python. Its versatility also extends to modifications for alternative cosmologies, making it adaptable to various theoretical frameworks. It is the most precise tool to calculate the matter power spectrum in the field, and, as it is implemented in the "CosmoBolognaLib", it is used in this work for calculations.

To understand the amount of spectra needed for the training model, the most robust method is to see when the model will converge at a particular accuracy. Therefore, it is best to compute a large amount of data and then choose the appropriate amount for the training. In this work, 200'000 spectra were calculated at the start. This number was chosen for several reasons:

- **Parameters and their ranges:** The size of the training set depends on the number of parameters and how much they vary. The more the parameters, the more degrees of freedom the function has. That can lead to degeneracy between parameters and higher complexity of the function. For these reasons, a larger dataset is required for the emulator to have the capability to grasp the features of the function.

- **Complexity of the function:** dataset is also dependent on the complexity of the function itself. In the case of matter power spectrum, the shape is relatively simple, except for the Baryonic Acoustic Oscillations (BAO). They are responsible for the wiggles in the function, and at $k \sim 0.1$ in the Fig. 3.2 it can be seen that in the case of a low Ω_m, the function gets very complex, therefore harder to emulate.

- **Literature:** Even though the parameters and complexity of the function are well understood, some reference is needed to estimate the needed amount of data. There have been many published works about the emulation of the matter power spectrum. For example, in the [69] they had a training set of 180'000 spectra.

Taking all of this into account, it was chosen to have 200'000 matter power spectra generated with "CAMB", which is implemented in "CosmoBolognaLib". Even though in the mentioned literature [69] they had more parameters, in this work the range of parameter values is higher, which gives a higher complexity of the function. Therefore, an even larger dataset could be considered.

3.3.3 Training

At this point, the training and testing dataset has been generated, and the main components of the neural network have been covered. Despite the general idea of choosing all the components being clear, it still requires experimenting with different configurations to reach a design for the neural network that satisfies the needed accuracy.

To evaluate the success of the training, one of the most general metrics is the loss. There are different ways to calculate it, but the most general, and also implemented in *CosmoPower*, is the Mean square error, which is the same as in the equation 3.3. This is a useful tool to compare different models between each other, but it does not provide information on where the largest errors are.

To have more information on the accuracy of the trained emulator, the percentage difference is used, as shown in the equation 3.5:

$$\text{Percentage Difference} = \left(\frac{|P_k^{(true)} - P_k^{(em)}|}{P_k^{(true)}} \right) \times 100\% \tag{3.5}$$

Where $P_k^{(true)}$ represents the matter power spectrum, either excluding or including BAO features, computed using CAMB for the full matter $P(k)$, and $P_k^{(em)}$ is the matter power spectrum obtained from the emulated model. After using this equation for each of the spectra in the testing dataset, a median can be taken at each wavenumber k. This gives the median percentage difference of a particular model. In this case, the median is used instead of the average for the reason not to give too much weight to the extreme cases, where the emulated power spectrum can have uncharacteristically low or high accuracy. This is a great tool to compare different models and to pick the one that gives the finest results.

3.3.4 Emulating matter $P(k)$ without BAO features

Even though $P(k)$ calculation without BAO features, which will be further denoted also as no wiggle $P(k)$, includes calculating the matter power spectrum itself, it is better to start with this function, as it is simpler. Therefore, it should be more straightforward to emulate and compare different models between each other.

There are multiple ways to obtain the matter power spectrum without BAO features — using spherical Gaussian filters on 3D power spectra, 1D Gaussian filters on a logarithmic scale, or basis spline. In this work, 1D Gaussian filters were used [77]. The method can be summarized in the Equation 3.6:

$$P_{nw}(k) = P_{approx}(k)\mathscr{F}[P(k)/P_{approx}(k)], \tag{3.6}$$

where $P(k)$ is the initial power spectrum which will be smoothed, $P_{approx}(k)$ is the initial broad band approximation curve, which is BAO feature free — it serves the purpose of reducing the amplitude range and $\mathscr{F}$ is the $1D$ Gaussian filter.

The training set was chosen to be of 200000 spectra, and the number of k was

hyperparameter	Value
Layers	4
Neurons	512
Batch size	512
Learning rate	1e-1 to 1e-7
Patience value	20
Max epochs	2000
Validation split	20%
Optimizer	AdamW

Table 3.2. The reference hyperparameters for the matter no wiggle power spectrum.

picked to be 700, in order to accurately remove the BAO features, as well as to improve the estimate of the slope of the $P(k)$ at the small and large scale. The Ω_m range was reduced from 0.1 to 0.5 from the analysis of the section 3.3.5.

The first thing to do would be to find the best hyperparameters for the model. As a reference model, the neural structure with the hyperparameters displayed in Table 3.2 was chosen. The reason for these specific hyperparameters is that in the literature [69] it was found that this model structure is the best for their emulated $P(k)$. The only change is the batch size, where it was reduced from 1024 to 512 to avoid overfitting. This model was compared with different models, by changing the number of spectra in the dataset, neurons in a layer, number of layers, different optimizers, and batch size. The result of this analysis can be seen in Fig. 3.3 .

There are many interesting features to be seen. To begin with, the dataset size impacts the accuracy of the model greatly. Thus, it solidifies the choice of picking the dataset that large.

Further on, also number of neurons shows a predictable decrease in the percentage difference. This is expected, as the neuron connections that are not important will simply have less weight given to them. Therefore, even if more neurons are chosen than optimally would be required, it usually does not have a negative effect.

The choice of layers is more complex. From Fig. 3.3 (c), two models have similar accuracy — with four and three layers. Even two layers show good accuracy. From this graph, it can be concluded that an insufficient number of layers can lead to underfitting, where the network lacks the capacity to capture the underlying patterns in the data. Conversely, an excessive number of layers can result in overfitting, where the network becomes overly complex and sensitive to noise in the training data, leading to poor generalization. It seems that the four layers are not large enough to be put in the category of overfitting. Even though the three-layer model overall performs better, it

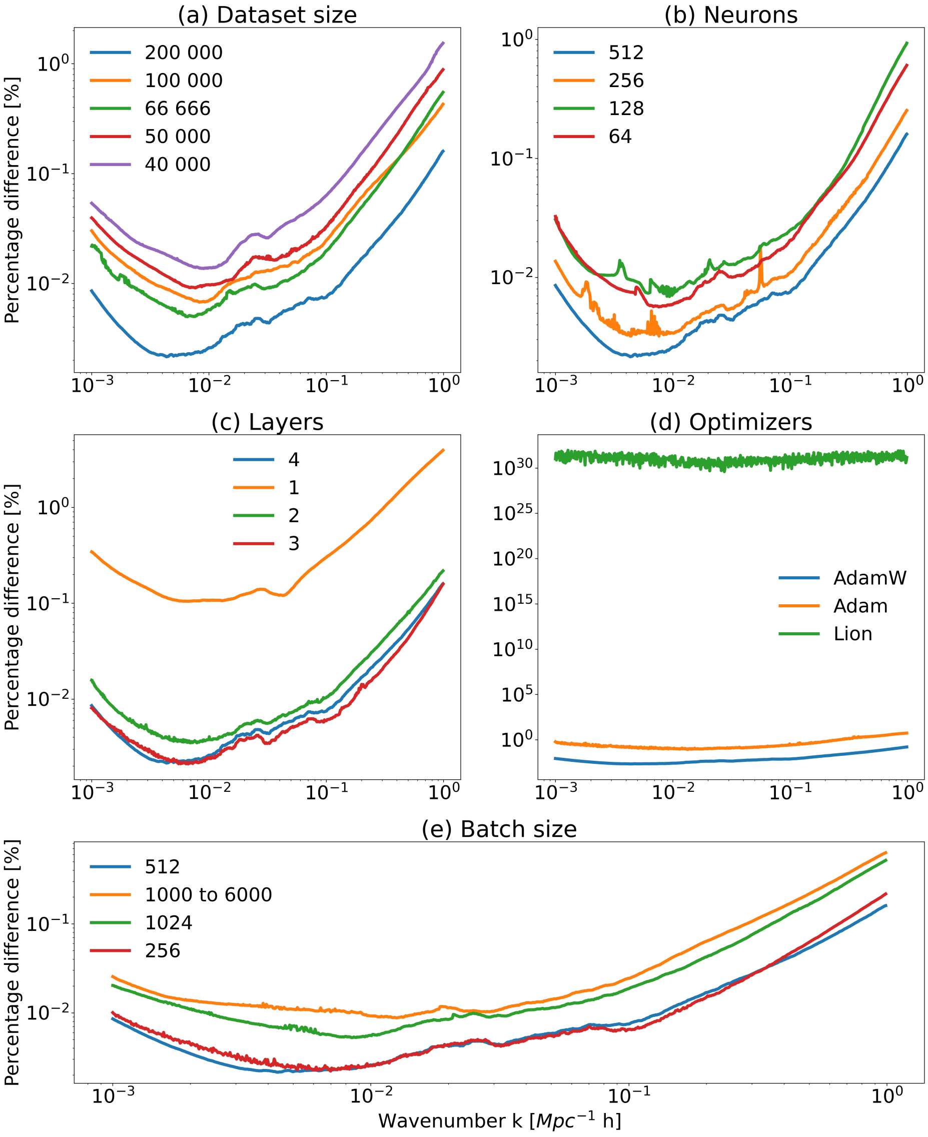


Figure 3.3. Comparison of hyperparameters; Median percentage difference calculated for: **(a)** Changing number of spectra in the dataset, **(b)** Changing the number of neurons in the layers, **(c)** Changing the number of layers, **(d)** Changing the optimizers, **(e)** Changing the batch size, where all of them are fixed, except the "1000 to 6000", which varies in each epoch. The reference one is shown in Table 3.2.

is worth noting that the slope of the percentage difference in the large scales is bigger than in the four-layer model. This indicates that also the actual slope of the function is estimated worse.

Optimizers have a very large impact. It can be seen that some optimizers can even diverge, therefore not producing a valuable model. AdamW is outperforming Adam and Lion, and it will be chosen as the optimizer also for other models.

Lastly, batch size, similar to the influence of layers, can lead to overfitting or underfitting, therefore a careful choice must be made. Here it can be seen that a large batch size is not performing better, as it most probably overfits the data. On the other hand, even though a batch size of 256 is a good model, it can be seen that at small and large wavenumber k it performs worse, which could be due to underfitting. This proves that the batch size of 512 is the most optimal.

hyperparameter	Reference model	batch size 256	3 layers	256 neurons	log10
Layers	4	4	3	4	4
Neurons	512	512	512	256	512
Batch size	256	256	512	512	512
Learning rate	1e-1 to 1e-7	1e-1 to 1e-7	1e-1 to 1e-7	1e-1 to 1e-7	1e-1 to 1e-7
Patience value	20	20	20	20	20
Max epochs	2000	2000	2000	2000	2000
Validation split	15%	15%	15%	15%	15%
Optimizer	AdamW	AdamW	AdamW	AdamW	AdamW
Log values	no	no	no	no	yes
Average percentage difference	5.53e-3	5.52e-3	4.99e-3	7.65e-3	3.00e-2

Table 3.3. The best hyperparameters for the matter power spectrum with no wiggles.

Apart from these, there are also not-so-trivial choices to be made. For example, the number of points in the spectra, therefore the number of wavenumber k in the given range. Training the model with the reference parameters in Table 3.2, the results can be seen in Fig. 3.4. It appears that even by reducing the number of wavenumber k by half, the effect is minimal. This means that in further analysis it can be reduced.

In Fig. 3.4 the influence of training the neural network when spectra are in logarithmic scale is displayed. This is a useful test because the spectra span in several orders of amplitude. By putting our data on a logarithmic scale, it changes the distribution and amplitude of our data, possibly allowing for the model to find the relationships between points more easily. Further on, as the models work on gradient descent, the model can

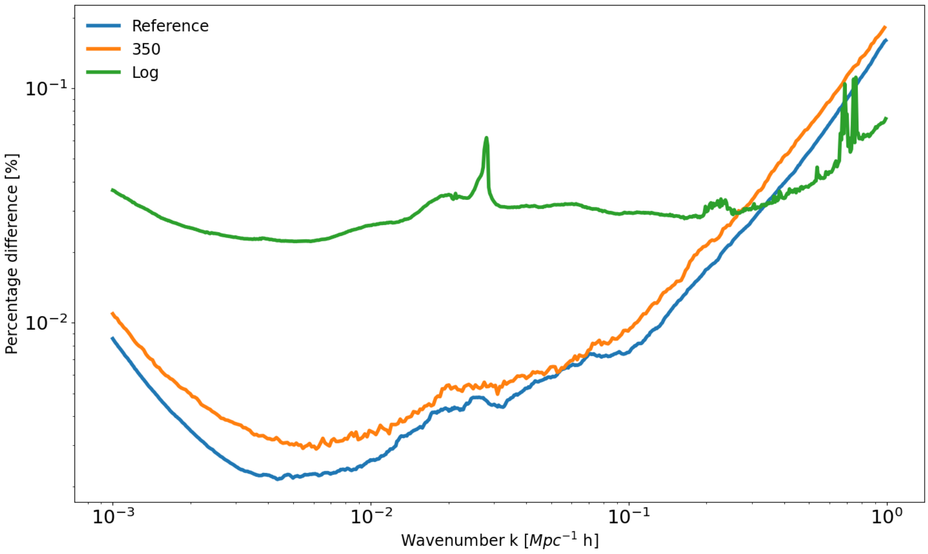


Figure 3.4. Tests for improving the emulator; Median percentage difference calculated for smaller amount of wavenumbers k. In this case 350 points. A second test was performed for a model that is trained on logarithmic data. Both of these are compared with the reference spectra in Table 3.2.

converge in a more stable fashion and more accurately. Nevertheless, an improvement was not found in this case, as the logarithmically trained model is significantly worse in this case. Although the slope of the percentage difference is more flat, thus, even though it starts with a worse accuracy of an order, at higher k the model performs even better. Hence, this idea can be tried in other models. It will be further seen in the section 3.4.3, that the different scaling of input data is significant, but to do so, modifications of the *CosmoPower* must be made.

To finalize the choice of the most appropriate emulator for the matter power spectrum of no wiggles, five of the best models were compared between each other in Fig. 3.5 and summarized in Table 3.3 together with their average percentage difference values. It can be seen that the three-layer model outperforms the rest, but for now, there is no information about how widely spread is the accuracy among different combinations of parameters.

To analyze that, the reference plot and three-layer model were plotted together with their percentile areas between $14th$ and $86th$ percentile in Fig. 3.6. From this, it can be observed that both models are extremely similar, but the three-layer model area is narrower, hence it is performing better. Therefore, the conclusion is that the best model

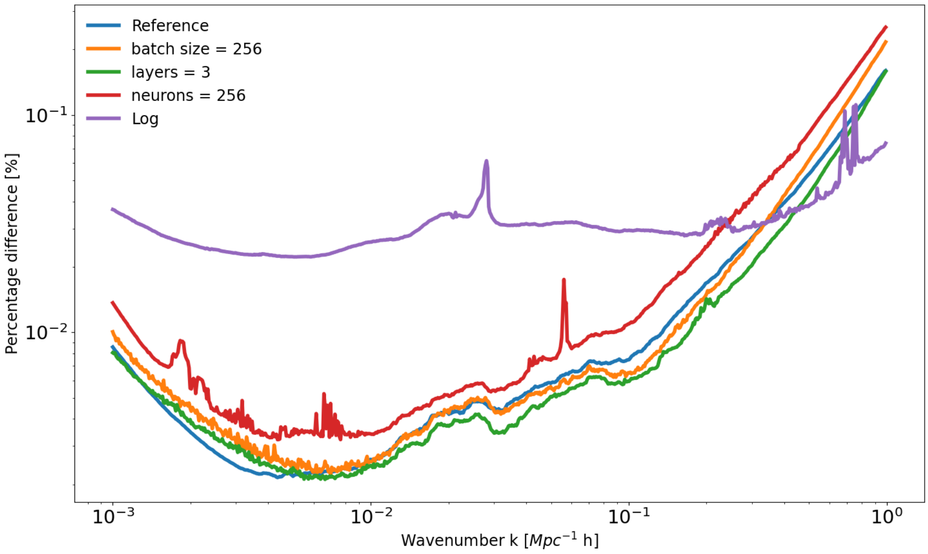


Figure 3.5. Best models for no wiggle emulator; Median percentage difference calculated for the best models of the hyperparameter and other value analysis of the neural network, summarized in 3.2.

to emulate the matter power spectrum with no wiggles is model "3 layers" in Fig. 3.5.

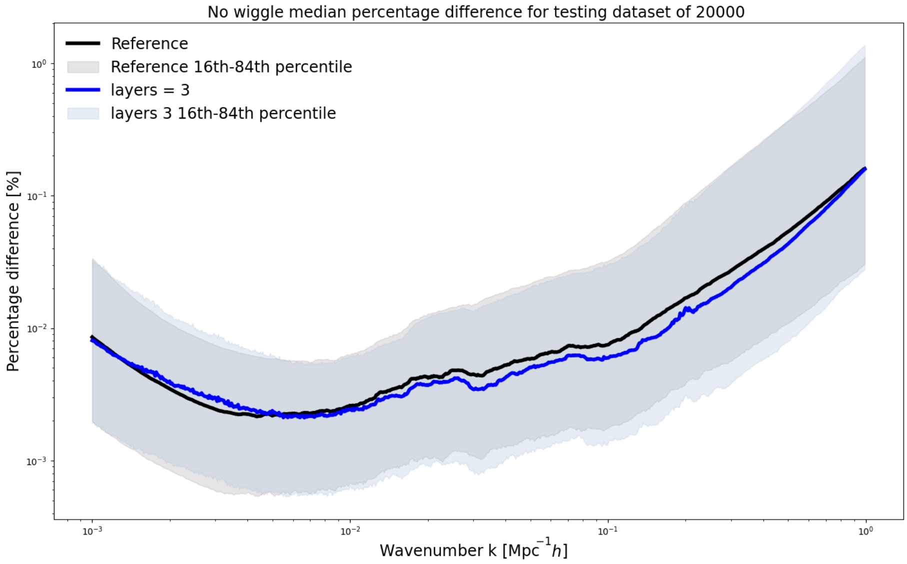


Figure 3.6. Two best model comparisons; Median percentage difference with percentiles calculated for the reference model and three-layer model for no wiggle $P(k)$.

3.3.5 Emulating matter $P(k)$

As with the no wiggle $P(k)$, also here it is important to find the best parameters. But before that, learning from the previous lessons of the analysis of no wiggle $P(k)$, the training set can be changed. It was seen that 700 points were not strictly necessary for high accuracy, therefore for matter $P(k)$ it is reduced to 500. The number of spectra is generated also of the number $200'000$, but it will be tested if convergence is reached for a smaller dataset. The parameter range for training the models is shown in Table 3.1. The analysis of the parameter ranges described below has been applied to no wiggle and only wiggle $P(k)$.

With that being said, now the best hyperparameters for the model must be found. To save some time, it is useful not to take the full training set, but still a large enough portion of it to have all the parameter space be adequately represented. In that case, the model design will be also applicable to the full dataset.

After defining the reference spectra seen in Table 3.4, now it can be explored what number of power spectra would be enough to reach a convergence of the accuracy of the model. To do so, percentage difference plots are created for the same model but with different sizes of training datasets seen in Fig. 3.7. It is clear that it follows the expected behavior — the bigger the dataset, the better the performance of the model. At the same time, it can be seen that convergence has been achieved with this particular model at $150'000$ spectra.

The other conclusion is that there is a steep increase in the percentage difference at the BAO scale, colored by the red region, which reduces the accuracy by an order, therefore at the largest scales, the difference is already of two orders. Even with that, the accuracy is still sub-percent, which is the aimed accuracy of the model.

To further analyze the reason for this steep increase in percentage difference, pa-

hyperparameter	Value
Layers	4
Neurons	512
Batch size	1024
Learning rate	1e-1 to 1e-7
Patience value	20
Max epochs	2000
Validation split	20%
Optimizer	AdamW

Table 3.4. The reference hyperparameters for the matter power spectrum.

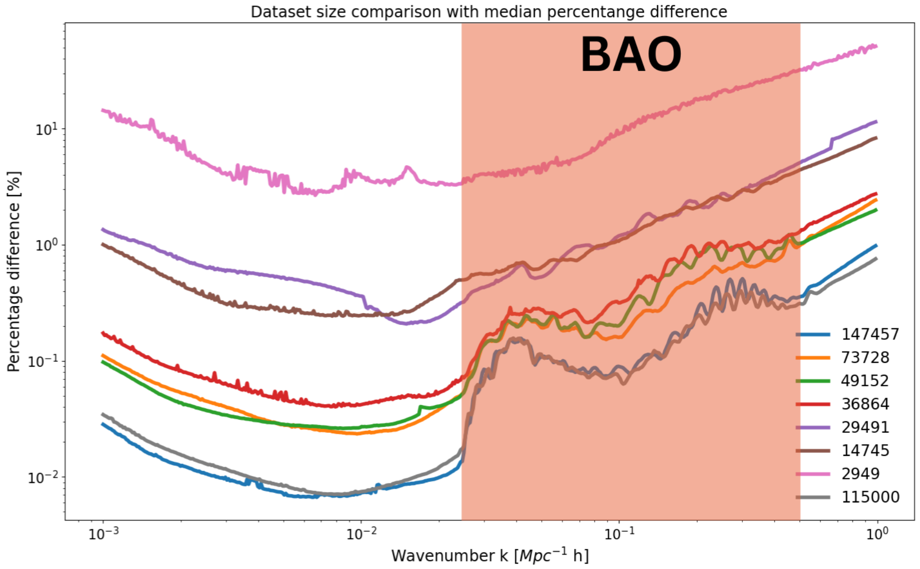


Figure 3.7. Dataset size comparison; Matter power spectrum trained on hyperparameters in Table 3.4 with different sizes of the training set. The red region indicates where the BAO features appear in the $P(k)$.

rameters were taken separately at different ranges and the percentage difference of the neural network was estimated, shown in Fig. 3.8.

