
Dissertation

submitted to the combined Faculties for the Natural Sciences and
for Mathematics of the Ruperto-Carola University of Heidelberg,
Germany for the degree of Doctor of Natural Sciences

Put forward by

Dipl.-Phys. Adrian Vollmer

Born in Ochsenfurt

Oral examination: 19.11.2014

**Efficient computation of nonlinear corrections to
the power spectrum using the time
renormalization group**

Referees: Prof. Dr. Luca Amendola
Prof. Dr. Matthias Bartelmann

Zusammenfassung. Wir entwickeln ein neues Verfahren, um mittels der Time Renormalization Group das heutige nichtlineare Materie-Powerspektrum bis zur BAO-Skala zu berechnen. Der große Vorteil unserer Methode ist eine signifikante Reduktion in der Laufzeit. Insgesamt erreicht unsere numerische Implementation eine Beschleunigung um den Faktor 50 im Vergleich zu ähnlichen Implementationen, sodass eine Laufzeit von nur sechs Sekunden erreicht wird.

Als erste Anwendung untersuchen wir, inwiefern nichtlineare Korrekturen von verschiedenen Quellen die Fehler beeinflussen, die man durch Verwendung der Fishermatrix erhält. Die Ergebnisse dieser Quellen mögen übereinstimmen, aber da die Fishermatrix von Ableitungen abhängt, ist es nicht sofort offensichtlich, dass wir hier ebenfalls Übereinstimmung erwarten können. Wir finden Hinweise darauf, dass erhebliche Unterschiede vorkommen können.

Unabhängig davon führen wir eine Fishermatrixanalyse für Beobachtungen schwacher Gravitationslinsen durch, die komplett modellunabhängig den anisotropischen Druck $\eta = -\Phi/\Psi$ einschränkt. Wir behandeln verschiedene Fälle, in denen wir unter anderem η als zeit- und skalenabhängig betrachten oder in denen es die Horndeski-Parameterisierung annimmt. Es zeigt sich, dass η durch zukünftige Euclid-artige Beobachtungen zusammen mit Supernovaedaten im besten Fall mit einer Genauigkeit von 1% gemessen werden kann, und im Horndeski-Fall mit höchstens 60%.

Abstract. We develop a new technique to compute the nonlinear matter power spectrum up to BAO scales at present time using the Time Renormalization Group. The great advantage of our method is a significant decrease in runtime. Overall, our numerical implementation yields a speedup of a factor of 50 when compared to similar implementations, leading to a runtime that is as little as six seconds.

As a first application, we investigate how sensitive the constraints obtained from the Fisher matrix are to the source of nonlinear corrections. The results from these sources may agree, but because the Fisher matrix depends on derivatives it is not immediately obvious that we can expect agreement here as well. We find hints that there may be substantial differences.

Unrelated to this, we perform a weak lensing Fisher matrix analysis on the anisotropic stress $\eta = -\Phi/\Psi$ in a completely model-independent way. We cover different cases where we, among others, allow η to vary with time and scale and where it takes the Horndeski parameterization. We find that, in the best case, η can be constrained by future Euclid-like surveys together with supernovae data to within 1%, and to within 60% or better in the Horndeski case.

Hydrogen is an odorless, colorless gas which, given enough time, turns into people.

Edward R. Harrison

Contents

| | | |
|----------|--|-----------|
| 1 | Introduction | 13 |
| 1.1 | Goal and motivation | 13 |
| 1.2 | Preview | 14 |
| 1.3 | Structure of this document | 15 |
| 1.4 | Acknowledgments | 15 |
| 1.5 | Nomenclature | 15 |
| 1.6 | Glossary | 16 |
| | | |
| I | Perturbations | 17 |
| | | |
| 2 | Review of perturbation theory in cosmology | 18 |
| 2.1 | General relativity | 18 |
| 2.1.1 | Einstein's field equations | 19 |
| 2.1.2 | The Friedmann-Lemaître-Robertson-Walker universe | 21 |
| 2.1.3 | Distance measures in cosmology | 24 |
| 2.2 | Fluid equations | 25 |
| 2.3 | Standard perturbation theory | 27 |
| 2.4 | Renormalized perturbation theory | 31 |
| 2.5 | Time renormalization group | 32 |
| | | |
| 3 | Numerics | 36 |
| 3.1 | Restructuring the differential equations | 36 |
| 3.2 | Symmetries | 38 |
| 3.3 | Tabulated functions | 39 |
| 3.4 | Momentum integration | 40 |
| 3.4.1 | p -Integration | 41 |
| 3.4.2 | q -Integration | 44 |
| 3.5 | Time integration | 48 |
| 3.6 | Initial conditions | 48 |
| | | |
| 4 | Implementation | 51 |
| 4.1 | Structure | 51 |
| 4.1.1 | Initialization | 51 |
| 4.1.2 | Linear run to estimate growth function | 52 |
| 4.1.3 | Nonlinear run | 52 |
| 4.1.4 | Spline interpolation | 53 |

| | | |
|-----------|---|-----------|
| 4.1.5 | Spline evaluation | 53 |
| 4.1.6 | Output | 55 |
| 4.1.7 | Options | 55 |
| 4.2 | Profiling and memory leaks | 58 |
| 4.3 | Algorithms | 59 |
| 4.3.1 | Spline interpolation | 59 |
| 4.3.2 | ODE integrator | 60 |
| 4.3.3 | Cutoff method | 60 |
| 4.4 | Other languages | 61 |
| 4.4.1 | C/C++ | 61 |
| 4.4.2 | MATHEMATICA | 61 |
| 4.4.3 | Other languages | 62 |
| 5 | Accuracy analysis and comparison | 64 |
| 5.1 | Parameter space | 64 |
| 5.1.1 | Extrapolation | 64 |
| 5.1.2 | Number of time steps | 65 |
| 5.1.3 | Number of sampling points | 65 |
| 5.1.4 | Sampling scheme | 68 |
| 5.1.5 | Additional index symmetry | 68 |
| 5.1.6 | Curvature tolerance and slope limit | 69 |
| 5.1.7 | Sampling density | 69 |
| 5.2 | Validating the results | 72 |
| 5.2.1 | Overall comparison | 72 |
| 5.2.2 | Comparison with CLASS | 74 |
| 5.2.3 | Comparison with Copter | 74 |
| 5.2.4 | Comparison to N -body simulations | 77 |
| 5.3 | Performance | 81 |
| II | Weak lensing | 84 |
| 6 | Review of weak lensing | 85 |
| 6.1 | The convergence power spectrum | 85 |
| 6.2 | Weak lensing Fisher matrix | 90 |
| 6.3 | Tomography | 91 |
| 7 | Application: Weak lensing forecast | 94 |
| 7.1 | Setup | 94 |
| 7.2 | Error discussion | 95 |
| 8 | Model-independent constraints on η | 99 |
| 8.1 | Observables | 99 |
| 8.2 | Binning | 101 |
| 8.3 | Weak lensing Fisher matrix | 102 |
| 8.4 | Projection on η | 104 |
| 8.5 | Discussion of the results | 108 |
| 8.6 | Exclusion plot for the Horndeski parameters | 109 |

| | | |
|----------|---|------------|
| 9 | Conclusion and outlook | 113 |
| A | Cosmomathica | 115 |
| B | Splines | 116 |
| C | Time variables and derivatives | 118 |
| D | Numerical antiderivatives | 119 |
| E | Differential equations in full form | 124 |
| F | Excerpts from select source files | 126 |
| | F.1 Definition of the integrand $K_i(k, q)$ | 126 |
| | F.2 The explicit TRG equations | 127 |

Chapter 1

Introduction

1.1 Goal and motivation

In October 2011 the royal Swedish academy of sciences acknowledged the importance of the discovery that the expansion of the universe happens at an accelerated rate by awarding the Nobel prize for physics to its discoverers, Saul Perlmutter, Brian P. Schmidt and Adam G. Riess. Observing an increase of the expansion rate came as a complete surprise to the majority of the cosmological community including the Nobel prize laureates themselves. What causes this mysterious acceleration has not yet been understood, but many different theories trying to explain this phenomenon exist. They range from postulating new forms of matter (Wetterich, 1988; Ratra and Peebles, 1988; Caldwell et al., 1998; Tsujikawa, 2013) over reconsidering the Copernican principle (Pascual-Sánchez, 1999; Enqvist, 2007) to modifying Einstein's field equations for gravity (Brans and Dicke, 1961; Buchdahl, 1970; De Felice and Tsujikawa, 2010). All of those theories are often loosely labeled as *dark energy*, which may be interpreted as an elusive substance making up roughly 70% of the energy content of the universe, but is also often used as an umbrella term for theories that may not necessarily suggest the existence of new forms of energy.

The accelerated expansion exposes a tremendous lack of knowledge regarding our models of fundamental physics. To understand the nature of dark energy and to find the theory that describes our world best, we need to compare the predictions of all theories to observations. One of the most anticipated upcoming surveys is Euclid, a space-based mission which has been approved by ESA in October 2011, in fact in the same week as the Nobel prize has been awarded. The space telescope is planned to launch in 2020 and will orbit the Sun-Earth Lagrangian point L2. It will measure the shape of billions of galaxies with a redshift of up to $z = 2$. The light coming from these galaxies has been slightly distorted by galaxy clusters and dark matter structures on its way to us due to the effect of gravitational lensing. Those distortions contain information that will allow us to accurately reconstruct the expansion history of the universe, giving us an insight of how dark energy influenced the structure formation.

In order to make forecasts about how precisely a survey like Euclid will be able to measure dark energy related cosmological parameters, we can use Markov Chain Monte Carlo (MCMC) methods or the Fisher information matrix. However, these tools require the knowledge of the matter power spectrum at many different points in parameter space. The matter power spectrum is a central quantity in modern cosmology, because

it describes the clumpiness of our universe. In technical terms, the power spectrum is the Fourier transform of the two-point correlation function of the matter density contrast.

To effectively use MCMC or the Fisher matrix, the theoretical computation of the power spectrum needs to be sufficiently fast, which would not be particularly challenging if we just used the power spectrum computed from linear theory. But many probes, such as weak lensing, require the power spectrum at scales where nonlinear effects are important, in particular the scale where baryonic acoustic oscillations (BAO) are visible. Computing the nonlinear power spectrum is a much more daunting task for which several approaches employing perturbation theory exist.

In this work, we focus on a novel implementation of one specific kind of perturbation theory called time renormalization group (TRG) to efficiently compute nonlinear corrections to the matter power spectrum. We describe the difficulties that come with the new technique and how to overcome them, then compare the results and the efficiency to existing algorithms as well as simulated N -body data. We apply our code to compute the weak lensing Fisher matrix for a vanilla Λ CDM model. Then we compare the impact of using different codes to compute the same intermediate quantities can have on the final result.

We also compute the weak lensing Fisher matrix, this time in a completely model-independent way for cosmological observables that can be measured without assuming an underlying model, in particular the anisotropic stress. Considering that the physics of the dark sector is completely unknown, making as few assumptions as possible about it is important when trying to constrain the properties of dark energy. Deviations from the anisotropic stress from unity would indicate that Λ CDM is not the correct model to describe our universe.

1.2 Preview

The module computing the nonlinear corrections has the following features:

1. Written in Fortran2003 for maximum performance
2. Parallelized with OPENMP
3. Interfaces provided for C/C++ and MATHEMATICA
4. Full control over speed and accuracy
5. No dependencies, makes no use of copyrighted source code
6. Tested with GCC 4.6.2 as well as Intel compilers 12.0.2

The code has been published on <https://www.github.org/User0815/trgfast> along with a side product of this work, a framework to unify different cosmological codes including CAMB and CLASS inside a MATHEMATICA package (see appendix A).

1.3 Structure of this document

The dissertation is divided into two parts. The first part focuses on perturbation theory (chapter 2), including the details of SPT and RPT as well as the TRG algorithm (section 2.5). We thoroughly elaborate on the numerical methods developed to achieve the low runtime we aimed for while not compromising on accuracy in chapter 3 and document the technical implementation of it in chapter 4. In chapter 5, we give a detailed analysis of our code and an in depth comparison with similar codes in terms of accuracy and efficiency.

In the second part, we apply this code to compute the weak lensing Fisher matrix for the standard Λ CDM model. After giving a short introduction to the necessary theory in chapter 6, we investigate how different codes used by the cosmology community to compute the matter power spectrum affect the uncertainties provided by the weak lensing Fisher matrix (chapter 7). Moving on from nonlinear corrections, we construct the weak lensing Fisher matrix for the anisotropic stress η in an entirely model-independent way in chapter 8. In this chapter, we find the constraints on η assuming it stays constant, varies with time, scale, or takes the form of the Horndeski parameterization.

Chapters 3 to 5, 7 and 8 contain original research. Parts of this work have been published or are about to be published:

- Chapter 8: Luca Amendola, Simone Fogli, Alejandro Guarnizo, Martin Kunz, Adrian Vollmer: *Model-independent constraints on the cosmological anisotropic stress*. Physical Review D, **89** (2014)(6), 63538.
- Section 8.6: Alejandro Guarnizo, Adrian Vollmer, Luca Amendola: *A cosmological exclusion plot for modified gravity*. To be published.
- Chapters 3 to 5: Adrian Vollmer, Luca Amendola: *An efficient implementation of the time renormalization group technique*. To be published.

1.4 Acknowledgments

First, I want to thank Luca for providing me with the opportunity of working as a PhD student at the ITP, as well as Matthias for accepting the duty of being the second referee. Thanks also to Eduard for always helping out with everything administrative. I'm also grateful for the initial ideas for this work provided by Riccardo. For the occasional illuminating conversation and guidance, I thank Iggy, Valerio, Guillermo, Gerasimos, and Miguel. Thank you for stimulating discussions at lunch, Alicia, Wessel, Valeria, Santiago, Frank, Emilio, Florian, Alejandro, and everybody else. And thanks, Olli, David, Alexander, Thomas, and Svea, for giving me something to look forward to on Fridays. Last but not least, I thank my parents for their support when I needed it.

1.5 Nomenclature

| | |
|--|----------------------------------|
| $A_{acd,bef}(k, \eta)$ | Source term in the TRG equations |
| $\gamma_{acd}(\mathbf{k}, \mathbf{q}, \mathbf{p})$ | Vertex function |

| | |
|--------------------------|---|
| f | Growth rate |
| $d_C(z)$ | Comoving distance |
| δ_D | Dirac delta |
| $P_{ab}(k)$ | Power spectrum for matter ($ab = 11$), velocity (22) and their cross-correlation (12) |
| $I_{acd,bef}(k)$ | Redefinition of the bispectrum |
| η | Number of e -foldings since a_{ini} ; anisotropic stress |
| $\eta_{\mu\nu}$ | Minkowski metric |
| k | Wave number |
| K_i | Integrand of A_i , i.e. $A_i = \int dq K_i(q)$ |
| τ | Conformal time |
| t | Cosmic time |
| δ_m | Matter density contrast |
| \mathcal{H} | Conformal Hubble parameter, $\mathcal{H} = aH$ |
| h | Hubble parameter in units of $100(\text{km/s})/\text{Mpc}$ |
| σ_8 | Matter power spectrum normalization |
| n_s | Scalar spectral index |
| $\Omega_x (\Omega_{x0})$ | Dimensionless density of fluid x , redshift/time dependent (now) |
| a | Scale factor, normalized to unity today |
| z | Redshift |
| \mathcal{L} | Likelihood; Lagrangian |
| Θ_α | Fisher matrix parameter |

1.6 Glossary

| | |
|---------------|-------------------------------------|
| BAO | Baryonic acoustic oscillations |
| MCMC | Markov Chain Monte Carlo |
| Λ CDM | Lambda Cold Dark Matter |
| FLRW | Friedmann-Lemaître-Robertson-Walker |
| ODE | Ordinary differential equation |
| PT | Perturbation theory |
| EdS | Einstein-de Sitter |
| CQ | Coupled Quintessence |
| TRG | Time Renormalization Group |

Part I

Perturbations

Chapter 2

Review of perturbation theory in cosmology

To put this work in the proper context, we begin with a brief review of different approaches of perturbation theory commonly used in cosmology. We first set the terminology by presenting the basic ideas from general relativity needed for perturbation theory and other necessary concepts, following mostly the text by Carroll (2003). We assume the reader is familiar with differential geometry. If not, we recommend either Spivak (1999) for an extensive introduction for the mathematically inclined or Oloff (2004) for a treatment (in German) with general relativity in mind while still being mathematically rigorous. Both of these were written by mathematicians – a standard text by physicists that also covers a lot of differential geometry (albeit less rigorously) is Misner et al. (1973).

Regarding the conventions used here: We use units in which $c = \hbar = 1$. Quantities of dimension length are measured in Mpc/h . The signature of the metric is $-+++$ and the Einstein sum convention is used. Greek indices run from 0 to 3, latin indices from 1 to 3 (or sometimes from 1 to 2). We may use the variables z , a , τ , or $\log a$ interchangeably to denote time dependency. Primes indicate a derivative with respect to $\log a$. The Fourier transform is defined as

$$\tilde{f}(\mathbf{k}) \equiv \frac{1}{(2\pi)^3} \int d^3\mathbf{x} e^{-i\mathbf{k}\mathbf{x}} f(\mathbf{x}). \quad (2.1)$$

Sometimes, we denote the Fourier transform with a tilde, but mostly we just replace the argument \mathbf{x} with \mathbf{k} and omit the tilde. The tilde is also used for integration variables, but the risk of confusion when considering the context is generally low.

2.1 General relativity

The standard model of cosmology, dubbed ΛCDM , posits that the universe is described by general relativity with a cosmological constant and filled with cold dark matter, baryonic matter, and radiation. Einstein's theory of general relativity is based on the well known Einstein equivalence principle, according to which local experiments cannot differentiate between acceleration and gravity. It can be stated as: *A sufficiently small patch of spacetime is governed by special relativity.* Since special relativity uses

Minkowski space to describe spacetime, it is natural to generalize this approach to a Lorentzian manifold to incorporate the Einstein equivalence principle, since Lorentzian manifolds look locally like Minkowski space.

A Lorentzian manifold is defined as a differential manifold equipped with a non-degenerate, smooth, symmetric metric tensor $g_{\mu\nu}$. If we want to measure distances on the manifold, i.e. turn the manifold into a metric space, we need a so called connection ∇ , which is a bilinear map acting on vector fields and can be interpreted as a covariant derivative (a derivative acting on tensor fields that is independent of the particular coordinate system one is using). The fundamental theorem of Riemannian geometry states that there is a unique torsion free connection on a Lorentzian manifold, the Levi-Civita-connection. It is characterized by the fact that it preserves the metric, i.e.

$$\nabla_{\rho}g_{\mu\nu} = 0. \quad (2.2)$$

Having a connection immediately gives us the notion of “straight lines” in a curved spacetime, which are called geodesics. Since freely falling particles move on straight lines in Minkowski space, it is natural to assume that freely falling particles will also move on these generalized straight lines when spacetime is curved. They are fully determined by the geodesic equation,

$$\nabla_{\dot{\gamma}}\dot{\gamma} = 0 \quad (2.3)$$

for a parameterized curve $\gamma(\lambda)$ with a given initial condition. It is justified to think of them as generalized straight lines in the sense that their acceleration vector is always perpendicular to the tangent space at any point of the curve – or that they always represent the shortest distance between two points. The sign of $\gamma^{\mu}\gamma_{\mu}$ can be used to classify curves as time like (negative sign), space like (positive sign), or null (zero).

Thus, if we are given a metric $g_{\mu\nu}$, the equations of motion for test particles are already fixed. The only question left is: What determines the metric?

2.1.1 Einstein’s field equations

To find the metric we need only two more properties that we want the final result to have. First, since we expect the matter distribution to determine the motion of test particles and we already set the framework of differential geometry, it makes sense to incorporate a tensor that describes the matter distribution. Fortunately, there already is a quantity that meets these requirements. It has been used in classical field theory and is called the energy-momentum tensor $T_{\mu\nu}$. It is defined as

$$T_{\mu\nu} = 2\frac{\delta\mathcal{L}_{\text{matter}}}{\delta g_{\mu\nu}} + g^{\mu\nu}\mathcal{L}_{\text{matter}}, \quad (2.4)$$

where $\mathcal{L}_{\text{matter}}$ is the Lagrangian for matter.

The last property is that for small masses and low speeds, the geodesic equation must reduce to Newton’s law of gravity, or the field equations must reduce to the Poisson equation. In the Newtonian limit, we assume that the metric is close to the one of Minkowski space with the metric $\eta_{\mu\nu}$, such that the difference is only a small perturbation:

$$g_{\mu\nu} = \eta_{\mu\nu} + h_{\mu\nu}. \quad (2.5)$$

If we neglect nonlinear orders of $h_{\mu\nu}$, further assume that the metric is static ($\partial_0 g_{\mu\nu} = 0$), and that the test particle moves slowly, i.e.

$$\frac{dx^i}{d\tau} \ll \frac{dt}{d\tau}, \quad (2.6)$$

the geodesic equation can be worked out to read

$$\frac{d^2 x^i}{dt^2} = \frac{1}{2} \partial_i h_{00}. \quad (2.7)$$

This becomes Newton's law of gravity if the time-time component of the metric takes the form

$$g_{00} = -(1 + 2\Phi) \quad (2.8)$$

with the gravitational potential Φ .

The potential is determined by Poisson's equation,

$$\nabla^2 \Phi = 4\pi G \rho, \quad (2.9)$$

where the second derivative of the potential is proportional to the matter distribution. This means that we expect a tensor depending on second derivatives of the metric to appear in the equation along with $T_{\mu\nu}$. The most obvious tensor of rank 2 that fits that description is the Ricci tensor $R_{\mu\nu}$, which is a contraction of the Riemann curvature tensor $R^\rho{}_{\sigma\mu\nu}$. It can be written in terms of the Christoffel symbols,

$$R^\rho{}_{\sigma\mu\nu} = \partial_\mu \Gamma^\rho{}_{\nu\sigma} - \partial_\nu \Gamma^\rho{}_{\mu\sigma} + \Gamma^\rho{}_{\mu\lambda} \Gamma^\lambda{}_{\nu\sigma} - \Gamma^\rho{}_{\nu\lambda} \Gamma^\lambda{}_{\mu\sigma}, \quad (2.10)$$

where the Christoffel symbols are defined as

$$\Gamma^\rho{}_{\mu\nu} \equiv \frac{1}{2} g^{\rho\sigma} (\partial_\nu g_{\sigma\mu} + \partial_\mu g_{\sigma\nu} - \partial_\sigma g_{\mu\nu}). \quad (2.11)$$

Thus, with $R_{\mu\nu} \equiv R^\rho{}_{\mu\rho\nu}$, we might be tempted to write

$$R_{\mu\nu} \propto T_{\mu\nu}, \quad (2.12)$$

and in fact Einstein himself considered this version as well. However, we postulate local conservation of energy and momentum, reflected by the fact that the divergence of $T_{\mu\nu}$ vanishes by construction, i.e. $\nabla^\mu T_{\mu\nu} = 0$. This would have to hold for $R_{\mu\nu}$ as well if eq. (2.12) were correct. But the Bianchi identity tells us that the divergence of the Ricci tensor does not vanish, instead we have

$$\nabla^\mu R_{\mu\nu} = \frac{1}{2} \nabla_\nu R, \quad (2.13)$$

where $R \equiv R^\mu{}_\mu$ is the Ricci scalar, defined as the trace of the Ricci tensor.

To satisfy local energy conservation, we simply rewrite eq. (2.13), remembering that we can use the metric to move indices up and down, to get

$$\nabla^\mu (R_{\mu\nu} - \frac{1}{2} R g_{\mu\nu}) = 0. \quad (2.14)$$

In the parenthesis we see another rank 2 tensor that depends on second derivatives of the metric tensor, and it is divergence free by construction. It is called the Einstein tensor and presents a better candidate to put in relation with the energy-momentum tensor. With it we finally arrive at the correct Einstein field equations:

$$R_{\mu\nu} - \frac{1}{2}Rg_{\mu\nu} = 8\pi GT_{\mu\nu}. \quad (2.15)$$

The proportionality constant can be determined by deriving the Newtonian limit and comparing the result to the Poisson equation. This result has been corroborated by Hilbert, who wrote down the equations even before Einstein by applying the principle of least action with the Ricci scalar as the Lagrangian,

$$S = \frac{1}{16\pi G} \int d^4x \sqrt{-g} R, \quad (2.16)$$

and, of course, by experimental verification a few years later.

What we did not mention until now, and what has not been seriously considered until decades after Einstein developed general relativity, is one of the central topics of this work: The possibility of a cosmological constant. The above arguments also hold if we add the term $\Lambda g_{\mu\nu}$ to the Einstein tensor, or simply -2Λ to the Lagrangian. Depending on which side of the Einstein equation we put the cosmological constant, it can be interpreted as a property of spacetime (being part of the Einstein tensor) or as a novel form of energy (being part of the energy-momentum tensor), famously dubbed *dark energy*. As it turned out, having the cosmological constant describes our universe better than not having it (Perlmutter et al., 1999; Riess et al., 1998; Planck Collaboration et al., 2013; Blanchard et al., 2006), so any credible model of the universe must have something that at least approximates a cosmological constant.

2.1.2 The Friedmann-Lemaître-Robertson-Walker universe

Einstein's field equations only have a handful of exact solutions. One of them is the so called Robertson-Walker metric, which applies to a spatially homogeneous and isotropic matter distribution and is precisely what we need for cosmology. It follows from the assumption of the Copernican principle, the idea that (at least on large enough scales) the universe looks the same everywhere. Also, at least from our point of view, the universe looks isotropic to great degree as evidenced by the near perfect regularity of the CMB (Harrison, 1970; Smoot et al., 1992). Thus, if the universe is homogeneous, it must be isotropic everywhere. Homogeneity is difficult to verify observationally, so we cannot be sure that it really applies, but it seems to be a reasonable assumption. Note that we do not assume homogeneity in time, i.e. there may be special moments in time – and indeed there are: the big bang, matter domination, possibly a big rip (Caldwell et al., 2003), etc.

Any spacetime where the spatial hypersurfaces are maximally symmetric (homogeneous and isotropic) can be described by the Robertson-Walker metric

$$ds^2 = -dt^2 + a^2(t) \left[\frac{dr^2}{1 - \kappa r^2} + r^2 d\Omega^2 \right], \quad (2.17)$$

where the evolution of the scale factor $a(t)$ is determined by Einstein's equation. The

sign of the curvature parameter κ determines the geometry: if it is positive, negative, or zero, the universe is open, closed, or flat, respectively. The coordinates here are comoving, a coordinate system in which observers at rest will see an isotropic universe. We can perform a change in variables

$$r \rightarrow \chi = S_\kappa^{-1}(r) \quad (2.18)$$

to obtain the metric in the form

$$ds^2 = -dt^2 + a^2(t) [d\chi^2 + S_\kappa(\chi)^2 d\Omega^2] , \quad (2.19)$$

where

$$S_\kappa(\chi) \equiv \begin{cases} r_0 \sin(\chi/r_0), & \kappa > 0 \\ \chi, & \kappa = 0 \\ r_0 \sinh(\chi/r_0), & \kappa < 0 \end{cases} . \quad (2.20)$$

Here, r_0 is the present curvature radius of the universe, which is given by

$$r_0 = \frac{1}{\sqrt{|\kappa|}} . \quad (2.21)$$

To find the behavior of the scale factor, we need to solve the field equations. The energy-momentum tensor for a universe filled with a perfect fluid which is homogeneous and comoving is

$$T_{\mu\nu} = (\rho + p)U_\mu U_\nu + pg_{\mu\nu} , \quad (2.22)$$

with the four-velocity $U^\mu = (1, 0, 0, 0)$. With this, the rather complicated Einstein equation reduces to only two ordinary differential equations, which are called the first and second Friedmann equation:

$$\left(\frac{\dot{a}}{a}\right)^2 = \frac{8\pi G}{3}\rho - \frac{\kappa}{a^2} , \quad (2.23)$$

$$\frac{\ddot{a}}{a} = -\frac{4\pi G}{3}(\rho + 3p) . \quad (2.24)$$

At this point we can define two more quantities which are central to cosmology. The Hubble parameter,

$$H \equiv \frac{\dot{a}}{a} , \quad (2.25)$$

and the equation of state,

$$w \equiv \frac{p}{\rho} . \quad (2.26)$$

The equation of state for different fluids can be computed in the framework of statistical mechanics. For pressureless matter (or dust), w vanishes. For radiation, it takes the value $w = \frac{1}{3}$. For vacuum energy it becomes $w = -1$. Another useful equation comes from the conservation of energy and momentum, which can be derived by either expanding $\nabla^\mu T_{\mu\nu} = 0$ or by eliminating \ddot{a} from the Friedmann equations. It reads

$$\frac{\dot{\rho}}{\rho} = -3(1+w)\frac{\dot{a}}{a} . \quad (2.27)$$

The solution gives the behavior of a fluid with a constant equation of state w , which is

$$\rho = \rho_0 a^{-3(1+w)}, \quad (2.28)$$

where ρ_0 is the density of that fluid today at $a = 1$. We further define the fractional energy density as

$$\Omega_x(a) \equiv \frac{\rho_x(a)}{\rho_{\text{crit}}(a)}, \quad (2.29)$$

where the critical energy density is defined as

$$\rho_{\text{crit}}(a) \equiv \frac{3H(a)^2}{8\pi G}, \quad (2.30)$$

which is derived from eq. (2.23) by setting $\kappa = 0$. Thus, this energy density corresponds to a flat universe. The x stands for the energy species and can include the cosmological constant with

$$\rho_\Lambda = \frac{\Lambda}{8\pi G} \quad (2.31)$$

and even a term resembling a ‘‘curvature energy density’’ with

$$\rho_k \equiv -\frac{3\kappa}{8\pi G a^2}. \quad (2.32)$$

However, this last definition is only a technical one, since curvature is not actually a form of energy. It is defined that way because it allows for a very compact form of the first Friedmann equation:

$$\sum_x \Omega_x = 1. \quad (2.33)$$

It also lets us write the curvature radius as

$$r_0 = \frac{1}{H_0 \sqrt{|\Omega_{k0}|}}. \quad (2.34)$$

A somewhat more useful form of the Friedmann equation is obtained when we explicitly reinstate the time dependence (by using the redshift z) of the individual fractional densities:

$$H(z) = H_0 \sqrt{\Omega_{r0}(1+z)^4 + \Omega_{m0}(1+z)^3 + \Omega_{k0}(1+z)^2 + \Omega_{\Lambda 0}}. \quad (2.35)$$

The radiation density Ω_r is often neglected, since it decreases the quickest due to the power of four and is only relevant in the very early stages of the universe.