It can be seen that the largest effect is from the Ω_m parameter, when it has a small value of around 0.1 or less, or from the ratio $\frac{\Omega_m}{\Omega_b}$, mainly when it is smaller than 1.2. From closer inspection, it is clear that the ratio is the main source of error, which is due to the dominant presence of the BAO. Another problem might appear at those scales, which are nearly nonphysical parameter combinations. That is also confirmed by separately plotting individual cases of small $\frac{\Omega_m}{\Omega_b}$, shown in Fig. 3.9. It can be seen that there are extreme cosmological parameter values that produce a nonphysical power spectrum in comparison to a well-behaved power spectrum. This means that the model is trained on nonphysical power spectra, which can greatly affect the final accuracy of the model. To see the effects of that, in one case all the spectra with $\Omega_m < 0.1$ were removed, and in the other case all spectra with $\frac{\Omega_m}{\Omega_b} < 1.2$ were removed. These models were trained and can be seen in Fig. 3.10. This clearly shows an improved model accuracy at high k and a similar accuracy at lower k. Even though the $\frac{\Omega_m}{\Omega_b} < 1.2$ model seems to be the best performing, as the difference is not significant, there is slightly worse accuracy at lower k and for easier implementation in the following tasks, it was

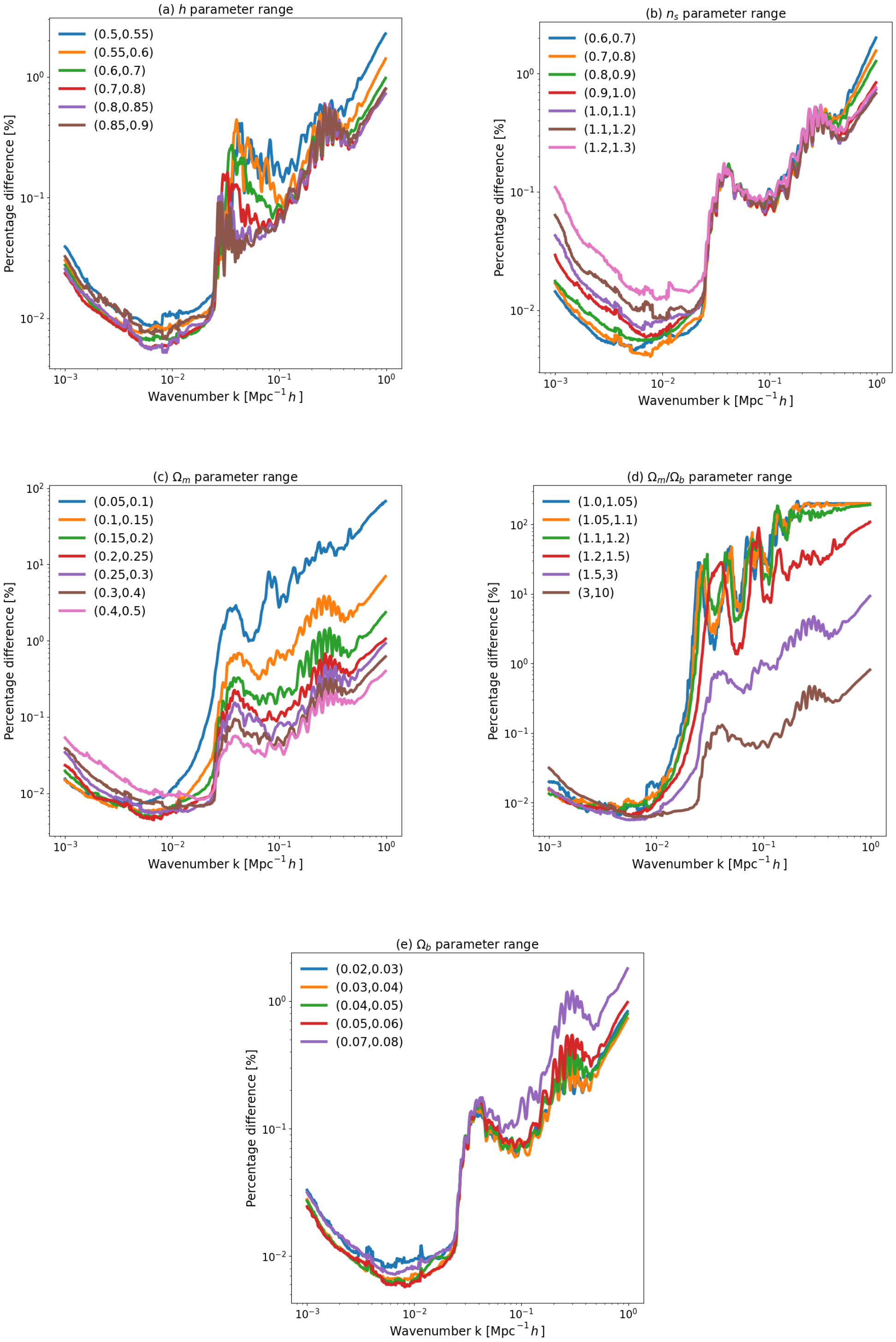
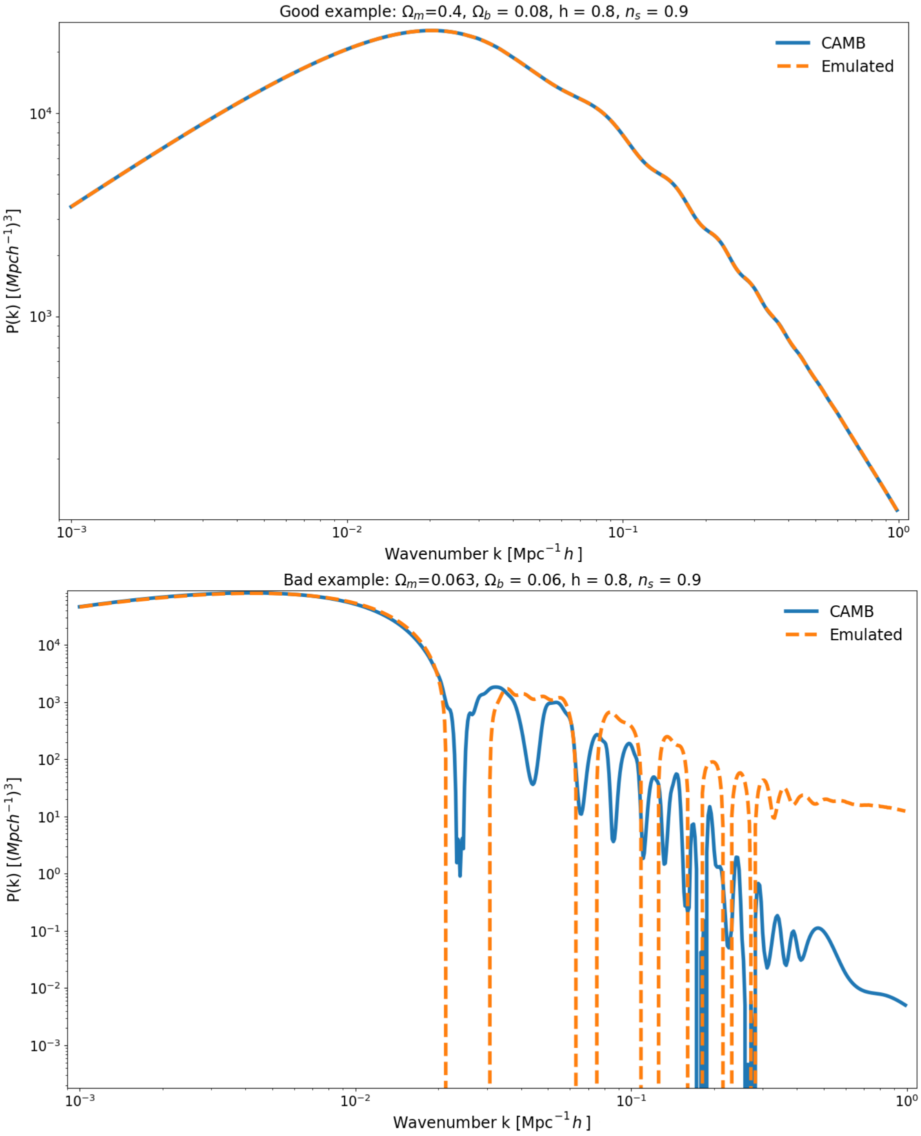


Figure 3.8. Identifying cosmological parameters that decrease emulator accuracy; Median percentage difference of the same model shown in Table 3.4, but slicing specific parameters in different ranges **(a)** Hubble constant h **(b)** Spectral index n_s **(c)** Matter density Ω_m **(d)** Matter and baryon density ratio $\frac{\Omega_m}{\Omega_b}$ **(e)** Baryon density Ω_b.



Good example: Ω_m=0.4, Ω_b = 0.08, h = 0.8, n_s = 0.9

Bad example: Ω_m=0.063, Ω_b = 0.06, h = 0.8, n_s = 0.9

Figure 3.9. Example of bad cosmological parameters for $P(k)$; Matter power spectrum calculated with CAMB by CBL compared with emulated matter power spectrum. Top figure: Well-emulated matter power spectrum compared to CAMB calculation; Bottom figure: Extreme cosmological parameter that breaks the CAMB and emulator calculation.

chosen to restrict the parameter range as $\Omega_m > 0.1$. This would also explain why in the Fig. 3.7 the dataset of $115'000$ points has even slightly higher accuracy at larger k - because it had fewer spectra with nonphysical parameter ranges, meaning low ratio of $\frac{\Omega_m}{\Omega_b}$. The conclusions of this analysis were also applied to the no wiggle $P(k)$ in section 3.3.4 and only wiggle $P(k)$ in section 3.3.6.

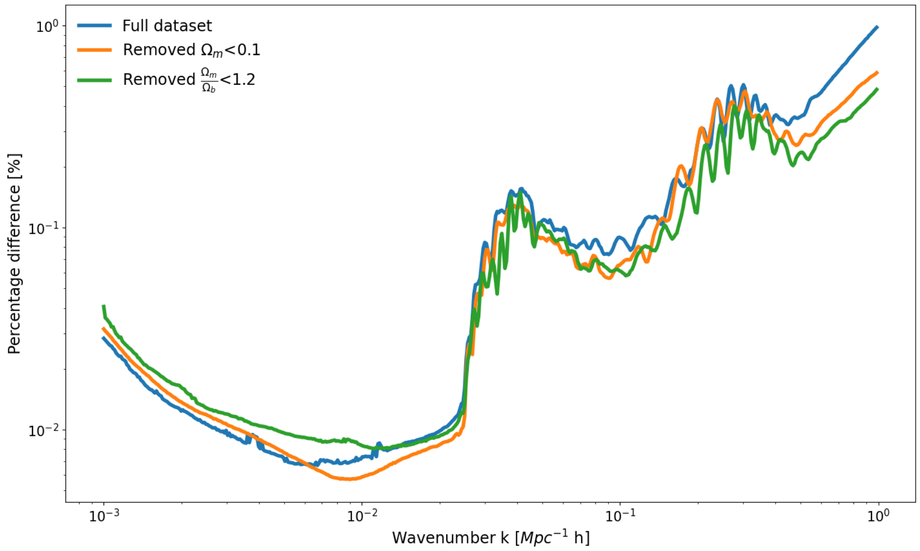


Figure 3.10. Emulator comparison with modified parameter ranges; Median percentage difference calculated for trained models for a dataset of 147457 spectra, dataset where spectra with $\Omega_m < 0.1$ were removed and dataset where spectra with $\frac{\Omega_m}{\Omega_b} < 1.2$ were removed.

Comparing the Fig. 3.10 and Fig. 3.5, it can be clearly seen that there is no steep increase at the BAO for no wiggle $P(k)$ percentage difference plots. This proves the hypothesis that it is due to the BAO wiggles. To reduce the effects of it, the number of points at this scale can be increased. Thus, we trained models where there are 150 points in k from 0.001 to 0.01, 300 points from 0.01 to 0.15 and 150 points from 0.15 to 1. In total, this gives 600 points in the spectra, where half of them are at the BAO scale. As this gives increased complexity of the non-linearity in the distribution, the number of spectra was increased to $250'000$ for the training of the models.

The best models together with two reference data can be seen in Table 3.5, and their median percentage difference is visualized in Fig. 3.11. The behavior of the best model with non-linear k distribution is interesting, as there is a significant increase in the accuracy at the BAO scale, but a reduction of it at small k. The most plausible reason is that there was a shift in the accuracy — the better emulated spectra at the BAO scale led to a worse estimation of the slope of the small scale, therefore significantly increasing the percentage difference. This could be an implication of overfitting. It is important to note that a similar analysis was made in section 3.3.4, but a better model was not found. Nevertheless, the obtained accuracy is good, as overall it is sub-percent.

To fully conclude that the Model nonlin-2 is the best, the percentile analysis was

hyperparameter	Model lin-1	Model lin-2	Model nonlin-1	Model nonlin-2
Training set spectra	147 457	115 000	250 000	250 000
Layers	4	4	4	4
Neurons	512	512	512	1024 to 256
Batch size	1024	1024	1024	1024
Learning rate	1e-1 to 1e-7	1e-1 to 1e-7	1e-1 to 1e-7	1e-1 to 1e-7
Patience value	20	20	20	20
Max epochs	2000	2000	2000	2000
Validation split	15%	15%	15%	15%
Optimizer	AdamW	AdamW	AdamW	AdamW
Non-linear k	no	no	yes	yes
Average percentage difference	1.27e-1	1.08e-1	8.93e-2	6.28e-2

Table 3.5. The best hyperparameters for the matter power spectrum.

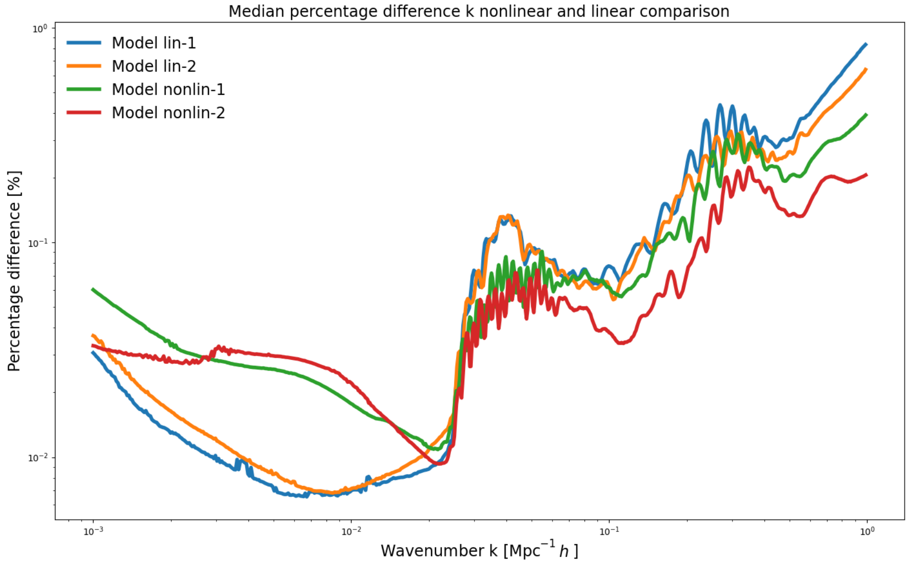


Figure 3.11. Best emulator comparison; Median percentage difference calculated for the best models emulating the matter power spectrum. Model parameters are shown in Table 3.5.

conducted in comparison to the reference model, shown in Fig. 3.12. Here it is clear that the Model nonlin-2 has a lot less scatter overall and is a better choice for the emulation of the matter power spectrum $P(k)$.

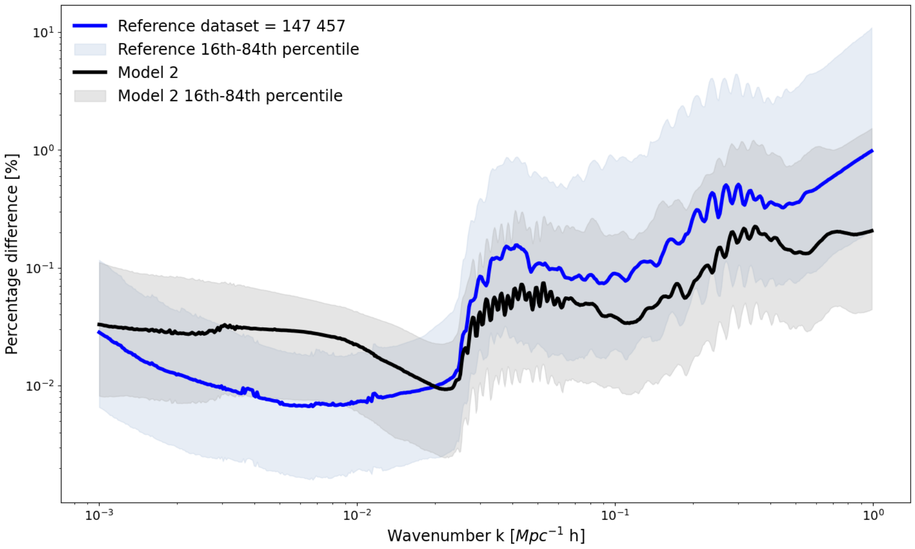


Figure 3.12. Best matter power spectrum emulator comparison with percentile regions; Median percentage difference with percentiles between 14th and 86th displayed for the two best models of the matter power spectrum emulation. Model 2 represents the "Model nonlin-2" in the Table 3.5.

3.3.6 Emulating matter $P(k)$ with only BAO features

The last step for emulating functions related to the matter power spectrum is the only BAO feature part. It is simply calculated as the subtraction between the matter $P(k)$ and no wiggle $P(k)$. Therefore, a rightful question would be — why emulate it?

In the last sections, it was shown that with the emulation of $P(k)$ and no wiggle $P(k)$ accuracy of 0.01% could be reached, which is very good for the required scientific case. When subtracting both functions, the amplitude changes by 2 orders or more. Thus, the percentage difference of the only wiggles would increase by two or more orders, if it would be calculated from the emulated spectra. With this reasoning, it is beneficial to emulate it and try to reach a higher accuracy.

A large dataset was created of $250'000$ spectra. It was calculated by picking a kernel size of 0.25 for the no-wiggle spectra. An extensive analysis was made to find the best hyperparameters of the model. The best five model parameters are displayed in Table 3.6. Here it can be seen that there is a new row “Wavenumber”, that wasn’t present in the previous tables. This means that “regular” trains the model, where wavenumber k goes from 0.001 to 1, instead “log” is executing the training, where the wavenumber is

hyperparameter	Model 1	Model 2	Model 3	Model 4	Model 5
Number of layers	5	4	4	4	4
Number of neurons	512	512	512	512	[1024, 512, 512, 256]
Batch size	1024	1024 to 10188	1024	1024	1024
Learning rate	1e-2 to 1e-7	1e-2 to 1e-7	1e-2 to 1e-7	1e-1 to 1e-7	1e-1 to 1e-7
Patience value	20	20	20	20	20
Max epochs	2000	2000	2000	2000	2000
Validation split	15%	15%	15%	15%	15%
Optimizer	AdamW	AdamW	AdamW	AdamW	AdamW
Wavenumber	log	regular	log	regular	regular
Average median percentage difference	2.32e-1	2.63e-1	2.88e-1	3.36e-1	3.73e-1

Table 3.6. The five best hyperparameters for the only wiggle matter power spectrum.

in logarithmic units, therefore from -3 to 0. The idea behind this choice is that the function is distributed more evenly, therefore the gradient calculation for the weights is more efficient.

The overall performance of all the tested models can be seen in the plot 3.13. Due to the abundance of models, it is only an indicative plot of the overall behavior of different models, and the best-performing emulators have been summarized in Table 3.6. Here the best model in gray color corresponds to Model 1. In this plot, there are also salmon color regions that specify the regions where the only wiggle $P(k)$ values are nearly zero, where an example is shown in Fig. 3.14. These regions are of little interest for the determination of the accuracy of the model, as the percentage difference will be always high in the case where the denominator in Equation 3.5 will be nearly zero. For this reason, the estimated accuracy is only determined between the salmon-colored regions. It is important to note that these regions were hand-picked, as there is no easy way to pick them — they differentiate from one spectrum to another due to different cosmology, and the function is only gradually departing from the zero value. A precise estimate is also not needed, as this is only indicative. From further analysis of the Fig. 3.13, the two peaks in the middle do not correspond to the highest or lowest amplitudes of the only wiggle $P(k)$, but they are also due to the zero crossings of the function.

Even though the accuracy of these models is worse by almost an order, it is again important to stress out the difference in amplitudes between the previous spectra and only wiggle $P(k)$. This is one of the reasons for the slightly worse accuracy model. The other is simply the fact that this function is more complex — that is also the reason why the best model is with 5 layers instead of 4.

With the best model in hand, it is possible to compare the accuracy between the emulated only wiggle $P(k)$ and the subtraction between emulated $P(k)$ and no wiggle $P(k)$. As expected, the emulated only wiggle $P(k)$ is better, as seen in Fig. 3.15.

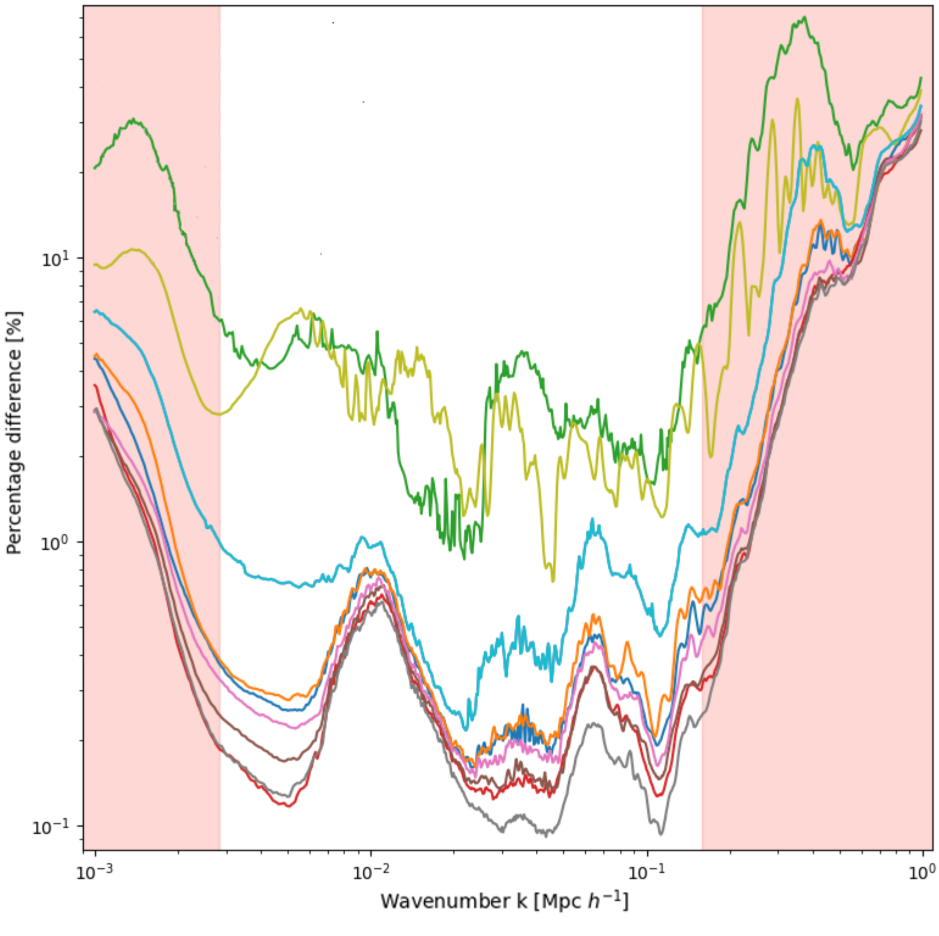


Figure 3.13. Only wiggle $P(k)$ emulator performances; Median percentage difference calculated for different models shown in Table 3.6 trained on full dataset of 250000 spectra of matter power spectrum with only BAO wiggle.

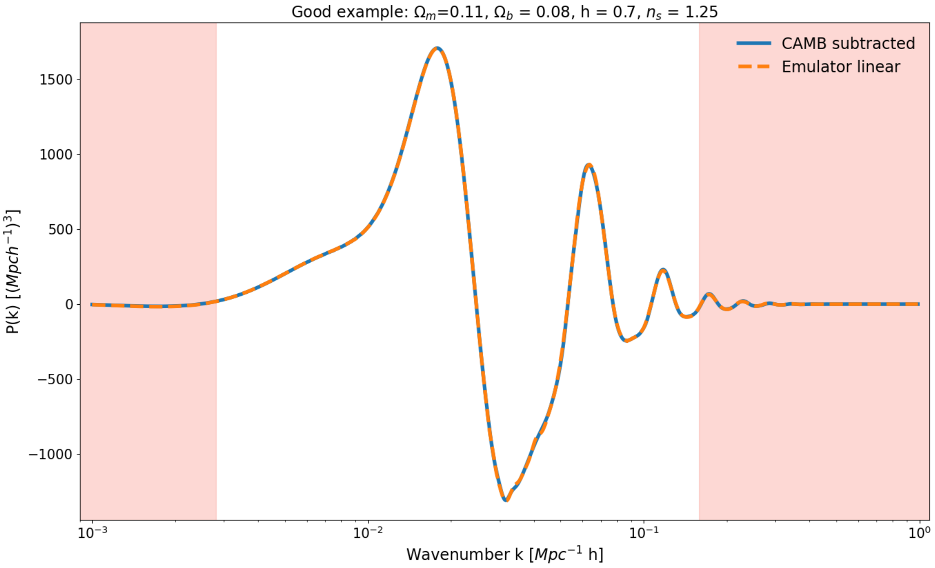


Figure 3.14. Only wiggle emulator example; Only wiggle matter power spectrum with the blue line indicating the calculation done with python CAMB, but the orange line is the emulated one. Salmon color regions are the same regions as in 3.6, indicating when the function is close to zero.

Although it is surprising that the difference is that large — it can reach more than two orders difference. This proves that even though the accuracy of $P(k)$ and no wiggles $P(k)$ models is good enough for their planned usage, their accuracy is not good enough to produce only wiggle $P(k)$ for the later on usage for three-point correlation function calculation.

The summary of the obtained emulators for the matter power spectrum, no wiggle power spectrum, and only wiggle power spectrum can be seen in Fig. 3.16. It can be seen that all of them have the desired accuracy of sub-percent. Only wiggle spectra have a deviation from this accuracy only due to the close to zero values in the spectra, which does not reflect the real accuracy well with the percentage difference formula. To visualize the results, in the top row of Fig. 3.16 can be seen the comparison between the emulated and calculated spectra. It can be concluded that all the emulators have been made successfully — they are tens of times faster, and they do not lose significant accuracy in their predictions.

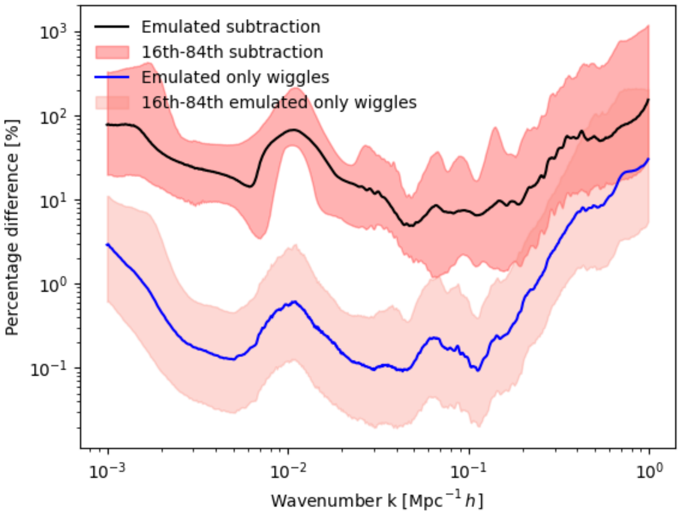


Figure 3.15. Comparison between the percentage difference calculated with emulated model for only wiggles $P(k)$ and subtraction between emulated $P(k)$ and no wiggle $P(k)$. The shaded regions imply percentile regions between 16% and 84%.

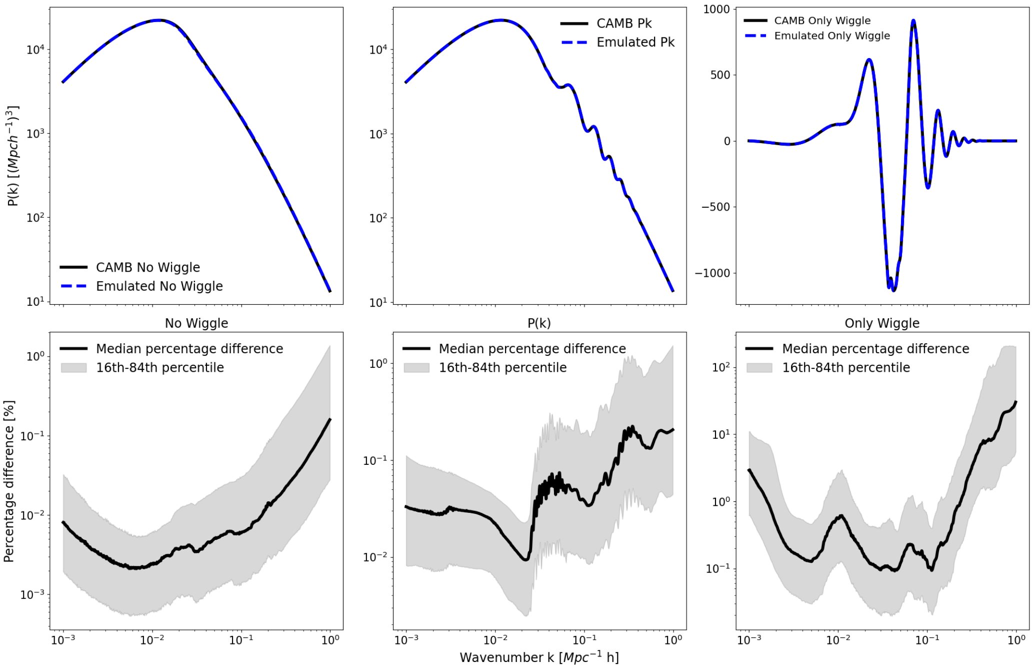


Figure 3.16. Summary of the emulated power spectra; In the top row: Example of emulated spectra and calculated spectra with CAMB. In the bottom row: comparison between the median percentage difference and the percentile regions between 16 % and 84%.

3.4 Emulating matter anisotropic three-point correlation function

This section will describe the main goal of the thesis — to build an emulator for the redshift space matter 3PCF. To do so, we must adapt the formalism described in the chapter 2.8. This has been already implemented in the Python package *Mod3l* [2] [28]. From the inputs of the function, it performs the spherical decomposition of the bispectrum and executes the 2D FFT-Log to obtain the multipoles of the 3PCF. It can calculate both the matter and galaxy 3PCF, with BAO and no BAO, as well as with template fitting or without it. In this work, the matter 3PCF code is exploited without the template fitting.

3.4.1 Preparation for the training

As with the power spectrum, the first step is to understand the computational resources, the possible data set that can be generated, and the cosmological parameters to consider. To begin with, producing one matter 3PCF model can take up to an hour on 48 CPUs, depending on the grid size and template fitting. Thus, it is more than $100'000$ times longer than the calculation of the power spectrum, which already speaks volumes for the complexity of this model. This fact already gives several constraints: it will be impossible to have as much data as in the power spectrum case, as it would require up to $9'600'000$ CPU hours, if each 3PCF calculation takes one hour. This is not feasible, therefore we will have a lot less data, which means that it would be also better to reduce the number of cosmological parameters as input parameters, in order to potentially increase the accuracy.

Instead of using DIFA OPH cluster, we need more computational resources. Therefore, we used the supercomputer center *Cineca*, which is the largest computing center in Italy and one of the largest in Europe. *Cineca's* supercomputers are equipped to handle tasks that range from large-scale simulations and data analysis to machine learning and artificial intelligence applications. It provides both CPU and GPU hardware for academic research. In our case, we acquired $65'000$ CPU hours on their supercomputer *Leonardo*, which is their latest addition to their complex of supercomputers.

Leonardo is composed of two main partitions: the Booster Module and the Data-centric Module. The Booster Module is equipped with 3456 computing nodes, each containing a single Intel Xeon Platinum 8358 CPU with 32 cores and four NVIDIA custom Ampere A100 GPUs. This configuration provides a total of $110'592$ cores and

[2] The code of Mod3l can be found: https://gitlab.com/antoniofarina/mod3l

powerful GPU acceleration for high-performance computing tasks. The Data-centric Module consists of 1,536 nodes, each powered by two Intel Sapphire Rapids CPUs with 56 cores, offering a total of $172'032$ cores.

These resources are valuable, as they allow for efficient use of available computational power and enable quick results through the ability to parallelize tasks across multiple nodes. We have $65'000$ CPU hours, and if one matter 3PCF calculation with template fitting requires around 48 CPU hours, then we could produce around 1300 3PCF. That would not be enough to make an accurate emulator for such a complex function. Consequently, we must constrain the input parameters of the model.

A major increase in time can be achieved by fixing our template parameters. This reduces the computational cost two times, hence it takes 24 CPU hours now. At the same time, these parameters are not perfectly known, therefore by assuming their value, our results will be biased towards these assumptions.

The other parameters, such as the grid size in the configuration space and Fourier space, were chosen through experiments on our own, as well as from the literature [28]. On top of that, some are pre-computed before, such as the power spectrum and no wiggle power spectrum with a given cosmology. These are the most important parameters, as the cosmology is encoded in them. Then there is also σ^2, which is the radial component of the displacement vector, $\Delta\sigma_8^2$, which is the transverse component of the displacement vector, and $f\sigma_8$, which is the growth factor multiplied by the variance at 8Mpc, which are also pre-computed with the dependence on the power spectrum.

N_k	N_x	k_{max}	k_{min}	Template parameters	α	ε	L_{order}	L_s
56	51	10	1e-4	[f, 1, 0, 0]	1	0	4	[0, 2]

Table 3.7. Input parameters for running the 3PCF model

Then there are the parameters in Table 3.7. N_k is the size of one side of the grid for the k_1 and k_2, but N_x are the number of values for the cosine of the angle between k_1 and k_2. Then we also set the k_{min} and k_{max}, which are the minimal and maximal values for the wavevector on the grid. The template parameters are, correspondingly, the growth rate of the fluctuations f, which is also pre-computed, as it depends on the chosen cosmology. The next one is the linear bias parameter b_1 set to one, and quadratic bias parameter b_2 and tidal bias parameter b_T, which are set to zero, which is a valid approximation based on literature [35]. The α and ε are the isotropic dilatation and anisotropic warping [51]. In the end, we have the L_{order}, which depicts in what order will the r_1 and r_2 be decomposed, and also L_s, which shows the $\hat{n}$ decomposition order. As can be seen, it has been chosen in favor of $L_s = [0, 2]$ to have both the isotropic

Cosmological parameter	Minimal value	Maximal value
Ω_m	0.2	0.5
h	0.55	0.85
A_s	1e-9	5e-9

Table 3.8. Cosmological parameters that are varied for the matter 3PCF emulator

and anisotropic case. Each of them will have 4 different multipole expansions, which is enough to capture the main features of the 3PCF.

The next choice for the user of the code is to select the $r_{\rm min}$, $r_{\rm max}$, and Δr. These are crucial parameters, as Δr are responsible for the resolution of the output. The range of r is trickier than for 2PCF because it is quite clear where the BAO should be located—around 100Mpc. However, in the case of 3PCF, it consists of three pairs, and each one can hold the BAO scale, even if the other two sides are not 100Mpc. Therefore, it is not obvious where exactly the influence of BAO becomes negligible for 3PCF, resulting in insignificant information. This value can be determined by experimenting with this range and comparing the matter 3PCF with the no-wiggle matter 3PCF. If at higher r the no-wiggle and full 3PCF converge, it can safely be assumed that the chosen range is appropriate. In this work, $r_{\rm max} = 200$ Mpc h^{-1} was found to capture most of the BAO information, but also $r_{\rm max} = 150$ Mpc h^{-1} is sufficient.

Coming back to computational costs, it can be seen that no other parameters can be changed to speed up the calculations significantly. Therefore, by fixing the template parameters, we chose to compute 3500 3PCF for the training set and 350 for the testing set, totaling $61'600$ CPU hours. Here the total hours are less than expected, due to the fast CPU's in *Leonardo* supercomputers, which could compute one 3PCF in 16 CPU hours. This left some computational resources for testing the pipeline before execution, as well as for running the training for the neural network.

Compared to the power spectrum analysis, we have 50 times less data. Looking back at Fig. 3.7, it might seem that this amount of data is insufficient for a sub-percent accuracy goal. For this reason, we reduce the number of cosmological parameters to three, and here the chosen parameters are Ω_m, h, and A_s, where we fix the cosmology at redshift $z = 0$. The cosmological parameter choice was made due to the potential of comparing the results with ongoing research for a 2PCF+3PCF emulator (Euclid Collaboration: Guidi et al., in prep). Also, the ranges cannot be as wide in this case, therefore we opted for values shown in Table 3.8.

To show the impact on the 3PCF itself, in Fig. 3.17 the Ω_m parameter is varied. This is only for the equilateral configuration, and it can be seen that the amplitude of

3PCF, depending on the multipole, can vary significantly. The same can be observed for h and A_s parameters. We can also see the BAO impact, which, unlike 2PCF, appears as wiggly behavior. A similar trend can also be in non-isosceles configurations, which will be explored further on.

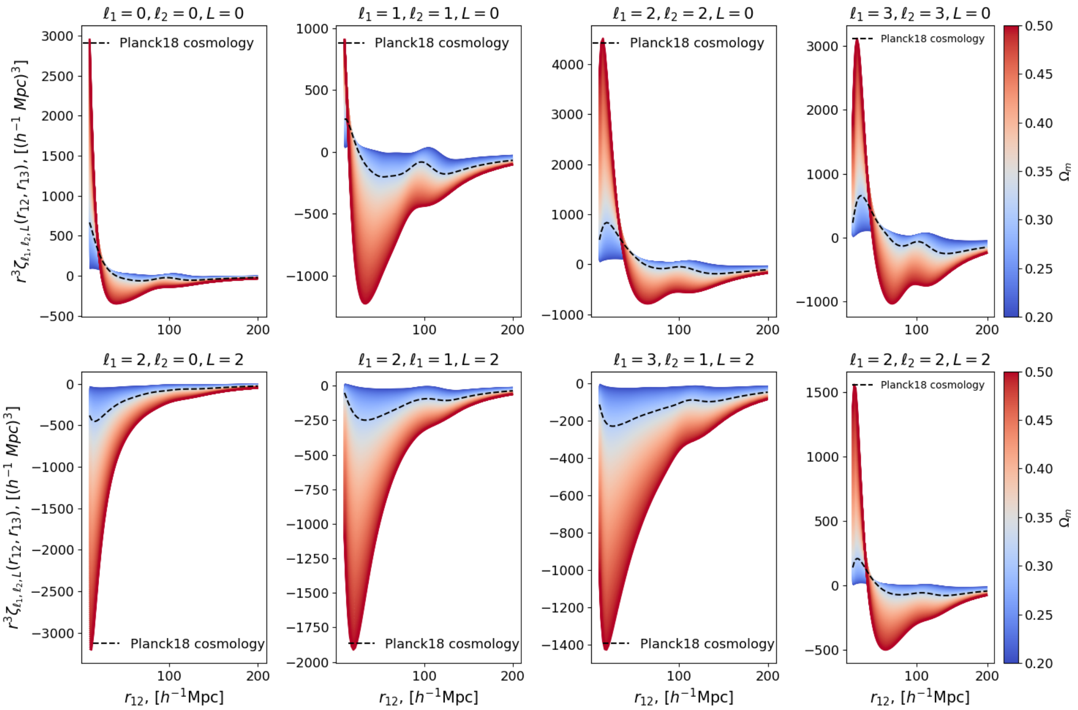


Figure 3.17. Cosmology influence on 3PCF multipoles; Matter anisotropic 3PCF, expanded into eight multipoles, restricted to isosceles triangle configurations $r_{12} = r_{13}$. In this case, the cosmological parameters h and A_s are fixed to *Planck18* values [3], while O_m is varied in the range shown in Table 3.8.

3.4.2 Pipeline for the training

As the groundwork for the input parameters, type of 3PCF, and number of cosmologies has been set, we will present the framework to produce the 3PCF emulator. The pipeline is visualized in the Fig. 3.18. The first step is to produce a set of 3500 cosmological parameters for the training set and 350 for the testing set, where the proper distribution of the parameter space is produced with LHS. The second and third step is to calculate the full power spectrum and no wiggle power spectrum with the previously produced cosmological parameters. Here the library *CosmoBolognaLib* is used, in the same manner as in the subsection 3.3.5 and 3.3.4. The only difference is also the growth rate

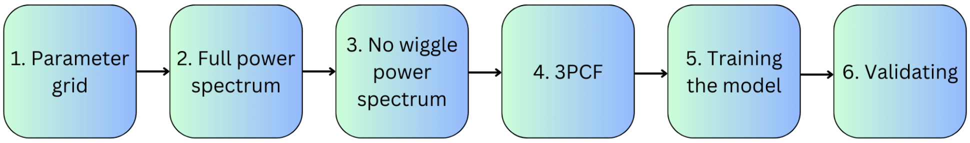


Figure 3.18. The pipeline for producing 3PCF emulator.

f calculation in the $2nd$ step, as it also requires the usage of *CosmoBolognaLib* with the input parameters.

The fourth step is generating the actual training set of the 3PCF, by leveraging the previously discussed input parameters and power spectrums. To do this, the package *Mod3l* is used. As there was a limitation of 24 hours for each submitted computation on *Leonardo*, as well as no more than 64 nodes per computation, a parallelization algorithm was implemented. The data was split into 100 equal parts for the training set, and each part was then separately calculated on the *Leonardo* cluster. After that, the 100 different parts were combined, in order to use them in the training segment. Thanks to the parallelization method, the data was generated in a single day.

The next steps are the training of models, measuring their performance, and then trying to tweak the hyperparameters to obtain an emulator with the highest accuracy. This will be further explored in the next segments.

3.4.3 Training models and a modification to *CosmoPower*

There are two possible approaches to the training. First, to train one model that will produce 8 multipoles. Second, training 8 models for each of the multipoles. The second option was chosen, as performing tests, we noticed higher accuracy for separate modeling. This is rational, as with a limited amount of training data and very different amplitudes for each of the multipole, neural networks can better predict them separately.

Further on, when training the models, we did not get a good accuracy for any of the models. It was discovered later that in the case of 3PCF, the *Cosmopower* package is not well suited for the training of the models. This is because the loss function is calculated with non-normalized data, therefore we obtained the absolute MSE. In the case of the power spectrum, it was not a big issue, as the amplitudes there were large, therefore, also the value of loss was larger and a better fit for adjusting. In the case of 3PCF, for most of the scales, the amplitude is nearly zero, thus the absolute MSE is small, and the loss is small too. This leads to a model that cannot find the absolute minimum for the loss by tuning the neural network parameters, due to the very small loss values initially.

In order to address this issue, we modified the *Cosmopower* package, to introduce the relative MSE, which can be achieved by normalizing the data. The specifics of the implementation can be found in Appendix A. This modification enables the normalization only for the loss calculation, while still producing the output of the real scale for the 3PCF.

Additionally to that, there can be different types of normalization, and we explored Gaussian and Min-Max differences. The Gaussian normalization has been already implemented in the *CosmoPower*, just not used for the loss calculation, which is of the form:

$$x' = \frac{x - \mu}{\sigma}, \tag{3.7}$$

where x' is the normalized data, x is data points, μ is the mean value of the data, but σ is the standard deviation. On the other hand, Min-Max is described by:

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}}, \tag{3.8}$$

where x_{min} and x_{max} are the minimal and maximal values of the data accordingly. This normalization had to be implemented additionally. Firstly, the modified *CosmoPower* was compared to the original one, which can be seen in Fig. 3.19: here it can be clearly seen that the internal normalization for the loss significantly improves the accuracy of the model for all scales. Not only that, but the accuracy stays relatively flat. Instead, the original version gradually increases.

This correction has not been applied to the power spectrum emulator in the section 3.3, as there the accuracy was already sub-percent. Nevertheless, this might be one of the reasons why there we saw a steep decrease in accuracy in the small scales.

The other option is the Min-Max normalization, which was implemented in the *CosmoPower* code. The comparison with the Gaussian can be seen in Fig. 3.20. It can be seen that the Gaussian normalization is the better one in this case, providing higher accuracy over all scales. For all the following models, this will be the implemented normalization.

With the normalization being dealt with, we can proceed to the training of the model. Through experimentation, a good reference model was found with a good accuracy. The hyperparameters of this model are shown in Table 3.9. The key difference for the matter power spectrum is the reduction in the number of layers and neurons, as well the optimizer, which in this case is*Adam*, that proved to be better.

To find the best model, these hyperparameters were tweaked, and the results for the *000* multipole configuration can be seen in Fig. 3.21. All of these models show

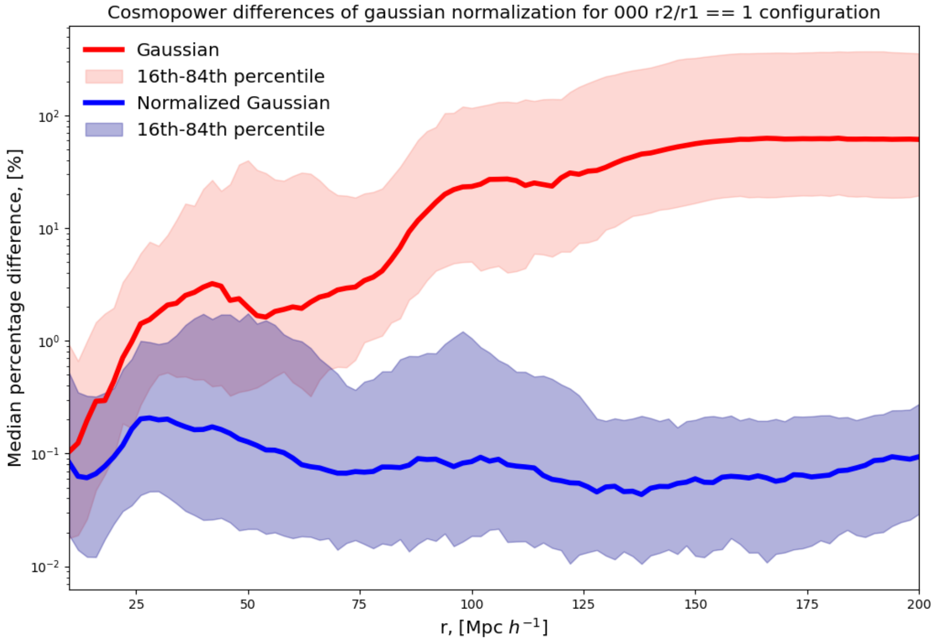


Figure 3.19. Comparison between modified *Cosmopower* with Gaussian normalization and the original *CosmoPower*. *Normalized Gaussian* is the modified version, while *Gaussian* is the original. The comparison is done with median percentage difference, and in particular for the *000* multipole, equilateral triangle configuration.

good accuracy, which is well within the sub-percent accuracy. Nonetheless, we can see a few good candidates that offer better performance. The ones with 128 neurons per layer and batch size of 128 are mildly better, but the best one is the model where the percentage for the validation set was reduced to 10%. Naturally, you would think that combining these would yield better results. Unfortunately, the combination of the best hyperparameters did not produce better results. Therefore, the best model that will be used for other multipoles, will be the one in Table 3.9, but with the validation set to 10%.

The reason behind a simpler architecture of the neural network in comparison to the power spectrum might have different reasons. In essence, the 3PCF is a more complicated function, as well as it has more triangle configurations than the wavevectors for $P(k)$. This would indicate that a more complex model is needed. But this is not the dominant factor — the size of the dataset is. As the dataset is around 50 times smaller, it is not sufficient to train it with an architecture with a large number of neurons and layers. It would only lead to overfitting, which would yield worse accuracy for the model.

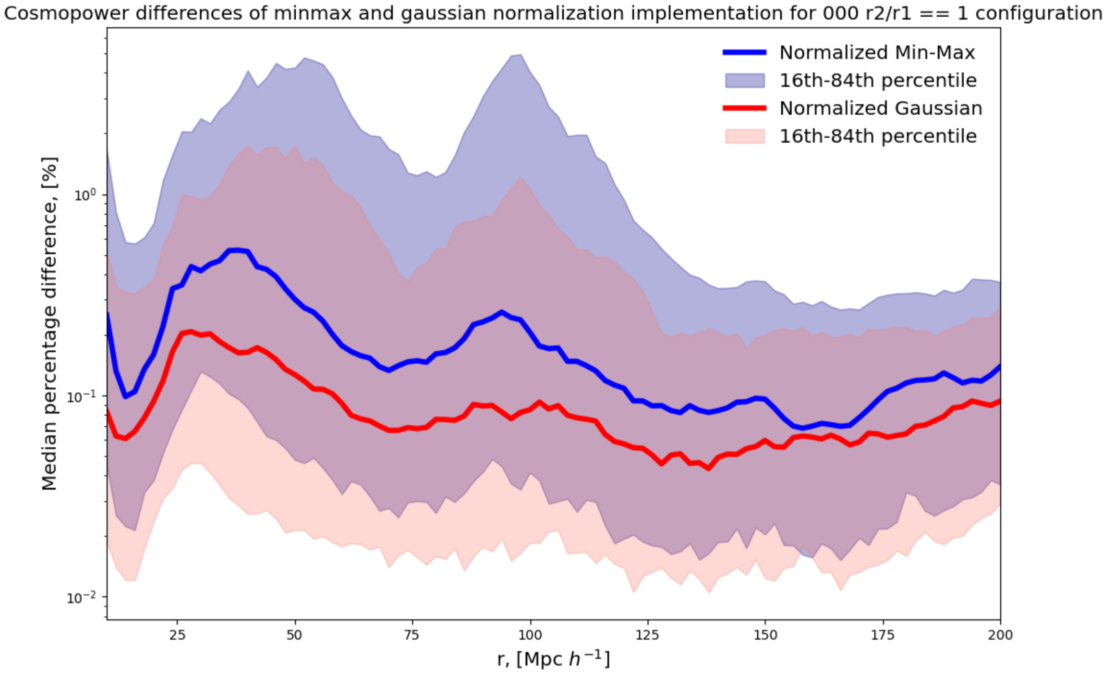


Figure 3.20. Comparison between Modified *Cosmopower* with Gaussian and Min-Max normalization; Comparison is done with the median percentage difference for the *000* multipole, equilateral triangle configuration.

hyperparameter	Value
Number of layers	3
Number of neurons	64
Batch size	256
Learning rate	1e-1 to 1e-7
Patience value	30
Max epochs	2000
Validation split	20%
Optimizer	Adam

Table 3.9. The reference hyperparameters for the matter 3PCF

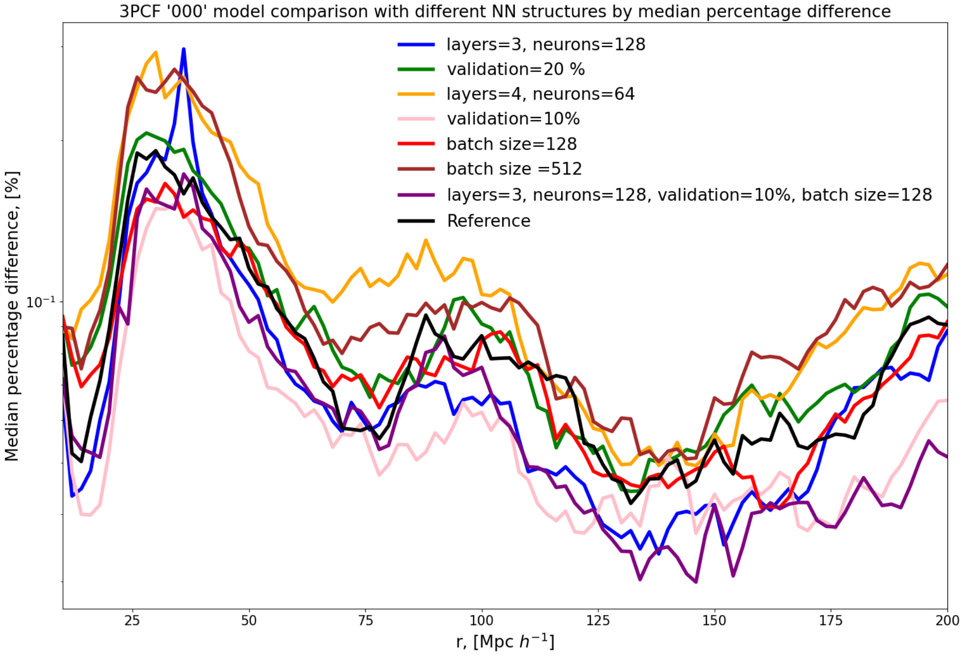


Figure 3.21. Comparison of different emulators for 3PCF; Exploration of different hyperparameter space around the reference model, shown in Table 3.9. The median percentage difference is reported for the *000* multipole.