Before we come to the end of this section, let us consider the Newtonian limit once more. When linearizing the Einstein equations, there can be many coordinate systems who describe the same manifold, but their linearization may be different. This is the problem of gauge transformations (Carroll, 2003, ch. 7). In this work, we will use only the so called Newtonian gauge, which reads

$$ds^2 = -(1 + 2\Psi)dt^2 + a^2(t)(1 + 2\Phi)d\mathbf{x}^2 \quad (2.36)$$

for a FLRW universe. In Λ CDM, we have $\Phi = -\Psi$. This equation also fixes our

convention regarding the signs of Ψ and Φ .

2.1.3 Distance measures in cosmology

In an expanding universe (or even in a static, curved universe – a valid, but unstable solution of the field equations), the concept of a distance becomes ambiguous. Unlike in Minkowski space, distances can take on different values depending on how you measure, or rather define them. For a rigorous and detailed treatment, Hogg (1999) (and references therein) is a recommendable resource, which this section is closely following. Since in practice the observable is always the redshift (and never, say, the scale factor or the conformal time since the big bang), we give all quantities as functions of z . Furthermore, we are almost always interested in the distance of an object to the observer, thus we will not cover distances between two objects at different non-vanishing redshifts. The cosmological redshift relates to the scale factor via

$$a = \frac{1}{1+z}, \quad (2.37)$$

which can be derived by considering the frequency of a photon along its null geodesic in an expanding universe (Carroll, 2003, ch. 8.5).

The most fundamental distance measure is the comoving distance. It stays constant even while the universe expands if it is measured between two comoving observers – hence the name. It equals the proper distance (the distance you would measure with a ruler) divided by the scale factor. We can obtain an expression for it by using the Robertson-Walker metric in eq. (2.19) for a radial null geodesic, such that $dt^2 = a(t)^2 d\chi^2$. Integrating and switching from the scale factor to the redshift gives

$$d_C(z) \equiv \int_0^z \frac{d\tilde{z}}{H(\tilde{z})}. \quad (2.38)$$

It is useful to define a similar quantity called the transverse comoving distance. It is defined such that two objects at the same redshift that are separated by a small angle $\delta\theta$ on the sky have the comoving distance $d_M(z)\delta\theta$ from each other. Here, $d_M(z)$ is the transverse comoving distance and given by

$$d_M(z) = S_\kappa(d_C(z)). \quad (2.39)$$

This can be seen when setting $d\chi = d\phi = 0$ in eq. (2.19). In a flat universe, we have $d_M(z) = d_C(z)$.

Sometimes we know the physical size of a structure (a so called *standard ruler*, for instance BAO), and by measuring its angular size on the sky we can infer its distance. This is the angular diameter distance $d_A(z)$. Specifically,

$$d_A(z)\delta\theta = S, \quad (2.40)$$

where S is the intrinsic physical size of the object and $\delta\theta$ the (small) angular size. This equation looks similar to the one for the transverse comoving distance, except the transverse distance between the two objects here is not the comoving one but the

proper distance. Since these two distances are related by a factor a , we get

$$d_A(z) \equiv \frac{d_M(z)}{1+z}. \quad (2.41)$$

When determining the distance to a light source of known luminosity (a so called *standard candle*, for instance type Ia supernovae) by measuring its flux, the quantity that appears in the corresponding expression is the luminosity distance. In particular, we have

$$\frac{L}{F} \equiv 4\pi d_L^2(z), \quad (2.42)$$

with the intrinsic luminosity L and flux F . In a static universe, the expression on the right hand side would be simply the area of the sphere at the distance of the source. However, due to the expansion, the flux is weakened by two things. First, the energy of the photons is reduced by a factor of a due to the cosmological redshift. Second, the frequency with which the photons pass the sphere is reduced by another factor of a , since the distance of two photons increases with a as the universe expands. Thus, we have

$$\frac{L}{F} = (1+z)^2 A, \quad (2.43)$$

where the area A of the sphere is obtained by integrating over the angle element using the transverse comoving distance. Putting it all together, we get

$$d_L(z) = (1+z)d_M(z). \quad (2.44)$$

The light-travel distance (sometimes also called “lookback distance”) equals the time it takes for a photon to travel from a point at redshift z to the observer. It can be written as

$$d_T(z) \equiv \int_0^z \frac{d\tilde{z}}{(1+\tilde{z})H(\tilde{z})}, \quad (2.45)$$

and we get it by simply integrating over the cosmic time from the time of emission t_{em} to today, i.e.

$$\int_{t_{\text{em}}}^{t_0} dt = \int_{a_{\text{em}}}^1 \frac{da}{aH(a)} \quad (2.46)$$

and again substituting the scale factor for the redshift. As a bonus, this gives the age of the universe for $a_{\text{em}} = 0$.

2.2 Fluid equations

The fluid equations govern a fluid consisting of dust particles and present the starting point when trying to find the time evolution of the matter power spectrum. We want to derive them in this section, largely following Bernardeau et al. (2002) and Bernardeau (2013). We are assuming all particles have the mass m for simplicity. We use comoving coordinates \mathbf{x} and the conformal time τ . A particle with velocity $\mathbf{u}(\mathbf{x}, \tau)$ then has the peculiar velocity

$$\mathbf{u}_p \equiv \mathbf{u} - \mathcal{H}\mathbf{x}, \quad (2.47)$$

where $\mathcal{H} \equiv d \ln a / d\tau = aH$ is the conformal Hubble parameter, and the momentum of such a particle can be written as

$$\mathbf{p} = am\mathbf{u}_p. \quad (2.48)$$

According to the Liouville theorem, the particle number density in phase space $f(\mathbf{x}, \mathbf{p}, \tau)$ (we will suppress the dependencies on space and time for clarity from here on in most cases, as they should be easily deducible) is conserved, i.e.

$$\frac{df}{d\tau} = \frac{\partial}{\partial \tau} f + \frac{d\mathbf{x}}{d\tau} \nabla f + \frac{d\mathbf{p}}{d\tau} \frac{\partial}{\partial \mathbf{p}} f = 0, \quad (2.49)$$

which immediately gives the Vlasov equation

$$\frac{\partial}{\partial \tau} f + \frac{\mathbf{p}}{ma} \frac{\partial}{\partial \mathbf{p}} f - am \nabla \Phi \frac{\partial}{\partial \mathbf{p}} f = 0. \quad (2.50)$$

Here we have used the equations of motion,

$$\frac{d\mathbf{p}}{d\tau} = -am \nabla \Phi, \quad (2.51)$$

and the definition of the momentum,

$$\frac{d\mathbf{x}}{d\tau} = \mathbf{u}_p = \frac{\mathbf{p}}{ma}. \quad (2.52)$$

Note that the gradient with respect to \mathbf{x} is written as ∇ , and the gradient with respect to \mathbf{p} as $\frac{\partial}{\partial \mathbf{p}}$.

Solving the Vlasov equation is obviously very difficult, since it is not only a partial differential equation in seven variables, but it is also nonlinear due to the fact that the gravitational potential Φ depends on f via the Poisson equation. The potential is sourced by the matter distribution and satisfies the Poisson equation,

$$\nabla^2 \Phi = \frac{3}{2} \Omega_m \mathcal{H}^2 \delta_m, \quad (2.53)$$

with the density contrast

$$\delta_m \equiv \rho / \langle \rho \rangle - 1. \quad (2.54)$$

The brackets indicate spatial averaging. However, we can greatly simplify the equations by defining moments of the particle number density f .

We start with the zeroth-order moment of the phase-space density, i.e. the mass density,

$$\rho \equiv \int d^3 \mathbf{p} f(\mathbf{x}, \mathbf{p}, \tau). \quad (2.55)$$

The first-order moment is the peculiar velocity flow,

$$\mathbf{v} = \frac{1}{\rho} \int d^3 \mathbf{p} \frac{\mathbf{p}}{ma} f(\mathbf{x}, \mathbf{p}, \tau), \quad (2.56)$$

and the second-order moment is the stress tensor,

$$\sigma_{ij} = \frac{1}{\rho} \int d^3\mathbf{p} \frac{p_j p_i}{m^2 a^2} f(\mathbf{x}, \mathbf{p}, \tau) - v_i v_j. \quad (2.57)$$

Besides the moments of f , we can also take the n th momentum of the Vlasov equation (eq. (2.50)) by multiplying with $(p_i/am)^n$ and integrating over \mathbf{p} . After some calculation, we finally arrive at the fluid equations, which consist of the continuity equation,

$$\frac{\partial \delta_m}{\partial \tau} + \nabla \cdot [(1 + \delta_m)\mathbf{v}] = 0, \quad (2.58)$$

representing conservation of mass, and the Euler equation,

$$\frac{\partial \mathbf{v}}{\partial \tau} + \mathcal{H}\mathbf{v} + \mathbf{v}\nabla \cdot \mathbf{v} = -\nabla\Phi - \frac{\mathbf{e}_i \nabla_j (\rho \sigma_{ij})}{\rho}, \quad (2.59)$$

describing conservation of momentum. Solving these equations (or at least obtaining the matter power spectrum from them) is the goal of perturbation theories.

We could now define even higher order moments, but in order to close the system in eqs. (2.58) and (2.59), we simply set the stress tensor to zero. This is called the single-streaming approximation and will eventually break down at larger and larger scales as nonlinearities increase with time. It is a very reasonable approximation at early times, but when small perturbations in the mass distribution start to collapse, we expect multiple streams. One way to visualize this is that the velocity is not a single-valued function of position anymore, as there can be particles with different velocities at the same position. This phenomenon is also dubbed *shell crossing* (Valageas, 2010) and will be a major limitation when computing nonlinear corrections to the matter power spectrum with perturbation theory. One of the advantages of N -body simulations is that they do not have this limitation

2.3 Standard perturbation theory

While standard perturbation theory (SPT) will not play a major role in the rest of this text, it is interesting to revisit the first ideas on cosmological perturbation theory as some concepts will reappear later on. Also, since TRG is a relatively new method, we want to know how it compares to the older methods and why improvements were needed in the first place. We will mostly follow the treatment by Bernardeau et al. (2002) and Jain and Bertschinger (1994).

At sufficiently large scales, we expect the universe to be completely homogeneous. In other words, we expect density fluctuations to vanish. Thus, we start out with neglecting nonlinear terms in eqs. (2.53), (2.58) and (2.59) and write

$$\frac{\partial \delta_m}{\partial \tau} + \theta = 0, \quad (2.60)$$

$$\frac{\partial \mathbf{v}}{\partial \tau} + \mathcal{H}\mathbf{v} = -\nabla\Phi, \quad (2.61)$$

where we defined the velocity divergence $\theta \equiv \nabla \cdot \mathbf{v}$. Any vector field is completely determined by its divergence and its vorticity (up to a constant), i.e. given a scalar

field s and a vector field \mathbf{c} , there exists a unique vector field \mathbf{F} (up to a constant) such that

$$\nabla \cdot \mathbf{F} = s, \quad \nabla \times \mathbf{F} = \mathbf{c}. \quad (2.62)$$

This follows from the Helmholtz decomposition (Panofsky and Phillips, 2012). There even exists an explicit expression for \mathbf{F} , namely

$$\mathbf{F} = -\nabla\phi + \nabla \times \mathbf{A} \quad (2.63)$$

with

$$\phi(\mathbf{x}) = \frac{1}{4\pi} \int \frac{s(\mathbf{x}')}{|\mathbf{x} - \mathbf{x}'|} d^3\mathbf{x}', \quad (2.64)$$

$$\mathbf{A}(\mathbf{x}) = \frac{1}{4\pi} \int \frac{\mathbf{c}(\mathbf{x}')}{|\mathbf{x} - \mathbf{x}'|} d^3\mathbf{x}'. \quad (2.65)$$

In our case with $\mathbf{F} = \mathbf{v}$, we call the divergence $\theta(\mathbf{x}, \tau)$ and the vorticity $\mathbf{w}(\mathbf{x}, \tau)$. If the velocity field were curl-free, we could simply carry on with only considering its divergence. And in fact, the curl does almost vanish. It can be shown that the vorticity field decays quickly, i.e. $\mathbf{w}(\tau) \propto a^{-1}$, such that initial vorticities (in the linear regime, at least) are removed by the expansion of the universe. Without neglecting the vorticity, finding a solution to the fluid equations would be much more challenging than it already is, so the tendency for the vorticity to decay gives us a good reason to only focus on the divergence field. However, doing so does contribute to the limitations of the fluid equations at increasingly nonlinear scales. See also (Pueblas and Scoccimarro, 2009; Valageas, 2010).

Applying the divergence to eq. (2.61) and plugging in the time derivative of eq. (2.60) gives us

$$\frac{\partial\theta}{\partial\tau} + \mathcal{H}\theta + \frac{3}{2}\Omega_m\mathcal{H}^2\delta_m = 0. \quad (2.66)$$

If we assume that the density contrast takes the form $\delta_m(\mathbf{x}, \tau) = D(\tau)\delta_m(\mathbf{x}, 0)$, where $D(\tau)$ is called the linear growth function, then the time derivative of eq. (2.66) becomes

$$\frac{d^2D}{d\tau^2} + \mathcal{H}\frac{dD}{d\tau} = \frac{3}{2}\Omega_m\mathcal{H}^2D. \quad (2.67)$$

This is a second-order differential equation and as such it has two independent solutions. Let $D_+(\tau)$ be the fast growing solution and $D_-(\tau)$ the slow growing or decaying solution. Together with the initial conditions specified by some functions $A(\mathbf{x})$ and $B(\mathbf{x})$, the solution takes the form

$$\delta_m(\mathbf{x}, \tau) = D_+(\tau)A(\mathbf{x}) + D_-(\tau)B(\mathbf{x}). \quad (2.68)$$

The evolution of the velocity divergence given by eq. (2.66) is then

$$\theta(\mathbf{x}, \tau) = -\mathcal{H}(\tau)(f(\tau)A(\mathbf{x}) + g(\tau)B(\mathbf{x})), \quad (2.69)$$

where we defined the growth rates

$$f(\tau) \equiv \frac{d \log D_+}{d \log a}, \quad g(\tau) \equiv \frac{d \log D_-}{d \log a}. \quad (2.70)$$

The exact expressions for the growth function and subsequently the growth rates depend on \mathcal{H} and cannot be determined before we choose a cosmology, and for some cosmologies, a closed expression might not even exist.

Moving on from linear to nonlinear perturbation theory, we write eqs. (2.60) and (2.66) as usual in Fourier space, i.e.

$$\frac{\partial \delta_{\mathbf{m}}(\mathbf{k}, \tau)}{\partial \tau} + \theta(\mathbf{k}, \tau) + \int d^3 \mathbf{k}_2 d^3 \mathbf{k}_1 \delta_D(\mathbf{k} - \mathbf{k}_2 - \mathbf{k}_1) \alpha(\mathbf{k}_2, \mathbf{k}_1) \theta(\mathbf{k}_2, \tau) \delta_{\mathbf{m}}(\mathbf{k}_1, \tau) = 0, \quad (2.71)$$

$$\begin{aligned} \frac{\partial \theta(\mathbf{k}, \tau)}{\partial \tau} + \mathcal{H} \theta(\mathbf{k}, \tau) + \frac{3}{2} \mathcal{H}^2 \Omega_m(\tau) \delta_{\mathbf{m}}(\mathbf{k}, \tau) \\ + \int d^3 \mathbf{k}_2 d^3 \mathbf{k}_1 \delta_D(\mathbf{k} - \mathbf{k}_2 - \mathbf{k}_1) \beta(\mathbf{k}_2, \mathbf{k}_1) \theta(\mathbf{k}_2, \tau) \theta(\mathbf{k}_1, \tau) = 0, \end{aligned} \quad (2.72)$$

where the mode coupling, a consequence of the original equations having nonlinear terms, is expressed in the newly defined functions

$$\alpha(\mathbf{k}_2, \mathbf{k}_1) \equiv \frac{(\mathbf{k}_1 + \mathbf{k}_2) \cdot \mathbf{k}_2}{k_2^2}, \quad \beta(\mathbf{k}_2, \mathbf{k}_1) \equiv \frac{(\mathbf{k}_1 + \mathbf{k}_2)^2 \mathbf{k}_1 \cdot \mathbf{k}_2}{2k_1^2 k_2^2}. \quad (2.73)$$

At this point we need to restrict ourselves to a Einstein-de Sitter (EdS) universe which is flat and only contains matter, i.e. $\Omega_m = 1$ and $\Omega_\Lambda = \Omega_k = 0$. We will see later how to extend perturbation theory to other models. In an EdS universe, the conformal Hubble parameter becomes

$$\mathcal{H}(\tau) = 2/\tau, \quad (2.74)$$

such that the growth functions are simply

$$D_+^{\text{EdS}} = a, \quad D_-^{\text{EdS}} = a^{-3/2}, \quad (2.75)$$

while the growth rate becomes constant, $f = 1$.

We now assume that the fields can be written as

$$\delta_{\mathbf{m}}(\mathbf{x}, \tau) = \sum_{n=1}^{\infty} \delta^{(n)}(\mathbf{x}, \tau), \quad \theta(\mathbf{x}, \tau) = \sum_{n=1}^{\infty} \theta^{(n)}(\mathbf{x}, \tau), \quad (2.76)$$

where $\delta^{(n)}$ and $\theta^{(n)}$ go as the n th power of the initial density field. This is the characteristic assumption of standard perturbation theory. As it turns out (Goroff et al., 1986), the solution to eqs. (2.71) and (2.72) takes the form

$$\delta_{\mathbf{m}}(\mathbf{k}, \tau) = \sum_{n=1}^{\infty} a^n(\tau) \delta_n(\mathbf{k}), \quad \theta(\mathbf{k}, \tau) = -\mathcal{H}(\tau) \sum_{n=1}^{\infty} a^n(\tau) \theta_n(\mathbf{k}, \tau), \quad (2.77)$$

where the terms in the sum are defined as

$$\delta_n(\mathbf{k}) = \int d^3 \mathbf{q}_1 \cdots \int d^3 \mathbf{q}_n \delta_D(\mathbf{k} - \mathbf{q}_1 \cdots \mathbf{q}_n) F_n(\mathbf{q}_1, \dots, \mathbf{q}_n) \delta_1(\mathbf{q}_1) \cdots \delta_1(\mathbf{q}_n), \quad (2.78)$$

$$\theta_n(\mathbf{k}) = \int d^3 \mathbf{q}_1 \cdots \int d^3 \mathbf{q}_n \delta_D(\mathbf{k} - \mathbf{q}_1 \cdots \mathbf{q}_n) G_n(\mathbf{q}_1, \dots, \mathbf{q}_n) \delta_1(\mathbf{q}_1) \cdots \delta_1(\mathbf{q}_n). \quad (2.79)$$

The kernels F_n and G_n depend purely on the mode coupling functions and are given recursively by (Goroff et al., 1986; Jain and Bertschinger, 1994)

$$F_n(\mathbf{q}_1, \dots, \mathbf{q}_n) = \sum_{m=1}^{n-1} \frac{G_m(\mathbf{q}_1, \dots, \mathbf{q}_m)}{(2n+3)(n-1)} [(2n+1)\alpha(\mathbf{k}_1, \mathbf{k}_2)F_{n-m}(\mathbf{q}_{m+1}, \dots, \mathbf{q}_n) + 2\beta(\mathbf{k}_1, \mathbf{k}_2)G_{n-m}(\mathbf{q}_{m+1}, \dots, \mathbf{q}_n)], \quad (2.80)$$

$$G_n(\mathbf{q}_1, \dots, \mathbf{q}_n) = \sum_{m=1}^{n-1} \frac{G_m(\mathbf{q}_1, \dots, \mathbf{q}_m)}{(2n+3)(n-1)} [3\alpha(\mathbf{k}_1, \mathbf{k}_2)F_{n-m}(\mathbf{q}_{m+1}, \dots, \mathbf{q}_n) + 2n\beta(\mathbf{k}_1, \mathbf{k}_2)G_{n-m}(\mathbf{q}_{m+1}, \dots, \mathbf{q}_n)], \quad (2.81)$$

with $F_1 = G_1 = 1$ and

$$\mathbf{k}_1 \equiv \sum_{i=1}^m \mathbf{q}_i, \quad \mathbf{k}_2 \equiv \sum_{i=m+1}^n \mathbf{q}_i. \quad (2.82)$$

At very early times, the scale factor was small and the first terms in eq. (2.77) dominate. The linear continuity equation gives us $\delta_1(\mathbf{k}) = \theta_1(\mathbf{k})$, such that $\delta_1(\mathbf{k})$ completely determines the linear fluctuations. Going to higher orders, we can plug the expressions from eq. (2.77) into the definition of the power spectrum, which gives us

$$P(k, \tau)\delta_D(\mathbf{k} + \mathbf{k}') \equiv \langle \delta_m(\mathbf{k}, \tau)\delta_m(\mathbf{k}', \tau) \rangle \quad (2.83)$$

$$= a^2(\tau)\langle \delta_1(\mathbf{k}, \tau)\delta_1(\mathbf{k}', \tau) \rangle + a^4(\tau)[\langle \delta_1(\mathbf{k}, \tau)\delta_3(\mathbf{k}', \tau) \rangle + \langle \delta_2(\mathbf{k}, \tau)\delta_2(\mathbf{k}', \tau) \rangle + \langle \delta_3(\mathbf{k}, \tau)\delta_1(\mathbf{k}', \tau) \rangle] + \mathcal{O}(\delta_1^6). \quad (2.84)$$

In this context, the brackets mean the ensemble average. As a side note, we point out that the power spectrum is also the Fourier transform of the two-point correlation function

$$\xi(\mathbf{x}) = \langle \delta_m(\mathbf{x})\delta_m(\mathbf{x} + \mathbf{x}') \rangle, \quad (2.85)$$

which is the result of the Wiener–Khinchine theorem (Khinchine, 1934). Depending on what convention is used for the Fourier transform, the definition for the power spectrum given here can differ by a factor of $(2\pi)^3$ when compared to other sources. For convenience, we define the power spectrum at different orders as

$$P_{m,n-m}(k)\delta_D(\mathbf{k} + \mathbf{k}') \equiv \langle \delta_m(\mathbf{k})\delta_{n-m}(\mathbf{k}') \rangle, \quad (2.86)$$

such that eq. (2.84) at second order becomes

$$P(k, \tau) = a^2(\tau)P_{11}(k) + a^4(\tau)[P_{22}(k) + 2P_{13}(k)]. \quad (2.87)$$

Here, P_{11} is the initial linear power spectrum, which is typically obtained numerically from Boltzmann codes that evolve the primordial power spectrum through the radiation dominated area until after decoupling (Bond and Efstathiou, 1984; Ma and Bertschinger, 1995). We now have all we need to compute the nonlinear power spectrum at second order within this framework. The remaining expressions in explicit form can be found in the appendix of Carlson et al. (2009) (originally from Makino

et al. (1992)) and read

$$2P_{13}(k) = \frac{2\pi}{252} P_{11}(k) k^3 \int dr P_{11}(kr) \times \left[\frac{12}{r^2} - 158 + 100r^2 - 42r^4 + \frac{3}{r^2} (r^2 - 1)^3 (7r^2 + 2) \log \left(\frac{1+r}{|1-r|} \right) \right], \quad (2.88)$$

$$P_{22}(k) = \frac{2\pi}{98} k^3 \int dr P_{11}(kr) \int_{-1}^1 dx P_{11} \left(k \sqrt{1+r^2-2rx} \right) \frac{(3r+7x-10rx^2)^2}{(1+r^2-2rx)^2}. \quad (2.89)$$

It is possible to draw a diagrammatic representation of the nonlinear corrections similar to Feynman diagrams, where the linear power spectrum corresponds to the tree level, second order perturbations correspond to one-loop corrections, and so on (Bernardeau et al., 2002). Sometimes it is useful to think of the corrections of different orders in terms of these diagrams.

Before we can conclude the summary of standard perturbation theory, we need to discuss the generalization of this result to cosmologies other than EdS. As has been shown in Bouchet et al. (1992), for $\Omega_m \neq 1$ the solutions cannot be separated in functions of τ and \mathbf{k} as in eq. (2.77), complicating this method. However, if we rewrite the ansatz in eq. (2.77) to read

$$\delta(\mathbf{k}, \tau) = \sum_{n=1}^{\infty} D_n(\tau) \delta_n(\mathbf{k}), \quad (2.90)$$

$$\theta(\mathbf{k}, \tau) = -\mathcal{H}(\tau) f \sum_{n=1}^{\infty} E_n(\tau) \theta_n(\mathbf{k}), \quad (2.91)$$

then the system becomes separable if $f = \sqrt{\Omega_m}$ (Scoccimarro et al., 1998), which is in fact a good approximation to Λ CDM where $f \approx \Omega_m^{5/9}$ (Carroll et al., 1992).

2.4 Renormalized perturbation theory

The problem with standard perturbation theory is that higher order corrections can become of the same magnitude in the nonlinear regime as lower ones in the nonlinear regime. It was argued in Crocce and Scoccimarro (2006a) that SPT breaks down already at weakly nonlinear scales of $k < 0.2h/\text{Mpc}$. Because there is no clear expansion parameter that is independent of the scale and redshift, the contributions at each order do not converge as expected. They become larger and there is strong cancellation between them, and to get meaningful results one would have to take the sum of all the terms (Crocce and Scoccimarro, 2006a, see fig. 1).

One method proposed as an attempt to fix this problem is renormalized perturbation theory (RPT) (Crocce and Scoccimarro, 2006a,b, 2008), which can also be described in a diagrammatic language. Unlike SPT, it does not only consider the fastest growing mode but takes the full time dependence into account. The authors of RPT develop their own diagrammatic language where each contribution corresponds to a whole set of diagrams. They show that the power spectrum can be resummed by reorganizing these diagrams such that a better behaved sum is obtained. Using this new language, the

nonlinear propagator is resummed such that it decays almost exponentially at nonlinear scales. By identifying terms in the expression for the power spectrum that make up the renormalized propagator and terms that account for mode-mode coupling, one can control the individual contributions and obtain a well behaved formalism.

More recently, another approach to resum the contributions to the power spectrum has been developed. Due to the similarity to techniques used in quantum field theory, it has been called the eikonal approximation (Bernardeau et al., 2012). This new approach helps with understanding and extending the results from RPT, as it separates contributions coming from the coupling of modes with a very different amplitude and modes with comparable amplitude. The eikonal approximation also makes it possible to incorporate more than just a single fluid, in particular baryonic and dark matter.

Since the focus of our research lies on the numerical implementation of solely the time renormalization group, we refer the interested reader to the articles cited in this section (and references therein) for further details on RPT.