The acquired architecture for the neural network was applied to all the multipoles, and the results of that can be seen in the Fig. 3.22. It can be concluded that all the multipoles have sub-percent accuracy, which is the threshold that we wanted to achieve. The worst accuracy seems to be for the *011* multipole, which might have required a better architecture. Having said that, finding the perfect architecture is not the point of the thesis, but rather achieving a certain accuracy of the models, which in this case is sub-percent.

To finalize, I present an example for the emulated matter 3PCF in all multipole configurations in Fig. 3.23. To emphasize, this is the first emulator for matter 3PCF in redshift space for isotropic and anisotropic multipoles. Here also the no-wiggle 3PCF is added, which, even though is not yet emulated, can help to distinguish the BAO influence on the 3PCF. More importantly, this example portrays the obtained results of the emulator accuracy in Fig. 3.22 — all the multipoles are predicted very well, indistinguishable from the human eye. With these eight emulators, many doors open for scientific analysis. Two of them are further analyzed in the next chapters.

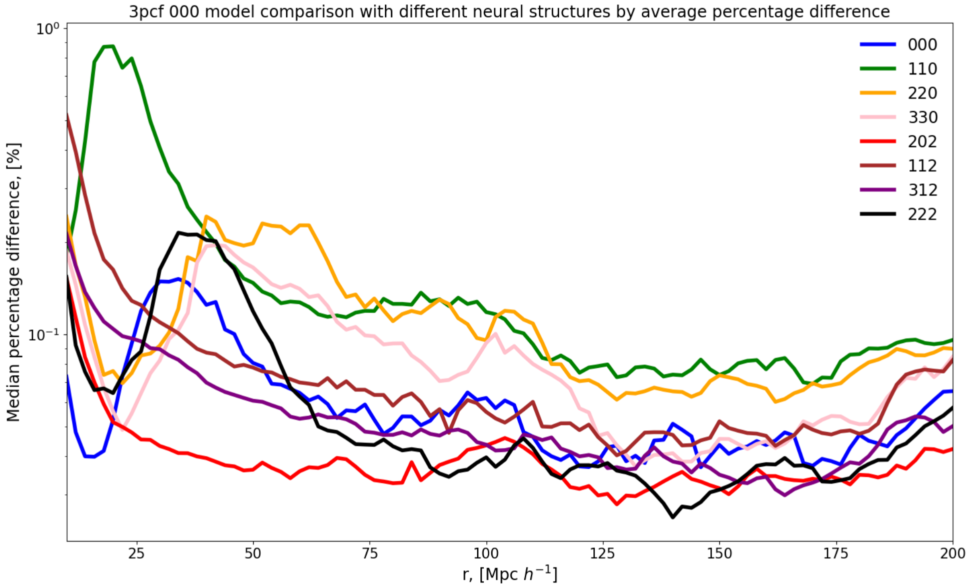


Figure 3.22. Emulator accuracy for each 3PCF multipole; Median percentage difference for the developed emulators, compared to all the multipoles. Labels are denoted as integers from 0 to 7, which are corresponded to the 8 possible multipole configurations.

3.4.4 Computational benefits from the emulators

Having built four emulators — three for the power spectrum and one for the 3PCF — we can now summarize the computational gain from it. In the case of the power spectrum, we will compare the time needed to compute it with the Python library *CosmoBolognaLib*, but for the 3PCF we will compare it with the Python package *Mod3l*. The emulator time is reported, by running it on CPU i7-8650U 4 cores, 1.9 GHz (only two of the cores were used). Instead, P(k) and 3PCF were computed on CPU Intel Xeon 8358 32 cores, 2,6 GHz. The emulator could not be tested on the same CPU due to the limited availability of the available computational resources attributed to the generation of training data. Thus, the comparison between these values is only indicative. The comparison can be seen in Table 3.10.

In the case of the power spectrum emulators, we can see that there is not a large gain from the emulators, as it is around 10 times faster. The computational time to compute the power spectrum with *CosmoBolognaLib* is small for starters, as it takes only a little longer than a second. Here it can also be seen that the full $P(k)$ takes the least amount of time, because the other functions have to compute the full $P(k)$ as well.

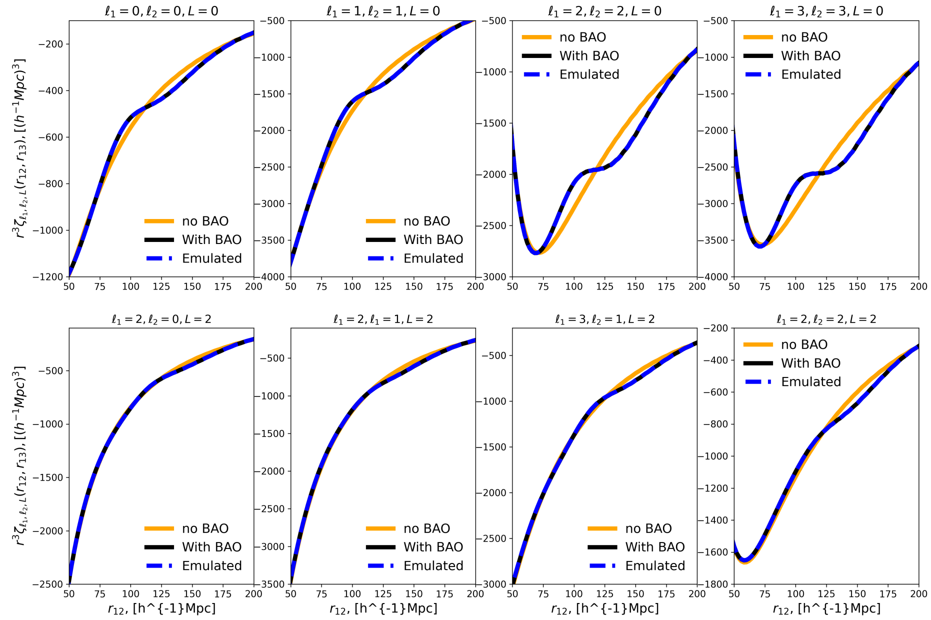


Figure 3.23. 3PCF emulator example; The matter 3PCF and no wiggle matter 3PCF acquired from *Mod3l*, compared with the emulated matter 3PCF.

Function	**Model CPU time (sec)**	**Emulator CPU time (sec)**	**Absolute difference (sec)**	**Relative difference**
Full $P(k)$	1.6	0.106	1.494	14.09
No wiggle $P(k)$	1.756	0.166	1.59	9.58
Only wiggle $P(k)$	1.64	0.14	1.5	10.71
3PCF	86400	0.0048	86399.9976	1.80×10^7

Table 3.10. Comparison of computation times and differences for various functions.

And the no wiggle is longer due to the larger number of wavenumbers k, as there we used 700 instead of 500. The times for the emulator are different due to the different neural network structures, and in these examples, the order is more important than comparing the exact time, as on these time scales it can even differ from the load on the CPU apart from the computations.

In the case of the 3PCF, the improvement is a lot more impressive. Only for a single computation with the *Mod3l*, it takes a full day of computation on a single CPU. Again, it depends on the type of CPU, so this has to be taken only as indicative time. The emulator is even faster than for the power spectrum, due to the simpler neural structure. Combining the long computation time with the short time for the emulator, it is more than 10 million times faster. This is the reason why an emulator is so beneficial in the case of cosmological analysis with 3PCF. Without losing much accuracy, now it does not require the usage of supercomputers, but a single laptop can easily suffice.

4

Cosmological information stored in 3PCF at BAO scales

Baryonic acoustic oscillations have become a standard ruler in the analysis of large-scale structures. They have been observed both in the power spectrum and two-point correlation function, and they have been used to better constrain the cosmological parameters, as the amplitude and the location of BAO hold rich cosmological information. In the 2PCF the BAO appears as a bump at around 100 Mpch^{-1}, but the shape and location in the 3PCF is not as straightforward.

One way to observe the BAO in the TripoSH decomposition of 3PCF is shown in Fig. 3.23, where it has been done by implying a constraint on the triangle sides. It can be observed that the BAO appears in every multipole, and in each one in a different shape. Thus, we can grasp the added complexity by going from 2PCF to 3PCF, which is yet to be understood and described.

For the existing and upcoming space missions analyzing the LSS clustering, it would be beneficial to understand more about the appearance of BAO and the advantage of constraining BAO properties from higher-order statistics, such as the 3-point correlation function. Which multipole holds the most information? Do anisotropic multipoles have a significant contribution to isotropic components? Which triangle configurations are the most informative? Do SNR and BAO features differ for different multipoles? There are various questions, which will be analyzed further in the following sections.

4.1 Methodology to unveil the cosmological information with BAO

As mentioned before, 3PCF is very rich in information since there are a multitude of triangle configurations. If in two-point statistics there is only one peak due to BAO at 100 h^{-1}Mpc, then in 3PCF it can appear in various triangle configurations. It is important to understand which of these triangle configurations are the most informative.

The simplest way, which is also similar to the 2PCF analysis, is to apply a constraint to one of the triangle sides, which allows reporting the 3PCF only with respect to one of the triangle sides. For example, it is interesting to fix one of the triangle sides to a certain distance, and then compare the full 3PCF to the no-wiggle one, which has been done in Fig. 4.1. It can be seen that in all the multipoles the BAO influence can be clearly distinguished. Furthermore, different conditions have different effects on the amplitude and the shape of the 3PCF. A lot of information can be gained by just investigating this plot. For example, it seems that the most informative is the condition, where one of the triangle sides is fixed to $r_{12} = 30$ Mpc h^{-1}. It already seems that the amplitude of the function is inversely proportional to the scale. By fixing one of the sides to a small scale, most of the triangles become squeezed. Therefore, here we have an indication that squeezed triangles might hold most of the information. This is due to

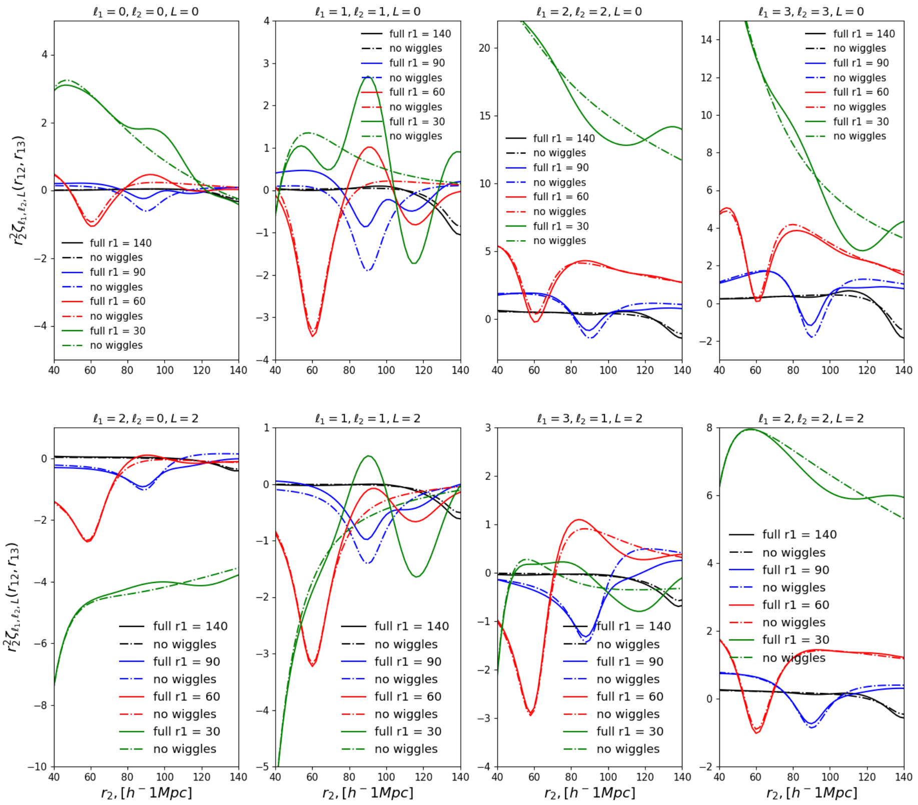


Figure 4.1. BAO appearance in 3PCF multipoles; The full 3PCF and no wiggle 3PCF are expanded into 8 multipoles, where one of the triangle sides has been fixed to either: 140, 90, 60 or 30 Mpc h^{-1}. The scale is zoomed in around the influence of BAO, which is from 40 to 140 Mpc h^{-1}.

the fact, that in the non-linear regime, when the density contrasts exceed the value of 1, the overdensities do not evolve independently anymore, thus increasing the clustering [10].

A similar analysis also can be done by implying a condition between the ratio of the two sides of the triangle, for example, $r_{12} = r_{13}$ or $r_{12} = 2r_{13}$. However, manually taking specific conditions, although useful for the interpretation of results, is laborious, time-consuming, as well as loses part of the analysis by not taking every possible triangle configuration into account. Therefore, this will be left only for the result interpretation, but the actual analysis will be done by constructing 2D matrices.

We will start by analyzing the 3PCF itself, by representing it on a 2D grid, where each point will be a triangle configuration. With the addition of the covariance matrix, we can construct Signal-to-noise ratio (SNR) figures, which will help us find the highest signal configurations. Furthermore, we will use a similar method to analyze the BAO. This can be done by either subtracting the full 3PCF with the no wiggle 3PCF or by calculating the only wiggle 3PCF. These figures will be addressed as the BAO detectability figures, as they should depict only the triangles where the BAO signatures are present. By performing this analysis, we will better understand the intricacies of 3PCF in different multipoles, as well as where to search for the BAO in the 3PCF.

4.2 Signal to noise ratio matrix

The signal-to-noise ratio is an instructive way to analyze the models, as it not only provides the theoretical predictions of the most informative triangle configurations, but also connects this knowledge with the observational properties, such as the expected detectability. In general, it can be expressed as:

$$SNR_i = \frac{\zeta_i}{\sigma_i}, \tag{4.1}$$

where we denote each triangle configuration with i, σ_i is the variance or the error for the triangle, while ζ_i is the 3PCF value for the specific triangle. To use this formula, we need the signal, which is the 3PCF, and the "noise", therefore the errors for each of the triangles. That can be acquired from the covariance matrix, which is more explained in the section 2.10.

We have visualized the 3PCF by implying different conditions, but, as discussed before, the most complete way is to represent all triangle configurations in all multipoles in sets of 2D matrices, as shown in the Fig. 4.2. This grid is formed in such a manner, that both triangle sides are on the grid axis, but the 3PCF value is indicated by color.

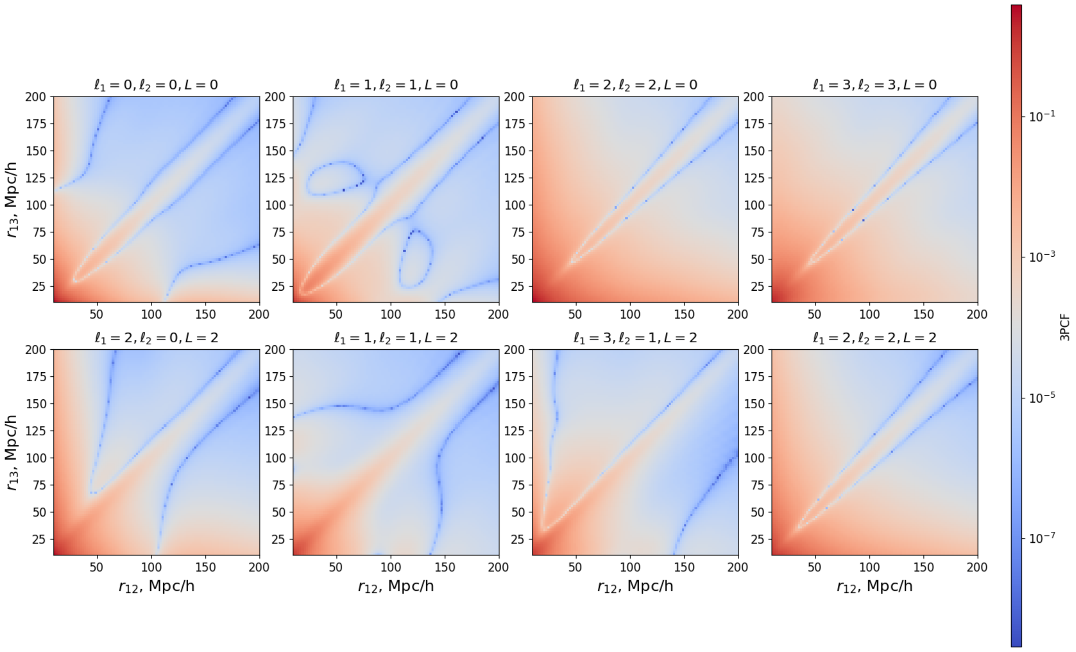


Figure 4.2. 3PCF for all triangle configurations in 8 multipoles; The value of the 3PCF is indicated by the color bar, while on the axis are the two lengths for triangle sides. This is calculated with the cosmological parameters according to *Plack18* results [3].

Here we can see some interesting features, for example, the high signal in the corner of each multiple. In this region, all the triangle sides are short, even the third one, as, if we take the example of $r_{12} = r_{13} = 10\ Mpc/h$, then the maximal side for the third is $r_{23} = 20\ Mpc/h$. It is clear that these structures do not contain the BAO scale, but they do have an increase in clustering due to the non-linear effects of the perturbation theory [83]. This is expected, as on small scales structures are interacting with each other, therefore clustering together.

Furthermore, in most of the multiples, there is also a large amplitude for the squeezed triangle configurations. These are triangles, where one of the sides is considerably longer than the other. In the plot, it can be roughly estimated to be the whole row or column up to $r = 50\ Mpc/h$. It can be also explained by the fact that one of the sides lies in the non-linear regime, which increases the overall probability of finding matter at these scales.

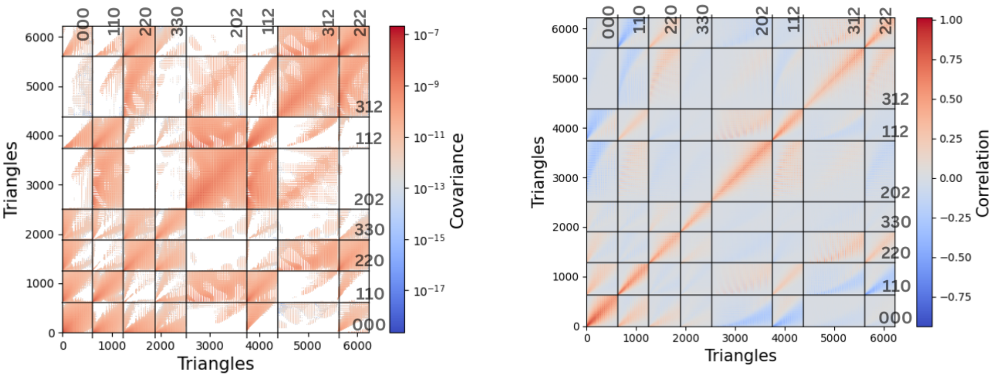
The other interesting feature in all the multipoles is the isosceles triangle configuration $r_{12} = r_{13}$. This is the case, because if two sides are equal, and we are rotating them, then at one point they align and make the third side 0 or close to it. It is a highly unlikely configuration, where the third side is nearly zero or zero, which greatly impacts the whole 3PCF, thus it produces a trough in the 3PCF. The rest of the function is nearly zero, therefore the absolute amplitude is significantly higher than in other configurations. The BAO feature on this configuration is hard to spot. Again, as the 3PCF is mostly negative for isosceles triangles, the bump of the BAO just makes it less negative, thus reducing the absolute amplitude. This, in turn, would result in a dimmer region around the BAO scales in Fig. 4.2. This effect can be better seen in Fig. 4.4.

Then there are the zero crossings, which can be spotted by the bright blue lines. The ones around the isosceles are due to the trough discussed before. Around this trough the 3PCF is slightly elevated above the zero line, thus it produces two zero crossings around the isosceles line. Then there are other features, that mostly arise from non-linear effects. But each multipole has its own pattern, and in the asymmetric case, for example, of multipole $\ell_1 = 2, \ell_2 = 0, L = 2$ we can see also different patterns on larger scales, where $r_{12} \approx 80\ Mpc/h$. This is due to the nature of expanding each of the triangle sides in a different order.

It is important to stress that, even though in each multipole we can spot similar behavior for either squeezed or isosceles triangles, there are many differences. Each of the multipoles portrays the information from matter clustering differently. The largest overall differences occur between the isotropic — the first row — and anisotropic — the second row — multipoles. The anisotropic are smaller in amplitudes, which is expected, as they are induced only by the RSD and AP effects. That is why they also exhibit slightly different patterns.

The other thing to interpret is the noise, which can be obtained from the covariance matrix. In the case of TripoSH basis with 8 multipoles, where $L = 0, 2$, we visualize the covariance and correlation matrix in Fig. 4.3. It is important to note that this covariance matrix has been made with a different grid, where $r_{min} = 10$, $r_{max} = 146$, and $\Delta r = 4$. It is a lot smaller than the previously used one due to the heavy computational time required to construct a covariance matrix. Furthermore, this is a Gaussian covariance matrix, even though the smaller scales do exhibit non-Gaussian behavior. Thus, the small-scale variance is underestimated [75], which makes it unreliable to use the covariance on small scales. Thus, a conservative approach will be applied to use only the scales that are bigger than 20 h^{-1}Mpc, whenever the covariance matrix must be used.

Back to Fig. 4.3: these figures contain crucial information about the system. Firstly, it is easier to interpret the correlation matrix, as it shows the correlation between different triangles between different multipoles. It can be seen on the right side of Fig.



(a) **(b)**

Figure 4.3. Covariance and correlation matrix of redshift space 3PCF multipoles; Each of the multipoles is expressed as: $\ell_1 = i\ell_2 = jL = ijk$ **(a)** Gaussian covariance matrix for the TripoSH basis with 8 multipole expansions, obtained from theoretical calculation described in section 2.10. In this example, each symmetric multipole has 630 independent triangles, but the two asymmetric multipoles *202* and *312* have 1225 triangles. **(b)** The corresponding correlation matrix.

4.3 that the diagonal has the values of unity, but the rest of the matrix spans from -1 to 1, depicting either the strength of anti-correlation or correlation between different triangles in different multipoles. For example, it is interesting to see that the multipole *000* is positively correlated to the isotropic multipoles, therefore *110*, *220* and *330*, but it is anti-correlated to the anisotropic multipoles, hence *112*, *202*, *312* and *222*. Then there is the *330* multipole that is weakly correlated to any of the other multipoles, or *220*, which is mostly positively correlated to other multipoles. This does not explicitly imply that the strong correlation means the degeneracy of signal between the multipoles. It is rather indicative of similar patterns in the 3PCF for positive correlation between triangles, and opposite behavior for anti-correlation.

Secondly, the covariance matrix shown on the left side of Fig. 4.3 allows obtaining the errors for each of the triangles, which is shown in Fig. 4.4. It also serves as a visualization of the previous discussion of the appearing trough at $r_{12} = r_{13}$. The BAO features are clearly distinguishable from the no-wiggle 3PCF and the full 3PCF in the isotropic multipoles. However, in the anisotropic case, the no-wiggle 3PCF remains within the error margins of the full 3PCF across all scales, making it impossible to differentiate the BAO signal.

With the help of the covariance matrix, we can extract the errors of each triangle configuration from its diagonal. With this in hand we can now slightly redefine the

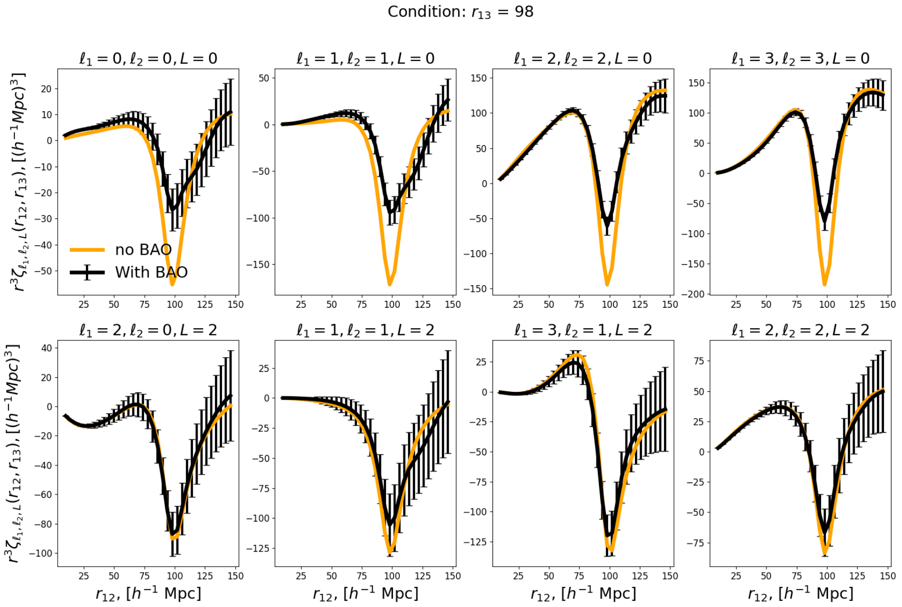


Figure 4.4. Example for errors in 3PCF; 3PCF is expanded in 8 multipoles, and the no wiggle 3PCF is included, in order to visualize the effect of the BAO better. Errors are obtained from the diagonal of the covariance matrix. The cosmology is *Planck18*.

equation 4.1 and use the general equation:

$$SNR_{ij}^{\text{full}} = \sqrt{\zeta_i Cov_{ij}^{-1} \zeta_j}. \tag{4.2}$$

We can now apply it for our 3PCF, but also taking into account that covariance is not accurate at scales below $r = 20\ Mpc/h$, therefore these scales are excluded from the SNR. The SNR of 3PCF can be seen in the Fig. 4.5. It can be seen that the overall structure has not changed much from Fig. 4.2, therefore the physical interpretation of the multiple regions of isosceles, squeezed and other triangle configurations stay the same. Nevertheless, the importance lies in the overall SNR of the multipoles. It can be clearly seen that the multipoles $\ell_1 = 2, \ell_2 = 2, L = 0$ and $\ell_1 = 3, \ell_2 = 3, L = 0$ has the highest SNR, but also the anisotropic multipole $\ell_1 = 2, \ell_2 = 2, L = 2$ has a good overall SNR. The summary of the average signal can be seen in Table 4.1.

It must be stressed that the span of the SNR is wide, for example, apart from the overall averaged values of each multipole, the small scale triangles in the lower left corner of the Fig. 4.5 have the average SNR of more than 500 for the $\ell_1 = 2, \ell_2 = 2, L = 0$ and $\ell_1 = 3, \ell_2 = 3, L = 0$ multipoles. The isosceles has a smaller average SNR of around 100 for the same multipoles. On the contrary, the signal in the top right corner has the

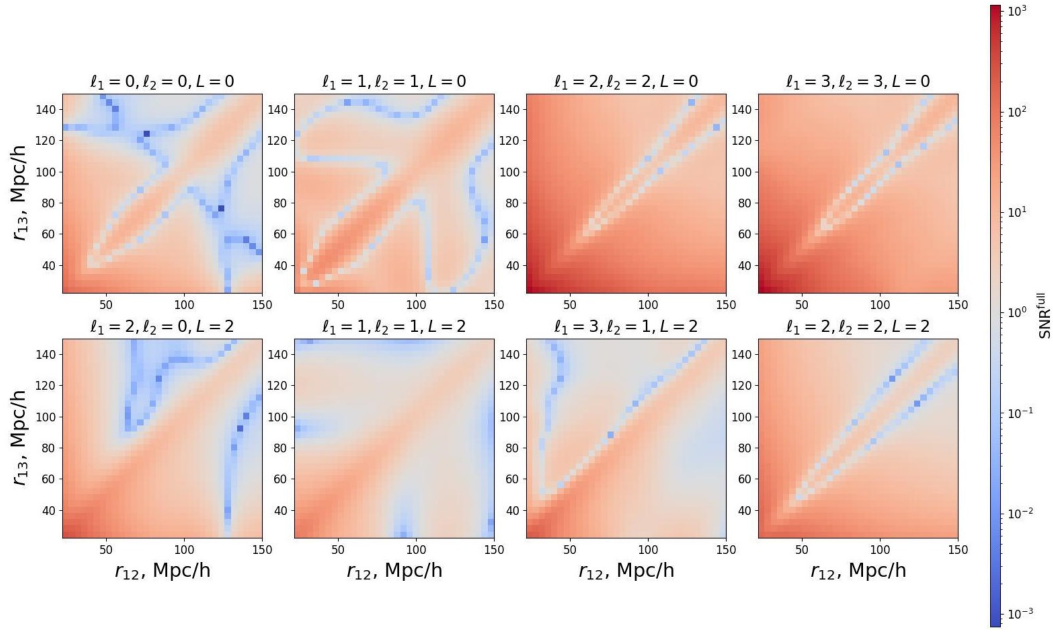


Figure 4.5. SNR matrix for 3PCF; The signal is the full 3PCF with the Planck cosmology, the error is from the diagonal of the covariance matrix.

Full 3PCF Average SNR

	$\ell_1 = 0,\, \ell_2 = 0$	$\ell_1 = 1,\, \ell_2 = 1$	$\ell_1 = 2,\, \ell_2 = 2$	$\ell_1 = 3,\, \ell_2 = 3$
$L = 0$	6.16	5.79	58.84	48.90
	$\ell_1 = 2,\, \ell_2 = 0$	$\ell_1 = 1,\, \ell_2 = 1$	$\ell_1 = 3,\, \ell_2 = 1$	$\ell_1 = 2,\, \ell_2 = 2$
$L = 2$	9.40	4.10	6.29	14.56

Table 4.1. Summary of the average full SNR over each of the multipoles.

average value of SNR below 10 for the same multipoles. The difference is more than 50 times compared to the small triangle configurations, which is very significant. Even though the small triangle configurations can still be affected by the Gaussian covariance matrix, the effect must be minor, due to the exclusion of the very small scales.

As discussed in the introduction and further chapters, BAO holds most of the information in these N-point correlation functions. In these figures, the BAO plays a minor role, and even in the SNR plots, it is not clearly distinguishable. Therefore, in the next section, we switch to a different method to analyze this feature separately.

4.3 BAO detectability

The BAO signal cannot be accurately distinguished from the SNR of matter 3PCF. To find the best triangle configurations for the appearance of BAO, we define a new metric: *BAO detectability.* It is defined by simply exchanging the 3PCF in the equation 4.2 with the subtraction between the full 3PCF and no wiggle 3PCF:

$$(\text{BAO detectability})_{ij} = \sqrt{\left(\zeta_i^{full)} - \zeta_i^{(NW)}\right) Cov_{ij}^{-1} \left(\zeta_j^{(full)} - \zeta_j^{(NW)}\right)}. \tag{4.3}$$

This requires the calculation of $\zeta^{(NW)}$, which at first glance might seem like a difficult task.

By observing the equation 2.51, we see that BAO comes into the equation only from the wiggle power spectrum $P_W(k)$. To remind, it is defined as $P_W(k) = P_{full}(k) - P_{NW}(k)$. Thus, if in the calculation of the 3PCF, we put as input P_{NW} instead of P_{full}, we get that $P_W(k) = 0$. This leaves us only with terms that include P_{NW} in equation 2.51. With this trick, we can easily calculate the no-wiggle 3PCF. By doing so, subtraction allows obtaining only the features due to BAO.

The resulting BAO detectability and the examples of the highest detectability regions are represented in Fig. 4.6. To better understand where the highest signal is, we have summarized the average BAO detectability signal for each multiple in Table 4.2.

The Fig. 4.6 and Table 4.2 are highly informative, and we can draw several conclusions from them. First and foremost, it seems similar to the SNR case: the best multipoles for BAO detectability are the isotropic ones, and the anisotropic multipoles have the signal several times smaller. The differences lie in the multipoles themselves, with the average values being similar across all isotropic multipoles except for the *000*, which has a lower value. Similarly, the anisotropic multipoles also show consistent values, except for the *202*, which stands out with a higher value. The BAO features appear in different regions for each of the multipole, therefore in those regions, the values are significantly higher than the overall average.

To expand on this, we see high detectability from the isosceles triangles for the isotropic and anisotropic multipoles. Also, on the isosceles line, the detectability is not uniform. It can be noticed that the strongest detectability is around the BAO scale - 100 h^{-1}Mpc, which is expected, as at least two of the triangle sides lie in the BAO range. This means that the BAO feature is automatically present in two thirds of the triangle, which gives significant differences with the no-wiggle 3PCF. From the example of the isosceles 3PCF in Fig. 4.6, it can be seen why it is not strictly localized around the BAO scale. It is due to the fact there are at least two crossings of both functions,

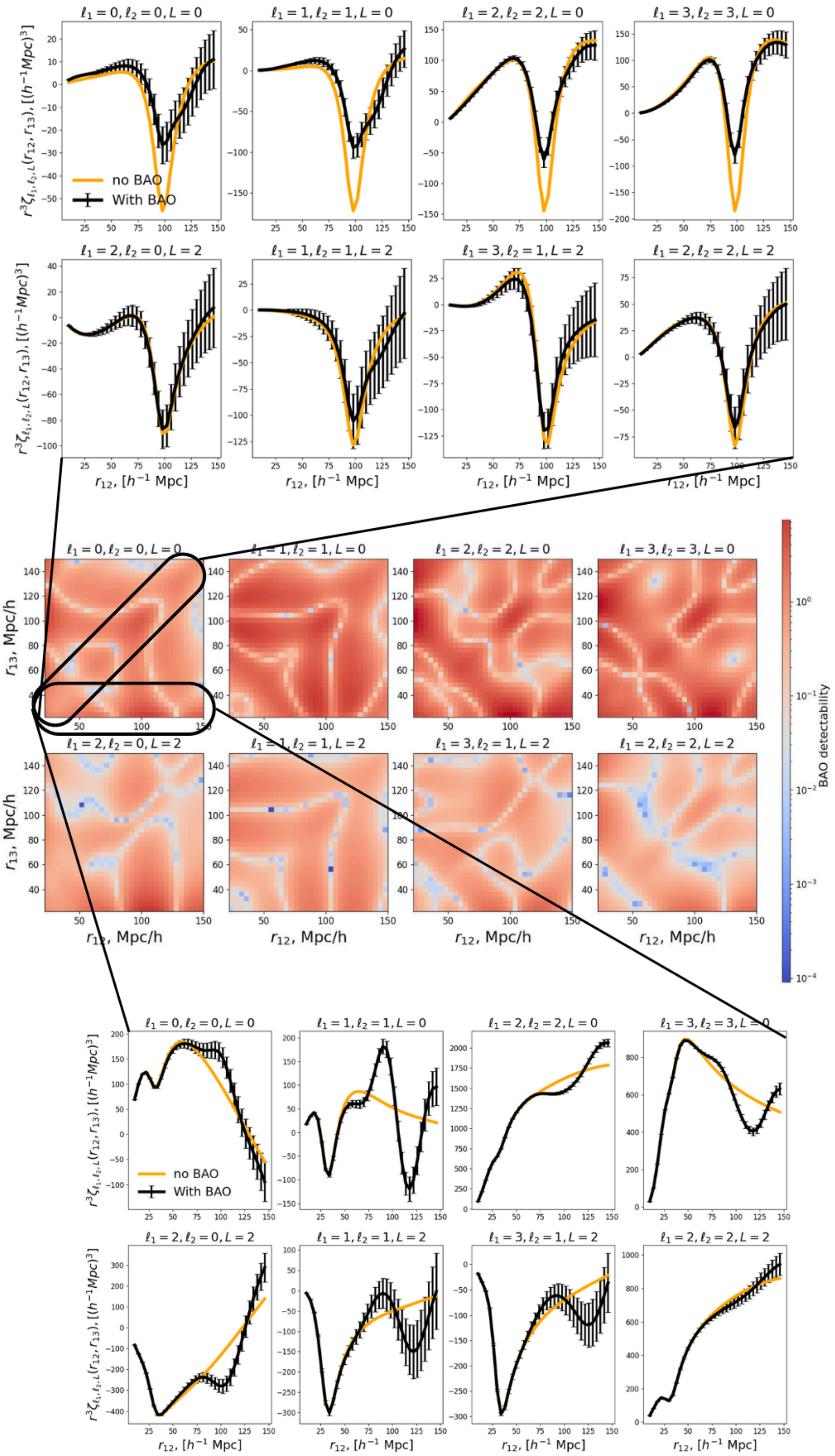


Figure 4.6. BAO detectability with 1D examples; Top: 3PCF with error bars and no wiggle 3PCF represented with the condition $r_{12} = r_{13}$. Middle: BAO detectability for each of the multipoles. Bottom: 3PCF with error bars and no wiggle 3PCF represented with the condition $r_{12} = 30\ Mpc$.

Average BAO detectability

	$\ell_1 = 0,\, \ell_2 = 0$	$\ell_1 = 1,\, \ell_2 = 1$	$\ell_1 = 2,\, \ell_2 = 2$	$\ell_1 = 3,\, \ell_2 = 3$
$L = 0$	0.94	2.31	2.04	2.14
	$\ell_1 = 2,\, \ell_2 = 0$	$\ell_1 = 1,\, \ell_2 = 1$	$\ell_1 = 3,\, \ell_2 = 1$	$\ell_1 = 2,\, \ell_2 = 2$
$L = 2$	0.56	0.58	0.43	0.36

Isosceles region

	$\ell_1 = 0,\, \ell_2 = 0$	$\ell_1 = 1,\, \ell_2 = 1$	$\ell_1 = 2,\, \ell_2 = 2$	$\ell_1 = 3,\, \ell_2 = 3$
$L = 0$	1.15	1.26	4.89	5.92
	$\ell_1 = 2,\, \ell_2 = 0$	$\ell_1 = 1,\, \ell_2 = 1$	$\ell_1 = 3,\, \ell_2 = 1$	$\ell_1 = 2,\, \ell_2 = 2$
$L = 2$	0.23	0.31	0.74	0.84

Squeezed region ($r_{13} \leq 34$ h^{-1}Mpc)

	$\ell_1 = 0,\, \ell_2 = 0$	$\ell_1 = 1,\, \ell_2 = 1$	$\ell_1 = 2,\, \ell_2 = 2$	$\ell_1 = 3,\, \ell_2 = 3$
$L = 0$	1.51	2.49	4.37	4.20
	$\ell_1 = 2,\, \ell_2 = 0$	$\ell_1 = 1,\, \ell_2 = 1$	$\ell_1 = 3,\, \ell_2 = 1$	$\ell_1 = 2,\, \ell_2 = 2$
$L = 2$	1.99	0.73	0.88	0.75

Table 4.2. Summary of the BAO detectability. Top table: Average BAO detectability over all triangle configurations among each multipole. Middle table: averaged isosceles region, ranging from $r_{12} = r_{13} = (90, 102)$ h^{-1}Mpc. Bottom table: averaged squeezed triangle region, where $r_{13} \leq 34$ h^{-1}Mpc.

which gives the apparent oscillating behavior. If the analysis was extended to higher r_{max}, there would be at least another crossing for the most multipoles. This means that the BAO features give an excess for the matter 3PCF at around the BAO scale, but departing from it, there is a reduced probability with respect to the no wiggle 3PCF. This is reasonable, as matter from higher or lower scales cluster at the BAO scale of around 100 h^{-1}Mpc.

However, the strongest signal for both isotropic and anisotropic cases is found in the squeezed triangle configurations, which occur when at least one of the sides has a small scale, such as 20 or 30 h^{-1}Mpc. The length of the other side, as seen in Fig. 4.6, varies for each multipole but is mostly around 80, 100, or 120 h^{-1}Mpc.

The physical explanation is that if the longer side is, for example, 100 h^{-1}Mpc, then the shorter the other side, the more triangle configurations will be approximately 100 h^{-1}Mpc. This is also true for other values of the long triangle side and is also the reason why the high BAO detectability does not extend far out from the squeezed triangle configurations, as with longer other triangle sides, fewer combinations yield the BAO scale. The exception, of course, is when one of the sides is around 100 h^{-1}Mpc. Then the detectability is extended also to longer scales for the other triangle side. The apparent gaps among the highest detectability regions in the squeezed triangle configuration regime are due to the crossing of both functions.

So far, all BAO detections have been consistent with physical expectations. However, there is a specific region in the lower left corner, where both triangle sides are small, that requires further attention. This region should have been absent from the BAO signal, as the combination of both sides, for example, $r_{12} = 30$ h^{-1}Mpc and $r_{13} = 30$ h^{-1}Mpc would give a maximum of only $r_{23} = 60$ h^{-1}Mpc. This is nowhere near the BAO scale, therefore there is no clear physical explanation for the high detectability. From further investigation, the difference between the full 3PCF and no-wiggle 3PCF is minimal, less than unity. This difference arises from the numerical effects, due to the finite grid size. But at the same time, at these scales, the error is also very small, around the same value or even smaller. Thus, the detectability is artificial and exists only due to numerical errors. Therefore, it is safe to ignore this part of the plot.

To further solidify the reasoning, the BAO detectability plot can also be obtained differently. It can be obtained from the only wiggle 3PCF, which is further explored in section 4.4.

To summarize, there are two regions of high BAO detectability — the isosceles and squeezed triangles. For the isosceles triangles, the peak of detectability is around the BAO scales, but the detectability is fairly significant for scales down to 60 h^{-1}Mpc and up to 150 h^{-1}Mpc. On the other hand, there are also multiple regions in the squeezed triangle configurations, where the strongest detectability is also around 100 h^{-1}Mpc. But, depending on the multipole, there is also prominent detectability, when one of the sides is around 60, 80, 120, and 140 h^{-1}Mpc.

4.4 Alternative method for BAO detectability

In the section 4.3, we adopted the most straightforward method to detect the BAO - subtract from the full 3PCF the no wiggle 3PCF, leaving only features that arise from BAO. The issue arises at small scales, where the accumulation of numerical errors in both functions produces a non-zero difference. The low error values make the small scales as a significant probe of BAO, which is not physical.

To deal with this issue, we can calculate the only wiggle 3PCF directly. This is useful, because due to the absence of additional operations, we reduce the accumulation of the numerical error. Furthermore, the amplitude there must be close to zero, thus the relative errors are a lot smaller than for the full or no wiggle 3PCF.

Firstly, we can examine why the no wiggle 3PCF calculation worked, by inserting both power spectrums as no wiggle. The template power spectrum can be decomposed

[71] as:

$$P^{(temp)}(\mathbf{k}) \longrightarrow [Z^{(1)}(\mathbf{k})]^2[\mathcal{D}^2(\mathbf{k})P_W(\mathbf{k}) + P_{NW}(\mathbf{k})], \tag{4.4}$$

where $[Z^{(1)}(\mathbf{k})]^2$ is the Kaiser factor, but the $\mathcal{D}(\mathbf{k})$ is the damping factor from IR flow and only wiggle power spectrum is defined as $P_W(\mathbf{k}) = P_{full}(\mathbf{k}) - P_{NW}(\mathbf{k})$. From this, we can use the equation 2.51. To remind, the input functions for the Python package *Mod3l* are $P_{full}(\mathbf{k})$ and $P_{NW}(\mathbf{k})$. Thus, if we put instead of full power spectrum the $P_{NW}(\mathbf{k})$, we are only left with the last term in the equation 2.51. This allows us to get the no-wiggle 3PCF by applying the 2D FFT-Log to the bispectrum.

For the only wiggle 3PCF, the situation is more complex. If we apply the same procedure by using $P_W(\mathbf{k})$ as the input for both power spectra, we end up with the last term in the bispectrum template equation. However, instead of $P_{NW}(\mathbf{k})$, we now have $P_W(\mathbf{k})$, which is not the term we want to obtain. What we need is the first term in the brackets: $\mathcal{D}(\mathbf{k}_1)\mathcal{D}(\mathbf{k}_2)\mathcal{D}(\mathbf{k}_{12})P_W(\mathbf{k}_1)P_W(\mathbf{k}_2)$. As a result, what we obtain is the only wiggle 3PCF, but without the damping factors from the IR flow.

However, for our purposes, only wiggle 3PCF without the damping factors is sufficient, as the primary goal is to test small-scale triangle configurations. The example for the generated only wiggle 3PCF in isosceles triangle configurations can be seen in Fig. 4.7. It can be seen that both of the functions are very different, and not only the amplitude differs. The number and position of peaks also differ greatly. This is due to the fact that in reality the two metrics, which are compared, are not the same, therefore there are also differences in the position of the peaks. The only wiggle 3PCF allows to isolate the contribution only by baryons, while the subtracted 3PCF also includes the non-linear interaction with cold dark matter, which is the major contributor to the differences between the two metrics.

Nevertheless, the small-scale amplitudes are nearly zero, which is exactly what was expected. Therefore, even though there are multiple inconsistencies, the purpose of the test has been achieved. The full BAO detectability can be seen in Fig. 4.8. It can be seen that, even though the details are quite different, the highest detectability regions are roughly the same. Furthermore, the bottom left corner is absent of BAO detection, which solidifies the reasoning that it is simply a numerical effect.

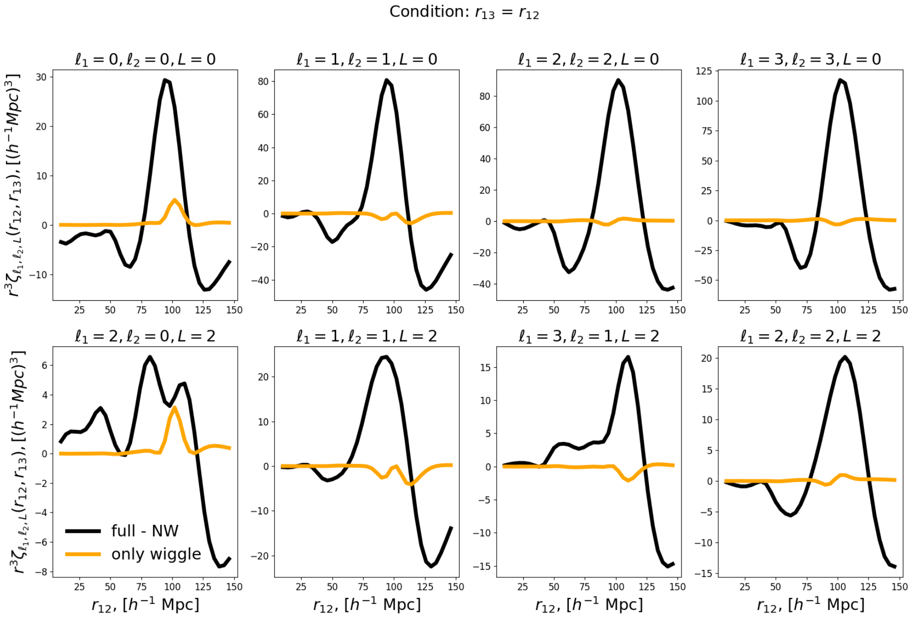


Figure 4.7. Comparison between only wiggle 3PCF and subtraction between full 3PCF and NW 3PCF. It is compared for only the isosceles triangle configurations.