2.5 Time renormalization group

While renormalized perturbation theory and similar methods present an improvement over standard perturbation theory, they also have some shortcomings. One of those shortcomings is the approximation by an Einstein-de Sitter cosmology, which only holds as long as $\Omega_m/f^2 = 1$, and this approximation fails for higher order terms (Bernardeau et al., 2002). Also, it is not possible to extend these methods to cosmologies where the linear growth function is not just time dependent, but also scale dependent. This happens, for example, in models with massive neutrinos (Lesgourgues and Pastor, 2006; Anselmi et al., 2011). However, the so called time renormalization group approach described in Pietroni (2008) (which we want to review in this section) promises to fix these issues while at the same time being much less technical as it does not require a deep knowledge of field theoretical methods. Like RPT, it is also based on the idea of resumming a partial set of diagrams.

The TRG technique is applicable to many non-standard cosmological models, such as those containing massive neutrinos, coupled quintessence, or modified gravity. In general, it covers all models in which the fluid equations take the following form:

$$\text{Continuity:} \quad \frac{\partial \delta_m}{\partial \tau} + \nabla \cdot ((1 + \delta_m) \mathbf{v}) = 0 \quad (2.92)$$

$$\text{Euler:} \quad \frac{\partial \mathbf{v}}{\partial \tau} + \mathcal{H}(\mathbf{v} + [\mathcal{A}\mathbf{v}]) + (\mathbf{v} \cdot \nabla) \mathbf{v} = -\nabla \phi \quad (2.93)$$

$$\text{Poisson:} \quad \nabla^2 \phi = \frac{3}{2} \mathcal{H}^2 \Omega_m (\delta_m + [\mathcal{B}\delta_m]) \quad (2.94)$$

Here, the brackets mean convolution, which become products after a Fourier transformation. We already presented these equations in section 2.2, but now they contain two extra terms, $\mathcal{A}(\mathbf{x}, \tau)$ and $\mathcal{B}(\mathbf{x}, \tau)$. Additionally, the matter density can now also be time dependent, i.e. it is $\Omega_m(\tau)$. In general, \mathcal{A} and \mathcal{B} are different from zero when particles' geodesics are modified. Examples can be scalar-tensor theories (Perrotta et al., 2004) or coupled dark energy models (Amendola, 2004), where \mathcal{B} corresponds to having an effective Newton's constant which is different than the default Newton's constant (more on that in section 5.2.4).

In Fourier space, the fluid equations including \mathcal{A} and \mathcal{B} now read

$$\frac{\partial \delta_m(\mathbf{k}, \tau)}{\partial \tau} + \theta(\mathbf{k}, \tau) + \int d^3\mathbf{q} d^3\mathbf{p} \delta_D(\mathbf{k} - \mathbf{q} - \mathbf{p}) \alpha(\mathbf{q}, \mathbf{p}) \theta(\mathbf{q}, \tau) \delta_m(\mathbf{p}, \tau) = 0, \quad (2.95)$$

$$\begin{aligned} \frac{\partial \theta(\mathbf{k}, \tau)}{\partial \tau} + \mathcal{H}(1 + \mathcal{A}(\mathbf{k}, \tau)) \theta(\mathbf{k}, \tau) + \frac{3}{2} \mathcal{H}^2 (1 + \mathcal{B}(\mathbf{k}, \tau)) \Omega_m(\tau) \delta_m(\mathbf{k}, \tau) \\ + \int d^3\mathbf{q} d^3\mathbf{p} \delta_D(\mathbf{k} - \mathbf{q} - \mathbf{p}) \beta(\mathbf{q}, \mathbf{p}) \theta(\mathbf{q}, \tau) \theta(\mathbf{p}, \tau) = 0. \end{aligned} \quad (2.96)$$

The mode coupling functions α and β have been defined in eq. (2.73). Instead of the physical power spectrum, it will be more convenient to study a closely related quantity, defined by the doublet

$$\begin{pmatrix} \varphi_1(\mathbf{k}, \eta) \\ \varphi_2(\mathbf{k}, \eta) \end{pmatrix} \equiv e^{-\eta} \begin{pmatrix} \delta_m(\mathbf{k}, \eta) \\ -\theta(\mathbf{k}, \eta)/\mathcal{H} \end{pmatrix}. \quad (2.97)$$

We just have to remember to convert it back at the end. The new time variable that we introduced here, η , is the e-folding time and defined by the number of e-folds of the scale factor after some initial moment in time, i.e.

$$\eta \equiv \log \frac{a}{a_{\text{ini}}}. \quad (2.98)$$

The initial scale factor (or redshift) is arbitrary in principle, but needs to be well inside the linear regime. Typically, values such as 35, 50, or 100 are used for the initial redshift. With this, eq. (2.96) becomes

$$\begin{aligned} \partial_\eta \varphi_a(\mathbf{k}, \eta) = -\Omega_{ab}(\mathbf{k}, \eta) \varphi_b(\mathbf{k}, \eta) \\ + e^\eta \int d^3\mathbf{q} d^3\mathbf{p} \gamma_{abc}(\mathbf{k}, -\mathbf{p}, -\mathbf{q}) \varphi_b(\mathbf{p}, \eta) \varphi_c(\mathbf{q}, \eta), \end{aligned} \quad (2.99)$$

with the vertex functions

$$\gamma_{121}(\mathbf{k}, \mathbf{p}, \mathbf{q}) = \frac{1}{2} \delta_D(\mathbf{k} + \mathbf{p} + \mathbf{q}) \alpha(\mathbf{p}, \mathbf{q}), \quad (2.100)$$

$$\gamma_{222}(\mathbf{k}, \mathbf{p}, \mathbf{q}) = \delta_D(\mathbf{k} + \mathbf{p} + \mathbf{q}) \beta(\mathbf{p}, \mathbf{q}), \quad (2.101)$$

$$\gamma_{121}(\mathbf{k}, \mathbf{p}, \mathbf{q}) = \gamma_{112}(\mathbf{k}, \mathbf{q}, \mathbf{p}). \quad (2.102)$$

For any other indices, γ_{abc} vanishes. One of the advantages of the TRG method is that all of the background evolution is now entirely encapsulated in the matrix Ω :

$$\Omega(\mathbf{k}, \eta) = \begin{pmatrix} 1 & -1 \\ -\frac{3}{2} \Omega_m(\eta) (1 + \mathcal{B}(\mathbf{k}, \eta)) & 2 + \frac{\mathcal{H}'}{\mathcal{H}} + \mathcal{A}(\mathbf{k}, \eta) \end{pmatrix}. \quad (2.103)$$

We can now write down the time evolution of the n -point functions iteratively by using the product rule and eq. (2.99) (with repeated indices being summed over and repeated momenta being integrated over):

$$\begin{aligned} \partial_\eta \langle \varphi_a \varphi_b \rangle = -\Omega_{ac} \langle \varphi_c \varphi_b \rangle - \Omega_{bc} \langle \varphi_a \varphi_c \rangle \\ + e^\eta \gamma_{acd} \langle \varphi_c \varphi_d \varphi_b \rangle + e^\eta \gamma_{bcd} \langle \varphi_a \varphi_c \varphi_d \rangle, \end{aligned}$$

$$\begin{aligned}
\partial_\eta \langle \varphi_a \varphi_b \varphi_c \rangle &= -\Omega_{ad} \langle \varphi_d \varphi_b \varphi_c \rangle - \Omega_{bd} \langle \varphi_a \varphi_d \varphi_c \rangle - \Omega_{cd} \langle \varphi_a \varphi_b \varphi_d \rangle \\
&\quad + e^\eta \gamma_{ade} \langle \varphi_d \varphi_e \varphi_b \varphi_c \rangle + e^\eta \gamma_{bde} \langle \varphi_a \varphi_d \varphi_e \varphi_c \rangle \\
&\quad + e^\eta \gamma_{cde} \langle \varphi_a \varphi_b \varphi_d \varphi_e \rangle, \\
\partial_\eta \langle \varphi_a \varphi_b \varphi_c \varphi_d \rangle &= \dots \\
&\quad \vdots
\end{aligned} \tag{2.104}$$

Next, we identify the 2-point correlation function with the power spectrum P_{ab} , the 3-point correlation with the bispectrum B_{abc} , and apply Wick's theorem (also known as Isserlis' theorem) to the 4-point correlation function. Wick's theorem states that the 4-point function can be written as the connected 4-point function plus a sum of products of the 2-point function. Note that this does not imply that ϕ_a has to be Gaussian, since we are still taking into account the full bispectrum, which would vanish in the Gaussian case. This gives us

$$\begin{aligned}
\langle \varphi_a(\mathbf{k}, \eta) \varphi_b(\mathbf{q}, \eta) \rangle &\equiv \delta_D(\mathbf{k} + \mathbf{q}) P_{ab}(\mathbf{k}, \eta), \\
\langle \varphi_a(\mathbf{k}, \eta) \varphi_b(\mathbf{q}, \eta) \varphi_c(\mathbf{p}, \eta) \rangle &\equiv \delta_D(\mathbf{k} + \mathbf{q} + \mathbf{p}) B_{abc}(\mathbf{k}, \mathbf{q}, \mathbf{p}; \eta), \\
\langle \varphi_a(\mathbf{k}, \eta) \varphi_b(\mathbf{q}, \eta) \varphi_c(\mathbf{p}, \eta) \varphi_d(\mathbf{r}, \eta) \rangle &\equiv \\
&\quad [\delta_D(\mathbf{k} + \mathbf{q}) \delta_D(\mathbf{p} + \mathbf{r}) P_{ab}(\mathbf{k}, \eta) P_{cd}(\mathbf{p}, \eta) \\
&\quad + \delta_D(\mathbf{k} + \mathbf{p}) \delta_D(\mathbf{q} + \mathbf{r}) P_{ac}(\mathbf{k}, \eta) P_{bd}(\mathbf{q}, \eta) \\
&\quad + \delta_D(\mathbf{k} + \mathbf{r}) \delta_D(\mathbf{q} + \mathbf{p}) P_{ad}(\mathbf{k}, \eta) P_{bc}(\mathbf{q}, \eta) \\
&\quad + \delta_D(\mathbf{k} + \mathbf{p} + \mathbf{q} + \mathbf{r}) Q_{abcd}(\mathbf{k}, \mathbf{q}, \mathbf{p}, \mathbf{r}, \eta)].
\end{aligned} \tag{2.105}$$

In the last equation, Q_{abcd} is the trispectrum, the connected part of the 4-point function (Verde and Heavens, 2001). The approximation made in TRG is to put Q_{abcd} to zero, leaving a closed system of equations determining the time evolution of the power spectrum and the bispectrum:

$$\begin{aligned}
\partial_\eta P_{ab}(\mathbf{k}, \eta) &= -\Omega_{ac}(\mathbf{k}, \eta) P_{cb}(\mathbf{k}, \eta) - \Omega_{bc}(\mathbf{k}, \eta) P_{ac}(\mathbf{k}, \eta) \\
&\quad + e^\eta \int d^3 \mathbf{q} [\gamma_{acd}(\mathbf{k}, -\mathbf{q}, \mathbf{q} - \mathbf{k}) B_{bcd}(\mathbf{k}, -\mathbf{q}, \mathbf{q} - \mathbf{k}; \eta) \\
&\quad + B_{acd}(\mathbf{k}, -\mathbf{q}, \mathbf{q} - \mathbf{k}; \eta) \gamma_{bcd}(\mathbf{k}, -\mathbf{q}, \mathbf{q} - \mathbf{k}, \eta)],
\end{aligned} \tag{2.106}$$

$$\begin{aligned}
\partial_\eta B_{abc}(\mathbf{k}, -\mathbf{q}, \mathbf{q} - \mathbf{k}; \eta) &= -\Omega_{ad}(\mathbf{k}, \eta) B_{dbc}(\mathbf{k}, -\mathbf{q}, \mathbf{q} - \mathbf{k}; \eta) \\
&\quad - \Omega_{bd}(-\mathbf{q}, \eta) B_{adc}(\mathbf{k}, -\mathbf{q}, \mathbf{q} - \mathbf{k}; \eta) \\
&\quad - \Omega_{cd}(\mathbf{q} - \mathbf{k}, \eta) B_{abd}(\mathbf{k}, -\mathbf{q}, \mathbf{q} - \mathbf{k}; \eta) \\
&\quad + 2e^\eta [\gamma_{ade}(\mathbf{k}, -\mathbf{q}, \mathbf{q} - \mathbf{k}) P_{db}(\mathbf{q}, \eta) P_{ec}(\mathbf{k} - \mathbf{q}, \eta) \\
&\quad + \gamma_{bde}(-\mathbf{q}, \mathbf{q} - \mathbf{k}, \mathbf{k}) P_{dc}(\mathbf{k} - \mathbf{q}, \eta) P_{ea}(\mathbf{k}, \eta) \\
&\quad + \gamma_{cde}(\mathbf{q} - \mathbf{k}, \mathbf{k}, -\mathbf{q}) P_{da}(\mathbf{k}, \eta) P_{eb}(\mathbf{q}, \eta)].
\end{aligned} \tag{2.107}$$

Just like RPT, this framework also allows an interpretation in terms of Feynman diagrams, which has been developed by Matarrese and Pietroni (2007). To put it in words used in field theory, the approximation scheme presented here includes the renormalization of the power spectrum while neglecting vertex renormalization. This

is reflected by the fact that the power spectrum is formally at one-loop order, because it contains an integration over the momentum \mathbf{q} , unlike the bispectrum, which can be said to be at tree level. Thus, this truncation is only valid at the first few orders. One would have to include additional loop corrections to make this truncation valid at higher orders. The goal is now to find the solution to eq. (2.107) numerically.

Chapter 3

Numerics

3.1 Restructuring the differential equations

The equations of motion for the power spectrum in eqs. (2.106) and (2.107) can be put in a much more suitable form for numerical integration. In particular, the 3D integral over \mathbf{q} in eq. (2.106) can be simplified as follows. In an isotropic universe, the power spectrum depends only on the magnitude k of \mathbf{k} , whereas the bispectrum and the gamma functions only depend on k , q , and on either $s \equiv |\mathbf{k} - \mathbf{q}|$ or the angle θ_{kq} between \mathbf{k} and \mathbf{q} . For the integral in eq. (2.106), we adopt a spherical coordinate system where \mathbf{k} points in the z -direction, such that the φ integration can be carried out immediately to yield a factor of 2π . Performing a change of variables

$$\theta_{kq} \rightarrow s = |\mathbf{k} - \mathbf{q}| = \sqrt{k^2 + q^2 - 2kq \cos \theta_{kq}} \quad (3.1)$$

with the transformed differential

$$ds = \frac{kq \, d(\cos \theta_{kq})}{|\mathbf{k} - \mathbf{q}|}, \quad (3.2)$$

the integration measure becomes

$$\int d^3\mathbf{q} = \frac{2\pi}{k} \int_0^\infty dq \, q \int_{|q-k|}^{q+k} ds \, s. \quad (3.3)$$

Note that the sum of all three arguments in the vertex functions appearing in eq. (2.106) are trivially zero, so the Dirac deltas disappear. With the substitution $\cos \theta_{kq} = (s^2 - k^2 - q^2)/(2kq)$ we get

$$\gamma_{121}(\mathbf{k}, \mathbf{q}, \mathbf{p})|_{\mathbf{p}=-(\mathbf{k}+\mathbf{q})} = \frac{q^2 + k^2 - s^2}{4q^2}, \quad (3.4)$$

$$\gamma_{222}(\mathbf{k}, \mathbf{q}, \mathbf{p})|_{\mathbf{p}=-(\mathbf{k}+\mathbf{q})} = \frac{k^2}{4s^2q^2} (k^2 - s^2 - q^2). \quad (3.5)$$

In order to match the notation in Pietroni (2008), we use the abbreviation

$$\tilde{\gamma}_{abc}(k, q, s) \equiv \gamma_{abc}(\mathbf{k}, \mathbf{q}, \mathbf{p})|_{\mathbf{p}=-(\mathbf{k}+\mathbf{q})} \quad (3.6)$$

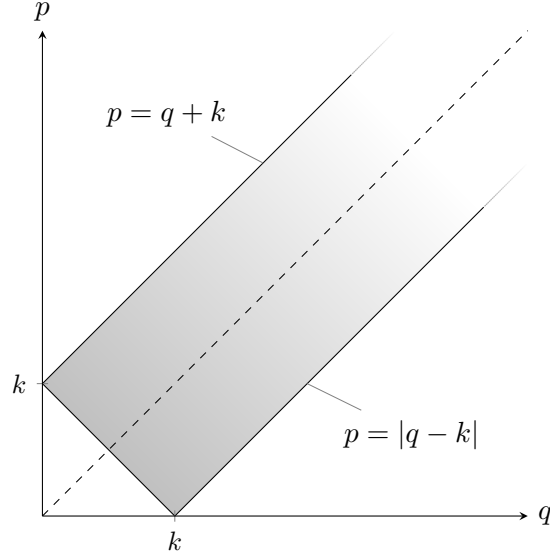


Figure 3.1 The integration domain of the integral in eq. (2.106) is enclosed by the curves given by $p = q + k$ and $p = |q - k|$. Since the integral is symmetric under exchange of p and q , only the lower half has to be evaluated. A “point of instability” is located at $(q, p) = (k, 0)$.

and similarly for $\tilde{B}(k, q, s)$. Also, we now rename s to p (not to be confused with the magnitude of \mathbf{p}). We notice that the integrand is invariant under exchange of q and p , since exchanging the last two indices and the last two arguments of either $\gamma_{abc}(\mathbf{k}, \mathbf{q}, \mathbf{p})$ or $B_{abc}(\mathbf{k}, \mathbf{q}, \mathbf{p})$ leaves these expressions invariant:

$$\gamma_{abc}(\mathbf{k}, \mathbf{q}, \mathbf{p}) = \gamma_{acb}(\mathbf{k}, \mathbf{p}, \mathbf{q}) \quad (3.7)$$

$$B_{abc}(\mathbf{k}, \mathbf{q}, \mathbf{p}) = B_{acb}(\mathbf{k}, \mathbf{p}, \mathbf{q}) \quad (3.8)$$

This can easily be checked by recalling the definitions of these terms in eq. (2.100). Because the last two indices are contracted, all combinations are present. Due to this symmetry, we can account for half the integration domain (see fig. 3.1) by a factor of 2 such that eq. (2.106) is now

$$\begin{aligned} \partial_\eta P_{ab}(k, \eta) = & -\Omega_{ac}(k, \eta)P_{cb}(k, \eta) - \Omega_{bc}(k, \eta)P_{ac}(k, \eta) \\ & + e^\eta \frac{4\pi}{k} \int_{k/2}^{\infty} q \, dq \int_{|q-k|}^q p \, dp \left[\tilde{\gamma}_{acd}(k, q, p) \tilde{B}_{bcd}(k, q, p; \eta) \right. \\ & \left. + \tilde{B}_{acd}(k, q, p; \eta) \tilde{\gamma}_{bcd}(k, q, p) \right]. \end{aligned} \quad (3.9)$$

The bispectrum appears in eq. (2.106) inside an integral while in eq. (2.107) it does not, which makes it difficult to work with in a numerical context. But because we are not interested in the bispectrum, we can multiply eq. (2.107) by the appropriate vertex

function, integrate it over $d^3\mathbf{q}$ and define the quantity

$$I_{acd,bef}(k) \equiv \int_{k/2}^{\infty} dq q \int_{|q-k|}^q dp p \frac{1}{2} \left[\tilde{\gamma}_{acd}(k, q, p) \tilde{B}_{bef}(k, q, p) + (q \leftrightarrow p) \right]. \quad (3.10)$$

Now eqs. (2.106) and (2.107) can be written as

$$\partial_{\eta} P_{ab}(k) = -\Omega_{ac} P_{cb}(k) - \Omega_{bc} P_{ac}(k) + e^{\eta} \frac{4\pi}{k} [I_{acd,bcd}(k) + I_{bcd,acd}(k)] \quad (3.11)$$

$$\begin{aligned} \partial_{\eta} I_{acd,bef}(k) = & -\Omega_{bg} I_{acd,gef}(k) - \Omega_{eg} I_{acd,bgf}(k) \\ & - \Omega_{fg} I_{acd,beg}(k) + 2e^{\eta} A_{acd,bef}(k), \end{aligned} \quad (3.12)$$

such that the integral of \tilde{B} is taking part in the differential equations as opposed to B . Dependence on η is understood. However, this only works when assuming that Ω_{ab} does not depend on k (which makes it possible to write it in front of the integral contained in $I_{acd,beg}$). Otherwise, terms like

$$\int_{k/2}^{\infty} dq q \hat{\Omega}(q) \int_{|q-k|}^q dp p \frac{1}{2} \left[\tilde{\gamma}_{acd}(k, q, p) \tilde{B}_{efg}(k, q, p) + (q \leftrightarrow p) \right], \quad (3.13)$$

$$\int_{k/2}^{\infty} dq q \int_{|q-k|}^q dp p \hat{\Omega}(p) \frac{1}{2} \left[\tilde{\gamma}_{acd}(k, q, p) \tilde{B}_{efg}(k, q, p) + (q \leftrightarrow p) \right], \quad (3.14)$$

would appear in eq. (3.12), which would have to be converted somehow to take the form of a multiple of eq. (3.10). A k -dependent Ω is needed for example in a massive neutrino cosmology, which is the topic of Lesgourgues et al. (2009). There, the authors also applied the TRG formalism and found that neglecting the k -dependence in eq. (2.107) while keeping it in eq. (2.106) causes only a negligible error.

For eq. (3.12), we define the nonlinear source term as

$$\begin{aligned} A_{acd,bef}(k) \equiv & \int_{k/2}^{\infty} dq q \int_{|q-k|}^q dp p \frac{1}{2} \left\{ \tilde{\gamma}_{acd}(k, q, p) [\tilde{\gamma}_{bgh}(k, q, p) P_{ge}(q) P_{hf}(p) + \right. \\ & \left. \tilde{\gamma}_{egh}(q, p, k) P_{gf}(p) P_{hb}(k) + \tilde{\gamma}_{fgh}(p, k, q) P_{gb}(k) P_{he}(q)] + (q \leftrightarrow p) \right\}. \end{aligned} \quad (3.15)$$

Finally, we define the integrands F and K (indices may be dropped occasionally) implicitly via

$$A_{acd,bef}(k) \equiv \int_{k/2}^{\infty} dq q \int_{|q-k|}^q dp p (F_{acd,bef}(k, q, p) + F_{acd,bef}(k, p, q)) \quad (3.16)$$

$$\equiv \int_{k/2}^{\infty} dq K_{acd,bef}(k, q). \quad (3.17)$$

3.2 Symmetries

The integral in eq. (3.16) is the major hurdle in obtaining the nonlinear power spectrum and a central part in the remainder of this chapter. It is imperative to understand the symmetries of this object. Since there are six indices that take the values 1 or 2, we have 64 different components. From the definition of the $\tilde{\gamma}_{acd}$ it is clear that only the

terms $A_{acd,bef}$ with $(acd) \in \{(112), (121), (222)\}$ are nonzero. That leaves $3 \times 2^3 = 24$ terms.

We will now show that $A_{acd,bef} = A_{adc,bfe}$. Due to eq. (3.7), the first vertex function appearing in eq. (3.15), $\tilde{\gamma}_{acd}(k, q, p)$, is equal to $\tilde{\gamma}_{adc}(k, p, q)$, i.e. q and p are switched. The same thing is true for the first term in the brackets if we switch the indices e and f . We just have to relabel the dummy indices g and h and switch the last two indices of the vertex function. Doing the same to the last two terms in the brackets turns them into each other while again switching q and p . Thus, $F_{acd,bef}(k, q, p) = F_{adc,bfe}(k, p, q)$, and because of the existence of the $(q \leftrightarrow p)$ term, we are done.

From this symmetry, it follows that all indices with $(acd) = (121)$ are accounted for by some index with $(acd) = (112)$, i.e. $2^3 = 8$ more components are not independent. Also, we can associate $(222, 121) \rightarrow (222, 112)$ and $(222, 221) \rightarrow (222, 212)$, hereby removing 2 more components.

Thus, the 14 remaining nonzero, independent components of $A_{acd,bef}$ and at the same time the components of I that take part in the equation are (in lexicographical order):

$$(acd, bef) \in \{(112, 111), (112, 112), (112, 121), (112, 122), \quad (3.18)$$

$$(112, 211), (112, 212), (112, 221), (112, 222), (222, 111), \quad (3.19)$$

$$(222, 112), (222, 122), (222, 211), (222, 212), (222, 222)\}. \quad (3.20)$$

This result has been verified by explicitly computing all components individually with MATHEMATICA. We may relabel the surviving indices in ascending order by a single index i , running from 1 to 14, because it is useful when using a programming language. For example, instead of $K_{acd,bef}(k, q)$, we may write $K_i(k, q)$ to keep consistency between source code and documentation. Note that in the original paper (Pietroni, 2008), the claim that there are only 12 independent components of $A_{acd,bef}$ is erroneously made.¹

These symmetries extend to $I_{acd,bef}$ as well, using the same arguments. Even the evolution equation for $\partial_\eta I_{acd,bef}$ is the same as the one for $\partial_\eta I_{adc,bfe}$, as you can see by inspecting eq. (3.12).

Concluding, the TRG equations are a system of 17 coupled, homogeneous, ordinary differential equations of first order and second degree, three for the power spectrum and 14 for the bispectrum. The function A has 14 independent components and depends on both k and η . For a typical interpolation of the power spectrum sampled at 1000 values of k and 100 time steps, we are looking at 1.4×10^6 evaluations of A . Even though we can assume $A \approx 0$ for low wave numbers (in the linear regime), this is a large amount of evaluations and clearly represents the bottle neck of the integration of the original differential equation.

3.3 Tabulated functions

When working on numerical implementations, functions that have no analytical expression (such as the power spectrum, which comes from simulations or numerical code) are typically represented by a set of tuples. Using the power spectrum $P(k)$ as an example,

¹Massimo Pietroni (Personal communications, May 2014)

we will call the set

$$\mathcal{T}(P, k_1, \dots, k_{N_k}) \equiv \{(k_i, P(k_i)) | 1 \leq i \leq N_k\} \quad (3.21)$$

the sampling points of $P(k)$, i.e. \mathcal{T} is defined as the tabulated version of a function after choosing N_k values for k . The way the k_i are spaced out is in general crucial in order to have an accurate representation of the function.

Conversely, we often want to convert a tabulated function into a continuous one. This is done via interpolation. Usually, piecewise defined polynomials are used in this case, such as splines. For details on splines, see appendix B.

3.4 Momentum integration

Regarding eq. (3.16), it is suggested in the appendix of Pietroni (2008) to perform a rotation in (q, p) space by $\pi/2$, such that the integration limits are independent of the variables, making the integral more suitable for numerical evaluation:

$$A_{acd,bef}(k) = \int_{k/\sqrt{2}}^{\infty} dx \int_0^{k/\sqrt{2}} dy \frac{x^2 - y^2}{2} \left[F \left(k, \frac{x+y}{\sqrt{2}}, \frac{x-y}{\sqrt{2}} \right) + (y \leftrightarrow -y) \right]. \quad (3.22)$$

However, leaving the integral as it is, i.e. *not* performing the rotation of the integral domain, enables us to perform the p - and the q -integration separately². By doing so, the p -integration of the third term in eq. (3.15) can be carried out analytically, as the power spectrum only depends on k and q and $\tilde{\gamma}_{fgh}(p, k, q)$ are rational functions of p . After fully expanding the integrand and integrating all terms that do not depend on $P_{ab}(p)$ (using MATHEMATICA), we can see that all the remaining p -dependent terms are of the form

$$C(q, k) p^m P_{ab}(p) \quad (3.23)$$

where $C(q, k)$ is some factor independent of p and m is an integer, which can take the values $-3, -1, 1, 3, 5$. We call these the “ m -moments” and define

$$P_{ab}^{(m)}(p) \equiv p^m P_{ab}(p). \quad (3.24)$$

By finding an antiderivative of the moments as an interpolation function, we effectively reduce the integral in eq. (3.16) to a one-dimensional one. In total there are only 13 different moments occurring in eq. (3.16):

$$\begin{aligned} &M_{-1,11}, M_{1,11}, M_{3,11}, M_{5,11}, \\ &M_{-3,12}, M_{-1,12}, M_{1,12}, M_{3,12}, M_{5,12}, \\ &M_{-3,22}, M_{-1,22}, M_{1,22}, M_{3,22}. \end{aligned} \quad (3.25)$$

Occasionally these may be labeled with a single index corresponding to this particular order. The moments and their antiderivatives need to be computed for every time step, as they contain the time dependent power spectrum. This is the crucial difference to other implementations of the TRG equations that leads to a substantial decrease in runtime.