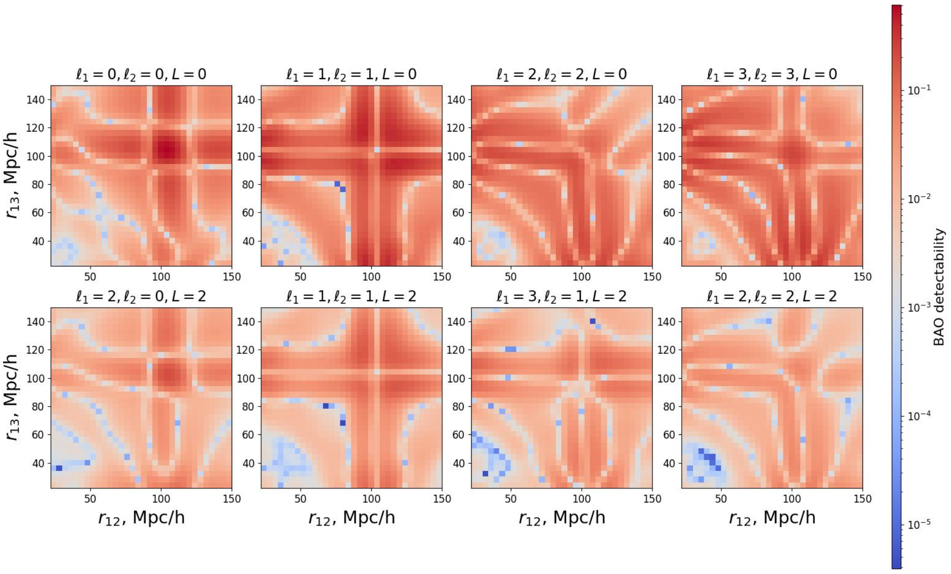


Figure 4.8. BAO detectability with only wiggle 3PCF.

5

Forecasting cosmological parameters with the anisotropic 3PCF

The final goal of this thesis is to forecast the cosmological parameters based on the representative volume of Stage IV surveys. Up until now, the constraining power of the isotropic component has been studied to some extent [66], however, the anisotropic 3PCF has been only developed recently [70]. Therefore, the constraining power of the anisotropic component has not been explored before, making it one of the main objectives of this section.

Furthermore, as we have discussed before, modeling the 3PCF is computationally expensive, thus the previously developed emulator for the anisotropic 3PCF is a crucial component in this Fisher analysis. Fisher forecasts require modeling the 3PCF with different cosmological parameters, which can quickly become computationally unfeasible. Instead, the emulator allows us to perform the forecasts in a matter of seconds.

Motivating from the previous section 4.2 of SNR and BAO detectability analysis, we will analyze how much the constraining power reduces, if we take only the highest SNR or BAO detectability triangle configurations for the forecasts.

Lastly, we will analyze separately the constraining power of the small scales. As discussed before, the squeezed triangles hold the most information in the multipoles. Furthermore, the errors for these scales are also smaller, due to the larger sampling size. The combination of that should give great constraining power, therefore it is important to understand the effect of small scales for the forecasts.

5.1 Fisher matrix formalism

Before analyzing observations or mock catalogs with newly developed theories, it is valuable to take a step back and understand how well we *expect* the developed methodology to constrain the cosmological parameters for a given experiment. This can be answered by calculating the Fisher matrix for our cosmological parameters, which, in the end, provides a forecast given a covariance. This is very useful, as it can give predictions of the performance of newly developed theories or signals, and indicate if it is useful to apply them or not.

5.1.1 Likelihood function

The basis for every statistical analysis is the likelihood function $\mathcal{L}(\theta)$. It represents the probability of observing data $\mathbf{d}$ given some theory, where the parameters for the theory are θ. Therefore, we can define it as:

$$\mathcal{L}(\theta) = P(\mathbf{d}|\theta). \tag{5.1}$$

The Likelihood function can take many forms, but the most popular is the Gaussian form [23], as in many cases in cosmology the data are distributed in Gaussian shape. This assumption simplifies the likelihood function and is often justified when dealing with large datasets due to the Central Limit Theorem. Thus, assuming two parameters — the mean ω and the standard deviation from the mean σ_ω, we can define it as:

$$\mathcal{L}\left(\{d_i\}_{i=1}^{m}|\omega,\sigma_\omega\right) = \frac{1}{(2\pi\sigma_\omega^2)^{m/2}} \exp\left\{-\frac{\sum_{i=1}^{m}(d_i-\omega)^2}{2\sigma_\omega^2}\right\}, \tag{5.2}$$

where m is the number of data points. In reality, we usually want to go from the opposite side — we want to refine our theory or the cosmological parameters, given some data. Thus, from the probability theory, we can express:

$$P(B,A) = P(B|A)P(A) = P(A|B)P(B). \tag{5.3}$$

Inserting this relation in the equation 5.1, we get the equation:

$$P(\omega,\sigma_\omega|\{d_i\}) = \frac{P(\{d_i\}|\omega,\sigma_\omega)P(\omega,\sigma_\omega)}{P(\{d_i\})}. \tag{5.4}$$

This is the pure form of the Bayes' theorem. It is important to notice that in the denominator we have the probability of data, which is independent of the parameters from the theory. Therefore, it can be regarded as the normalization factor for the Gaussian Likelihood function, as it does not change either the mean or the standard deviation of our probability. Secondly, the probability of our theory $P(\omega,\sigma_\omega)$ can be denoted as our prior knowledge of the parameters. Therefore, we can simplify the equation 5.4 as:

$$P_{posterior}(\omega,\sigma_\omega|\{d_i\}) \propto \mathcal{L}(\{d_i\}_{i=1}^{m}|\omega,\sigma_\omega)P_{prior}(\omega,\sigma_\omega). \tag{5.5}$$

This is one of the most important equations in cosmology, as it allows constraining the cosmological parameters, based on the information from the previous experiments. From this formalism, we can also develop a theoretical framework not only to constrain parameters from observations but also to forecast them from possible experiments.

5.1.2 Fisher matrix

We want to have a quantifiable way to estimate the uncertainties from a given experiment. For example, if we want to build a new telescope, like the Roman Space Telescope, we need to see if with the given observations we can achieve the needed precision for the constraints of cosmological parameters. To do so, the general formula is:

$$\mathcal{F}_{\alpha\beta} = \left\langle \frac{\delta^2 ln\mathcal{L}}{\delta\lambda_\alpha \delta\lambda_\beta} \right\rangle |_{\lambda_\gamma = \bar{\lambda}_\gamma}, \tag{5.6}$$

where $\mathcal{L}$ is the likelihood, but the λ_α and λ_β are the cosmological parameters. This equation represents how fast would the observational data change by the given cosmological parameters of a given theory. This is for an arbitrary amount of cosmological parameters, therefore it is better to take a step back and explore the case of a single cosmological parameter with a Gaussian Likelihood function. We will start with the definition of chi-squared [23], which is a measure between the observed and expected outcomes. The definition is as follows:

$$\chi^2(\{\lambda_\alpha\}) = \sum_r \frac{[\hat{T}(r) - T^{theory}(r, \{\lambda_\alpha\})]^2}{Var(\hat{T}(r))}, \tag{5.7}$$

where r is an arbitrary parameter, which the function $T(r)$ depends on. $\hat{T}(r)$ is the observed values of the function, $T^{theory}(\lambda_\alpha)$ is the expected values from theory, and the $Var(\hat{T}(r))$ is the expected uncertainty from the given experiment. The λ_α are the cosmological parameters in our experiment, and from now on we assume that it is only one parameter. The $\bar{\lambda}_\alpha$ is the fiducial value for the universe, and χ^2 is expected to be at the minimum there.

Now, coming back to the definition of the Fisher matrix, we can estimate how fast does the χ^2 value deviates from the fiducial value. If it is a rapid change, then the errors will be naturally small, whereas the opposite is true if χ^2 changes slowly. We can show it by expanding the χ^2 around the fiducial value, acquiring:

$$\chi^2(\lambda) = \chi^2(\bar{\lambda}) + \mathcal{F}(\lambda - \bar{\lambda})^2, \tag{5.8}$$

where λ is the single cosmological parameter and $\mathcal{F}$ is the Fisher matrix, which is the expression of:

$$\mathcal{F} = \frac{1}{2}\frac{\delta^2\chi^2}{\lambda^2}. \tag{5.9}$$

By inserting the equation 5.7 in the equation 5.9, we obtain the equation:

$$\mathcal{F} = \sum_r \frac{1}{\mathrm{Var}\left[\hat{T}(\ell)\right]}\left[\left(\frac{\partial T^{\mathrm{theory}}(r,\lambda)}{\partial\lambda}\right)^2 + \left(T^{\mathrm{theory}}(r,\lambda) - \hat{T}(r)\right)\frac{\partial^2 T^{\mathrm{theory}}(r,\lambda)}{\partial\lambda^2}\right]. \tag{5.10}$$

Due to the fact that we are working with the expected results, we can simplify the equation 5.10, since T^{theory} should be the same as the observed values $\hat{T}$. Thus, we are left only with the first term in the brackets, yielding:

$$F = \langle\mathcal{F}\rangle = \sum_r \frac{1}{\mathrm{Var}\left[\hat{T}(\ell)\right]}\left(\frac{\partial T^{\mathrm{theory}}(r,\lambda)}{\partial\lambda}\right)^2, \tag{5.11}$$

where $\langle\mathcal{F}\rangle$ is an ensemble of averages over multiple iterations of the experiment. Thus, F is the Fisher information. To use it for multiple cosmological parameters, we can generalize it to the form:

$$F_{\alpha\beta} = \sum_r \frac{1}{\mathrm{Var}\left[\hat{T}(r)\right]}\frac{\partial T^{\mathrm{theory}}(r,\{\bar{\lambda}_\gamma\})}{\partial\bar{\lambda}_\alpha}\frac{\partial T^{\mathrm{theory}}(r,\{\bar{\lambda}_\gamma\})}{\partial\bar{\lambda}_\beta}. \tag{5.12}$$

Now it is clear that the equation does not depend on the observational data, but only on the expected errors and theoretical model. Now, by performing the inverse of the Fisher matrix, we can obtain the uncertainties of our parameters. For example, the uncertainty of the parameter λ_1 is simply $\sqrt{(F^{-1})_{11}}$. Thus, we can simply state it as:

$$Cov_{\alpha\beta} = F^{-1}_{\alpha\beta}, \tag{5.13}$$

Where the $Cov_{\alpha\beta}$ is the covariance of our cosmological parameters.

5.2 Cosmological constraints from the matter 3PCF

To compute the Fisher matrix, we need the appropriate covariance matrix, where its calculation is explained in the section 2.10. It must be done on a TripoSH basis, which is appropriate for the considered analysis. In this work, we will consider the expected full *Euclid DR3* covered light-cone volume of $43(\mathrm{h}^{-1}\mathrm{Gpc})^3$. Same as in the section 4.2, the grid for the triangle sides is smaller due to the high computational costs. To remind, it is $r_{min} = 10$, $r_{max} = 146$ with $\Delta r = 4$. Hence, to use this covariance, we will have to re-scale also the computed 3PCF.

Now we will walk through the algorithm of obtaining the forecasts.

1. **Fiducial cosmology:** As explained before, we need to assume a fiducial cosmology, which is the assumed true value for the chosen cosmological parameters. In this case, as it was also done for emulators, we will choose the *Planck18* results [3].

2. **Emulation:** After choosing the fiducial values, we also need some offset from them, in order to compute the numerical derivatives for the equation 5.12. For that reason, we vary each of the three cosmological parameters Ω_m, h, and A_s by 10% around the fiducial value, obtaining six 3PCF with different cosmologies.

3. **Re-scaling:** As mentioned before, we have the constructed emulator for a different grid of triangles, compared to the computed Gaussian covariance. Therefore, we have to find the values of the 3PCF at the new triangle configurations. In our case we do not need to interpolate, as we have the same $R_{min} = 10Mpc\text{h}^{-1}$ and the Δr goes from 2 to 4, therefore we can pick every second element. We have constructed also a general code, where interpolation is included with the Python library *scipy.interpolate.interp2d.*

4. **Threshold:** In the forthcoming analysis, we want to explore how the constraining power changes due to different types of thresholds. Hence, we implement also the option to use the threshold based both on conditions by SNR or BAO detectability or by triangle sides. This allows us to see the importance of certain triangle configurations for forecasts of the parameters.

5. **Computing numerical derivatives of the triangles:** To compute the numerical derivatives, we have various options. To have second-order accuracy, we chose the central numerical derivative of the form:

$$\frac{\delta T^{\text{theory}}(r, \lambda_\alpha)}{\delta \lambda_\alpha} = \frac{T^{\text{theory}}(r, \lambda_\alpha + \Delta\lambda_\alpha) - T^{\text{theory}}(r, \lambda_\alpha - \Delta\lambda_\alpha)}{2\Delta\lambda_\alpha}, \tag{5.14}$$

   where $\Delta\lambda_\alpha$ is 10% of the fiducial value. This was chosen, for the deviation to be small enough to still have an accurate numerical derivative.

6. **Fisher matrix:** With the covariance and the derivatives in hand, it is possible to compute the Fisher matrix, using the equation 5.12. Each element is calculated separately and then combined to create a 3 × 3 matrix.

7. **Covariance matrix:** To estimate the uncertainties of the parameters, we have to get the covariance matrix. That is simply done by inverting the Fisher matrix, and thus the uncertainties are on the diagonal.

8. **Visualize:** To visualize, we take two of the parameters and show their joint constraints. For example, if we constrain λ_1, then we integrate over λ_2: $P(\lambda_1) = \int d\lambda_2 P(\lambda_1, \lambda_2)$. This gives us elliptical constrains, as we also have assumed Gaussian Likelihoods, therefore in our case we will have three figures showing the combination of Ω_m, h and A_s constrains.

With this simplified algorithm, it is possible to get forecasts for the cosmological parameters, where the input is the computed theoretical Gaussian covariance and the emulated 3PCF for the calculation of derivatives.

5.2.1 Forecasts for stage IV spectroscopic surveys

Before presenting the results from the described algorithm, I will address an issue encountered with the theoretical Gaussian covariance matrix. This is the first time it has been applied to the specific case of anisotropic 3PCF at redshift zero. Through detailed analysis, we observed discrepancies in the off-diagonal elements of the covariance matrix.

A comparison with the numerical covariance matrix, particularly from the Pinocchio simulation [57], revealed that while the diagonal eigenvalues behaved in the same manner, the full covariance matrix displayed negative eigenvalues, suggesting potential issues. As a result, the current analysis is based solely on the diagonal elements of the covariance matrix. Validation of the full covariance matrix is ongoing, and future work will incorporate it to repeat the analysis with the complete covariance structure. However, this approach is sufficient for now, as we are primarily focused on comparing the impact of different multipoles with different conditions, given the same type of uncertainties.

Once again, these calculations are done for a survey with volume $43(\mathrm{h}^{-1}Gpc)^3$. Similarly, as we did in the previous section 4.2, we are excluding every triangle configuration with $r \leq 20\ h^{-1}$Mpc, due to the issues of the Covariance in the small scales. Firstly, we will analyze the constraining power of each of the isotropic multipoles separately. This can be seen in the top row of Fig. 5.1. Here the conclusions are similar to ones from the SNR analysis, as the *220* and *330* multipoles hold the best parameter constraints. It can be seen that the *330* has the best constraints, and the *000* has the worst. At the same time, every multipole apart from *000* has similar forecasts, which was also the case for the average SNR in Table 4.1.

Similarly as for the isotropic components, also the anisotropic can be analyzed separately, which is done in the middle row in Fig. 5.1. Here, as already shown before, the constraints are a lot worse than from the isotropic multipoles. Nevertheless, it seems that the asymmetric multipoles have the best constraints among the others, even though the *112* is very similar to *312* multipole. The *202* has the best constraining power, approximately matching it with the worst isotropic multipole: *000*.

The results from the separate multipole forecasts do not perfectly correlate neither with the SNR Table 4.1 nor the BAO detectability Table 4.2, as one could expect higher SNR or BAO detectability for the multipoles to match a better constraining power. There is a considerable correlation between them, for example, the average BAO detectability values were approximately equal for the *110*, *220*, and *330* multipoles and similarly for the anisotropic multipoles. This is to be expected, as BAO is rich in cosmological information, and it is sensitive to different cosmologies. However, the Tables were produced for a single cosmology - the Planck cosmology - but the forecasts are estimated by varying the cosmology. Therefore, it is informative to take into consideration Fig. 3.17, which portrays how cosmology affects the 3PCF. The overall amplitude of the 3PCF multipole variation from changing the Ω_m values strongly correlates with the constraining power. For example, looking at the anisotropic multipoles, *202* has clearly the largest changes from different cosmologies, while other multipoles have fairly similar changes, which also reflects the forecasts of the cosmological parameters.

One of the most important forecasts in this work is the comparison between the multipole *000*, isotropic and isotropic with anisotropic multipole forecasts, which can be seen in the bottom row of the Fig. 5.1. It can be seen that there is a very significant improvement by not only considering the *000* multipole, but also the rest of the three isotropic multipoles. The surprising part is also the considerable improvement by adding the anisotropic components. As seen in section 4.2, the anisotropic multipoles do not have a large SNR, but nevertheless, they give a significant contribution to the constraining power. Before drawing conclusions, it must be stressed that this is with a diagonal covariance, therefore for the full covariance the results might vary. However, this indicates the significance of including the anisotropic signal in the future analysis of the LSS clustering.

Another interesting feature in Fig. 5.1 is the rotation of the ellipse seen in all the forecasts. This effect is also observed in the following figures of this chapter. The tilt of the ellipses reflects the correlation between the parameters. If the tilt is 0 or 90 degrees, the two parameters are either degenerate with each other or simply independent. In

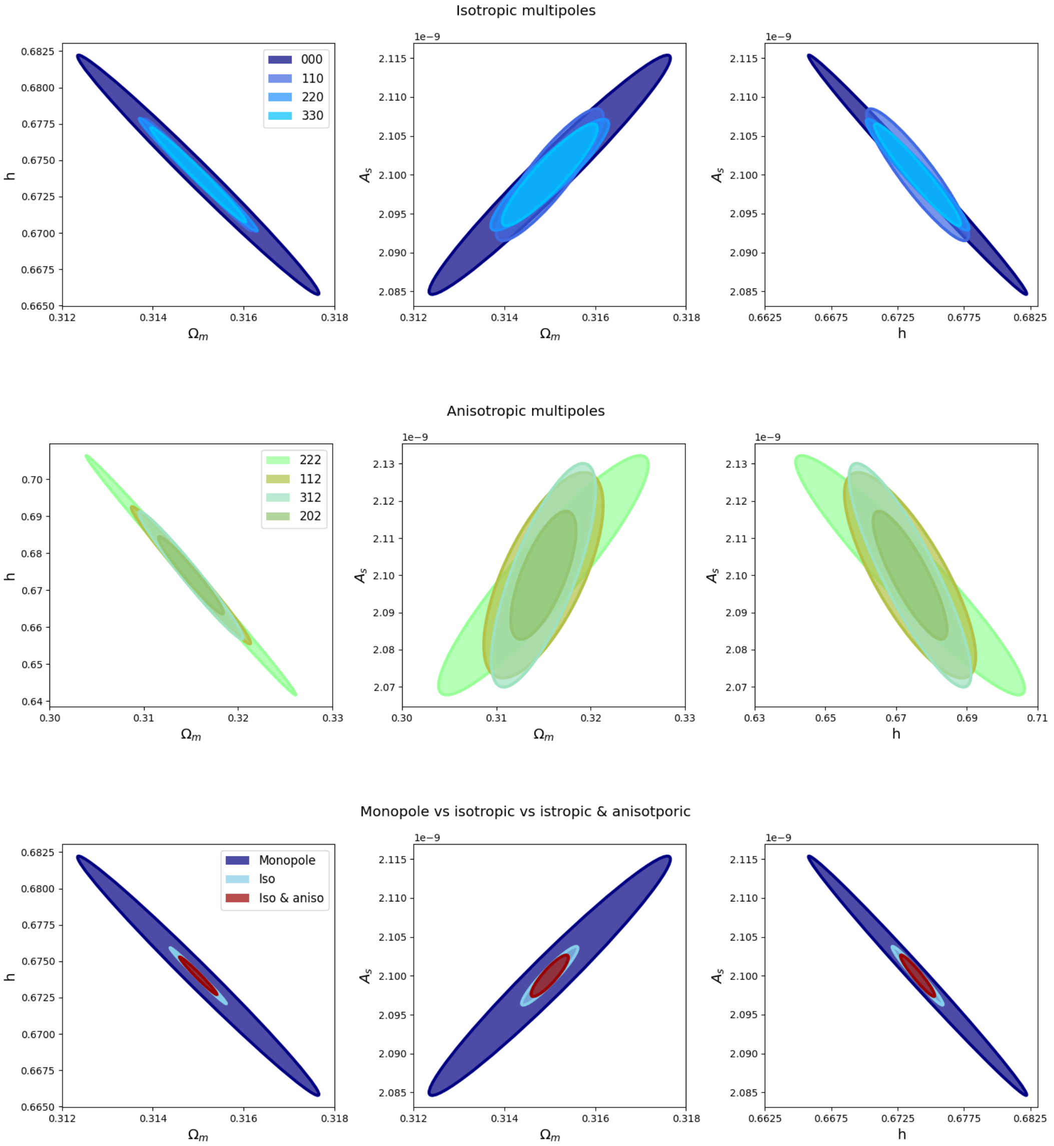


Figure 5.1. Forecasts from different multipoles; Performed for Ω_m, h and A_s cosmological parameters. Top row: Forecasts considering separately the isotropic multipoles. Middle row: Forecasts considering separately the anisotropic multipoles. Bottom row: Forecasts considering either the *000* multipole, all isotropic or isotropic and anisotropic multipoles.

the figure, we can see that the better the constraints, the more the parameters become independent of each other. This is physically reasonable, as different multipoles have different sensitivity to the cosmological parameters [65]. The most informative ones can reduce the dependence between the cosmological parameters, thus enabling us to probe changes in the 3PCF for each parameter separately. This effect can also be seen in other Fisher analysis works in the literature [80].

Moving on, we can also explore the effects of adding different kinds of constraints to the considered triangle configurations in the multipoles. For example, from the previous section 4.2, we can select only the triangles with the highest SNR or BAO detectability. In this case, we choose to report BAO detectability thresholds, as BAO is more sensitive to cosmology, thus it should have better constraining power. The effect of that can be seen in the Fig. 5.2.

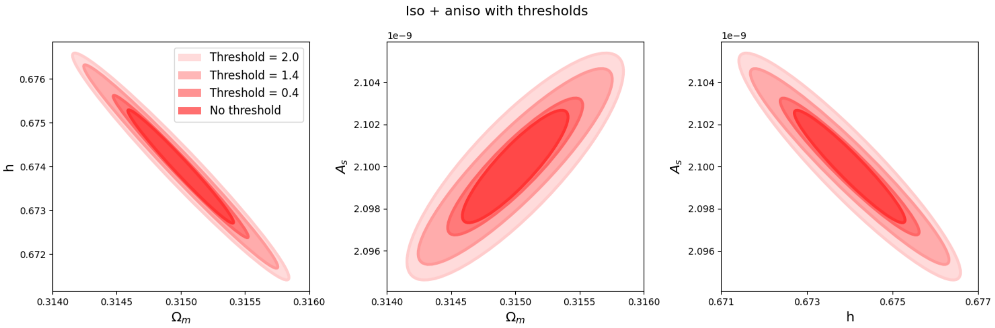


Figure 5.2. Forecasts by applying threshold; Comparison of different threshold for BAO detectability, which is used to select the triangle configurations for the forecasts. Forecasts are done, using the isotropic & anisotropic multipoles.

Statistically, the more triangles, the better the constraints. The interesting part is how much worse the constraining power becomes. We chose four different thresholds, where the threshold is 2.0, as it takes 1500 of the highest signal points, which is attributed to little more than 15% of the initial number of triangles, which is 9800. The threshold 1.4 is for 2*000* points, and then the threshold of 0.4 attributes to half of the points, thus 4900.

It can be seen that by almost reducing the points by half, we do not lose a lot in terms of constraining power. Furthermore, having only 15% of the initial triangles, we still have fairly good constraints. Thus, by reducing the number of triangles by more than 6 times, we only have a reduction of the area of the constraining ellipse by less than 5 times. This means that by selectively choosing the highest BAO detectability triangle configurations, we can obtain good constraints and greatly reduce the computational time.

As we saw in the section 4.2, the highest signal comes from the squeezed triangles. To see the effect of them, it is valuable to pick a certain r_{min}, reducing the amount of squeezed triangles. From that, we will be able to see the importance of them for cosmological parameter constraints. We will start with the isotropic component, as seen in the Fig. 5.3. The effect of removing squeezed triangles is great, compared to the number of triangles removed. In this case, $r_{min} = 30\ h^{-1}$Mpc attributes to 88% of total triangles, while the $r_{min} = 40\ h^{-1}$Mpc has 71% and $r_{min} = 50\ h^{-1}$Mpc has 61% of the total triangles. This means that even with the majority of triangles, we see a considerable reduction of the constraining power, more precisely, the area increases almost 8 times, comparing the no r_{min} with $r_{min} = 42\ h^{-1}$Mpc. This is another proof of the importance of squeezed triangles for the cosmological parameter constraining.

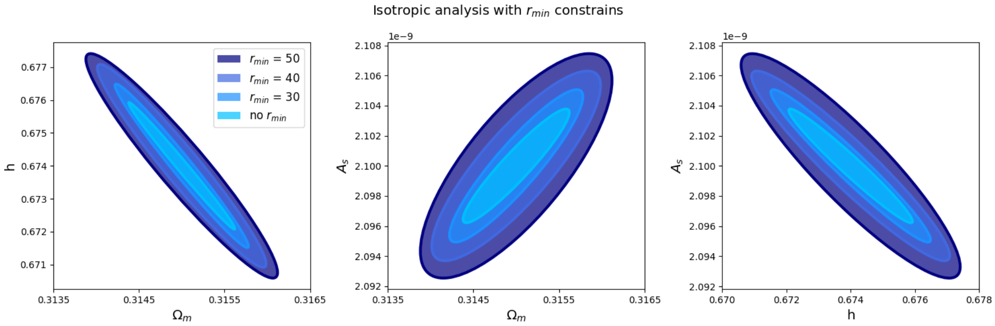


Figure 5.3. Isotropic multipole forecasts by applying r_{min}; The forecasts are done by inducing different r_{min} of values: 30 h^{-1}Mpc, 40 h^{-1}Mpc, 50 h^{-1}Mpc and no r_{min}. No r_{min} is the general case applied in the previous forecasts, where we remove triangles with $r \leq 20$ Mpc/h due to underestimated errors of covariance.

The same can be done also with the combination of isotropic and anisotropic multipoles, which can be seen in Fig. 5.4. The behavior is exactly the same as for the isotropic case, solidifying the importance of including squeezed triangle configurations for the cosmological parameter constraining.

To see how applying an r_{min} condition affects different combinations of multipoles, we explore the $r_{min} = 42\ h^{-1}$Mpc in the Fig. 5.5. It is interesting to compare it with Fig. 5.1, where all triangles, except the ones below the scale of 20 h^{-1}Mpc, are considered for the multipoles. It can be seen that by applying the r_{min} condition, the isotropic and isotropic & anisotropic constraints are nearly indistinguishable. This indicates that squeezed triangles give a significant contribution to the combination of isotropic and anisotropic multipoles. If they are not considered, then considering only the isotropic component is sufficient to obtain the best constraining power. This is an important

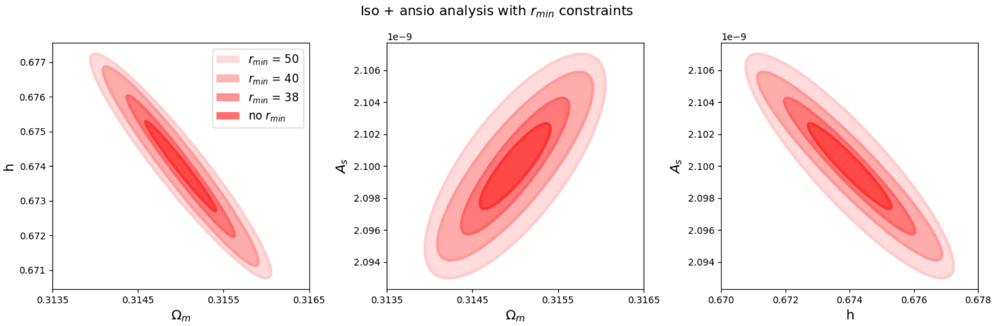


Figure 5.4. Isotropic and anisotropic multipole forecasts by applying r_{min}; The forecasts are done by inducing different r_{min} of values: 18 h^{-1}Mpc, 30 h^{-1}Mpc, 42 h^{-1}Mpc and no r_{min}.

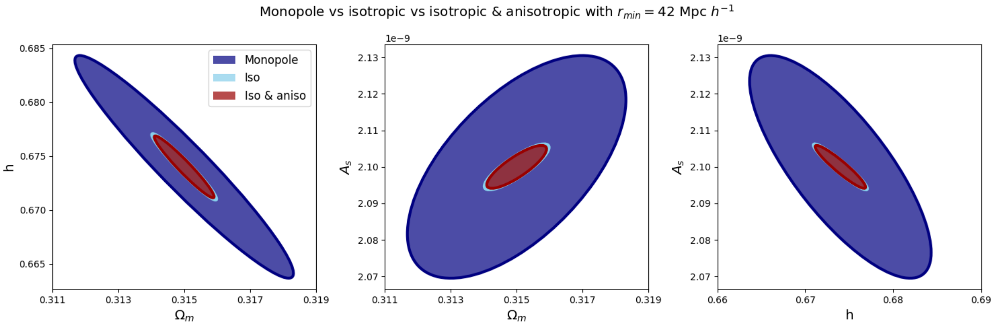


Figure 5.5. Different multipole comparison with $r_{min} = 42$ h^{-1}Mpc; Comparison between *000* multipole, isotropic and isotropic & anisotropic multipole combination, when the condition of $r_{min} = 42$ h^{-1}Mpc has been applied.

aspect of future surveys and simulations, when choosing the r_{min} value, as it has a great impact on the parameter constraints. Furthermore, as of now, the current modeling is usually performed from 40 h^{-1}Mpc [75], which, as seen in the figure, is not the scale where the anisotropic component can give a significant improvement. This is crucial, because excluding smaller scales from the modeling notably reduces the constraining power of the redshift space 3PCF, especially for the anisotropic multipoles.

Finally, we can also consider a different volume for our survey and see if that changes the behavior of the constraints. In particular, we chose the $8(Gpc\ \mathrm{h}^{-1})^3$ volume, which corresponds to one redshift bin for the Euclid survey [44]. Generating the same figures reported above, we will show only one example, which is the comparison between the isotropic and isotropic & anisotropic forecasts, seen in Fig. 5.6. It can be concluded that, even though the constraints have significantly increased, as expected with a lower volume, the behavior is exactly the same - the inclusion of the anisotropic component gives the same improvement over only isotropic constraining power. Therefore, the conclusions drawn for the squeezed triangles and inclusion of the anisotropic multipoles are independent of the volume of the survey.

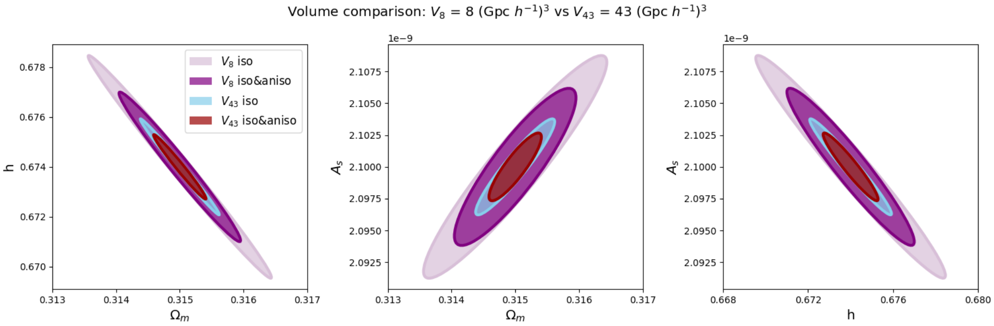


Figure 5.6. Forecast with smaller considered volume; Comparison between isotropic and isotropic & anisotropic forecasts between a possible survey of volume of either $V_{43} = 43(Gpc\ \mathrm{h}^{-1})^3$ or $V_8 = 8(Gpc\ \mathrm{h}^{-1})^3$.

Table 5.1. The predicted 68% confidence Level error for Fisher forecast on cosmological Parameters with the volume $43(Gpc\ \mathrm{h}^{-1})^3$, considering different conditions for the triangle configurations.

	h	Ω_m	A_s
Planck18	6.74×10^{-1}	3.15×10^{-1}	2.10×10^{-9}
Multipoles		**All triangles**	
000	1.75×10^{-3}	5.44×10^{-3}	1.01×10^{-11}
Isotropic	4.11×10^{-4}	1.28×10^{-3}	2.50×10^{-12}
Iso & aniso	2.85×10^{-4}	8.87×10^{-4}	1.80×10^{-12}
Threshold		**Iso & aniso with thresholds for BAO detectability**	
0.42	3.48×10^{-4}	1.08×10^{-3}	2.15×10^{-12}
1.43	5.00×10^{-4}	1.54×10^{-3}	3.07×10^{-12}
1.96	5.58×10^{-4}	1.71×10^{-3}	3.56×10^{-12}
$\mathbf{r}_{\mathrm{min}}$		**Iso & aniso with** r_{min}	
30	4.16×10^{-4}	1.34×10^{-3}	2.81×10^{-12}
40	6.01×10^{-4}	1.90×10^{-3}	3.90×10^{-12}
50	6.97×10^{-4}	2.14×10^{-3}	4.64×10^{-12}

The most important constraints have been summarized in the Table 5.1. Both BAO detectability and r_{min} portray similar information, as the highest BAO detectability is mostly for the squeezed triangles. The bottom line is that squeezed triangles are the most informative triangle configuration. On top of that, if the squeezed triangles are included, the inclusion of anisotropic multipoles to the isotropic gives a significant improvement in the constraining ability. However, the lack of squeezed triangles erases the benefit of the anisotropic multipoles.

To conclude, the main takeaways from the forecast analysis are the following. The anisotropic component on its own does not have a significant constraining power, but combined with the isotropic component it gives a considerable improvement for the cosmological parameter forecasts over isotropic multipoles. However, this is only true if the squeezed triangle configurations are included. If all triangles, that have one of the sides smaller than 42 h^{-1}Mpc, are excluded, then the improvement by adding the anisotropic multipoles in the forecasts analysis is negligible. To add to this point, BAO detectability threshold analysis was performed, which proved that less than 20% of the total number of triangle configurations can still give good results if they have high BAO detectability. Thus, it is important to stress that not all triangles have the same sensitivity for cosmological parameters in each of the multipoles, and computing time can be greatly reduced by only taking the ones with the highest BAO detectability. Finally, all of this analysis is independent of the chosen volume, and all these conclusions can also be applied to a smaller volume.

6 Conclusions and future prospects

In modern cosmology, we have come a long way to find a model, that can accurately describe our Universe - the ΛCDM model. It is simpler compared to other models [64], as it can be described only by six cosmological parameters. With the help of ΛCDM model, we have managed to describe a large portion of the history of our Universe, while measuring the cosmological parameters with unprecedented accuracy [3]. Despite the successes of this model, we are still left in the dark about 95% of our Universe, as we do not know the nature of dark energy - the largest component in the Universe -, as well as about dark matter - the largest gravitational component of our Universe. Moreover, we also lack knowledge of the luminous component, as we are still unclear about the formation of large-scale structures (LSS) - the vast cosmic network of galaxies, clusters, and voids that span the Universe.

A deep understanding of the LSS of the Universe and its information content involves studying the evolution of cosmic structures from the initial conditions set during the early stages of the Universe [4]. This requires knowledge of the role played by cosmological parameters such as the spectral index, scalar amplitude of matter density fluctuations, and the density parameters for matter, dark matter, baryons, dark energy, and radiation in tracing the formation and development of the LSS themselves, and the complex interactions between the different components of the Universe. The formation of LSS is influenced by both linear and non-linear mechanisms, and to describe it accurately, we must understand the interactions between visible and dark matter, as well as the tidal forces that shape the distribution of galaxies. This process involves parameters such as the linear, non-linear, and tidal galaxy bias, in addition to standard cosmological parameters including the Hubble constant and the density fractions of the Universe's various components.

A key role in tracing the expansion history is represented by the Baryonic Acoustic oscillations (BAO). In the early Universe, BAO were frozen at around 100 h^{-1}Mpc near the epoch of recombination. BAO have a critical role in the evolution of clustering of galaxies and matter, as well as in the analysis of large-scale structures. Firstly, it solidifies the theories for the thermal history of the early Universe, as well as the proposed gravitational instability theories. Secondly, the BAO serve as a standard ruler in cosmology, enabling precise measurements of cosmic distances and helping to constrain cosmological parameters and the expansion history. Beyond improving these constraints, BAO also help break degeneracies between parameters, particularly at high redshifts. This makes BAO an especially powerful and reliable tool for cosmological analysis, as they are largely resistant to astrophysical and observational systematics [20].

The BAO features have been mostly analyzed from the two-point statistics [26]. However, with the recent missions, such as Euclid and DESI[44, 1], and upcoming surveys, such as the Roman Space telescope [24], which have subpercent measurement accuracy of the BAO measurements, higher-order statistics represent a powerful tool to extract cosmological information from the BAO signal. For example, to fully exploit the BAO features of the reconstructed power spectrum [26, 79], unreconstructed three- and four-point statistics are needed [59]. Furthermore, three-point statistics allow breaking degeneracies between cosmological parameters, include the description of the non-Gaussian density field, provide valuable insights into the non-linear correlations between galaxies on larger scales, as well as serve as an independent method to constrain cosmological parameters.

Comparing bispectrum with 3PCF, it is evident that bispectrum is easier to model, due to the advantages of the Fourier space, as well as to conduct measurements. This is the reason why it has had continuous theoretical developments, as well as applications to the data of galaxy surveys [38]. However, it holds several disadvantages with respect to its counterpart in configuration space - 3PCF -, for example, mode coupling in Fourier space. In recent years there have been significant efforts to advance the theoretical side of 3PCF [70, 75, 28]. It has come to the point, where 3PCF can be modeled in redshift space, and be separated in its isotropic and anisotropic components [71].

Due to the recency of these developments, there is little knowledge of the BAO appearance in the multipoles of 3PCF [66], especially in the anisotropic component. Furthermore, the influence of including the anisotropic component for constraining the cosmological parameters is also unknown. It is difficult to address these questions, as a single model calculation can take up to 48 CPU hours. Therefore, methods to speed up the computation of 3PCF models must be explored to answer the stated questions, and also enable the usage of Markov-chain Monte Carlo (MCMC) methods to constrain cosmological parameters with the anisotropic 3PCF. Otherwise, it could take up to $480'000$ CPU hours to run an MCMC with 10000 points.

With this Thesis, we aim to address the previously discussed questions by developing a robust and versatile emulator for the 3PCF. For the first time, we generate an emulator for the anisotropic 3PCF, only recently presented in the literature [71] and implemented in a Python package *Mod3l* [28], and use it to assess the sensitivity and power of the anisotropic 3PCF to detect BAO and to derive constraints on cosmological parameters. Here below we summarize the main results.

Emulating the matter anisotropic 3CPF To handle the issue of large computational costs, we developed an emulator for the matter anisotropic 3PCF. It significantly speeds up the calculation of a single model, which allows us to perform further cosmological analysis of the 3PCF. The following steps were performed:

- Before emulating the 3PCF, three emulators for the power spectrum were developed of the cosmological parameters Ω_m, Ω_b, h, and n_s. the emulators for the power spectrum were developed for several reasons. Firstly, the power spectrum is well analyzed in the literature, as well as there have been multiple emulators developed for it. Due to its simplicity, the power spectrum proved to be the perfect target to assess and optimize the parameters of the neural network with the Python library *CosmoPower* [69]. Secondly, to calculate the redshift space 3PCF, the full and no-wiggle power spectrum are needed as an input. Hence, it is useful to understand these functions with great detail, as well as speed up the calculation for the input parameters. Lastly, the analysis of the cosmological parameters allowed us to understand better the influence of them, as well as explore the available range that can be chosen.

- Many different hyperparameters were tested for the neural network to find the best-performing emulator. It was understood that each function is unique, therefore the best hyperparameters for the power spectrum might not be the best for the no-wiggle power spectrum. Nevertheless, each emulator has an accuracy down to 0.01%, which was tested with the method of percentage difference. This method proves to be simple yet effective, as it allows for assessing accuracy across different modes and identifying potential issues. For instance, it was crucial in redefining the lower limit of Ω_m, as the function failed at and beyond the BAO scale when Ω_m was too low.

- Using the *CosmoPower* package for the 3PCF, the emulator did not reach the needed accuracy. In *CosmoPower*, there was no normalization of the data before the calculation of loss, which did not allow us to reach the global minimum for the loss function. Thus, we made a modification for it, which is further described in the Appendix A. With this in hand, the performance of trained emulators greatly increases, making it possible to create an accurate emulator for the anisotropic

3PCF.

- The next step was to emulate the matter anisotropic 3PCF. This is the first emulator for it, which was made on the foundation of the Python package *Mod3l*. This emulator allows to vary the cosmological parameters Ω_m, h, and A_s at redshift zero. The range for the triangle sides is from 10 to 200 with the step of 2 h^{-1}Mpc. It takes around 0.004 seconds to compute one of these models. It has made the calculations more than 10 million times faster, while still maintaining sub-percent accuracy, which can be seen in Table 3.10. Thus, it gives the possibility to execute cosmological analysis with only a laptop at hand.

Assessing the sensitivity to BAO of the anisotropic 3PCF The SNR for 3PCF was studied to understand the physical meaning of the observed features in the anisotropic 3PCF, as well as to find the triangle configurations with the highest signal. For the first time, the anisotropic redshift space 3PCF multipoles were analyzed using a newly developed BAO detectability method, identifying the best triangle configurations that maximize the BAO signal. In particular, the following steps have been done:

- A Gaussian covariance matrix was produced for this work, in order to calculate the SNR and use the BAO detectability metric. It simulates a volume of 43 $(\mathrm{h}^{-1}\mathrm{Gpc})^3$, which corresponds to Stage IV spectroscopic surveys, such as Euclid [44] or DESI [1]. Due to the large differences between the Gaussian and non-Gaussian covariance matrix for the small scales, we constrain the triangle configurations, that satisfy the condition $r > 20$ h^{-1}Mpc.

- An SNR analysis of the redshift space anisotropic 3PCF was performed. We were able to distinguish that the highest SNR for the 3PCF is from the squeezed triangles. This is expected, as one or more sides are in the non-linear regime, where there is an excess of clustering. On top of that, isosceles triangles show a strong signal because the 3PCF calculation involves integrating over the angle between the triangle sides. Consequently, this integral includes a triangle configuration, where the two sides are parallel and pointing in the same direction. This is a highly improbable configuration, as the third side's length approaches zero. This produces a trough in the 3PCF function, which has a large absolute amplitude, thus producing a strong signal. Instead, the BAO features in SNR analysis are hardly noticeable, therefore a different method must be adapted.

- To detect the best triangle configurations for the BAO, we developed a new metric, defined as *BAO detectability*. It is built as the SNR, but instead of the 3PCF in the numerator, it is the difference between the full and the no-wiggle 3PCF. This allows us to visualize which triangle configurations detect BAO features the best. We found that in the case of 3PCF, the BAO signatures are strongest for the squeezed triangles. To be more specific, the highest detectability is found when one of the two sides are 100 and 30 h^{-1}Mpc, respectively. Depending on the multipole, also other configurations show a noticeable detectability, namely (30,60), (30,80), (30,120), and (30,140) h^{-1}Mpc. This is because there are multiple crossings of the BAO scale when the 3PCF is calculated by integrating over the angle between the sides. Additionally, the strong clustering signal at small scales amplifies the BAO signal, which is true even when both sides are not on the BAO scale. The second highest detectability configuration is given by isosceles triangles when both sides are around 100 h^{-1}Mpc. This is because, at any triangle combination, at least two of the sides will contain the BAO scale, thus also giving an excess of signal. It is important to note that each multipole has different SNR and BAO detectability features, especially when comparing the isotropic with the anisotropic ones. For example, there are two asymmetric multipoles due to the expansion order of the triangle side, which have different features compared to the other multipoles.

- An alternative approach to analyzing the BAO detectability was also explored, by using the only-wiggle 3PCF instead of subtracting the full 3PCF with the no-wiggle 3PCF. This is a different metric, as the only-wiggle is only due to the baryonic component. The analysis was performed with this metric, to ascertain, that the triangles, which sides did not include the BAO scale, did not have BAO detectability. It also provides an additional tool to analyze BAO appearance in anisotropic 3PCF, but further work must be done to use it independently.

Forecasts for a Stage IV spectroscopic survey Finally, Fisher forecasts analysis was performed for the parameters Ω_m, h, and A_s by varying the multipoles and other conditions. This is a crucial analysis, as it provides insights into the potential constraining abilities of different multipoles for the next leading Stage IV surveys provided by DESI, Euclid, or the Roman Space telescope. Furthermore, most analysis in the literature has been done for only the isotropic multipoles [67, 29], but anisotropic multipoles have been left less explored. This allows us to find the multipoles with the most constraining power, as well as portray the improvement by including the anisotropic multipoles to the isotropic ones. The analysis can broken down into these conclusions:

- We determined that the *330* multipole had the best constraining power among the isotropic multipoles, although *220* and *110* are very similar. The anisotropic multipoles have a significantly worse constraining power in general, but among them, the best one is the *202* multipole.

- The most important result was obtained from the analysis of the combination of various multipoles. We discovered that the combination of isotropic and anisotropic multipoles gives a significant improvement over isotropic multipoles. The improvement is noteworthy only when small-scale triangle configurations are considered, specifically satisfying the condition of $r_{min} < 42\ h^{-1}$Mpc. This is crucial for the future modeling of the 3PCF, which mostly opt for $r_{min} \approx 40h^{-1}$Mpc [75], which would not yield considerably better constraints by including the anisotropic multipoles.

- Regarding the smaller scales, it was discovered that they hold the most constraining power. The isosceles triangles also have similar properties. This is established by the analysis of different thresholds of BAO detectability and choosing r_{min} to reduce the number of squeezed triangles. Consequentially, excluding 85% of triangles with low BAO detectability from the analysis only increased the area of constraining ellipses by less than five times, while removing 29% of triangles, all of which were squeezed ones, increased the area by almost eight times. This solidifies the significant constraining power of the small-scale triangle configurations.

- We find that for a stage IV spectroscopic survey, using only the isotropic part of the 3PCF provides constraints on $\Omega_m = 0.315 \pm 0.00138$. This is improved to $\Omega_m = 0.315 \pm 0.000887$ when adding also the anisotropic component. This is extremely promising since this analysis is derived from 3PCF constraints only, paving the way to a significant improvement when also the 2PCF is added to the analysis.

To summarize, in this work we developed the first-ever emulator for the matter anisotropic 3PCF. It speeds up the calculation by more than ten million times, while maintaining subpercent accuracy. Additionally, a pioneering exploration of the BAO features for the anisotropic 3PCF was performed, finding high detectability in isosceles and squeezed triangle configurations. Lastly, forecasts for the cosmological parameters Ω_m, h, and A_s were done. We found out that the inclusion of anisotropic multipoles makes a significant contribution to the constraining power of these cosmological parameters.

6.1 Future prospects

The work in this Thesis revolves around the unexplored territory of the redshift space anisotropic 3PCF. Even though a lot of novel work has been done, there is even more to be followed. The next steps will be as follows:

- **Full covariance matrix**: the Fisher forecasts were performed with only the diagonal elements of the covariance matrix, due to the issue of the off-diagonal elements. To confirm and solidify the conclusions from the SNR, BAO detectability, and Fisher forecast analysis, we must take into account the full correlation matrix. Then it will be a proper analysis in the configuration space of the mentioned metrics, which will be done after the Covariance pipeline is validated.

- **Only wiggle 3PCF**: the BAO detectability had the problem of producing a signal at the small scale triangle configurations, which cannot cross the BAO features. In order to solve this numerical issue, we can compute the only wiggle 3CPF, which should be the same as the subtraction between the full and no-wiggle 3PCF. In order to do so, some modifications to the Python package *Mod3l* must be made, further explained in Chapter 4.4.

- **MCMC**: In order to fully test the abilities of the developed emulator, the Markov chain Monte Carlo (MCMC) method must be used on simulation or even survey data to constrain the cosmological parameters. This would give more realistic constraints of the cosmological parameters, which would be more concrete than the Fisher analysis.

- **Galaxy anisotropic 3PCF emulators**: As of now, the emulator was built for matter, therefore the next step is to extend it to the galaxy 3PCF emulators, as that is the observed luminous tracer in the surveys. It requires a large amount of computational resources, which were not available to us during this work. However, we recently acquired more computational resources from *Cineca*, which are planned to be utilized for this purpose.

- **More emulators**: It would be useful to include the dependence on the fiducial cosmology, by involving the AP parameters. Additionally, emulating with more cosmological parameters and at different redshifts could prove useful to different kinds of future surveys.

- **2PCF + 3PCF**: The combination of the 2PCF for the Fisher forecasts or MCMC will significantly increase the constraining power on cosmological parameters and break degeneracies between them.