²Riccardo Catena (Personal communications, October 2011)

3.4.1 p -Integration

We can now carry out the p -integration first, but there are several difficulties one has to take care of when doing so. They will now be discussed in detail.

Point of instability

Note that the power spectrum for small k is proportional to k^{n_s} , with $n_s \approx 1$, such that the (-3) -moment goes roughly as p^{-2} for small p . As a consequence, the antiderivative for this moment diverges at $p = 0$, and so does the integral over the $1/p$ term in the analytically integrable part. Since every integration includes a point where $q = k$ such that the lower limit of the p -integral vanishes, this is potentially a source of numerical instabilities. This becomes especially important for large k , because the integrands are less and less well behaved there, as we can see in fig. 3.4. We know that the other terms in the integrand of the q -integral must lead to the cancellation of diverging terms, because otherwise this solution to the TRG equations would be meaningless. And indeed in fig. 3.4 we see that the q -integrand stays finite, but this is not obvious in numerical terms. We avoid this issue by carefully choosing the sampling points needed to evaluate the q -integral such that we always have $q \neq k$ (see section 3.4.2).

Catastrophic cancellation

Additionally, we replace the p -integral by a difference of the antiderivative at possibly (in the case of $k \ll q$) nearby values. Differences of large, almost equal numbers are prone to lead to a loss of significance due to the finite working precision of computers³. In short, the expression $(x+a) - (x+b)$ evaluates to something other than the expression $a-b$ if x is sufficiently larger than a or b , as is demonstrated by the following C code:

```

1 double x = 1e20, a = 4, b = 2;
2 printf("%g, %g\n", (x+a)-(x+b), a-b);
3 /* => 0, 2 */

```

Thus, associativity does not necessarily hold in finite precision arithmetic. This becomes a problem when the constants of integration are chosen arbitrarily.

Suppose we integrate a function $f(x)$ that goes to zero sufficiently fast as $x \rightarrow \infty$. Then we can always write any antiderivative of $f(x)$ as $F(x) + C$ where $F(x \rightarrow \infty) \rightarrow 0$. Unless we choose $C = 0$, we will have to evaluate expressions of the form

$$(F(x_1) + C) - (F(x_2) + C). \quad (3.26)$$

When $x_1 \approx x_2$ and $F(x_1)$ as well as $F(x_2)$ become much smaller than C , that expression will evaluate to zero, whereas $F(x_1) - F(x_2)$ may very well evaluate to a nonzero value. Note that this effect can also occur as $x \rightarrow 0$ (in log-space it will look like an asymptote), and hopefully only one of the two cases will occur. See fig. 3.3 for an illustration.

Thus, the constant of integration should be chosen such that nonzero asymptotes are avoided. Specifically, when computing

$$M_{m,ab}(p_0, p) \equiv \int_{p_0}^p \tilde{p}^m P_{ab}(\tilde{p}) d\tilde{p}, \quad (3.27)$$

³Typically “double” precision.

we choose $p_0 = 0$ if $m > 0$ and $p_0 = \infty$ if $m < 0$. This choice works, as we can see in fig. 3.2. The only nonzero asymptotes are for $m = -1$ and $k \rightarrow 0$ and for $m = 1$ and $k \rightarrow \infty$. These are unavoidable, since for these two curves we always get a nonzero asymptote, either on the IR end or the UV end. Fortunately, those two curves approach values which do not exceed 10^5 , which is small enough such that catastrophic cancellation is not a problem.

Of course, when actually integrating the function given by a finite table of values of the power spectrum, we replace ∞ with the largest k -value in the table. However, zero is kept as such, even though tables of the power spectrum typically do not include $k = 0$. But since $P(k) \propto k^{n_s}$ for small k , we can simply extrapolate analytically. This way, we can be certain that all asymptotes in the integral of the moments approach 0 and no catastrophic cancellation should take place.

In the integrand of A , more nontrivial cancellations are involved, even in between moments of different degree. These cancellations are not exact when using a computer, and even small remaining terms can be amplified due to terms containing up to seventh powers of q . To avoid the numerical noise this can introduce, it is important to only interpolate the three power spectra and keep the prefactor p^m instead of interpolating the moments themselves. Let us denote by $\mathcal{S}\{f(x)\}$ the spline representation of the function $f(x)$ with a given set of sampling points to phrase this more formally: Because of the nature of spline interpolation, $p^m \mathcal{S}\{P_{ab}(p)\}$ does not necessarily equal $\mathcal{S}\{p^m P_{ab}(p)\}$ between sampling points. These differences may be small, but they are big enough to cause issues in our case.

Spline integration in log-space

When integrating the moments, it proved to be most efficient and accurate to simply analytically integrate the splines piecewise and in log-space. Numerical integrators or ODE solvers could not produce the required accuracy reliably and are much slower. Since cubic splines⁴ are just polynomials of third degree, we need to integrate

$$\int dx e^{x(m+1)} [y_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3] \quad (3.28)$$

for the i -th spline interval with its coefficients y_i , b_i , c_i and d_i . Here, $x \equiv \log(p)$ with $x_i < x < x_{i+1}$ represents the wavenumber in log-space, such that we get one factor of $\exp(x)$ from the change of variables, and m factors of $\exp(x)$ from the moment. Performing the integral in eq. (3.28) for $m \neq -1$ yields

$$e^{\check{m}x} [\check{y}_i + \check{b}_i(x - x_i) + \check{c}_i(x - x_i)^2 + \check{d}_i(x - x_i)^3] \quad (3.29)$$

where the coefficients are

$$\check{y}_i = \check{m}^{-4}(-\check{m}^2 b_i + 2\check{m} c_i - 6d_i + \check{m}^3 y_i) \quad (3.30)$$

$$\check{b}_i = \check{m}^{-3}(\check{m}^2 b_i - 2\check{m} c_i + 6d_i) \quad (3.31)$$

$$\check{c}_i = \check{m}^{-2}(\check{m} c_i - 3d_i) \quad (3.32)$$

$$\check{d}_i = \check{m}^{-1} d_i \quad (3.33)$$

⁴See appendix B for more details.

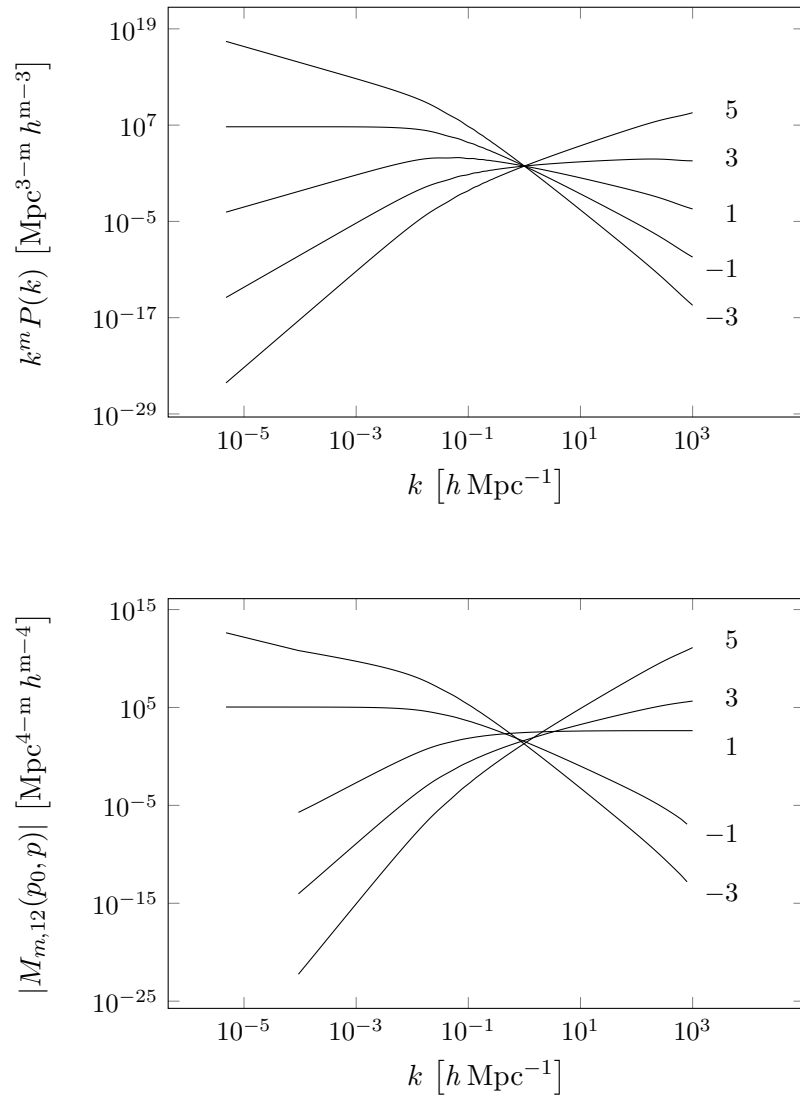


Figure 3.2 The five relevant moments $k^m P(k)$ for $m = -3, -1, 1, 3, 5$ (top) and their antiderivatives (bottom). Their behavior for small and large values of k determines the constant of integration. The curves are labeled with their respective value for m .

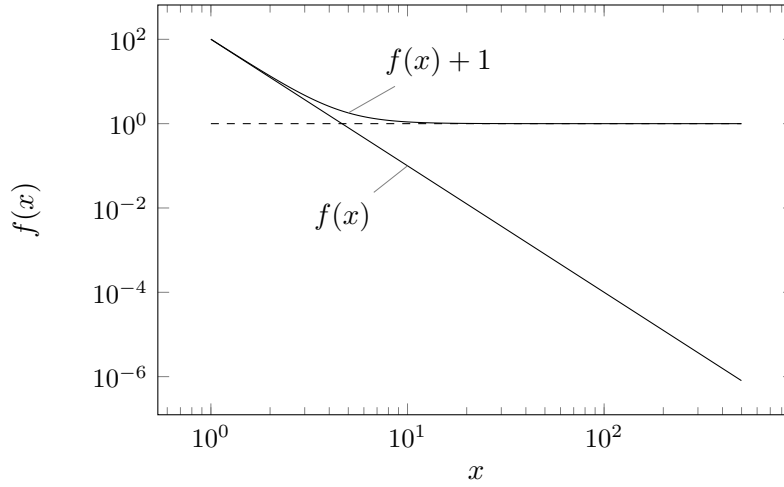


Figure 3.3 Catastrophic cancellation can be avoided if the asymptotes of the integrals approach 0. As an example, this figure illustrates the problem with a function $f(x)$ that quickly approaches zero as $x \rightarrow \infty$. In a finite precision arithmetic, as $f(x)$ approaches the smallest representable number, $f(x) + 1$ will evaluate to 1 and all precision is lost, even though the information could have been easily preserved by simply using $f(x)$. Thus, the choice of the constant of integration is essential when dealing with finite precision.

$$\tilde{m} = m + 1. \quad (3.34)$$

For $m = -1$ we get

$$y_i x + \frac{1}{2} b_i (x - x_i)^2 + \frac{1}{3} c_i (x - x_i)^3 + \frac{1}{4} d_i (x - x_i)^4. \quad (3.35)$$

The integration constant for each interval needs to be chosen such that the splines are continuous at the sampling points. The integration for the first interval is arbitrary in principle, but as discussed earlier we need to choose it such that the antiderivative is zero at the first sampling point if $m > 0$ or zero at the last sampling point if $m < 0$ in order to decrease numerical noise.

3.4.2 q -Integration

After we integrated the moments numerically, the integrand is now independent of p and we can carry out the q integration in eq. (3.17). This can be done straightforwardly with the trapezoidal rule. We argue below why this is preferable to a standard integrator. Since all 14 components of the integrand behave similarly (see fig. 3.4), we will use the same set of sampling points for all of them. When plotting the integrands for different k , we can observe that they are all well-behaved for low k , but start to look increasingly similar to the function $f(q) = 1/(k - q)$ as k enters the nonlinear regime. Note, however, that the function is continuous at all times and does not have a zero near $q = k$, it just changes its sign very rapidly. Since a lot of cancellation happens in this region, it is important that we make sure to get an accurate estimate of the integral there.

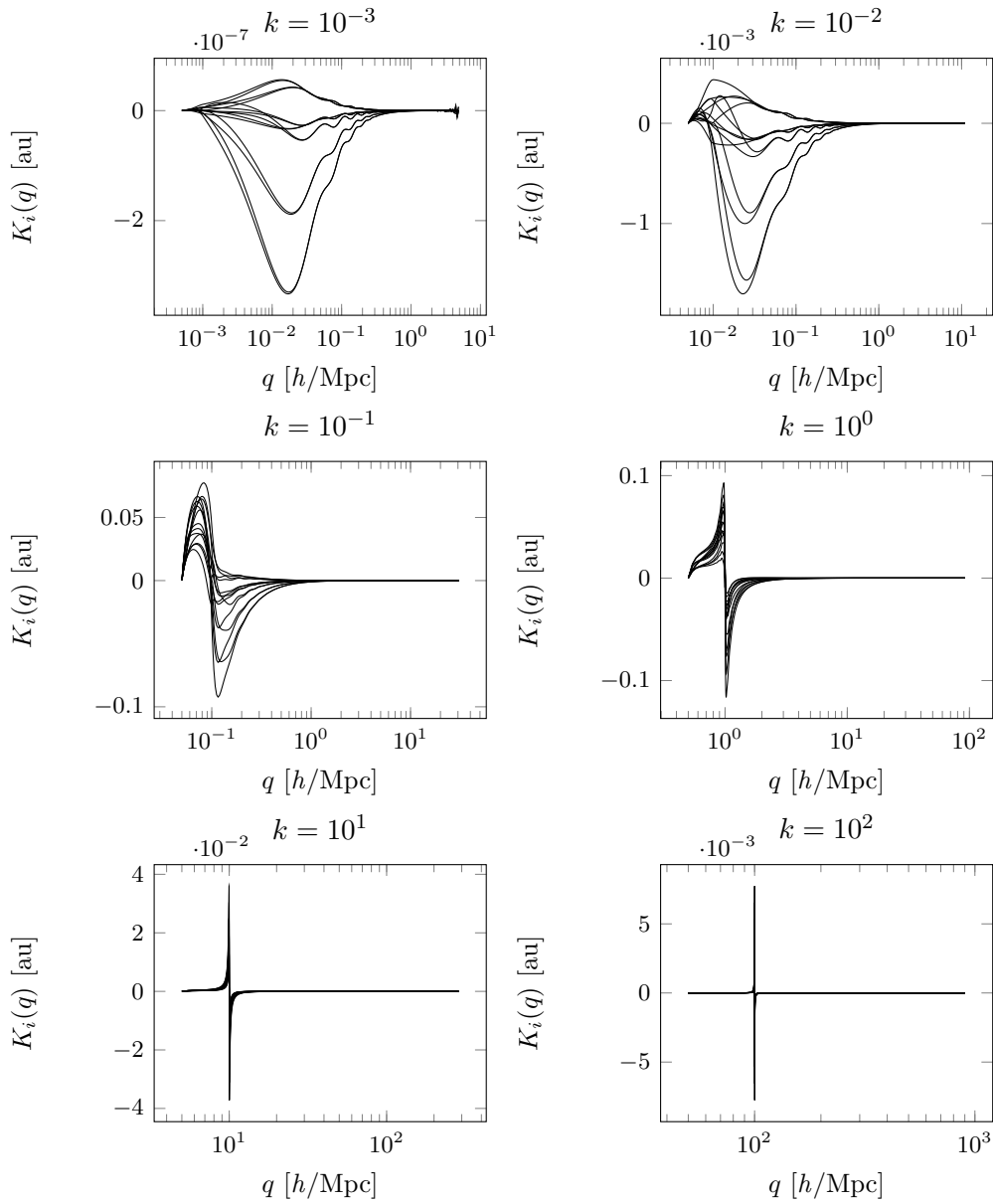


Figure 3.4 All 14 integrands (in arbitrary units) after carrying out the p -integration at different values for k . As k increases, the integrands approach a function with a vertical asymptote and a singularity at k .

The upper limit of the integral is technically infinity. Due to the nature of numerical integration, we need to define a cutoff that acts as the upper limit of the integral, which we call k_{\max} . We discuss in section 4.3.3 how to determine a suitable value for the cutoff.

Custom sampling

One might consider to use a general purpose integrator like the globally adaptive algorithm `dqage` from QUADPACK, which provides control over the desired accuracy and can deal with oscillating integrands, singularities, etc. but makes it impossible to reuse the sampling points without heavy modification of existing code. Since they are also designed to suit a very general class of integrands, they are too slow for our purposes.

Instead, we use the knowledge we have about the integrand to restrict the sampling scheme to two classes depending on some threshold k_{th} : For $k < k_{\text{th}}$ we use logarithmically equidistant points between $k/2$ and k_{\max} , and for $k > k_{\text{th}}$ we first translate the function to the left by transforming $q \rightarrow q - k$. Then we use logarithmically equidistant points from 0 to $-k/2$ and from 0 to $k_{\max} - k$ before undoing the transform (see fig. 3.5). We refer to these sampling schemes as “asymmetric scheme” and “symmetric scheme” respectively. This way the sampling density is much higher where the curvature of the integrand is large, yielding a good approximation of the real value of the integral.

Of course, we need to determine a suitable value for the threshold k_{th} and, in case of the second scheme, the distance from k to the sampling point closest to k .

This strategy produces a significant speed-up in runtime but has the disadvantage of not being able to give an estimate on the error. However, the error must decrease with increasing the sampling rate and we find that the result of the integration is stable for the chosen values.

IR and UV limit

Due to the nature of interpolating functions, they can only cover a finite interval. When trying to evaluate the function outside of its domain, we have to use extrapolation. In our case, this happens in both directions, which we need to deal with differently.

We will discuss the IR limit first. When performing the q integration, there is a point where $q \approx k$, at which we will have to evaluate the antiderivatives at $|q - k| \approx 0$, which is not covered by the interpolating function. Here we can make use of the fact that the power spectrum for low k is a power law:

$$P_{ab}(k) \propto k^{n_s}, \quad \text{if } k \ll 1 \text{ Mpc}^{-1}, \quad (3.36)$$

which means that the moments can be integrated analytically in this regime. We realize that the moments for $m < -1$ have a singularity at $p = 0$, however it is a nonessential singularity and becomes a removable singularity when taking all terms in the integrand into consideration.

The UV limit is more troublesome. Since the upper limit of the integral in eq. (3.16) is infinity, we would have to evaluate the antiderivatives outside of their domain. There are a few options on how to account for this: Setting the power spectrum to zero, keeping it constant after the last defined sampling point, or extrapolating it smoothly via a power law or exponential law. We discuss differences of these variants in section 5.1.1.

Another issue are numerical instabilities, which cause the UV end of the power

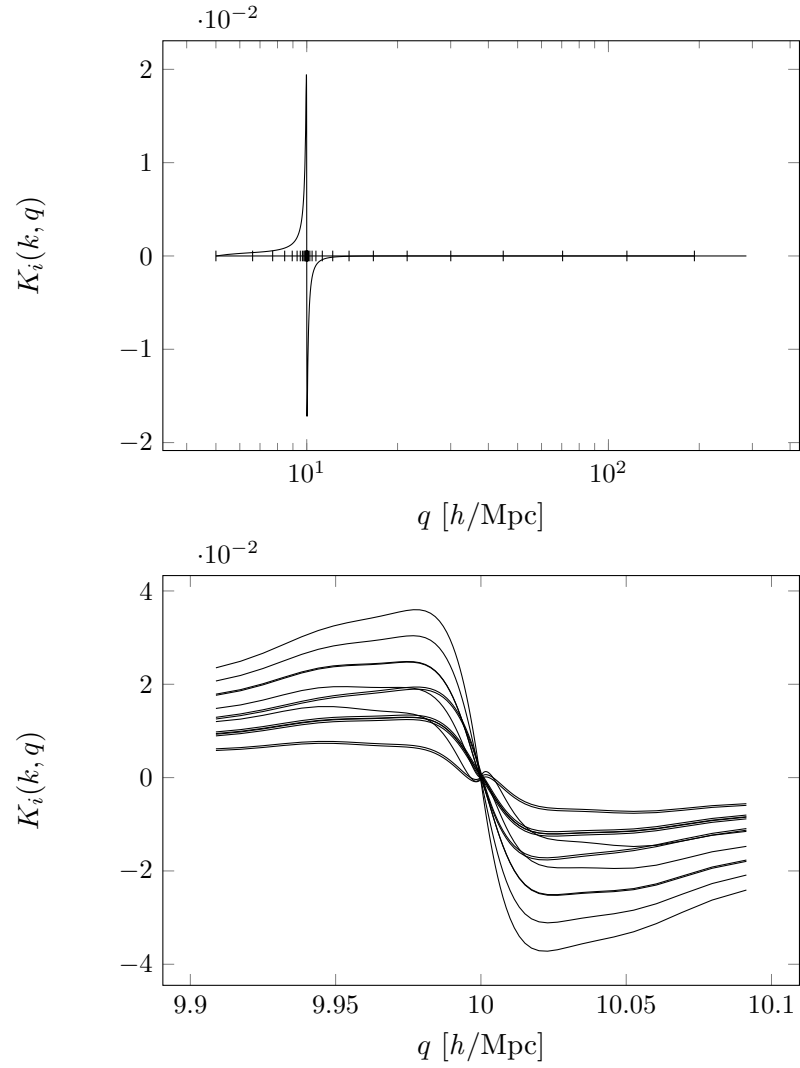


Figure 3.5 Here, the integrands for $k = 10h/\text{Mpc}$ are displayed. The sampling points, which are distributed symmetrically around $q = k$, are drawn in the top panel together with one representative integrand at $\eta = 0$. The bottom panel shows all 14 integrands zoomed in around a region close to k . The integrands are well resolved with this sampling method.

spectrum to oscillate. As soon as these oscillations are introduced, they quickly amplify and cause one of the three power spectra to take on negative values. More specifically, some values of the tabulated power spectrum become `-inf` or `nan`, causing the program to crash.

Even when these oscillations are absent, as the power spectrum evolves, the P_{22} component takes on very small values, approaching zero as the time evolution progresses. These numerical artifacts are the manifestation of a limitation of the fluid equation themselves, which neglect the vorticity of the velocity field and the velocity dispersion as has been discussed in sections 2.2 and 2.3.

To work around these issues, we decrease the maximum wavenumber k_{\max} at which the power spectrum is defined as the time η increases. Audren and Lesgourgues (2011) did it for CLASS by employing a technique they call “double escape”, which essentially consists of throwing out the last four points in each time step. In our implementation, however, it is handled differently, since the number and positions of the sampling points are given by the user. After all, if the algorithm works in a stable way for one set of sampling points, it should yield the same result if the density of the sampling points is doubled, for instance, which would not be the case if we would only throw out the last two sampling points in each time step. We will describe in section 4.3.3 how we determine k_{\max} at each time step.

3.5 Time integration

The TRG equations are differential equations where the power spectrum and bispectrum are the dependent variables and η the independent variable, forming a coupled system of 17 equations. However, since the power spectrum and bispectrum are also scale dependent, we have to interpret the equations either as an initial value problem where the initial value is a vector-valued function of k with 17 components, or, since this function is realistically given in tabulated form, as a vector with $17 \times N_k$ components, where N_k is the number of sampling points. The latter interpretation is much more suited for numerical computation due to the fact that standard ODE integrators assume that the dependent variable is a vector in \mathbb{R}^n .

We choose a fourth order Runge-Kutta code from SLATEC⁵ to integrate the TRG equations.

3.6 Initial conditions

The matter power spectrum at sufficiently early times can be assumed to be linear. It can be obtained from CAMB or CLASS. The velocity power spectrum P_{22} and the cross power spectrum P_{12} can be derived from the matter power spectrum. To do so, we note that the power spectrum can also be written as (Amendola and Tsujikawa, 2010, p. 31)

$$P_{ab}(\mathbf{k}, \eta) = |\varphi_a(\mathbf{k}, \eta)\varphi_b(\mathbf{k}, \eta)|, \quad (3.37)$$

⁵The SLATEC *Common Mathematical Library* is a collection of hundreds of Fortran 77 routines in the public domain and freely available at netlib.org. It contains popular sub-packages such as QUADPACK or BLAS.

such that we get

$$P_{12}(\mathbf{k}, \eta) = \sqrt{P_{11}(\mathbf{k}, \eta)} |\varphi_2(\mathbf{k}, \eta)|, \quad (3.38)$$

$$P_{22}(\mathbf{k}, \eta) = |\varphi_2(\mathbf{k}, \eta)|^2. \quad (3.39)$$

These are not the physical power spectra, but defined as in eq. (2.97), so they become

$$P_{22}(k, z) = \left(\frac{\partial \sqrt{P_{11}(k, z)}}{\partial \eta} \right)^2, \quad (3.40)$$

$$P_{12}(k, z) = \sqrt{P_{11}(k, z)} \frac{\partial \sqrt{P_{11}(k, z)}}{\partial \eta}. \quad (3.41)$$

Here we used the continuity equation (2.92), which reads in the linear regime

$$\theta = \nabla \mathbf{v} = -\frac{\partial \delta_m}{\partial \tau}, \quad (3.42)$$

as well as the fact that

$$\frac{\partial}{\partial \tau} = \mathcal{H} \frac{\partial}{\partial \eta}. \quad (3.43)$$

In order to compute the velocity power spectrum P_{22} and the cross power spectrum P_{12} , we note that the z -dependence at early times is entirely captured by the growth function $G(z)$ (called $D_+(a)$ in the previous chapter) via $P_{11}(k, z) \propto G^2(z)$, which allows us to write:

$$\frac{\partial \sqrt{P_{11}(k, z)}}{\partial \eta} = \frac{\sqrt{P_{11}(k, z)}}{G(z)} \frac{\partial G(z)}{\partial \eta} = \sqrt{P_{11}(k, z)} \frac{\partial \log G(z)}{\partial \eta}. \quad (3.44)$$

Thus, knowing the value of the derivative of $\log G(z)$ with respect to η at the initial redshift is necessary to get the initial power spectra. It can be determined from the background cosmology. The final equations then become

$$P_{22}(k, z) = P_{11}(k, z) f(z)^2, \quad (3.45)$$

$$P_{12}(k, z) = P_{11}(k, z) f(z), \quad (3.46)$$

where we have introduced the growth rate

$$f(z) \equiv \left. \frac{\partial \log G(a)}{\partial \log a} \right|_{a=1/(1+z)}. \quad (3.47)$$

Note that in Λ CDM the approximation

$$f(z) \approx \Omega_m(z)^\gamma \quad (3.48)$$

with the growth index $\gamma \approx 6/11$ holds (Wang and Steinhardt, 1998). Typically, z_{ini} is well in the matter dominated era and we have $f(z) \approx 1$, such that we can simply set $P_{11} = P_{12} = P_{22}$ at z_{ini} .

The bispectrum vanishes if one chooses Gaussian initial conditions. However, in one-loop perturbation theory (Bernardeau et al., 2002), the initial bispectrum does

not vanish, but it can be approximated by setting $I(0, k) = 2A(0, k)$ (Audren and Lesgourgues, 2011). If the results are to be compared to the ones from one-loop SPT, these are the initial conditions one should use.

Chapter 4

Implementation

The previously discussed techniques for solving the TRG equations have been implemented in Fortran. This chapter covers the technical details of our research.

The code has been developed in the Fortran2003 standard. Special care has been taken to ensure it compiles with both the latest Intel compiler as well as the latest stable GNU compiler on a Linux system.

4.1 Structure

To separate different aspects of the code from each other, it has been divided into five modules described in table 4.1. What follows is a step-by-step overview of how the program operates.

One-dimensional arrays are indicated by a trailing underscore.

4.1.1 Initialization

A run begins by calling the function

```
1 function time_evolution(  
2     eta_,  
3     k_,  
4     ps_,  
5     O_eta_,  
6     OmegaBulk,  
7     options_,  
8     growth_out,  
9     O_k_) result(ps_out)
```

which starts with calling `init_ode`. This routine is responsible for allocating arrays which will contain the growth function and a list of all k values along with their logarithm, set the initial values of various variables, set the options, as well as interpolate the background functions.

Here, `eta_` is an array of type double (like all other arguments) whose first value equals η_{ini} and the other values mark the points in time at which the nonlinear power spectrum will be computed. Thus, in the most simple case, we could have `eta_ = {0, 4.610}`, corresponding to $z_{ini} = 100$ and $z_{fin} = 0$. The argument `k_` is an array of

| Module | Description |
|-------------------------|--|
| <code>options</code> | Defines the various options and their default values |
| <code>tools</code> | Provides a set of routines that are nonspecific to the TRG algorithm, such as reading 2D arrays from disk, binary search, spline interpolation, etc. |
| <code>background</code> | Handles the interpolation of the background functions Ω_{ab} |
| <code>nonlinear</code> | Computes the quantity $A_{acd,bef}(\eta, k)$ |
| <code>ode</code> | Solves the TRG equations |

Table 4.1 Modules of the TRGFAST program and their description.

length N_k containing the wave numbers at which the initial linear power spectrum `ps_` (also an array of length N_k) at $z = 0$ is given.