Many of these steps require high computational power to begin with. For that, we have submitted two class C projects to *Cineca*, and both of them got accepted, thus acquiring 200′000 CPU hours in total for the continuation of the work on the redshift space 3PCF.

Acknowledgments

This research was supported by computational resources provided by Cineca under the project *IsCb2*. The author(s) would like to thank Cineca for providing access to the high-performance computing facilities that contributed to the results presented in this work.

We acknowledge the use of computational resources from the parallel computing cluster of the Open Physics Hub (https: //site.unibo.it/openphysicshub/en) at the Department of Physics and Astronomy in Bologna.

I would like to express my gratitude to my supervisor, Michele Moresco, for entrusting me with this project. It was a pleasure meeting every week, and I appreciated the insightful discussions about our work and future prospects, and the thoughtful guidance you provided throughout this journey. Your support and involvement made this process enjoyable and enriching - thank you!

I am also truly thankful to my co-supervisor, Massimo Guidi. You have helped me from start to finish in every aspect, providing helpful tips, explaining difficult concepts, suggesting interesting scientific analyses, and much more. Thank you for the massive involvement, I greatly appreciate it!

I am also grateful for all the help from my co-supervisor Sofia Contarini. You provided the basis of this thesis and also guided me with any issues I had. Thank you for your continuous support!

I

Appendix

A
Modified CosmoPower

To increase the speed and accuracy of the predictions, we added an internal normalization. This could not be done just by normalizing the input data, as then there would be a problem of the rescaling of the output of the data. Therefore, the following change was done to one of the functions in the *CosmoPower* code:

Listing A.1. TensorFlow prediction function for the output

```
def predictions_tf(self,
                   parameters_tensor
                   ):
    r"""
    Prediction given tensor of input parameters,
    fully implemented in TensorFlow

    Parameters:
        parameters_tensor (Tensor):
            input parameters

    Returns:
        Tensor:
            output predictions
    """
    outputs = []
    layers = [tf.divide(tf.subtract(parameters_tensor, self.parameters_mean),
        self.parameters_std)]
    for i in range(self.n_layers - 1):

        # linear network operation
        outputs.append(tf.add(tf.matmul(layers[-1], self.W[i]), self.b[i]))

        # non-linear activation function
        layers.append(self.activation(outputs[-1], self.alphas[i],
            self.betas[i]))

    # linear output layer
    layers.append(tf.add(tf.matmul(layers[-1], self.W[-1]), self.b[-1]))
    # rescale -> output predictions
    return layers[-1]
```

The rescaling of the data was done in the *training* function itself, where the following code was added after line 606:

Listing A.2. Gaussian normalization

```
training_features = tf.divide(tf.subtract(training_features,
    self.features_mean), self.features_std)
```

The normalization in this code is the gaussian one, therefore it must be changed if Min-Max normalization is intended to be used:

Listing A.3. MinMax normalization

```
training_features = tf.divide(tf.subtract(training_features,
    self.features_min), tf.subtract(self.features_max, self.features_min))
```

Here as the code was intended for the Gaussian normalization, then the *mean* and *std* parameters had to be changed to *min* and *max*.

Bibliography

[1] Behzad Abareshi, J Aguilar, S Ahlen, Shadab Alam, David M Alexander, R Alfarsy, L Allen, C Allende Prieto, O Alves, J Ameel, et al. Overview of the instrumentation for the dark energy spectroscopic instrument. *The Astronomical Journal*, 164(5): 207, 2022.

[2] Elcio Abdalla et al. Cosmology intertwined ii: The hubble constant tension. *Journal of High Energy Physics*, 2022. doi: 10.1088/1475-7516/2022/11/004.

[3] Nabila Aghanim, Yashar Akrami, Mark Ashdown, Jonathan Aumont, Carlo Baccigalupi, Mario Ballardini, Anthony J Banday, RB Barreiro, N Bartolo, S Basak, et al. Planck 2018 results-vi. cosmological parameters. *Astronomy & Astrophysics*, 641:A6, 2020.

[4] Yashar Akrami, Frederico Arroja, M Ashdown, J Aumont, Carlo Baccigalupi, M Ballardini, Anthony J Banday, RB Barreiro, Nicola Bartolo, S Basak, et al. Planck 2018 results-x. constraints on inflation. *Astronomy & Astrophysics*, 641: A10, 2020.

[5] Yashar Akrami, M Ashdown, Jonathan Aumont, Carlo Baccigalupi, M Ballardini, Anthony J Banday, RB Barreiro, Nicola Bartolo, S Basak, K Benabed, et al. Planck 2018 results-vii. isotropy and statistics of the cmb. *Astronomy & Astrophysics*, 641: A7, 2020.

[6] Erwan Allys, Tanguy Marchand, J-F Cardoso, Francisco Villaescusa-Navarro, Shirley Ho, and Stéphane Mallat. New interpretable statistics for large-scale structure analysis and generation. *Physical Review D*, 102(10):103506, 2020.

[7] Lauren Anderson, Eric Aubourg, Stephen Bailey, Florian Beutler, Vaishali Bhardwaj, Michael Blanton, Adam S Bolton, Jon Brinkmann, Joel R Brownstein, Angela Burden, et al. The clustering of galaxies in the sdss-iii baryon oscillation spectro-

scopic survey: baryon acoustic oscillations in the data releases 10 and 11 galaxy samples. *Monthly Notices of the Royal Astronomical Society*, 441(1):24–62, 2014.

[8] Adam Andrews, Jens Jasche, Guilhem Lavaux, and Fabian Schmidt. Bayesian field-level inference of primordial non-gaussianity using next-generation galaxy surveys. *Monthly Notices of the Royal Astronomical Society*, 520(4):5746–5763, 2023.

[9] Nicola Bartolo, Eiichiro Komatsu, Sabino Matarrese, and Antonio Riotto. Non-gaussianity from inflation: Theory and observations. *Physics Reports*, 402(3-4): 103–266, 2004.

[10] Francis Bernardeau, S Colombi, E Gaztanaga, and R Scoccimarro. Large-scale structure of the universe and cosmological perturbation theory. *Physics reports*, 367(1-3):1–248, 2002.

[11] Gianfranco Bertone and Dan Hooper. History of dark matter. *Reviews of Modern Physics*, 90(4):045002, 2018.

[12] Guilherme Brando, Bartolomeo Fiorini, Kazuya Koyama, and Hans A Winther. Enabling matter power spectrum emulation in beyond-λcdm cosmologies with cola. *Journal of Cosmology and Astroparticle Physics*, 2022(09):051, 2022.

[13] Carmelita Carbone, Olga Mena, and Licia Verde. Cosmological parameters degeneracies and non-gaussian halo bias. *Journal of Cosmology and Astroparticle Physics*, 2010(07):020, 2010.

[14] Peter Coles and Francesco Lucchin. *Cosmology: The origin and evolution of cosmic structure*. John Wiley & Sons, 2003.

[15] Dark Energy Survey Collaboration:, T Abbott, FB Abdalla, J Aleksić, S Allam, A Amara, D Bacon, E Balbinot, M Banerji, K Bechtol, et al. The dark energy survey: more than dark energy–an overview. *Monthly Notices of the Royal Astronomical Society*, 460(2):1270–1299, 2016.

[16] Planck Collaboration, Y Akrami, F Arroja, M Ashdown, J Aumont, C Baccigalupi, M Ballardini, AJ Banday, RB Barreiro, N Bartolo, et al. Planck 2018 results. ix. constraints on primordial non-gaussianity. *Astronomy & Astrophysics*, 641:1–47, 2020.

[17] Matthew Colless, Gavin Dalton, Steve Maddox, Will Sutherland, Peder Norberg, Shaun Cole, Joss Bland-Hawthorn, Terry Bridges, Russell Cannon, Chris Collins,

et al. The 2df galaxy redshift survey: spectra and redshifts. *Monthly Notices of the Royal Astronomical Society*, 328(4):1039–1063, 2001.

[18] James W Cooley and John W Tukey. An algorithm for the machine calculation of complex fourier series. *Mathematics of computation*, 19(90):297–301, 1965.

[19] Edmund J Copeland, Mohammad Sami, and Shinji Tsujikawa. Dynamics of dark energy. *International Journal of Modern Physics D*, 15(11):1753–1935, 2006.

[20] Andrei Cuceu, James Farr, Pablo Lemos, and Andreu Font-Ribera. Baryon acoustic oscillations and the hubble constant: past, present and future. *Journal of Cosmology and Astroparticle Physics*, 2019(10):044, 2019.

[21] Kyle S Dawson, David J Schlegel, Christopher P Ahn, Scott F Anderson, Éric Aubourg, Stephen Bailey, Robert H Barkhouser, Julian E Bautista, Alessandra Beifiori, Andreas A Berlind, et al. The baryon oscillation spectroscopic survey of sdss-iii. *The Astronomical Journal*, 145(1):10, 2012.

[22] Eleonora Di Valentino, Olga Mena, Supriya Pan, Luca Visinelli, Weiqiang Yang, Alessandro Melchiorri, David F Mota, Adam G Riess, and Joseph Silk. In the realm of the hubble tension—a review of solutions. *Classical and Quantum Gravity*, 38(15):153001, 2021.

[23] Scott Dodelson and Fabian Schmidt. *Modern cosmology*. Academic press, 2020.

[24] Tim Eifler, Hironao Miyatake, Elisabeth Krause, Chen Heinrich, Vivian Miranda, Christopher Hirata, Jiachuan Xu, Shoubaneh Hemmati, Melanie Simet, Peter Capak, et al. Cosmology with the roman space telescope–multiprobe strategies. *Monthly Notices of the Royal Astronomical Society*, 507(2):1746–1761, 2021.

[25] Daniel J Eisenstein and Wayne Hu. Baryonic features in the matter transfer function. *The Astrophysical Journal*, 496(2):605, 1998.

[26] Daniel J. Eisenstein, Idit Zehavi, David W. Hogg, Rom'an Scoccimarro, Michael R. Blanton, Robert C. Nichol, Ryan Scranton, et al. Detection of the baryon acoustic peak in the large-scale correlation function of sdss luminous red galaxies. *The Astrophysical Journal*, 633:560 – 574, 2005. URL https://api.semanticscholar.org/CorpusID:4834543.

[27] Xiao Fang, Tim Eifler, and Elisabeth Krause. 2d-fftlog: efficient computation of real-space covariance matrices for galaxy clustering and weak lensing. *Monthly Notices of the Royal Astronomical Society*, 497(3):2699–2714, 2020.

[28] Antonio Farina, Alfonso Veropalumbo, Enzo Branchini, and Massimo Guidi. Modeling and measuring the anisotropic halo 3-point correlation function: a coordinated study. *arXiv preprint arXiv:2408.03036*, 2024.

[29] Enrique Gaztanaga and R Scoccimarro. The three-point function in large-scale structure: redshift distortions and galaxy bias. *Monthly Notices of the Royal Astronomical Society*, 361(3):824–836, 2005.

[30] M. Guidi, A. Veropalumbo, E. Branchini, A. Eggemeier, and C. Carbone. Modelling the next-to-leading order matter three-point correlation function using FFTLog. *Journal of Cosmology and Astroparticle Physics*, 2023(8):066, August 2023. doi: 10.1088/1475-7516/2023/08/066.

[31] Massimo Guidi. *A new model for three-point statistics to probe Galaxy Clustering in the nonlinear regime*. PhD thesis, Università degli studi Roma Tre, 2023.

[32] Hong Guo, Cheng Li, YP Jing, and Gerhard Börner. Stellar mass and color dependence of the three-point correlation function of galaxies in the local universe. *The Astrophysical Journal*, 780(2):139, 2013.

[33] Alan H Guth. Inflationary universe: A possible solution to the horizon and flatness problems. *Physical Review D*, 23(2):347, 1981.

[34] AJS Hamilton. Uncorrelated modes of the non-linear power spectrum. *Monthly Notices of the Royal Astronomical Society*, 312(2):257–284, 2000.

[35] Kai Hoffmann, Julien Bel, and Enrique Gaztanaga. Linear and non-linear bias: predictions vs. measurements. *Monthly Notices of the Royal Astronomical Society*, page stw2876, 2016.

[36] Wayne Hu. Acoustic oscillations, Accessed 2024. URL https://background.uchicago.edu/~whu/intermediate/acoustic.html. University of Chicago.

[37] Dragan Huterer and Daniel L Shafer. Dark energy two decades after: observables, probes, consistency tests. *Reports on Progress in Physics*, 81(1):016901, 2017.

[38] Mikhail M Ivanov, Oliver HE Philcox, Giovanni Cabass, Takahiro Nishimichi, Marko Simonović, and Matias Zaldarriaga. Cosmology with the galaxy bispectrum multipoles: Optimal estimation and application to boss data. *Physical Review D*, 107(8):083515, 2023.

[39] ADH Science & Justice. Bao and cmb: Understanding the sound of the universe, Accessed 2024. URL https://adh-sj.info/bao_cmb.php. ADH Science & Justice.

[40] Neerav Kaushal, Francisco Villaescusa-Navarro, Elena Giusarma, Yin Li, Conner Hawry, and Mauricio Reyes. Necola: toward a universal field-level cosmological emulator. *The Astrophysical Journal*, 930(2):115, 2022.

[41] Martin Kerscher. Statistical analysis of large-scale structure in the universe. In *Statistical Physics and Spatial Statistics: The art of analyzing and modeling spatial structures and pattern formation*, pages 36–71. Springer, 2000.

[42] Juliana Kwan, Katrin Heitmann, Salman Habib, Nikhil Padmanabhan, Earl Lawrence, Hal Finkel, Nicholas Frontiere, and Adrian Pope. Cosmic emulation: fast predictions for the galaxy power spectrum. *The Astrophysical Journal*, 810(1): 35, 2015.

[43] Stephen D Landy and Alexander S Szalay. Bias and variance of angular correlation functions. *Astrophysical Journal, Part 1 (ISSN 0004-637X), vol. 412, no. 1, p. 64-71.*, 412:64–71, 1993.

[44] Rene Laureijs, Jérôme Amiaux, S Arduini, J-L Augueres, J Brinchmann, R Cole, M Cropper, C Dabin, L Duvet, A Ealet, et al. Euclid definition study report. *arXiv preprint arXiv:1110.3193*, 2011.

[45] Antony Lewis and Sarah Bridle. Cosmological parameters from cmb and other data: A monte carlo approach. *Physical Review D*, 66(10):103511, 2002.

[46] Felipe A Marín, Chris Blake, Gregory B Poole, Cameron K McBride, Sarah Brough, Matthew Colless, Carlos Contreras, Warrick Couch, Darren J Croton, Scott Croom, et al. The wigglez dark energy survey: constraining galaxy bias and cosmic growth with three-point correlation functions. *Monthly Notices of the Royal Astronomical Society*, 432(4):2654–2668, 2013.

[47] Jérôme Martin, Christophe Ringeval, Roberto Trotta, and Vincent Vennin. The best inflationary models after planck. *Journal of Cosmology and Astroparticle Physics*, 2014(03):039, 2014.

[48] Federico Marulli, Alfonso Veropalumbo, and Michele Moresco. Cosmobolognalib: C++ libraries for cosmological calculations. *Astronomy and Computing*, 14:35–42, 2016.

[49] Ali Masoumi, Alexander Vilenkin, and Masaki Yamada. Inflation in random gaussian landscapes. *Journal of Cosmology and Astroparticle Physics*, 2017(05): 053, 2017.

[50] Michele Moresco, Alfonso Veropalumbo, Federico Marulli, Lauro Moscardini, and Andrea Cimatti. C3: Cluster clustering cosmology. ii. first detection of the baryon acoustic oscillations peak in the three-point correlation function of galaxy clusters. *The Astrophysical Journal*, 919(2):144, 2021.

[51] Nikhil Padmanabhan and Martin White. Constraining anisotropic baryon oscillations. *Physical Review D—Particles, Fields, Gravitation, and Cosmology*, 77(12): 123540, 2008.

[52] PJE Peebles and Edward J Groth. Statistical analysis of catalogs of extragalactic objects. v-three-point correlation function for the galaxy distribution in the zwicky catalog. *Astrophysical Journal, vol. 196, Feb. 15, 1975, pt. 1, p. 1-11. NSF-supported research.*, 196:1–11, 1975.

[53] Saul Perlmutter, Goldhaber Aldering, Gerson Goldhaber, Richard A Knop, Peter Nugent, Patricia G Castro, Susana Deustua, Sebastien Fabbro, Ariel Goobar, Donald E Groom, et al. Measurements of ω and λ from 42 high-redshift supernovae. *The Astrophysical Journal*, 517(2):565, 1999.

[54] Oliver HE Philcox, Mikhail M Ivanov, Marko Simonović, and Matias Zaldarriaga. Combining full-shape and bao analyses of galaxy power spectra: a 1.6% cmb-independent constraint on h0. *Journal of Cosmology and Astroparticle Physics*, 2020(05):032, 2020.

[55] Anna Pugno, Alexander Eggemeier, Cristiano Porciani, and Joseph Kuruvilla. The streaming model for the three-point correlation function and its connection to standard perturbation theory. *arXiv e-prints*, art. arXiv:2408.10307, August 2024. doi: 10.48550/arXiv.2408.10307.

[56] Adam G Riess, Alexei V Filippenko, Peter Challis, Alejandro Clocchiatti, Alan Diercks, Peter M Garnavich, Ron L Gilliland, Craig J Hogan, Saurabh Jha, Robert P Kirshner, et al. Observational evidence from supernovae for an accelerating universe and a cosmological constant. *The astronomical journal*, 116(3):1009, 1998.

[57] Luca A Rizzo, Francisco Villaescusa-Navarro, Pierluigi Monaco, Emiliano Munari, Stefano Borgani, Emanuele Castorina, and Emiliano Sefusatti. Simulating cosmolo-

gies beyond λcdm with pinocchio. *Journal of Cosmology and Astroparticle Physics*, 2017(01):008, 2017.

[58] Ariel G Sánchez, Andrés N Ruiz, Jenny Gonzalez Jara, and Nelson D Padilla. Evolution mapping: a new approach to describe matter clustering in the non-linear regime. *Monthly Notices of the Royal Astronomical Society*, 514(4):5673–5685, 2022.

[59] Marcel Schmittfull, Yu Feng, Florian Beutler, Blake Sherwin, and Man Yat Chu. Eulerian BAO reconstructions and N -point statistics. *Physical Review D*, 92(12): 123522, December 2015. doi: 10.1103/PhysRevD.92.123522.

[60] Roman Scoccimarro, HMP Couchman, and Joshua A Frieman. The bispectrum as a signature of gravitational instability in redshift space. *The Astrophysical Journal*, 517(2):531, 1999.

[61] Daniel Scolnic, Adam G Riess, Yukei S Murakami, Erik R Peterson, Dillon Brout, Maria Acevedo, Bastien Carreres, David O Jones, Khaled Said, and Cullan Howlett. The hubble tension in our own backyard: Desi and the nearness of the coma cluster. *arXiv preprint arXiv:2409.14546*, 2024.

[62] Douglas Scott. The standard model of cosmology: A skeptic's guide. In *Gravitational Waves and Cosmology*, pages 133–153. IOS Press, 2020.

[63] S Shankaranarayanan and Joseph P Johnson. Modified theories of gravity: Why, how and what? *General Relativity and Gravitation*, 54(5):44, 2022.

[64] Ke Shi, YF Huang, and Tan Lu. A comprehensive comparison of cosmological models from the latest observational data. *Monthly Notices of the Royal Astronomical Society*, 426(3):2452–2462, 2012.

[65] Zachary Slepian and Daniel J Eisenstein. Computing the three-point correlation function of galaxies in time. *Monthly Notices of the Royal Astronomical Society*, 454(4):4142–4158, 2015.

[66] Zachary Slepian and Daniel J Eisenstein. Modelling the large-scale redshift-space 3-point correlation function of galaxies. *Monthly Notices of the Royal Astronomical Society*, 469(2):2059–2076, 2017.

[67] Zachary Slepian and Daniel J Eisenstein. A practical computational method for the anisotropic redshift-space three-point correlation function. *Monthly Notices of the Royal Astronomical Society*, 478(2):1468–1483, 2018.

[68] Zachary Slepian, Daniel J Eisenstein, Florian Beutler, Chia-Hsun Chuang, Antonio J Cuesta, Jian Ge, Héctor Gil-Marín, Shirley Ho, Francisco-Shu Kitaura, Cameron K McBride, et al. The large-scale three-point correlation function of the sdss boss dr12 cmass galaxies. *Monthly Notices of the Royal Astronomical Society*, 468(1): 1070–1083, 2017.

[69] Alessio Spurio Mancini, Davide Piras, Justin Alsing, Benjamin Joachimi, and Michael P Hobson. Cosmopower: emulating cosmological power spectra for accelerated bayesian inference from next-generation surveys. *Monthly Notices of the Royal Astronomical Society*, 511(2):1771–1788, 2022.

[70] Naonori S Sugiyama, Shun Saito, Florian Beutler, and Hee-Jong Seo. A complete fft-based decomposition formalism for the redshift-space bispectrum. *Monthly Notices of the Royal Astronomical Society*, 484(1):364–384, 2019.

[71] Naonori S Sugiyama, Shun Saito, Florian Beutler, and Hee-Jong Seo. Towards a self-consistent analysis of the anisotropic galaxy two-and three-point correlation functions on large scales: application to mock galaxy catalogues. *Monthly Notices of the Royal Astronomical Society*, 501(2):2862–2896, 2021.

[72] Istvań Szapudi. Three-point statistics from a new perspective. *The Astrophysical Journal*, 605:L89–L92, April 2004. doi: 10.1086/420748.

[73] István Szapudi and Alexander S Szalay. A new class of estimators for the n-point correlations. *The Astrophysical Journal*, 494(1):L41, 1998.

[74] Tilman Tröster, Ariel G Sánchez, Marika Asgari, Chris Blake, Martín Crocce, Catherine Heymans, Hendrik Hildebrandt, Benjamin Joachimi, Shahab Joudaki, Arun Kannawadi, et al. Cosmology from large-scale structure. *Astronomy & Astrophysics*, 633:L10, 2019.

[75] A Veropalumbo, A Binetti, E Branchini, M Moresco, P Monaco, A Oddo, AG Sánchez, and E Sefusatti. The halo 3-point correlation function: a methodological analysis. *Journal of Cosmology and Astroparticle Physics*, 2022(09):033, 2022.

[76] Alfonso Veropalumbo, Iñigo Sáez Casares, Enzo Branchini, Benjamin R. Granett, Luigi Guzzo, Federico Marulli, Michele Moresco, Lauro Moscardini, Andrea Pezzotta, and Sylvain de la Torre. A joint 2- and 3-point clustering analysis of the

VIPERS PDR2 catalogue at z 1: breaking the degeneracy of cosmological parameters. *Monthly Notices of the Royal Astronomical Society*, 507(1):1184–1201, October 2021. doi: 10.1093/mnras/stab2205.

[77] Zvonimir Vlah, Uroš Seljak, Man Yat Chu, and Yu Feng. Perturbation theory, effective field theory, and oscillations in the power spectrum. *Journal of Cosmology and Astroparticle Physics*, 2016(03):057, 2016.

[78] David H. Weinberg. Reconstructing primordial density fluctuations – I. Method. *Monthly Notices of the Royal Astronomical Society*, 254(2):315–342, 01 1992. ISSN 0035-8711. doi: 10.1093/mnras/254.2.315. URL https://doi.org/10.1093/mnras/254.2.315.

[79] Martin White. Reconstruction within the Zeldovich approximation. *Monthly Notices of the Royal Astronomical Society*, 450(4):3822–3828, July 2015. doi: 10.1093/mnras/stv842.

[80] Victoria Yankelevich and Cristiano Porciani. Cosmological information in the redshift-space bispectrum. *Monthly Notices of the Royal Astronomical Society*, 483 (2):2078–2099, 2019.

[81] Insu Yi, Ethan T Vishniac, and Shin Mineshige. Generation of non-gaussian fluctuations during chaotic inflation. *Physical Review D*, 43(2):362, 1991.

[82] Donald G. York, J. Adelman, John E. Jr. Anderson, Scott F. Anderson, James Annis, Neta A. Bahcall, J.A. Bakken, Robert Barkhouser, Steven Bastian, Eileen Berman, et al. The sloan digital sky survey: Technical summary. *The Astronomical Journal*, 120(3):1579–1587, 2000. doi: 10.1086/301513.

[83] Sihan Yuan, Tom Abel, and Risa H Wechsler. Robust cosmological inference from non-linear scales with k-th nearest neighbour statistics. *Monthly Notices of the Royal Astronomical Society*, 527(2):1993–2009, 2024.