The array `0_eta_` contains values of η in ascending order at which the background functions are defined. They are contained in `OmegaBulk`, which is a three-dimensional array. The element `OmegaBulk(a,b,c)` corresponds to $\Omega_{2c}(\eta_a, k_b)$. If Ω does not depend on k , then the second dimension of the array has size 1 and the optional argument `0_k_` is to be omitted. The next argument, `options_`, is an array of length 21 that contains all options as outlined in section 4.1.7.

The result, the power spectrum at the requested redshifts as well as the growth function, will be stored in the two arrays `ps_out` and `growth_out`, respectively. The element `ps_out(a,b,c)` corresponds to $P_{b-1}(\eta_c, k_a)$ if $b > 1$ and k_a else, where η_c is equivalent to one of the requested redshifts. Since the nonlinear power spectrum is not defined at all wave numbers where the initial power spectrum was defined, these elements are marked as irrelevant by making them negative, so remember to discard them. Similarly, `growth_out(i,1)` is a list of redshifts and `growth_out(i,2)` the corresponding values of the growth function.

4.1.2 Linear run to estimate growth function

After all necessary arrays have been allocated and properly initialized, the TRG algorithm is run while setting $A(k) = 0$, i.e. we evolve the power spectrum linearly. The linear equations are invariant under scaling of the power spectrum as can be seen in eqs. (3.11) and (3.12), where scaling factors cancel out when $A(k)$ vanishes. Thus, it is sufficient to know the evolution of the power spectrum at a single wavenumber, since the linear growth function

$$G(z)^2 = P(k, z)/P(k, 0) \quad (4.1)$$

is scale-independent. Subsequently, we pass only 17 equations to the ODE integrator (of which 14 stay constant at zero at all times) and extract the values of $P_{11}(\eta, k)$ at different η to obtain the growth function. Then the linear power spectrum at z_{ini} is computed by scaling the power spectrum from today using the growth function.

4.1.3 Nonlinear run

The actual run is straight forward. The initial conditions for the bispectrum are set and the ODE integrator is called to produce solutions at the requested redshifts. The integrator calls `f_ode` at each time step to get the value of the derivatives of the

dependent variables, which in turn computes $A_{acd,bef}$ at that point in time. This is the point where we parallelize the computation, since this is the inner most loop that carries out a computation that depends on results from earlier computations (namely the power spectrum at another time step), i.e. loops surrounding this one need to be executed in series. The relevant lines of code are:

```

1  !$OMP PARALLEL DO
2      do ki=ki_linear,ki_max
3          A_all_k(ki,:) = A(k_(ki))
4      end do
5  !$OMP END PARALLEL DO

```

This loop precomputes all needed values of A (between a scale considered as linear and the maximum wave number) and writes them into an array.

At the beginning of each time step, before the function A can be called, the nonlinear module needs to be initialized by passing the current power spectrum. This means needed arrays are allocated, the new k_{\max} is determined, and the moments are integrated. Now A can be called for arbitrary wave numbers until the cleanup routine is called at the end of the time step, which deallocates all arrays in the nonlinear module.

4.1.4 Spline interpolation

Just like the power spectrum is always shown in a log-log-plot for convenience, it is also more convenient to perform the spline interpolation in log-space, i.e. instead of $P(k)$ we interpolate $\log P(k)$. Otherwise, numerical artifacts in the form of oscillations will manifest in the interpolated power spectrum. They can be suppressed by choosing sampling points with higher density, but it is more efficient to simply perform the interpolation in log-space. For an illustration, see fig. 4.1. This inevitably leads to a lot of calls to the `exp` and `log` functions, slowing down the computation, which can be seen when profiling the code (see section 4.2).

4.1.5 Spline evaluation

When evaluating a spline at value x , most of the time is typically spent finding the right interval i such that $x_i < x < x_{i+1}$, especially if the spline interpolates many sampling points. To find the interval, we use a binary search algorithm, which executes in $\mathcal{O}(\log n)$ time, where n is the number of sampling points, given that they are ordered (Press et al., 2007). And in fact, for the majority of the time our code is executing a binary search (see section 4.2). That is not surprising, considering it consists mostly of integrating interpolated functions, which leads to a large number of evaluations of splines.

To speed this up even more, we take advantage of the fact that we often need to evaluate similar splines in the same interval. For instance, if we need the value of all three power spectra at a particular value k , we pass the index obtained from the first call of the power spectrum function to subsequent calls, which can then skip the binary search altogether. This method has yielded a significant speed up.

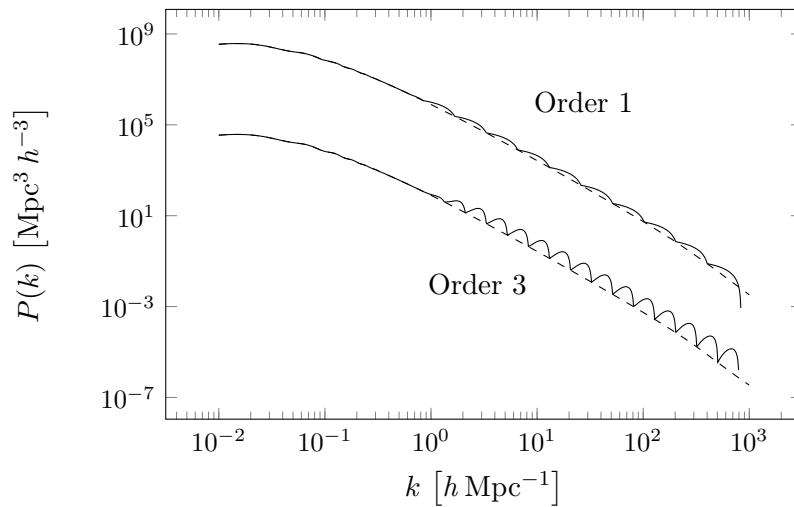


Figure 4.1 Unwanted oscillations can appear in the power spectrum if the interpolation is not performed in log-space. The solid curves corresponds to regular interpolation, while the dashed curves corresponds to interpolation performed in log-space, yielding a much better result. Reducing the interpolation order to one (here shown for a power spectrum with a different normalization) yields a better yet still insufficient result. Here, the power spectrum for $k < 1h/\text{Mpc}$ has been sampled at a higher frequency to account for the BAO wiggles, suppressing the interpolation oscillations as well. For this figure, the MATHEMATICA function `Interpolation` has been used, which does not necessarily employ splines but instead some other piecewise polynomial interpolation, but the issue remains.

| Parameter | Default value | Meaning |
|------------|---------------|--|
| p_1 | False | Use default values |
| p_2 | 0.001 | The end of the linear scale k_{lin} |
| p_3^* | 0.01 | The threshold wave number k_{th} |
| p_4 | 200 | Number of sampling points N_s |
| p_5 | 0.01 | Tolerance for the RK solver ϵ_{RK} |
| p_6 | 0.001 | Threshold for the RK solver Δ_{RK} |
| p_7 | 30 | Number of time steps N_t |
| p_8 | False | Linear mode |
| p_9 | 100 | Desired number of sampling points for $G(z)$ |
| p_{10} | 2 | Verbosity level |
| p_{11} | 0 | Output tag |
| p_{12} | False | Power spectrum requested at all times |
| p_{13} | False | Raw power spectrum requested at all times |
| p_{14} | False | Integrated moments requested at all times |
| p_{15} | False | Antiderivative differences requested at all times |
| p_{16} | False | $A(k)$ requested at all times |
| p_{17} | False | $K(k, q)$ requested at all times |
| p_{18}^* | 1.0 | Scale factor for $A(k, \eta)$ |
| p_{19}^* | 10.0 | Curvature tolerance |
| p_{20}^* | 20.0 | Slope limit |
| p_{21} | False | Force index symmetry |
| p_{22}^* | 0 | Extrapolation method |
| p_{23}^* | 10^{-4} | Offset |

Table 4.2 All parameters required for the time evolution of the nonlinear power spectrum are listed here. Starred parameters are merely for debugging purposes and should not be altered by the user.

4.1.6 Output

Finally, the contents of the array containing the three nonlinear power spectra at several redshifts up to z_{fin} can be extracted from the TRG module. Intermediate results may have been written to the disk during the run if the corresponding options have been set accordingly.

4.1.7 Options

The Fortran module takes a series of parameters and outputs the three nonlinear power spectra at the desired redshift. The initial conditions (as discussed in section 3.6) need to be provided in tabulated form, as a 2D array in double precision of dimension $2 \times N_k$, where N_k is the number of sampling points. Additionally, we need the background functions Ω_{21}, Ω_{22} in tabulated form. An overview of the remaining options can be found in table 4.2. Examples of how to use the module via Fortran, C, and MATHEMATICA have been included in the package.

Note that we use variables of type “double” for all parameters. Integers are simply cast onto doubles and booleans correspond to a negative value for “False” and a non-

negative value for “True”. The reason for this practice is that it allows us to simply pass all options as an array of doubles. When interfacing our code with code written a different language (such as C or MATHEMATICA), using more complicated structures can get complicated. For instance, MATHEMATICA does not have a `struct` statement (which would otherwise be more appropriate) like C does, but arrays are universal. We will now describe the function of each parameter in detail.

p_1 : Use default values If this parameter is set to “True”, TRGFAST will ignore all other values and set all other parameters to their predefined default value.

p_2 : k_{lin} This value defines the boundary between the linear and the nonlinear regime. If $k < k_{\text{lin}}$, the value of $A(k)$ will be assumed to be zero and not be computed in order to save computation time. Increasing this parameter decreases the run time, but if it is set too high the final power spectrum will have a discontinuity at k_{lin} , because nonlinear corrections were not applied to a region that is already nonlinear.

p_3 : Threshold k_{th} at which we change the sampling scheme See section 3.4.2. Only for debugging purposes.

p_4 : Number of sampling points N_s The number of sampling points for $K(k, q)$ to use for the trapezoidal rule. Can decrease numerical noise while increasing computation time.

p_5 : RK Tolerance ϵ_{RK} This parameter specifies the tolerance (i.e. the local relative error) for the RK integrator. Must be smaller than 10^{-2} . For more details, see `rksuite.doc`¹.

p_6 : RK Threshold Δ_{RK} This parameter specifies the threshold for the RK integrator. Must be greater than 0. For more details, see `rksuite.doc`.

p_7 : Number of time steps N_t This parameter defines the number of time steps used for the nonadaptive RK integrator. If this value is less than 0, the adaptive RK integrator will be used (see section 4.3.2).

p_8 : Number of sampling points for $G(z)$ TRGFAST also outputs the linear growth function between z_{ini} and z_{fin} . This parameter defines at how many redshifts the value of $G(z)$ will be computed.

p_9 : Linear mode If this parameter is set to “True”, the algorithm will be run with $A(k) = 0$, i.e. the time evolution is entirely linear.

p_{10} : Verbosity An integer defining the level of verbosity. 0: Only show errors. 1: Show errors and warnings. 2: Also show additional useful information. 3: Show everything (for debugging, not recommended).

¹<http://www.netlib.org/ode/rksuite/rksuite.doc>

p_{11} : Output tag If intermediate results have been requested, they are written to a file located in `/tmp/trgfast-xxxxx`, where `xxxxx` stands for a five digit integer defined by p_{10} .

p_{12} : Power spectrum requested at all times If this parameter is set to “True”, the three-component power spectrum will be written to disk at each time step. Mainly for debugging purposes, it will not only be defined at the relevant wave numbers, but also between the sampling points and beyond, i.e. extrapolated in both directions.

p_{13} : Raw power spectrum requested at all times If this parameter is set to “True”, the three-component power spectrum will be written to disk at each time step. In this case, only values defined at the sampling points are written.

p_{14} : Integrated moments requested at all times If this parameter is set to “True”, the integrated moments

$$\int dp p^m P_{ab}(p) \quad (4.2)$$

at each time step will be written to disk.

p_{15} : Antiderivative differences requested at all times If this parameter is set to “True”, the functions

$$\int_{|q-k|}^k dp p^m P_{ab}(p) \quad (4.3)$$

at each time step will be written to disk for various values of k (mainly for debugging purposes).

p_{16} $A(k)$ requested If this parameter is set to “True”, the source term $A(k)$ will be written to file at each time step.

p_{17} : $K(k, q)$ requested If this parameter is set to “True”, the integrand $K(k, q)$ will be written to file at each time step for various values of k . Note that this will generate a lot of files.

x_{18} : Scale factor for $A(k, \eta)$ A real number which is multiplied with the function $A(k, \eta)$. Only for debugging purposes.

x_{19} : Curvature tolerance This parameter defines σ_c (cf. section 4.3.3).

x_{20} : Slope limit This parameter defines s_{22} (cf. section 4.3.3).

x_{21} : Force index symmetry If this parameter is set to “True”, we will assume the additional symmetry as outlined in section 5.2.3 exists in order for the result to be compatible with Copter.

x_{22} : **Extrapolation method** This parameter defines which extrapolation method to use. 0: Power law. 1: Zero. 2: Exponential. 3: Constant. (See section 5.1.1.)

x_{23} : **Offset** This parameter specifies the distance between the first sampling point in either direction from the near-singularity when performing the symmetric sampling of the integrand $K(q, k)$.

4.2 Profiling and memory leaks

This section covers the technical part of the optimization of the code.

To identify bottlenecks in the implementation and to maximize efficiency, we employ the profiler GNU `gprof` (version 2.22) (Graham et al., 1982). A profiler enables us to see how much time a program spends on executing a particular function. Let us examine an excerpt from the output we get from applying `gprof` to the binary produced by the Intel compiler.

Flat profile:

Each sample counts as 0.01 seconds.

| % | cumulative | self | seconds | calls | name |
|-------|------------|------|----------|-------|--------------------------------------|
| 18.12 | 6.40 | 6.40 | | | exp.L |
| 17.92 | 12.73 | 6.33 | 17068227 | | nonlinear_mp_eval_integrated_moment_ |
| 15.98 | 18.38 | 5.65 | 19710481 | | tools_mp_eval_int_splines_ |
| 12.54 | 22.81 | 4.43 | | | pow.L |
| 8.80 | 25.92 | 3.11 | | | log.L |
| 7.20 | 28.46 | 2.55 | 2884397 | | tools_mp_bsearch_ |
| 4.67 | 30.11 | 1.65 | 688487 | | nonlinear_mp_kdq_ |
| 3.47 | 31.34 | 1.23 | 4121421 | | tools_mp_eval_interpolation_ |
| 1.57 | 31.89 | 0.56 | 2987494 | | tools_mp_lsearch_ |

Shown here are the functions in which the program spends most of the time. We learn that 18% of the time are spent computing exponentials. This is due to the fact that we interpolate the power spectrum and other quantities in log-space, but when evaluating the integral these values need to be converted back, leading to a large number of calls of `exp` and `log`.

Next on the list is the function that evaluates an integrated moment at wavenumber q , which first takes care of the UV and IR limit, and if the value for q is within the range of where the splines are defined, calls the function that evaluates an integrated spline. When evaluating a spline at q , the only challenging part is to find in which interval the value of q lies. As mentioned before, this is done by performing a binary search, here called `bsearch`, which takes up the most time after `exp`, `pow`, and `log`. During the development of TRGFAST, `bsearch` was much higher up the list and could be identified as a bottleneck. By applying the methods outlined in section 4.1.5, this issue was remedied. The rest of the functions take up a negligible amount of time (less than 1%).

Another potential technical issue are memory leaks. Memory leaks typically happen when a program allocates memory for its own use and fails to properly free it upon termination of the program. This can lead to a reduction of available memory on the system and cause serious issues, especially if the same code is executed again and again. Since this may be the case when using TRGFAST for MCMC or the Fisher matrix, we checked for memory leaks using the freely available tool Valgrind² (Seward and Nethercote, 2005).

Valgrind was called using the arguments

```
valgrind --tool=memcheck --leak-check=yes ./driver
```

and parallel computing was disabled in TRGFAST.

When using the GNU Fortran compiler, Valgrind gave the following output.

```
HEAP SUMMARY:
  in use at exit: 0 bytes in 0 blocks
  total heap usage: 1,645,286 allocs, 1,645,286 frees,
    3,921,522,222 bytes allocated
```

```
All heap blocks were freed -- no leaks are possible
```

We see that memory management has been sufficiently careful such that no memory leaks appear.

When using the Intel compiler, Valgrind gave a similar result.

```
HEAP SUMMARY:
  in use at exit: 264 bytes in 2 blocks
  total heap usage: 347,132 allocs, 347,130 frees, 535,400,609
    bytes allocated
```

```
LEAK SUMMARY:
  definitely lost: 0 bytes in 0 blocks
  indirectly lost: 0 bytes in 0 blocks
  possibly lost: 0 bytes in 0 blocks
  still reachable: 264 bytes in 2 blocks
```

Due to differences in how the compilers translate the source code to machine code (possibly when optimizing certain routines), we see a small region in memory marked as “still reachable.” This, however, does not necessarily pose a problem, as these blocks are not lost and can thus be freed by the system in principle. Should this prove to be a problem in the future, one can always fall back to the GNU Fortran compiler.

4.3 Algorithms

4.3.1 Spline interpolation

Finding the coefficients of a cubic spline interpolation function is as simple as finding the solution to a set of linear equations (see appendix B). We use the routine contained in `spline.f` (available at <http://www.netlib.org>), written in Fortran 77 by George Elmer Forsythe, Michael A. Malcolm, and Cleve B. Moler.

²<http://www.valgrind.org>

4.3.2 ODE integrator

The code supports the use of two different integrators. One is an adaptive fourth order Runge-Kutta integrator written in Fortran 77 by R.W. Brankin, I. Gladwell, and L.F. Shampine in 1991. `rksuite.f` (available at <http://www.netlib.org>). Here, the step size is variable and adapts in order to keep the local relative error below a specified tolerance ϵ_{RK} .

The other one is also a fourth order Runge-Kutta, but with a fixed step size. The approximate solution for the ODE $\dot{y} = f(t, y)$ with initial value $y_0 = y(t_0)$ is obtained by the following algorithm (Press et al., 2007):

$$y_{n+1} = y_n + \frac{1}{6}h(k_1 + 2(k_2 + k_3) + k_4), \quad (4.4)$$

$$t_{n+1} = t_n + h, \quad (4.5)$$

where

$$k_1 = f(t_n, y_n), \quad (4.6)$$

$$k_2 = f\left(t_n + \frac{1}{2}h, y_n + \frac{h}{2}k_1\right), \quad (4.7)$$

$$k_3 = f\left(t_n + \frac{1}{2}h, y_n + \frac{h}{2}k_2\right), \quad (4.8)$$

$$k_4 = f(t_n + h, y_n + hk_3). \quad (4.9)$$

Usually, the integrator with adaptive step size is preferable. However, in our case the numerical instabilities at the UV end of the power spectrum cause the adaptive algorithm to decrease the step size to very small values, resulting in a large number of evaluations of $A(k)$. The computational effort may be wasted if that part of the power spectrum is discarded later anyway. On top of that, the integrator seems to be very sensitive with regards to the choice of parameters, compiler or compiler flags. Small changes in one parameter or even just switching the compiler or setting different compiler flags can lead to a very large increase in the number of evaluations of $A(k)$, i.e. the step size for the time evolution in the adaptive algorithm becomes extremely small. Thus, we opt to use the nonadaptive RK integrator. As we will see later, the accuracy of the result will not suffer due to this choice.

4.3.3 Cutoff method

As mentioned in section 3.4.2, we will describe the criteria to determine k_{max} at each time step in this section. There are two causes for instabilities in the time evolution which ultimately lead to a crash of the program. One is numerical inaccuracies, which cause the UV end of the power spectrum to oscillate and take on unrealistic values, such as a sudden spike at last sampling point k_{max} . Another is the inherent limit of the fluid equations, which becomes apparent when $P_{22}(k)$ becomes negative because the approximations made when deriving the fluid equations break down.

First we tackle the numerical noise. Since we want to avoid uneven behavior such as oscillations and kinks in the power spectrum, we discard sampling points at the UV end one by one until the curvature between the last three sampling points of all three components varies only by a limited amount. This happens at each time step.

Conveniently, the coefficient of the quadratic term a_2 (cf. eq. (B.1)) in the splines gives a good measure of the curvature. In particular, we demand that the standard deviation of the last two values for a_2 of all three components of the power spectrum does not exceed a certain limit σ_c .

The other issue of one of the power spectrum components becoming negative can be avoided by imposing that the magnitude of the slope at the UV end of $P_{22}(k)$ stays below another limit, which we will call s_{22} . The optimal values for σ_c and s_{22} will be determined in section 5.1.6.

During the time evolution, we then simply set

$$\partial_\eta P_{ab}(k) = 0 \quad (4.10)$$

and

$$\partial_\eta I_{acd,bef}(k) = 0 \quad (4.11)$$

if $k > k_{\max}$, which takes these scales effectively out of the equation. This cutoff method appears to lead to a stable time evolution without discarding too much information. At $\eta = \eta_{\text{fin}}$, we typically have $k_{\max} \approx 0.8h/\text{Mpc}$. A comprehensive analysis can be found in chapter 5.

4.4 Other languages

Because the Fortran module was designed as a library, it is easily usable from within related programming languages. We will describe how to do that for some of the most popular ones.

4.4.1 C/C++

Both the GNU and the Intel Fortran compiler have been designed to be interoperable with their respective C compiler. The Intel Fortran compiler is even interoperable with the GNU C compiler. This means that within certain limitations we can call Fortran functions from C and the other way round – but we will only do the former. Not all data structures can be passed between the two, but for our purposes it is sufficient to pass pointers to arrays of doubles. An example of how to use the Fortran module from within a C program has been included in the code base. Thus, the routines could be used easily from within future versions of both CAMB (Lewis et al., 2000) or CLASS (Blas et al., 2011). For details, we refer to the GNU Fortran Manual³ and the Intel® Fortran Compiler XE 13.1 User and Reference Guides⁴. After writing a wrapper library that provides an interface between the Fortran module and C code, it is almost trivial to use it in C++ as well.

4.4.2 MATHEMATICA

After wrapping the Fortran module in a C program, it is almost straight forward to call it via MATHLINK from MATHEMATICA. MATHLINK is an interface for communication

³<http://gcc.gnu.org/onlinedocs/gfortran>

⁴<http://software.intel.com/sites/products/documentation/doclib/stdxe/2013/composerxe/compiler/fortran-mac>

between the MATHEMATICA kernel and (typically) a C program. This way, we have the convenience of MATHEMATICA at our disposal without sacrificing efficiency. For details, see Hahn (2011). Together with the source code of TRGFAST, we deliver a MATHEMATICA package that wraps the module. It provides the function

```
trgfast[ps, zini, zfin, Omega21, Omega22]
```

where the parameters have to be defined as follows.

ps The linear matter power spectrum at $z = 0$ in tabulated form as a list of pairs $\{k_i, P(k_i)\}$.

zini The initial redshift at which time evolution shall start.

zfin A redshift or a list of redshifts at which the nonlinear power spectrum is desired.

Omega21 The background function $\Omega_{21}(\eta)$. It needs to take one real numbers and must return a real number, i.e. `Omega21[eta]` must evaluate to a real number when `eta` is a real number.

Omega22 Same as above, but for $\Omega_{22}(\eta)$.

The other parameters corresponding to p_1 through p_{21} (see table 4.2 can be set using options, e.g. `Nq->250`. For details on how see all options and how to use the MATHEMATICA package in general, see the example notebook accompanying the package.

We can only pass pointers to variables or arrays of types like integer, double, etc. via the MATHLINK, not functions or pointers to functions. While it is possible to evaluate the background functions in the MATHEMATICA kernel, the overhead produced by the program communicating back and forth with the kernel to compute the values needed is significant. Thus, we simply pass the background functions in tabulated form down the MATHLINK.

The return value of the function is a list of two values. The first one is another list with each element containing one power spectrum at a given redshift in the form of

$$\{\{k_1, P_{11}(k_1), P_{12}(k_1), P_{22}(k_1)\}, \dots\}. \quad (4.12)$$

The second one is the growth function in the form of

$$\{\{z_1, G(z_1)\}, \dots\}. \quad (4.13)$$

4.4.3 Other languages

The ubiquity of C code allows interfacing with many other programming languages. Arguably one of the more popular languages used in cosmology is Python. For Python, there is the `ctypes` library, which provides the capability of calling functions written in and compiled by C.

Julia⁵ is a new high-level dynamic programming language designed for scientific applications developed at the Massachusetts Institute of Technology (Bezanson et al.,

⁵<http://julialang.org/>

2012). It is meant to be a free replacement for languages like Matlab, Python, or MATHEMATICA but promises to be comparable to C performance-wise (Lubin and Dunning, 2013) and is optimized to run in parallel. Julia can call functions from C or Fortran libraries directly, allowing simple use of TRGFAST.

An example of how to call TRGFAST from within Python or Julia may be included in a future version of TRGFAST.

Chapter 5

Accuracy analysis and comparison

This chapter will be quite technical as it covers a detailed analysis of the sensitivity of the code with respect to the individual parameters as well as a comparison to N -body simulations, CLASS, Copter, and Halofit. It serves to convince ourselves that the nonlinear power spectrum produced by the code is meaningful and not dominated by numerical noise.

5.1 Parameter space

Our algorithm contains parameters that influence the total runtime as well as the accuracy of the result. Their optimal value has to be determined empirically. As an indicator of the stability of the result with respect to changes in these parameters we will use the maximum wave number k_{\max} of where the resulting nonlinear power spectrum is defined and its value near the end of the computable scale, $P_0 \equiv P_{11}(0.6h/\text{Mpc})$. We will now discuss the influence of the individual parameters on the final result.

5.1.1 Extrapolation

When performing the q -integration, the power spectrum may be evaluated outside its range, in which case we need to extrapolate. The most obvious choices for a power spectrum sampled at wave numbers between k_{\min} (the first sampling point) and k_{\max} (the last sampling point) are:

1. Setting the power spectrum to zero:

$$P(k > k_{\max}) = 0. \quad (5.1)$$

2. Having a constant, continuous power spectrum:

$$P(k > k_{\max}) = P(k_{\max}). \quad (5.2)$$

3. Smoothly extrapolating with a power law:

$$P(k > k_{\max}) = ak^b, \quad (5.3)$$

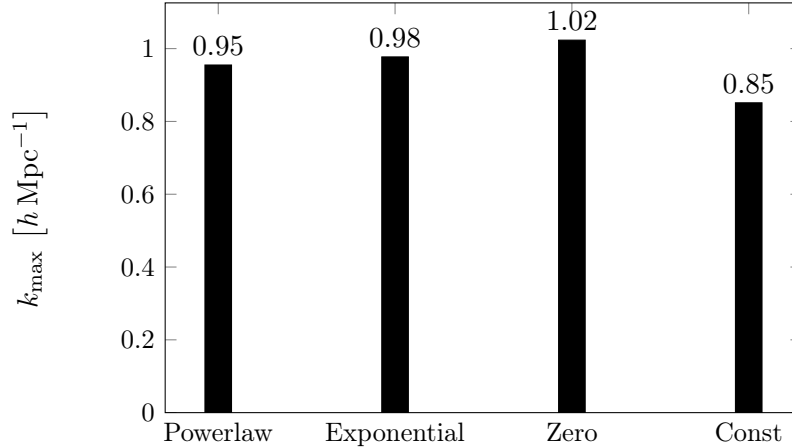


Figure 5.1 The maximum wavenumber for different extrapolation methods. Its value is only slightly sensitive to the extrapolation method used.

where a and b are determined by demanding continuity and differentiability at $k = k_{\max}$.

4. Extrapolating with a Gaussian (just like Copter):

$$P(k > k_{\max}) = a \exp\left(-4 \left(\frac{k - k_{\max}}{k_{\max}}\right)^2\right), \quad (5.4)$$

where a is determined by demanding continuity. Differentiability is neglected here.

In fig. 5.1, we can see that the extrapolation scheme is almost irrelevant.

5.1.2 Number of time steps

When using the nonadaptive RK integrator, we need to specify a step size, or equivalently, the number of steps N_t for the time evolution. Note that for every time step, the algorithm evaluates $A(k)$ four times. If we assume that the result for $N_t = 100$ is sufficiently close to the real nonlinear power spectrum, we observe in fig. 5.3 that already for $N_t = 20$, i.e. 80 evaluations of $A(k)$, we have agreement at the sub-0.1% level. Thus, we choose the default value to be $N_t = 30$.

5.1.3 Number of sampling points

When performing the q -integration, we use a simple trapezoidal rule as discussed in section 3.4.2. The final result is sensitive to the number of sampling points. Too few of them and the result will be inaccurate. Too many and the computation will take more time than necessary.

We find that $N_q \approx 150$ is sufficient to reach 1% accuracy (see figs. 5.4 and 5.5, but we can afford to increase the number of sampling points to $N_q = 200$).

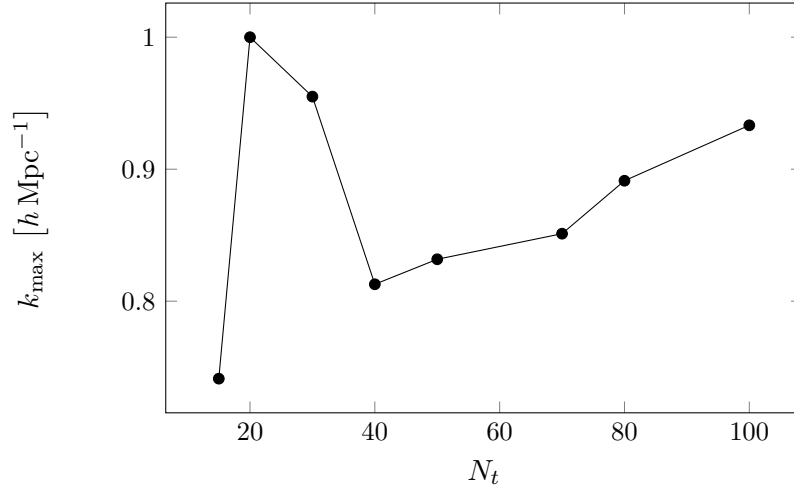


Figure 5.2 The maximum wave number depending on the number of time steps N_t . More time steps lead to a more stable evolution, leading to an increase in k_{\max} . But the value for k_{\max} is always acceptable.

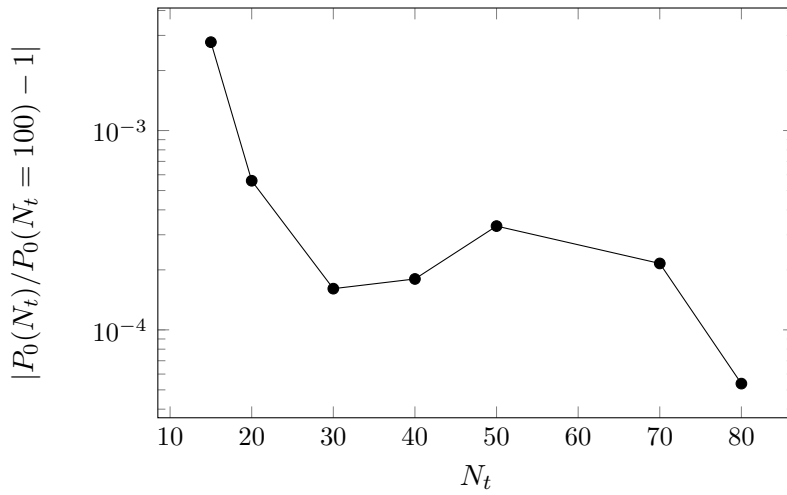


Figure 5.3 The amplitude of the power spectrum depending on the number of time steps N_t . Already 20 steps are enough to have a better accuracy than 0.1%, assuming the result for 100 steps is the exact one.

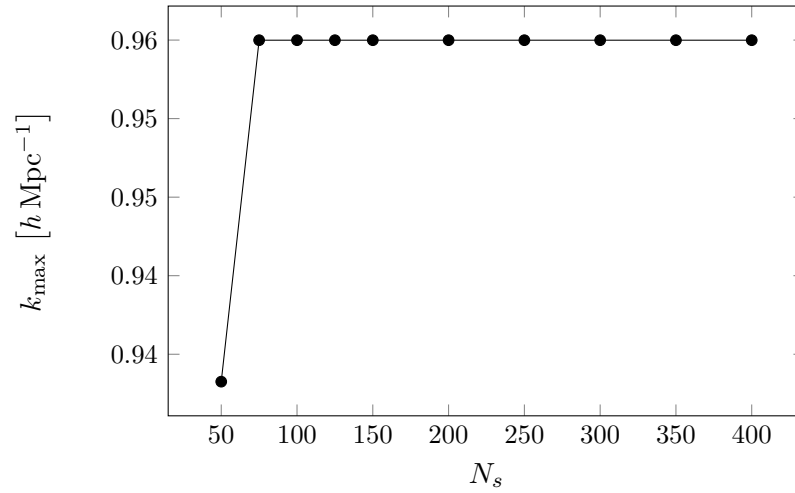


Figure 5.4 The maximum wave number depending on the number of sampling points N_s for the integrand $K(q)$. It is almost entirely independent of N_s .

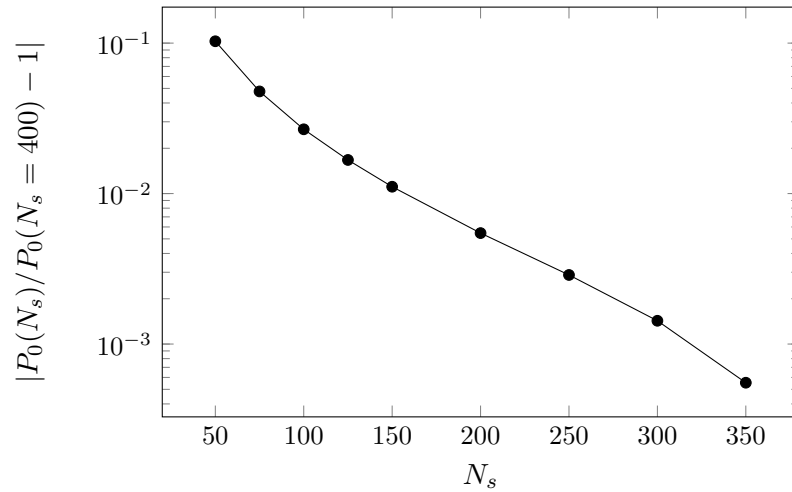


Figure 5.5 The amplitude of the power spectrum depending on the number of sampling points N_s . As expected, the more sampling points we use, the better the approximation becomes, assuming that the real result is sufficiently close to the ones using $N_s = 400$. Agreement at the 1% level is reached for $N_s \approx 150$.

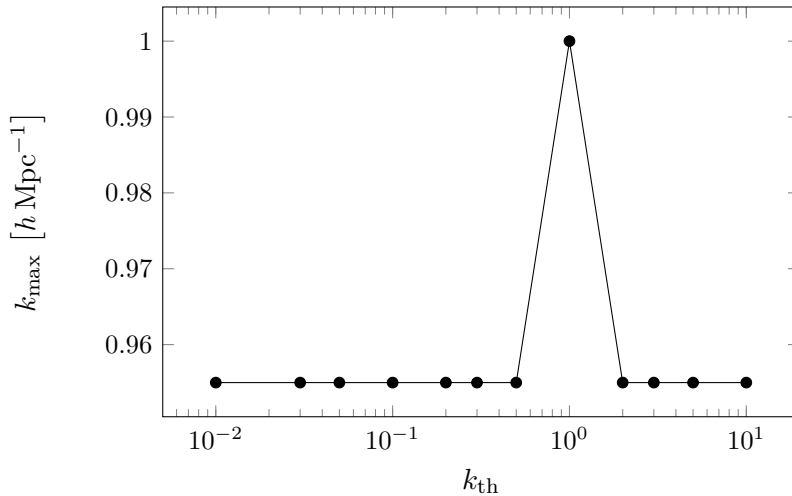


Figure 5.6 The maximum wave number depending on k_{th} . Again, varying this parameter does not affect k_{th} .

5.1.4 Sampling scheme

The parameter k_{th} defines the scale at which we switch the asymmetric sampling scheme to the symmetric one (see section 3.4.2). Unlike the cases for N_s and N_t , more (or less) is not necessarily better. We expect the optimal value for k_{th} to lie somewhere in the middle.

By looking at fig. 5.7 we first notice that the choice of the sampling scheme matters. While k_{max} stays mostly constant (see fig. 5.6), the amplitude of the power spectrum clearly varies as we increase k_{th} . If $k_{\text{th}} = k_{\text{lin}}$, the symmetric sampling scheme is always applied. On the other hand, if $k_{\text{th}} = k_{\text{max}}(\eta = 0)$, the asymmetric sampling scheme is always applied. Since the slope of the curve in fig. 5.7 is much steeper if we always applied the asymmetric scheme, it is clearly beneficial to at least switch to the symmetric scheme at some point.

What is unexpected is that the curve in fig. 5.7 plateaus for low values of k_{th} instead of somewhere in the middle. It implies that always using the asymmetric scheme works. One possible explanation is that the integrands $K(q)$ at small scales are fairly well behaved (see fig. 3.4) and the precise sampling scheme is not as important. We define the default value to be $0.01h/\text{Mpc}$, a scale where the result is barely sensitive to changes in this parameter.

5.1.5 Additional index symmetry

To compare our results with those from Copter (see section 5.2.3), we included an option that forces the last two indices of I and A to be symmetric, i.e. $I_{acd,b21} \equiv I_{acd,b12}$ and similarly for A . The default value here is “False”, since this symmetry does not hold in the TRG approximation. We want to point out the fact that if we do assume the symmetry, the result is closer to the one from N -body simulations, as can be seen in fig. 5.8. This is indeed expected, as including the symmetry corresponds to neglecting a certain type of diagram in the resummation scheme, making the approximation more

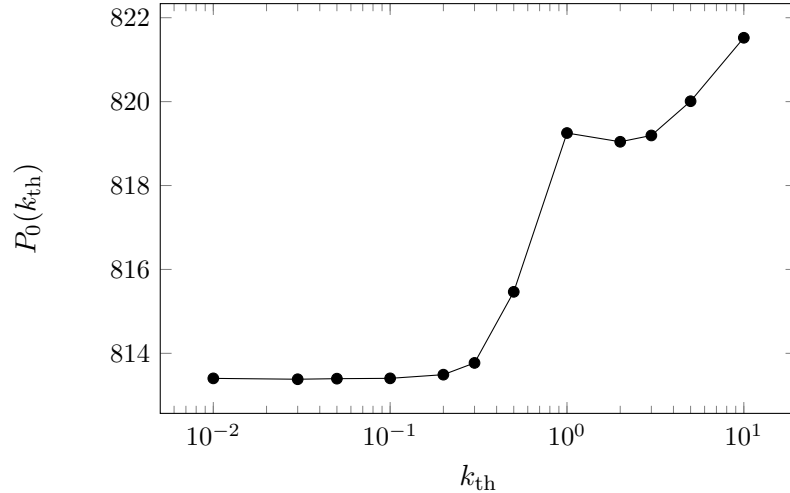


Figure 5.7 The amplitude of the power spectrum depending on k_{th} . The kink at $k \approx 1h/\text{Mpc}$ comes from the fact that the final nonlinear power spectrum is cut off at around this scale.

accurate¹. When we use the term “TRG-12”, we are referring to the Copter compatibility mode in TRGFAST, i.e. we do assume the additional symmetry and only have 12 TRG equations (the symmetry makes two equations redundant). “TRG-14” then stands for the CLASS compatibility mode, i.e. the symmetry is not assumed and we solve all 14 TRG equations.

5.1.6 Curvature tolerance and slope limit

As part of the cutoff method, we need to define an acceptable variation in the curvature of the power spectrum at the UV end (see section 4.3.3). We expect that the smaller we make σ_c and s_{22} , the more information will have been discarded at the end of a run, since the cutoff criterion is “stricter” in this case. Indeed, as can be seen in figs. 5.9 and 5.11, k_{max} increases as we loosen the curvature and slope restrictions. Figures 5.10 and 5.12 show a plateau around $\sigma_c = 10$ and $s_{22} = 20$, so these values will be the default for these parameters.

One could argue that the fact that we have a plateau in fig. 5.12 makes having a slope limit unnecessary, since we might as well set the slope limit to infinity without k_{max} being affected. But if we switch from the nonadaptive RK integrator to the adaptive one, having a finite slope limit becomes important again or else the algorithm will have trouble with convergence.

5.1.7 Sampling density

Ideally, the final result should not be sensitive to the sampling density of the linear power spectrum. As a test, we take a power spectrum sampled at 807 sampling points and form three new spectra before applying the nonlinear corrections to it: first we keep only every second sampling point, then only every third, and finally we keep only

¹Massimo Pietroni (Personal communications, May 2014)

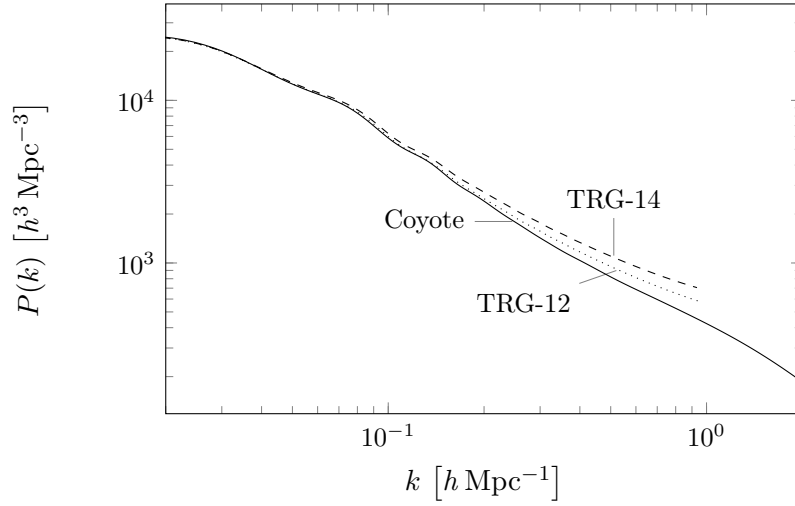


Figure 5.8 The dashed curve is the TRGFAST result without and the dotted curve is the `trgresult` with assuming the additional index symmetry. The latter is much closer to the N -body power spectrum (solid curve).

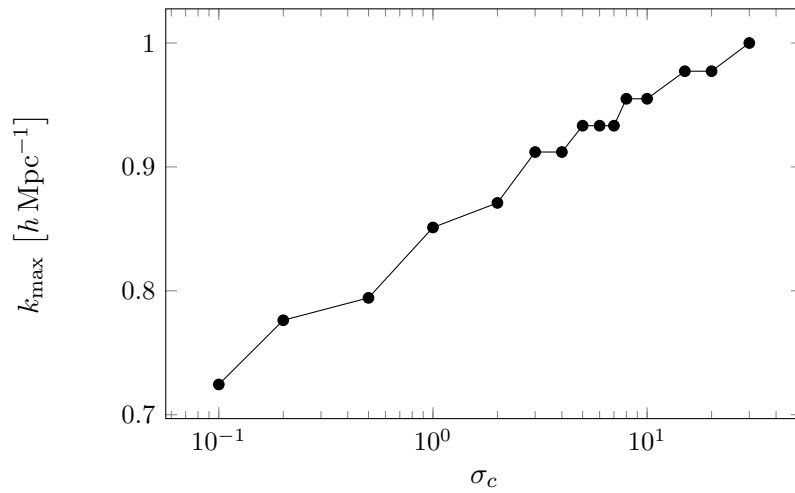


Figure 5.9 The maximum wave number as a function of the curvature tolerance σ_c . An increase in σ_c means a less strict cutoff procedure, thus an increase in k_{\max} .

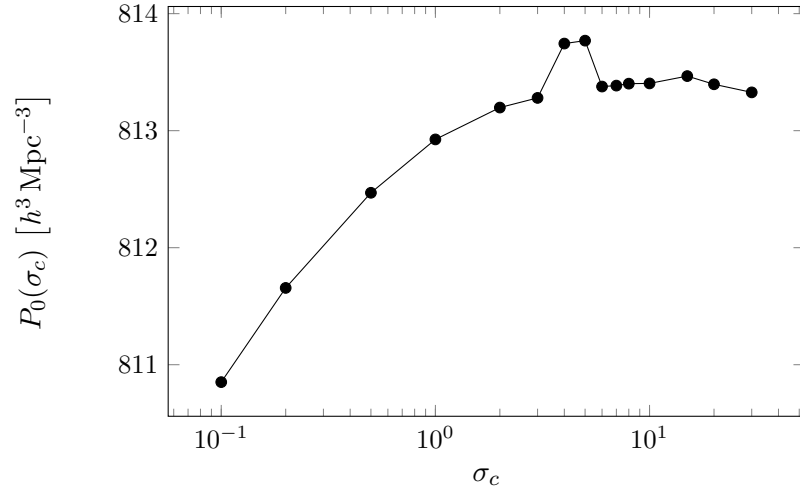


Figure 5.10 The amplitude of the power spectrum as a function of the curvature tolerance σ_c . There is a plateau near $\sigma_c = 10$. The bump near $\sigma_c = 5$ is unexplained.

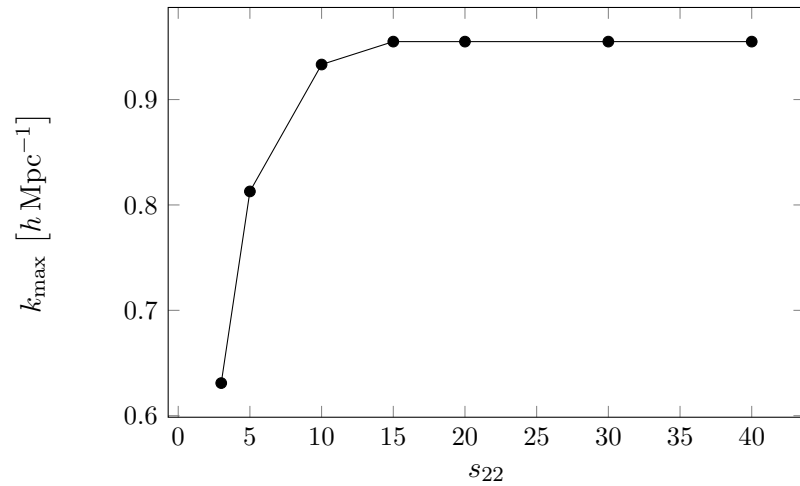


Figure 5.11 The maximum wave number as a function of the slope limit s_{22} . An increase in s_{22} means a less strict cutoff procedure, thus an increase in k_{\max} . At some point, the cutoff is entirely determined by σ_c and changing s_{22} has no effect anymore.

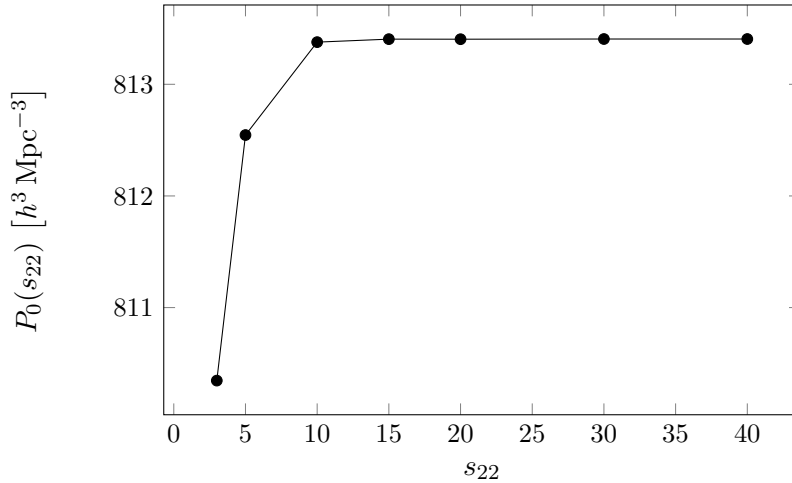


Figure 5.12 The amplitude of the power spectrum as a function of the slope limit s_{22} . Just like in fig. 5.11, the parameter does not affect the result after $s_{22} = 15$.

every fourth sampling point. Just as expected, fig. 5.13 shows that the cutoff method is so robust that k_{max} is barely effected by the sampling density. This is important, because the run time is roughly proportional to the number of sampling points, so one should aim to keep that number as low as possible.

5.2 Validating the results

Now that we confirmed that the algorithm is stable with regards to numerical parameters, we make sure that the result is also correct. With the parameters determined in the previous section, we obtain the power spectra shown in fig. 5.14. We are now ready to compare the spectra from TRGFAST to a number of other sources.

5.2.1 Overall comparison

Nonlinear corrections can be obtained from a number of sources. In this section we present a direct comparison of some popular codes as well as our TRG code. We will consider N -body simulations to be “correct”, i.e. we assume they equal what we would get from observations. In the next section, we will perform a more detailed comparison of the individual codes. Besides our own TRG code, we investigate the following sources:

- CLASS² (v2.0) by Audren and Lesgourgues (2011); Blas et al. (2011), where the same algorithm has been implemented in C, albeit with the 2D integral. However, the according module has been removed in v2.1 and is planned to be replaced in a later version.

²<http://class-code.net>

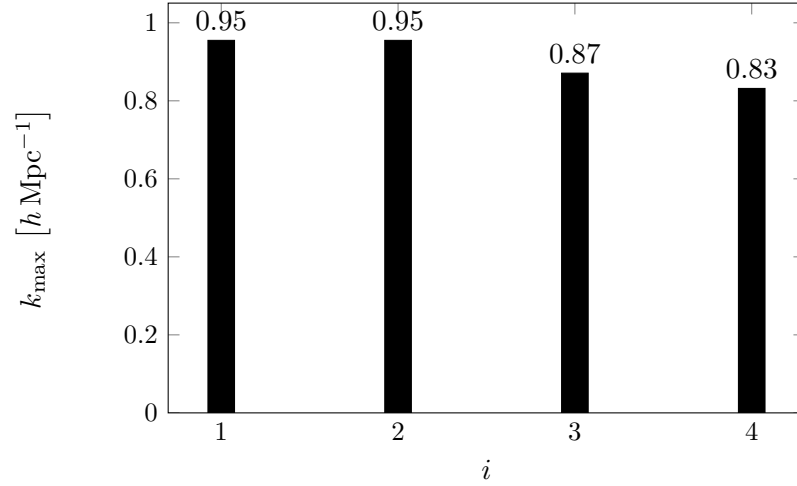


Figure 5.13 The maximum wavenumber barely depends on the sampling density. The bar chart shows the cases where only every i th sampling point was used.

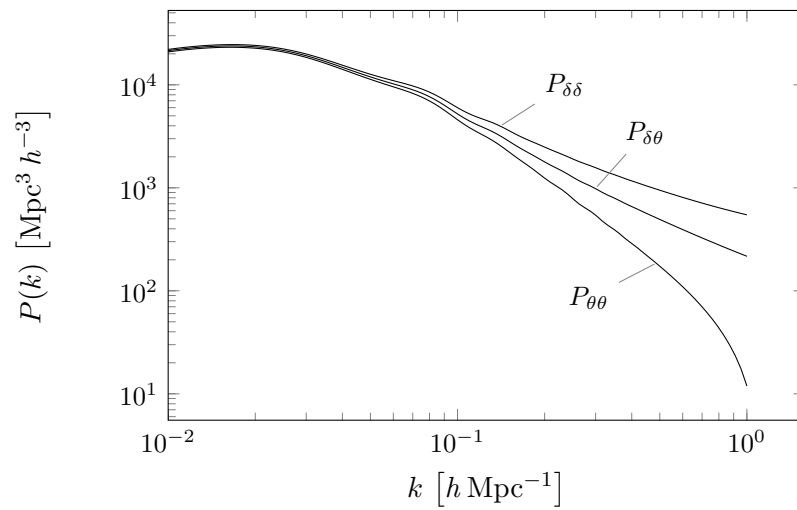


Figure 5.14 The power spectrum at $z = 0$ computed by TRGFAST, from top to bottom: $P_{\delta\delta}$, $P_{\delta\theta}$, $P_{\theta\theta}$.

- Copter³ (v0.8.7) by Carlson et al. (2009), also using the TRG algorithm with the 2D integral in C++.
- The Halofit fitting formula by Smith et al. (2003), revised by Takahashi et al. (2012).
- The Coyote interpolator⁴ (FrankenEmu v1.0) (Heitmann et al., 2009, 2010; Lawrence et al., 2010; Heitmann et al., 2013), which produces the nonlinear matter spectrum for a portion of the Λ CDM parameter space based on nearly 1,000 N -body simulations.

We chose a flat Λ CDM cosmology using values as determined by Planck⁵ (Planck Collaboration et al., 2013) and divided each power spectrum by a linear spectrum with the BAO “wiggles” removed, obtained by using the description by Eisenstein and Hu (1999). The result can be seen in fig. 5.15. We notice three pairs of curves: Unsurprisingly, the Halofit corrections match N -body simulations very well by design even in highly nonlinear scales. The other two pairs correspond to the power spectrum computed by using the TRG algorithm with and without enforcing the extra symmetry. Our code can match either curve well while surpassing the maximum wave number of CLASS by more than a factor of 2.

5.2.2 Comparison with CLASS

Since CLASS implemented the exact same algorithm, we expect to get identical results for a Λ CDM model, and we can in fact observe that the results are reproduced almost exactly (less than 1% difference), as can be seen in fig. 5.16. Thus, the entire analysis in Audren and Lesgourgues (2011) applies to our code as well.

However, we would expect the difference to vanish completely in the linear regime, which is not the case here: Even at $k = 0.01h/\text{Mpc}$ the relative difference is around 0.1%. This appears to be an issue from the CLASS implementation, since their own nonlinear power spectrum does not perfectly match their linear power spectrum at low wave numbers (see fig. 5.17).

5.2.3 Comparison with Copter

Copter is a C++ library developed by Carlson et al. (2009) where a number of algorithms from different kinds of perturbation theory are implemented, including SPT, RPT, LPT, and TRG, the latter being referred to as “FWT” (as in “Flowing With Time”, the title of Pietroni (2008)). The TRG implementation is very straight forward with no recognizable efforts for efficiency, resulting in run times of the order of 30 minutes on M1 (see table 5.1) despite running in parallel.

However, it needs to be noted that Copter imposes an additional symmetry (see section 5.1.5), resulting in only 12 instead of 14 independent equations. This is not mentioned in the accompanying paper, but only in the source code as a comment in the file `FlowingWithTime.cpp`:

³<http://mwhite.berkeley.edu/Copter/>

⁴<http://www.hep.anl.gov/cosmology/CosmicEmu/emu.html>

⁵ $\Omega_m = 0.3175$, $\Omega_b = 0.0490$, $h = 0.6711$, $\sigma_8 = 0.8344$, $n_s = 0.9624$

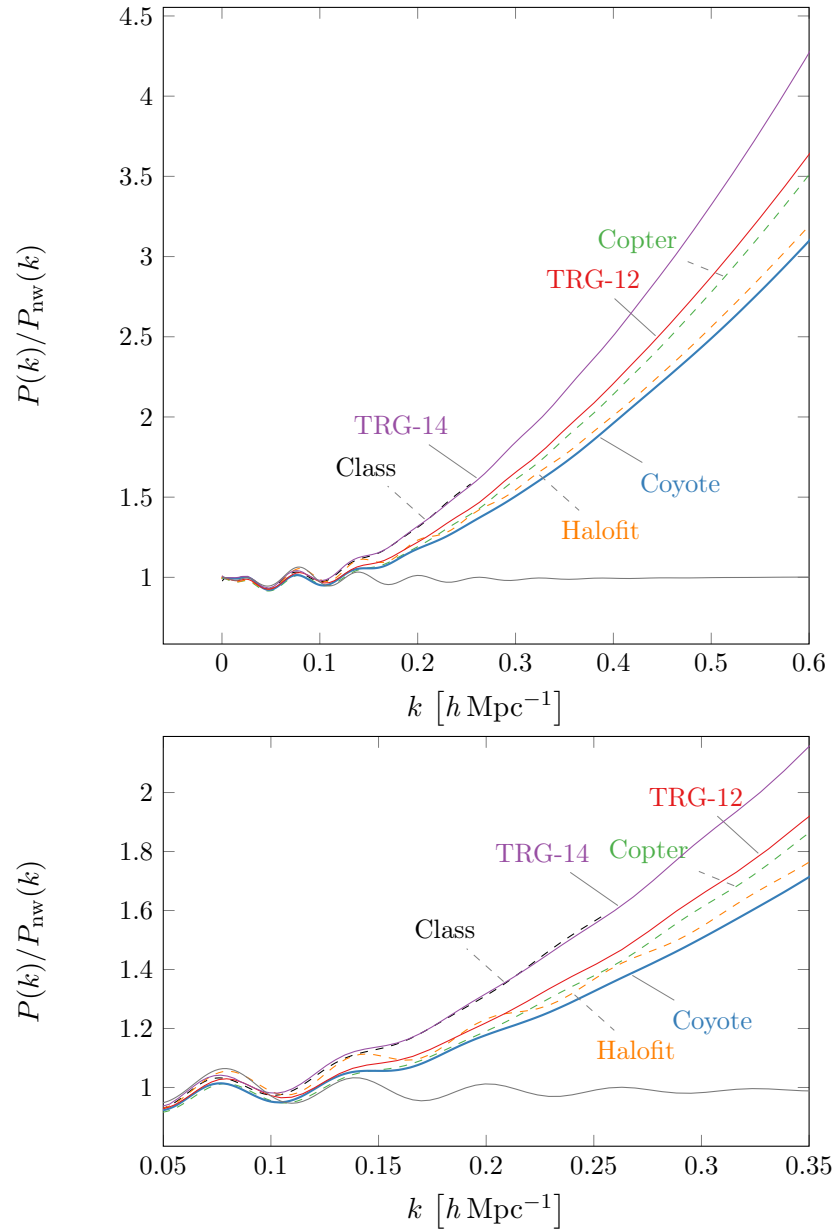


Figure 5.15 A comparison of the power spectrum at $z = 0$ obtained from different sources divided by a no-wiggle power spectrum from Eisenstein and Hu (1999). The sources are Coyote (blue, thick), TRG-12 (red), Copter (green, dashed), TRG-14 (purple), CLASS (black, dashed), Halofit (orange, dashed), and the linear power spectrum (gray). The bottom panel is simply a zoomed-in view on the BAO scale.

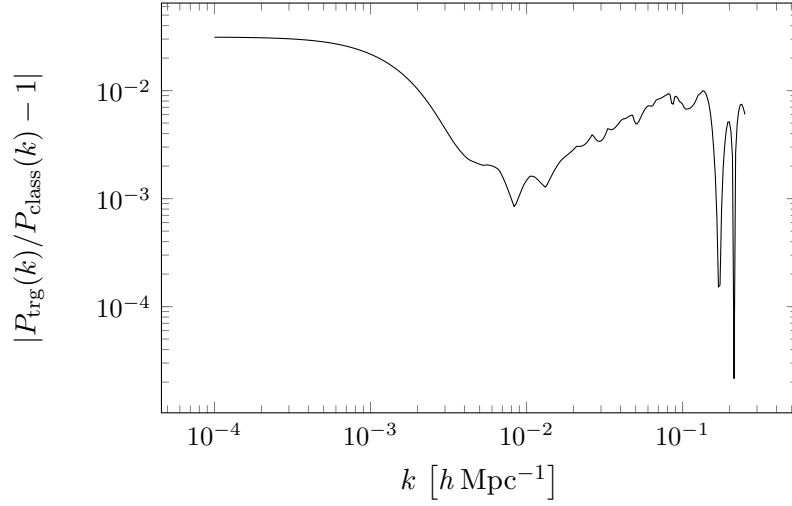


Figure 5.16 The nonlinear power spectrum computed by TRGFAST at $z = 0$ compared to the one computed by CLASS. Both agree up to 1%.

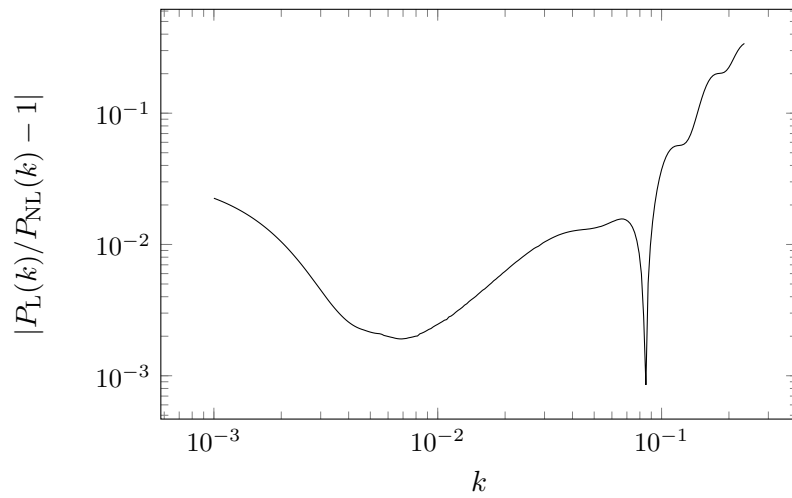


Figure 5.17 CLASS produces a linear and nonlinear power spectrum at $z = 0$ that only match at the 2% level in the linear regime.

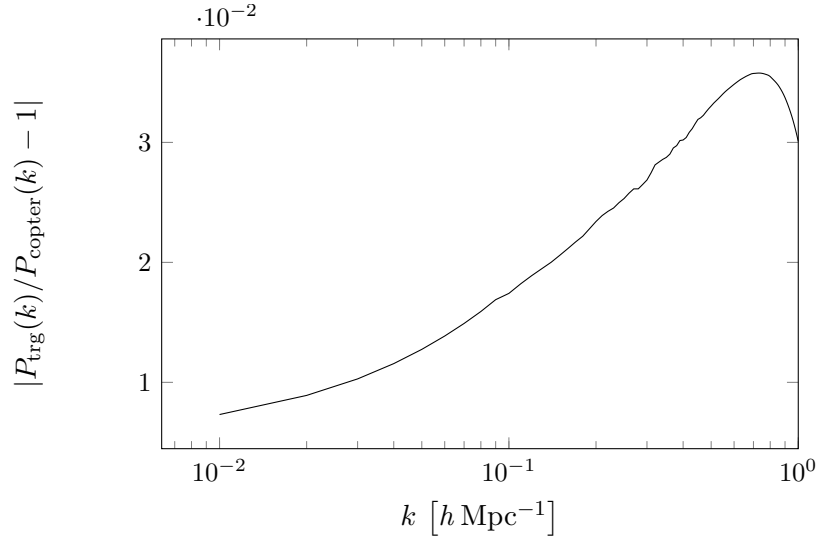


Figure 5.18 The nonlinear power spectrum computed by TRGFAST at $z = 0$ compared to the one computed by Copter. The power spectra agree up to 3.5%.

It's not obvious from the definition that $I_{acd,bef}(k)$ is symmetric in its last two indices ef . This result follows from the fact that it is initially symmetric ($I_{acd,bef} = 0$ at $\eta = 0$) and the equations of motion preserve this symmetry.

This statement appears to be false, considering that the equations of motion only preserve this symmetry if $A_{acd,bef}$ preserves it, which is only the case if $P_{11} = P_{12} = P_{22}$. Since the power spectra are only approximately equal in the linear regime, the symmetry is broken at later times. Indeed, the resulting nonlinear corrections differ if this symmetry is assumed. In order to still be able to compare the results, we can enable an option to enforce this symmetry in our code.

Figure 5.18 shows that the result from TRGFAST agrees with the result from Copter at the 3.5% level. This discrepancy may stem from the difference in power spectrum extrapolation, cutoff method or numerical noise in either Copter's code or our code.

5.2.4 Comparison to N -body simulations

N -body simulations come closest to actual data of different cosmological models. We will use them to verify the correctness of our implementation and demonstrate the general capabilities of the TRG technique.

Λ CDM

The first choice for a comparison with N -body data is the classic flat Λ CDM model. We use the data that has been computed by Sato and Matsubara (2011) for the redshifts $z = 3, 2, 1, 0.5, 0.3, 0$ based on the Wilkinson Anisotropy Probe 7 yr (WMAP7) parameters⁶ (Komatsu et al., 2010). They ran GADGET2 (Springel, 2005) with 1024^3

⁶ $\Omega_m = 0.265$, $\Omega_b = 0.0448$, $h = 0.71$, $\sigma_8 = 0.80$, $n_s = 0.963$

particles in boxes with sides $1h^{-1}$ Gpc. To estimate the uncertainty of the power spectrum, 30 realizations of the initial conditions based on a linear power spectrum from CAMB (Lewis et al., 2000) were made. Figure 5.19 shows their result together with the TRGFAST power spectrum for the redshifts mentioned above, both divided by a no-wiggle power spectrum (Eisenstein and Hu, 1999). In fig. 5.20, we plotted $k_{\%}$ as a function of the redshift, where $k_{\%}(z)$ is defined as the wavenumber where the two power spectra at that redshift first diverge by more than a given percentage. In the case of $z = 0.5$, the power spectra match so well that $k_{\%}$ would be unreasonably large, so we omitted that case. The TRG result matches the N -body data well until the redshift becomes smaller than unity. This is consistent with Audren and Lesgourgues (2011, fig. 7). After all, TRGFAST computes the same quantity as CLASS with great precision, as has been shown in section 5.2.2.

It is somewhat surprising that all three curves in fig. 5.20 decrease at $z = 3$, since we would expect the power spectra to agree better at higher redshifts, meaning that $k_{\%}$ would strictly increase. However, this is in line with fig. 5.22, where the growth functions from TRGFAST and N -body are compared and the largest discrepancy is for large redshifts. Of course, that figure is for coupled quintessence and not Λ CDM (the linear growth function from Sato and Matsubara (2011) is unfortunately not available), but qualitative differences to the Λ CDM case are unexpected. Ultimately, this comes down to the fact that by design the growth functions match exactly at $z = 0$, so that any possible deviations have to be at larger redshifts.

Coupled Quintessence

Since one of the advantages of TRG is the flexibility in regard to cosmological models, we are interested in its performance for models other than Λ CDM. For a first nonstandard cosmological model, we choose the coupled quintessence (CQ) model (Wetterich, 1994; Amendola, 1999, 2000). To the best of the authors knowledge, it appears that before the completion of this dissertation, no attempts at applying the TRG method to models with CQ have been made.

Quintessence has been known to provide a viable alternative to the cosmological constant for quite some time (Wetterich, 1988; Ratra and Peebles, 1988; Caldwell et al., 1998; Tsujikawa, 2013). Quintessence and variants such as k-essence may even offer possible solutions to what is known as the cosmological constant problem and the coincidence problem (Weinberg, 1989; Carroll, 2001). The idea is that a light scalar field ϕ can mimic a cosmological constant despite being dynamical if certain conditions are met. This field can couple to ordinary matter via an extra term in the Lagrangian of the form

$$-m_{\psi}^2 \exp(-C\phi/M_{\text{p}}^2) \partial_{\mu}\psi \partial^{\mu}\psi, \quad (5.5)$$

where ψ is the ordinary matter field and C the coupling constant. The Planck mass is $M_{\text{p}}^2 \equiv 8\pi G$.

Let us consider the energy-momentum tensor $T_{\mu\nu(\phi)}$ for the field and $T_{\mu\nu(\text{m})}$ for matter. Their sum needs to be locally conserved, such that we could have

$$\nabla_{\mu} T_{\nu(\phi)}^{\mu} = -CT_{\text{m}} \nabla_{\nu} \phi, \quad \nabla_{\mu} T_{\nu(\text{m})}^{\mu} = +CT_{\text{m}} \nabla_{\nu} \phi. \quad (5.6)$$

Other, more complicated couplings are also possible, but this is the simplest one and the one we will be considering here. For convenience, the coupling constant is redefined

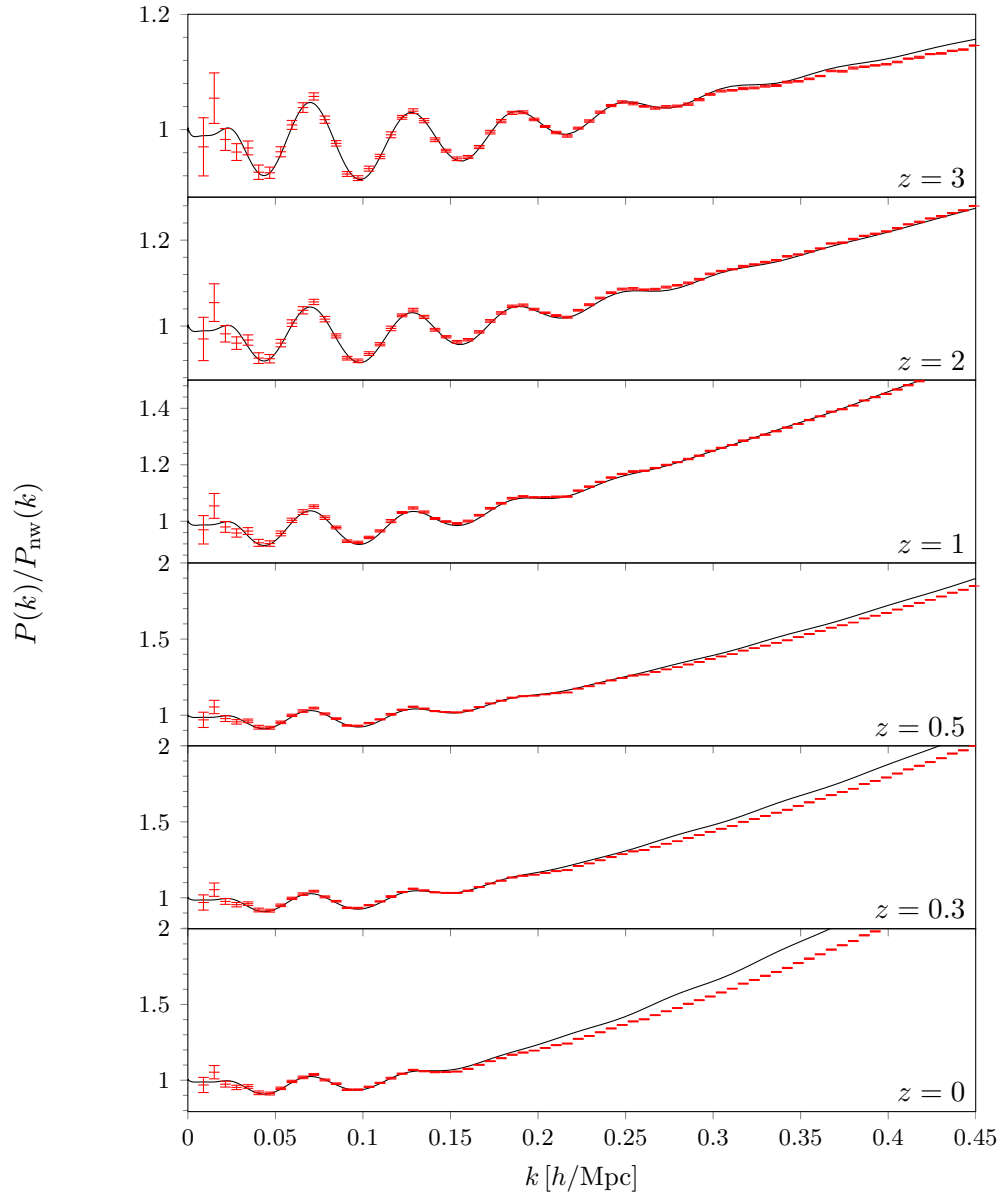


Figure 5.19 Comparison of the TRGFAST result with Λ CDM N -body data (Sato and Matsubara, 2011) for different redshifts. Each power spectrum has been divided by a no-wiggle spectrum from Eisenstein and Hu (1999). Agreement is good for redshift $z = 1$ and higher, then some discrepancy is noticeable just as in Audren and Lesgourgues (2011, fig. 7). Note the difference in the scale of the y direction for some panels!

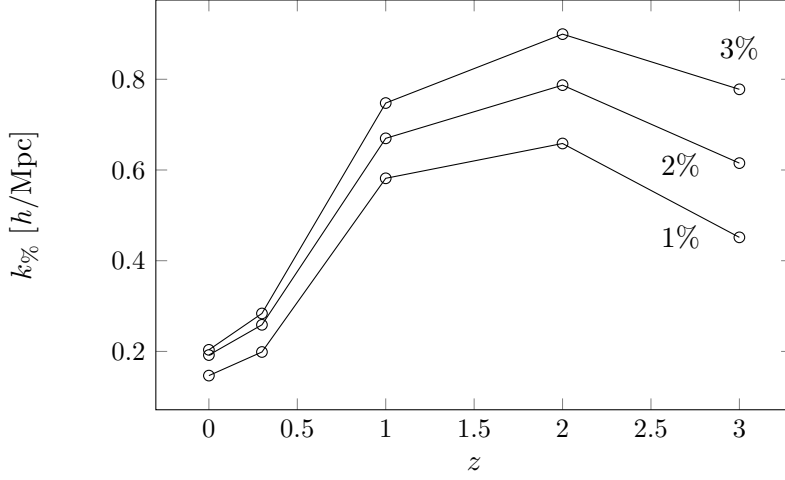


Figure 5.20 These curves indicate at which wave number $k_{\%}$ the power spectra from TRGFAST and Λ CDM N -body simulations (Sato and Matsubara, 2011) start to deviate by more than 1%, 2%, and 3% for each redshift. We left out the case for $z = 0.5$ because the power spectra coincidentally agree too well to have a meaningful value for $k_{\%}$ here.

as

$$\beta \equiv \sqrt{\frac{3}{2}} M_{\text{p}} C. \quad (5.7)$$

As mentioned above, the potential is assumed to read

$$V(\phi) = V_0 e^{-\sqrt{2/3} \alpha \phi / M_{\text{p}}} \quad (5.8)$$

with the potential parameter α . The evolution of the background functions are derived by Amendola (2004). Using the notation from section 2.5, we can identify

$$\mathcal{A} = -2\beta \sqrt{\Omega_{\text{kin}}}, \quad \mathcal{B} = \frac{4}{3} \beta^2, \quad (5.9)$$

such that the background functions in eq. (2.103) take the form

$$\Omega_{21}(a) = -\frac{3}{2} \Omega_{\text{m}}(a) \left(1 + \frac{4}{3} \beta^2 \right) \quad (5.10)$$

$$\Omega_{22}(a) = 3 + \frac{\partial \log H(a)}{\partial \log a} - 2\beta \sqrt{\Omega_{\text{kin}}(a)}. \quad (5.11)$$

Note that technically we should also include the term (Pietroni, 2008)

$$\mathcal{B}(\mathbf{k}, \tau) = \Omega_{\text{DE}} \delta_i^{\text{lin}} / (\Omega_{\text{m}} \delta_{\text{m}}^{\text{lin}}), \quad (5.12)$$

however, we will neglect this term since the quintessence perturbations are much smaller than the matter perturbations and they are not included in the N -body simulation either.

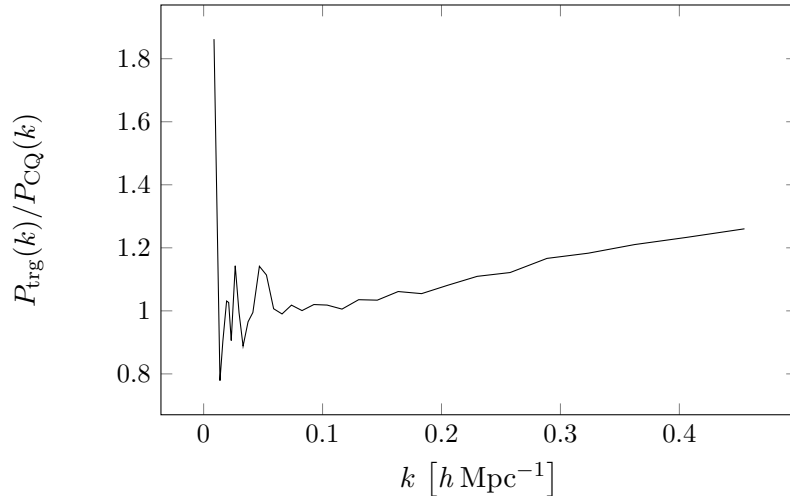


Figure 5.21 The relative difference of the nonlinear power spectrum obtained from TRGFAST and from CoDECS at $z = 0$. The difference is comparable to the Λ CDM case.

Several large N -body simulations have been performed using the CQ model with different coupling strengths and different potentials as part of a project called “CoDECS”⁷ (Baldi et al., 2010; Baldi, 2011). It uses a modified version of GADGET-2 (Springel, 2005) and features a box size of $L = 80\text{Mpc}/h$ and 2×512^3 particles. The data set we will be using from CoDECS has the label “EXP003-L”. Here, the coupling constant is $\beta = 0.15$ and the potential parameter is $\alpha = 0.1$.

We ran the code with the initial linear power spectrum from CoDECS and the background functions from above. We also use the symmetry discussed in section 5.2.3. The results are displayed in fig. 5.21, where the ratio of the nonlinear power spectrum from TRGFAST and the N -body simulation has been plotted, both at redshift $z = 0$. Unfortunately, the CoDECS data do not include error bars on the power spectrum. However, N -body simulations naturally cannot constrain the power spectrum very well on large scales due to their finite volume. This is why we see large fluctuations for low k in fig. 5.21. On larger scales, we have a discrepancy of up to 20%.

Figure 5.22 shows the comparison of the growth functions. They agree at the 1% level for all relevant redshifts, which gives reason to believe that the background evolution was implemented correctly and that the 20% discrepancy in the power spectrum has to come from the nonlinear corrections. The difference is comparable to the one for the Λ CDM case.

5.3 Performance

The advantage of TRGFAST compared to other implementations of the TRG algorithm is its relatively low runtime. Since we were able to reduce a 2D integral to a series of 1D integrals, the runtime should be lower by an order of magnitude. In this section we compare the performance by TRGFAST with the one by CLASS and Copter. Also, we

⁷Simulation data publicly available at <http://www.marcobaldi.it>.

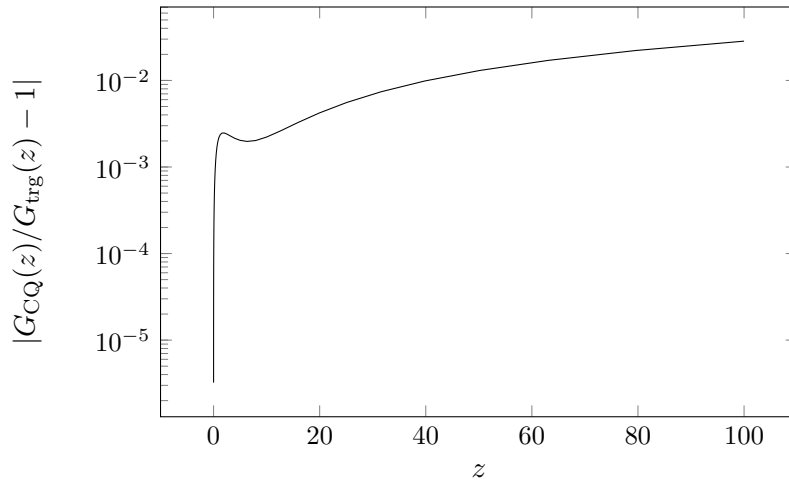


Figure 5.22 The relative difference of the growth function obtained from TRGFAST and from CoDECS. The discrepancy is less than 2% for all relevant redshifts.

| Name | Specs |
|------|---|
| M1 | 1 dual-core, Pentium(R) Dual-Core CPU E5300 @ 2.60GHz, 3.6GB RAM, Linux 3.2.0-4-amd64 #1 SMP Debian 3.2.54-2 x86_64 GNU/Linux |
| M2 | 24 cores, AMD Opteron(TM) Processor 6234 @ 2.40GHz, 64GB RAM, Linux 3.2.0-0.bpo.4-amd64 #1 SMP Debian 3.2.54-2 bpo60+1 x86_64 GNU/Linux |

Table 5.1 Machines used for benchmarking the TRGFAST code.

analyze how the runtime scales with number of cores being used.

The parallelization analysis was done using the GNU/Linux tools `time` (to measure the runtime) and `taskset` (to restrict the number of available cores). There are three interesting quantities. First, the *wall time*, i.e. the physical time that has passed during the run of the process (or the time that clock on the *wall* measures). Second, the *user time*, which is the amount of CPU-seconds that the process itself used. And third, the *system time*, which is the amount of CPU-seconds spent by the system on behalf of the process. We measured these quantities along with the average CPU load for a different number of cores (up to 24) by running TRGFAST three times and taking the average.

The results are shown in fig. 5.23. Just as we would expect in an ideally parallelized code, the wall time is inversely proportional to the number of cores. The user time stays almost constant, and the system time goes slightly up with the number of cores, since the kernel has to coordinate the communication between the cores. We notice that the binary produced by the Intel compiler is comparable in speed to the one produced by the GCC compiler.

On M2 with all cores enabled we have a total runtime of less than 6 seconds. Compare this to CLASS at 513 seconds (however, CLASS does not fully parallelize and uses no more than 14 cores at a time) or Copter at 369 seconds. On a typical desktop machine such as M1, TRGFAST completes the run in 38 seconds, CLASS in 1882 seconds and Copter in 2047 seconds. This speedup of approximately a factor of 50 is the major

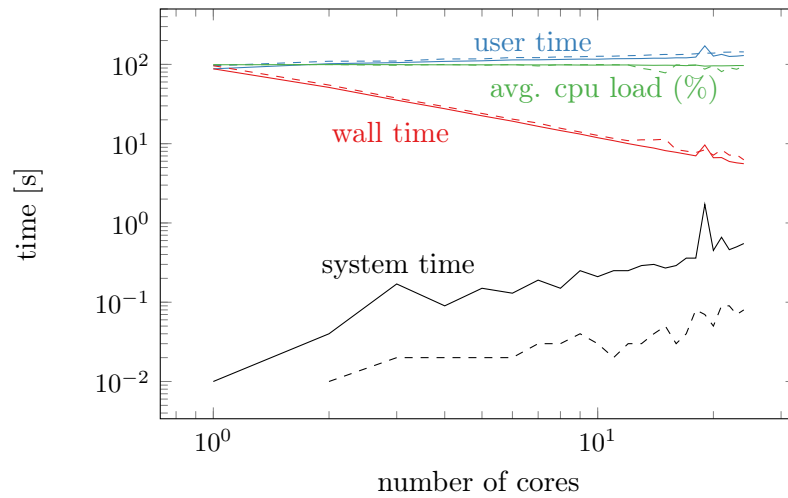


Figure 5.23 Runtime of TRGFAST for different number of cores run on M2 with a total of 24 cores. Shown are the wall time (red), user time (blue), system time (black), and average CPU load in percent (green). The test was done using a binary compiled with the Intel compiler (solid) and the GCC compiler (dashed). Each measurement was averaged over three runs. The options for TRGFAST were set to their default values.

achievement of this research.

All of this was done using the default values for all parameters. Of course, it is possible to adjust some parameter, for instance decrease the number of time steps to decrease the runtime even further, but the accuracy of the final result may suffer.

Part II

Weak lensing

Chapter 6

Review of weak lensing

To set the notation and present the basic methods of the statistics of weak gravitational lensing for the following chapters, we include a short summary here. For an extensive review, see e.g. Bartelmann and Schneider (1999). Carroll (2003) has a shorter, but more pedagogical approach. The weak lensing Fisher matrix formalism is covered in Amendola and Tsujikawa (2010, ch. 4.11 and 14.4). Note that in this and the following chapter, we will *not* use the Einstein sum convention.

6.1 The convergence power spectrum

Gravitational lensing is the phenomenon which can be observed when light passes through a local curvature perturbation in space, for instance caused by a star, galaxy cluster, or dark matter halo. One can distinguish between strong, weak, and micro lensing, where weak lensing is characterized by only being observable statistically over a sufficiently large patch of the sky. Due to the gravitational influence of (dark) matter structures in the path of a light ray emitted from background sources, the shapes of galaxies appear distorted by a small, but on average measurable amount, resembling the image you would see when looking through a sheet of uneven glass. Analyzing these distortions gives us insights about the distribution and evolution of the large scale structure in the universe, which both depend on several cosmological parameters from the dark sector. Thus weak lensing is an excellent probe of dark energy.

In a weak lensing survey, what we measure is the convergence power spectrum, which is a measure for the correlation of ellipticity between neighboring galaxies. Most galaxies appear elliptic due to their intrinsic shape, but we mostly expect the direction of those ellipses to be uncorrelated. Only if an object such as a massive cluster in the foreground acts as a lens, the ellipticity of the individual galaxies changes slightly, but this time they are all aligned around the lens. Hence, these added ellipticities do correlate and this correlation is the signal that future surveys are trying to find.

To derive all the expressions we need for the convergence power spectrum, we start with the geodesic equation

$$\frac{dk^\mu}{d\lambda} + \Gamma_{\alpha\beta}^\mu k^\alpha k^\beta = 0. \quad (6.1)$$

where

$$k^\mu = \frac{dx^\mu}{d\lambda} \quad (6.2)$$

is the momentum vector of a photon whose curve is parameterized by the affine parameter λ . Since photons move on null trajectories, we can further impose that k^μ is a null vector satisfying

$$k^\mu k_\mu = 0. \quad (6.3)$$

Our goal is to find the perturbed trajectory (at least for the two transverse components), so as usual we write the momentum vector as the sum of the background value and its perturbation:

$$k^\mu = \hat{k}^\mu + \delta k^\mu. \quad (6.4)$$

After plugging in the Christoffel symbols (see e.g. Amendola and Tsujikawa (2010, p. 9)) into eq. (6.1), the solution for the geodesic equation for $\mu = 0$ at the background level reads

$$\hat{k}^0 = \frac{d\tau}{d\lambda} \propto a^{-2}. \quad (6.5)$$

The metric for our purposes is the perturbed Robertson-Walker metric for a flat universe in the Newtonian gauge using the conformal time, which reads

$$ds^2 = a^2(t) [-(1 + 2\Psi)d\tau^2 + (1 + 2\Phi)dr^2] \quad (6.6)$$

if we neglect the angular part. For photons, the line element vanishes such that we get (at linear order)

$$(1 + \Psi)d\tau = (1 + \Phi)dr, \quad (6.7)$$

which we can rearrange to yield (again discarding nonlinear terms)

$$\frac{d\tau}{d\lambda} = (1 + \Phi - \Psi) \frac{dr}{d\lambda}. \quad (6.8)$$

This equation can be used together with eq. (6.5) to write down the geodesic equation for the transverse components labeled by $i = 1, 2$:

$$\frac{d^2 x^i}{d\lambda^2} + 2\mathcal{H} \frac{d\tau}{d\lambda} \frac{dx^i}{d\lambda} = \left(\frac{d\tau}{d\lambda} \right)^2 \partial_i \psi, \quad (6.9)$$

where we defined the lensing potential

$$\psi \equiv \Phi - \Psi. \quad (6.10)$$

We also use the shorthand notation for the partial derivative,

$$\partial_i \equiv \frac{\partial}{\partial x^i}. \quad (6.11)$$

Equation (6.9) can further be simplified to

$$\frac{d^2 x^i}{dr^2} = \partial_i \psi. \quad (6.12)$$

Because the displacement x^i is small, we can also write it in terms of the angular displacement θ^i , such that

$$\frac{d^2}{dr^2} (r\theta^i) = \partial_i \psi. \quad (6.13)$$

If we wanted to be precise, we would have to use the transverse comoving distance here, so d_M instead of r (see section 2.1.3), but since we are assuming a flat universe for now, the comoving distance will do. We can replace it with the appropriate expression later. The solution to the last equation is

$$\theta^i = \theta_0^i + \frac{1}{r} \int_0^r dr'' \int_0^{r''} dr' \partial_i \psi(r' \theta_0^1, r' \theta_0^2, r'), \quad (6.14)$$

which can be integrated to yield

$$\theta^i = \theta_0^i + \int_0^r dr' \left(1 - \frac{r'}{r}\right) \partial_i \psi(r' \theta_0^1, r' \theta_0^2, r'). \quad (6.15)$$

We can now do an expansion of θ^i around θ_0^i up to linear order to describe the separation of two nearby light rays, one at θ_0^i and one at $\theta_0^i + \Delta\theta_0^i$, and get the difference of their displacement

$$\Delta\theta^i = \Delta\theta_0^i + \Delta\theta_0^j \int_0^r dr' \left(1 - \frac{r'}{r}\right) r' \partial_{ij} \psi(r' \theta_0^1, r' \theta_0^2, r'). \quad (6.16)$$

This equation can be written in matrix form, i.e.

$$\Delta\theta^i = A_{ij} \Delta\theta_0^j, \quad (6.17)$$

where we defined

$$A_{ij} \equiv \frac{\partial \theta_s^i}{\partial \theta_0^j} = \delta_{ij} + D_{ij}. \quad (6.18)$$

It allows us to easily describe the transformation of entire images instead of single light rays. The distortion tensor D_{ij} in the transformation matrix can be parameterized by what we call the convergence κ_{wl} and the shear γ_i , such that it takes the form

$$D_{ij}(r_s) = \int_0^{r_s} dr' \left(1 - \frac{r'}{r_s}\right) r' \partial_{ij} \psi = \begin{pmatrix} -\kappa_{\text{wl}} - \gamma_1 & -\gamma_2 \\ -\gamma_2 & -\kappa_{\text{wl}} + \gamma_1 \end{pmatrix}. \quad (6.19)$$

The convergence is a measure for the magnification of the image and takes the form

$$\kappa_{\text{wl}} = -\frac{1}{2} \int_0^r dr' \left(1 - \frac{r'}{r}\right) r' (\partial_{11} + \partial_{22}) \psi, \quad (6.20)$$

while the shear describes the distortion and is written as

$$\gamma_1 = -\frac{1}{2} \int_0^{r_s} dr' \left(1 - \frac{r'}{r}\right) r' (\partial_{11} - \partial_{22}) \psi, \quad (6.21)$$

$$\gamma_2 = -\int_0^{r_s} dr' \left(1 - \frac{r'}{r}\right) r' \partial_{12} \psi. \quad (6.22)$$

The shear is tightly coupled to the ellipticity of galaxies, which is what is actually measured. For details we refer to the resources mentioned at the beginning of this section, and we just give the relevant result, which is at first order

$$\varepsilon_i = 2\gamma_i, \quad (6.23)$$

where ε_i is the ellipticity defined in terms of the components of the quadrupole moment of the image.

It may be surprising that the shear is the observable, but what we use for the Fisher matrix is the *convergence* power spectrum, even though the convergence itself is not observable directly. As we will see in a moment, this can be traced back to the fact that the shear power spectra are proportional to the convergence power spectrum. Since the convergence power spectrum depends on an integral over the matter power spectrum and the Hubble function, it is sensitive to the cosmological parameters we are aiming to constrain.

We start out with generalizing the distortion tensor for a thin lens to a smooth matter distribution. It is done by summing up all contributions along the line of sight, i.e.

$$\mathcal{D}_{ij} = \int_0^\infty dr n(r) D_{ij}(r), \quad (6.24)$$

where $n(r)$ is the radial galaxy distribution function normalized to unity. It is straightforward to verify that we can rewrite eq. (6.24) to read

$$\mathcal{D}_{ij} = \int_0^\infty dr w(r) \partial_{ij} \psi, \quad (6.25)$$

with the weight function

$$w(r) \equiv \int_r^\infty dr' \left(1 - \frac{r}{r'}\right) r n(r'). \quad (6.26)$$

Thus, we can put the convergence from eq. (6.20) into a form which will be more convenient later:

$$\kappa_{\text{wl}} = -\frac{1}{2} \int_0^\infty dr w(r) (\partial_{11} + \partial_{22}) \psi. \quad (6.27)$$

The convergence is defined on the two dimensional sky, but contains the gravitational potential which is defined on three dimensional space. The radial component is integrated out. If our goal is to relate the convergence power spectrum to the matter power spectrum (which traces the power spectrum of the potential), we need to find a way to express the power spectrum of one quantity that is an integral over another quantity in terms of the power spectrum of that other quantity. Fortunately, this is exactly what Limber's theorem does, which states that the power spectrum of F , which is defined as the weighted integral of another function f ,

$$F(\theta^1, \theta^2) = \int_0^\infty dr w(r) f(\theta^1 r, \theta^2 r, r), \quad (6.28)$$

can be written as

$$P_F(q) = \int_0^\infty dr \frac{w(r)^2}{r^2} P_f\left(\frac{q}{r}\right). \quad (6.29)$$

We use P_F to denote the power spectrum of any field F . Applying Limber's theorem to eq. (6.27), we get

$$P_{\kappa_{\text{wl}}}(q) = \frac{1}{4} \int_0^\infty dr \frac{w(r)^2}{r^2} P_{\sum_i \partial_{ii} \psi}\left(\frac{q}{r}\right) \quad (6.30)$$

$$= \frac{1}{4} \int_0^\infty dz \frac{W(z)^2}{H(z)} P_{\Sigma_i \partial_{ii} \psi} \left(\frac{q}{r} \right) \quad (6.31)$$

with the window function

$$W(z) \equiv \frac{w(r(z))}{r(z)}. \quad (6.32)$$

The last step is to find the power spectrum of $\partial_{ij}\psi$, which is the square of the Fourier transform, i.e.

$$P_{\partial_{ij}\psi} = k_i^2 k_j^2 |\tilde{\psi}|^2. \quad (6.33)$$

Remember that in the absence of anisotropic stress, we have $\psi = 2\Phi$, and we can use the Poisson equation in Fourier space, which reads

$$k^2 \tilde{\psi} = 3a^2 H(a)^2 \Omega_m \delta_m(a) \quad (6.34)$$

This gives us the power spectrum of $\partial_{ii}\psi$ in terms of the matter power spectrum:

$$P_{\Sigma_i \partial_{ii} \psi}(k) = k^4 P_\psi = 9H(z)^4 \Omega_m(z)^2 / (1+z)^4 P_{\delta_m}(k). \quad (6.35)$$

Putting it all together, and replacing q with the multipole ℓ/π , finally yields the power spectrum for the convergence,

$$P_{\kappa_{\text{wl}}}(\ell) = \frac{9H_0^3}{4} \int_0^\infty \frac{W(z)^2 E(z)^3 \Omega_m(z)^2}{(1+z)^4} P_{\delta_m}(z, k(z, \ell)) dz, \quad (6.36)$$

where

$$k(z, \ell) = \frac{\ell}{\pi d_M(z)}. \quad (6.37)$$

The full expression for the window function is

$$W(z) = \int_z^\infty \frac{d\tilde{z}}{H(\tilde{z})} \left(1 - \frac{d_M(z)}{d_M(\tilde{z})} \right) n(d_M(\tilde{z})), \quad (6.38)$$

where we replaced r with d_M to emphasize that this is the transverse comoving distance and generalize to non-flat cosmologies. Note that because of $n(z)dz = n(r)dr$ the factor $H(z)$ is absorbed when we replace $n(r(z))$ with $n(z)$, the galaxy density in redshift space.

In order to relate this to the shear power spectrum, we still need the power spectrum for the non-diagonal terms $\partial_{ij}\psi$. Similar to the expression in eq. (6.33), by using eq. (6.35), we get

$$P_{\partial_{ij}\psi} = k_i^2 k_j^2 P_\psi = 9H(z)^4 \Omega_m(z)^2 / (1+z)^4 \frac{k_i^2 k_j^2}{k^4} P_{\delta_m}(k). \quad (6.39)$$

Since the shear is related to the distortion tensor via

$$\gamma_1 = \frac{1}{2}(D_{22} - D_{11}), \quad \gamma_2 = -D_{12}, \quad (6.40)$$

and after applying Limber's theorem to the distortion tensor again to get the power spectra for γ_1 and γ_2 (as we did for κ_{wl}), we get (Amendola and Tsujikawa, 2010,

ch. 14.4)

$$P_{\gamma_1} = (P_{D_{11}} + P_{D_{22}} - 2P_{D_{12}})/4 = c_1 P_{\kappa_{\text{wl}}}, \quad (6.41)$$

$$P_{\gamma_2} = P_{D_{12}} = c_2 P_{\kappa_{\text{wl}}} \quad (6.42)$$

with

$$c_1 = (k_1^2 - k_2^2)^2 / (4k^4), \quad c_2 = k_1^2 k_2^2 / k^4. \quad (6.43)$$

This concludes the derivation of the convergence power spectrum. We will now introduce the Fisher matrix formalism before moving on to weak lensing tomography.

6.2 Weak lensing Fisher matrix

A comprehensive reference for the foundations of Bayesian statistics in cosmology, which we will summarize here, can be found in Hobson et al. (2009) and references therein. Here, we focus on the Fisher matrix formalism. Its strengths come into play when the likelihood for a model with a lot of parameters is involved in the problem we are trying to solve. In particular, the Fisher matrix can be a valuable tool to forecast how well future observation will constrain cosmological parameters.

The Fisher matrix for a model parameterized by the parameters Θ_α is defined as the negative expectation value of the Hessian of the log-likelihood, i.e.

$$F_{\alpha\beta} \equiv - \left\langle \frac{\partial \log \mathcal{L}}{\partial \Theta_\alpha \partial \Theta_\beta} \right\rangle, \quad (6.44)$$

where the expectation value is to be understood as an average over the data distribution. If the likelihood is Gaussian, all the information about the likelihood is contained in the Fisher matrix. In the Gaussian case, it equals the negative Hessian of the log-likelihood at its maximum, i.e.

$$F_{\alpha\beta} = \left. \frac{\partial \log \mathcal{L}}{\partial \Theta_\alpha \partial \Theta_\beta} \right|_{\Theta = \Theta_0}. \quad (6.45)$$

Here, the parameters Θ_0 where the log-likelihood has its maximum, is called the *fiducial model*. This approach also works if the likelihood can be approximated by a Gaussian reasonably well. When considering future experiments, the Fisher matrix gives us constraints on the model assuming that the experiment will measure the fiducial values. The Cramér-Rao bound guarantees that the errors derived from the Fisher matrix represent a best-case scenario, i.e. the actual errors can only be larger (Kendall and Stuart, 1967).

Another useful property of the Fisher matrix formalism is the convenient way to perform operations on the likelihood:

1. the square of the fully marginalized errors of the parameters lie on the diagonal of the inverse of the Fisher matrix, i.e. $\sigma_i^2 = (F^{-1})_{ii}$,
2. the inverse of the square of the error of one parameter with all other parameters fixed is a diagonal element of the Fisher matrix, i.e. $\bar{\sigma}_i^2 = 1/F_{ii}$,
3. marginalization of a parameter (i.e. integrating out a parameter from the likelihood) becomes striking out the corresponding column and row from the inverse,

4. fixing a parameter becomes striking out the corresponding column and row from the Fisher matrix, and
5. applying a prior or combining one Fisher matrix with one from another experiment means simply summing up the two matrices.

There is a more useful way to express the Fisher matrix. For a set of Gaussian data contained in a vector \mathbf{x} with mean $\boldsymbol{\mu}$ and covariance C , the likelihood is

$$\mathcal{L} \propto \frac{1}{\sqrt{\det C}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^T C^{-1}(\mathbf{x} - \boldsymbol{\mu})\right). \quad (6.46)$$

The covariance matrix is defined as

$$C = \langle D \rangle, \quad (6.47)$$

where we defined the data matrix

$$D = (\mathbf{x} - \boldsymbol{\mu})(\mathbf{x} - \boldsymbol{\mu})^T. \quad (6.48)$$

Then the log-likelihood becomes

$$-2 \log \mathcal{L} = \log \det C + \text{Tr}(C^{-1}D) = \text{Tr}(\log C + C^{-1}D). \quad (6.49)$$

Since we will take the derivative of this expression in a moment, we already left out the proportionality constant. Note that we also used the matrix identity

$$\log \det C = \text{Tr} \log C \quad (6.50)$$

as well as the fact that the argument in the exponential function in eq. (6.46) is a scalar, which means that it is identical with its trace. The cyclic property of the trace then yields the second term in eq. (6.49). Taking the second derivative and the expectation value afterwards gives us the Fisher matrix,

$$F_{\alpha\beta} = \frac{1}{2} \text{Tr} A^\alpha A^\beta + C^{-1} \langle \partial_{ij} D \rangle, \quad (6.51)$$

with $A^\alpha \equiv C^{-1} \partial C / \partial \theta_\alpha$ (Tegmark et al., 1997). Here, we also used that $\langle \mathbf{x} \rangle = \boldsymbol{\mu}$ and $\langle \partial_i D \rangle = 0$ as well as the fact that the first derivatives of the likelihood vanish at its maximum.

Now that we have the expression for the Fisher matrix, we can compute it for the power spectrum. Since we will do this for multiple redshift bins, we need to first talk about weak lensing tomography.

6.3 Tomography

It has been shown that one can increase the information extracted from a weak lensing survey by dividing the observed galaxies into bins in redshift space (Hu, 1999). Typically, we define roughly 3-8 redshift bins and correlate the signal coming from each bin. We only have to replace the window functions in eq. (6.36). The convergence power

spectrum for the bin pair with indices i and j is then

$$P_{ij}(\ell) = \frac{9}{4} H_0^3 \int_0^\infty \frac{(1+z)^2 \Omega_{m0}^2 W_i(z) W_j(z)}{E(z)^2} P_{\delta_m}(z, k(z, \ell)) dz. \quad (6.52)$$

Here we defined $W_i(z)$, the weak lensing window function for the i th bin¹,

$$W_i(z) = \int_z^\infty \left(1 - \frac{d_M(z)}{d_M(\tilde{z})}\right) n_i(\tilde{z}) d\tilde{z}. \quad (6.53)$$

In this expression, $n_i(z)$ equals the galaxy density $n(z)$ if z lies inside the i th redshift bin and zero otherwise. The overall galaxy density is as usual modeled as (Ma et al., 2006)

$$n(z) \propto z^a \exp(-(z/z_p)^b), \quad (6.54)$$

where we take $a = 2$, $b = 3/2$. These values have been determined empirically (Amara and Réfrégier, 2007). It is straightforward to verify that the parameter z_p is related to the median of the galaxy distribution z_* by

$$z_p = z_*/1.412. \quad (6.55)$$

The $n_i(z)$ are then smoothed with a Gaussian to account for the photometric redshift error (Ma et al., 2006) and normalized such that

$$\int_0^\infty n_i(z) dz = 1. \quad (6.56)$$

Including the noise due to intrinsic galaxy ellipticity, the covariance matrix becomes (Hu, 1999)

$$C_{ij} = P_{ij} + \gamma_{\text{int}}^2 \tilde{n}_i^{-1} \delta_{ij}, \quad (6.57)$$

with the intrinsic ellipticity $\gamma_{\text{int}} = 0.22$ (Amara and Réfrégier, 2007; Hu and Tegmark, 1999) and the number of all galaxies per steradian in the i -th bin, \tilde{n}_i . It can be written as

$$\tilde{n}_i = n_\theta \frac{\int_{z_i}^{z_{i+1}} n(z) dz}{\int_0^\infty n(z) dz}, \quad (6.58)$$

where n_θ is the areal galaxy density, an important parameter that defines the quality of a weak lensing experiment. To convert from galaxies per square arcminute (the unit in which this quantity is typically given) to galaxies per steradian, we need to multiply by $3600(180/\pi)^2$.

Finally, we can write down the weak lensing Fisher matrix. Plugging in the covariance matrix in eq. (6.51) and writing out the elements explicitly, we get (for details, see Hu and Tegmark (1999))

$$F_{\alpha\beta}^{\text{WL}} = f_{\text{sky}} \sum_{\ell, i, j, m, n} \frac{(2\ell + 1)}{2} \frac{\partial P_{ij}}{\partial \Theta_\alpha} C_{jm}^{-1} \frac{\partial P_{mn}}{\partial \Theta_\beta} C_{ni}^{-1}. \quad (6.59)$$

Note that it is the components of the inverse, not the inverse of the components that

¹We would like to point out that it can be computed most efficiently by slightly rearranging the terms and using the method outlined in appendix D.

is referenced here. This is for a weak lensing survey that covers a fraction of the sky f_{sky} . The Fisher matrix is a sum over all multipoles ℓ with $2\ell + 1$ modes per multipole.

Using the more compact expression gives us

$$F_{\alpha\beta}^{\text{WL}} = f_{\text{sky}} \sum_{\ell} \frac{(2\ell + 1)}{2} \text{Tr}[A^{\alpha} A^{\beta}], \quad (6.60)$$

where we have defined the matrices $A^{\alpha}(\ell)$ as

$$A^{\alpha}(\ell) \equiv C^{-1} \partial C / \partial \Theta_{\alpha}. \quad (6.61)$$

To save computation time, the sum is typically not evaluated at every single ℓ , but instead we divide the ℓ -space into bins (Eisenstein et al., 1999). The term that is being summed over is then assumed to be constant inside these ℓ -bins. Then the expression for the Fisher matrix becomes

$$F_{\alpha\beta}^{\text{WL}} = f_{\text{sky}} \sum_{\mu} \Delta\ell_{\mu} \frac{(2\bar{\ell}_{\mu} + 1)}{2} \text{Tr}[A^{\alpha} A^{\beta}], \quad (6.62)$$

with $\bar{\ell}_{\mu}$ being the value of the center of the μ th ℓ -bin and $\Delta\ell_{\mu}$ its width.

Chapter 7

Application: Weak lensing forecast

As a first application of our TRG code, we will compute the weak lensing Fisher matrix for a Λ CDM model including nonlinear corrections. Our model is parameterized by the following five quantities:

$$\Theta = (\Omega_b, \Omega_c, h, n_s, \sigma_8). \quad (7.1)$$

These will be the Fisher matrix parameters that we want to constrain in this chapter. As before, we will assume that the survey measuring the convergence power spectrum will be Euclid.

Because in practice, Fisher matrix analyses always involve numerical codes for generating the power spectrum (linear or nonlinear), we are also interested in how weak lensing Fisher matrices from these different approaches compare. The Fisher matrix depends on derivatives of the convergence power spectrum, so it is not enough for a code to generate a realistic power spectrum, its dependency on the cosmological parameters is equally important.

7.1 Setup

We are particularly interested in the effect of nonlinear corrections on the constraints provided by the Fisher matrix. To this end, we run the same calculation for three different cases: One without nonlinear corrections, one with Halofit+ (Takahashi et al., 2012) corrections, and one with corrections from our `trgfast` code. Additionally, we differentiate between two more cases: The linear power spectrum is computed either by CAMB (Lewis et al., 2000) or by using the formula for the transfer function from Eisenstein and Hu (1999) (we will just refer to it as “EHTF” from now on, as in “Eisenstein and Hu’s Transfer Function”). Thus, we are dealing with six different methods in total to compute the power spectrum.

For an accurate comparison, we only use the transfer function at redshift zero from the output of CAMB, not the linear power spectrum. CAMB does this by using the synchronous gauge. Just like in the EHTF case, we normalize with our value of σ_8 (see eq. (8.15)) and use the growth function to compute the power spectrum at earlier times. The growth function is obtained by using Copter, which calculates it by solving the linear fluid equations eqs. (2.58) and (2.59). For the survey parameters we use the

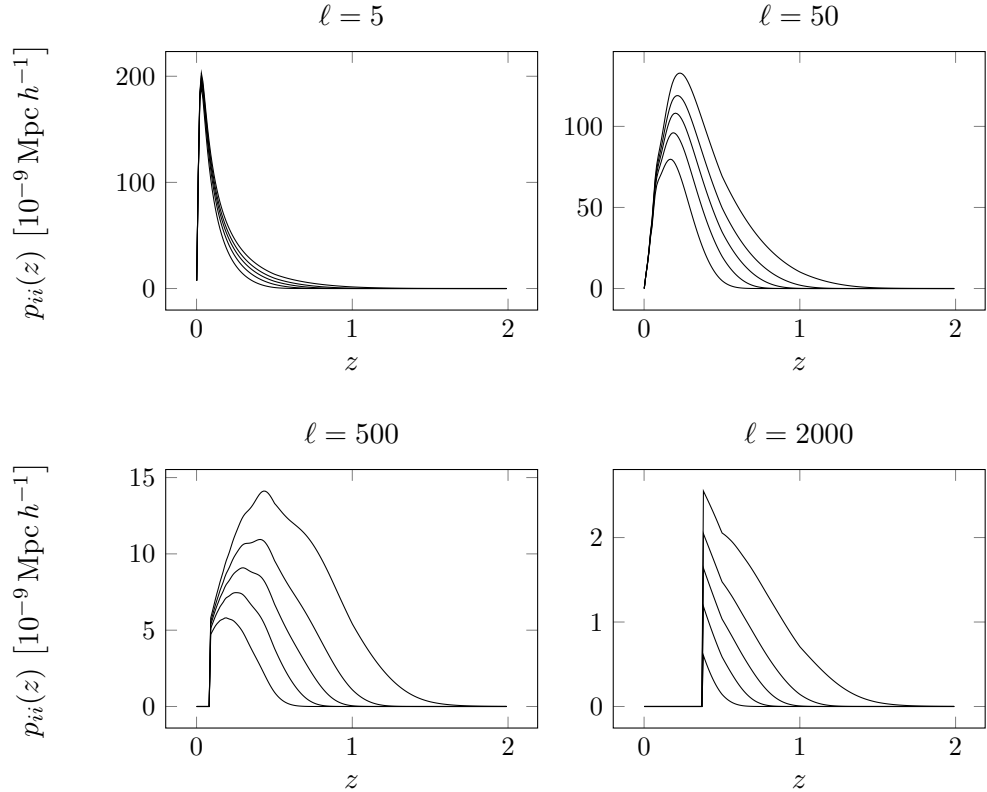


Figure 7.1 The integrands for the convergence power spectrum using `trgfast` for the matter power spectrum at various ℓ . Because the power spectrum is cut off at some k , the integrands for large ℓ vanish at low redshift. Each panel contains curves corresponding to $i = 1, \dots, 5$.

ones shown in table 8.1 and the maximum multipole is $\ell_{\text{max}} = 2000$. Redshift space is divided into five bins, which all contain the same number of galaxies. The fiducial model is the same as in chapter 8.

7.2 Error discussion

Before we discuss the errors, we want to point out one particular issue with our setup: The power spectrum obtained from `trgfast` has a maximum scale k_{max} at which it is defined. However, the integration limits in eq. (6.52) require the power spectrum to be defined for any positive scale k . If $k > k_{\text{max}}$, we assume the power spectrum vanishes. This effectively changes the lower integration limit in eq. (6.52), as you can see in fig. 7.1 where we plotted the integrands for different values of ℓ . `Halofit` does not have this limitation, such that we can formally use the nonlinear power spectrum at arbitrary k , but this clearly does not mean it can be trusted in these highly nonlinear regimes.

To gauge how much of the difference between using `Halofit` and `trgfast` stems from the higher integration limit, we also include an additional case called *HaloCut*, where the power spectrum comes from `Halofit` but is cut off at the same wave number k_{max}

| | Linear | | | Halofit | | trgfast | | HaloCut | |
|------------|----------|--------|------|---------|------|---------|------|---------|------|
| | fiducial | abs. | rel. | abs. | rel. | abs. | rel. | abs. | rel. |
| Ω_b | 0.049 | 0.0036 | 7.4% | 0.0044 | 9.% | 0.0051 | 10.% | 0.005 | 10.% |
| Ω_c | 0.2678 | 0.0031 | 1.2% | 0.0064 | 2.4% | 0.0085 | 3.2% | 0.0036 | 1.3% |
| h | 0.6704 | 0.02 | 3.% | 0.025 | 3.7% | 0.031 | 4.7% | 0.014 | 2.1% |
| n_s | 0.9619 | 0.019 | 2.% | 0.02 | 2.1% | 0.012 | 1.3% | 0.013 | 1.4% |
| σ_8 | 0.8347 | 0.0099 | 1.2% | 0.011 | 1.3% | 0.017 | 2.1% | 0.0067 | 0.8% |

Table 7.1 Fully marginalized Fisher matrix constraints (absolute and relative errors) on the fiducial model of our choice. Here, the linear power spectrum comes from CAMB.

| | Linear | | | Halofit | | trgfast | | HaloCut | |
|------------|----------|--------|-------|---------|------|---------|------|---------|------|
| | fiducial | abs. | rel. | abs. | rel. | abs. | rel. | abs. | rel. |
| Ω_b | 0.049 | 0.0035 | 7.2% | 0.0054 | 11.% | 0.0055 | 11.% | 0.0062 | 13.% |
| Ω_c | 0.2678 | 0.0037 | 1.4% | 0.0081 | 3.% | 0.0094 | 3.5% | 0.0086 | 3.2% |
| h | 0.6704 | 0.01 | 1.5% | 0.036 | 5.3% | 0.037 | 5.6% | 0.052 | 7.8% |
| n_s | 0.9619 | 0.013 | 1.4% | 0.018 | 1.8% | 0.013 | 1.3% | 0.024 | 2.5% |
| σ_8 | 0.8347 | 0.0048 | 0.58% | 0.012 | 1.4% | 0.018 | 2.1% | 0.011 | 1.3% |

Table 7.2 Same as table 7.1, but now the linear power spectrum comes from EHTF.

as the corresponding power spectrum from trgfast.

This makes 12 cases in total, for each of which we compute the Fisher matrix. The fully marginalized errors that we obtain from the procedure outlined above are shown in tables 7.1 to 7.3. For a visual comparison, see fig. 7.2.

We immediately notice that errors from the same quantity can differ substantially. Surprisingly, when comparing the blue (Halofit) and purple (HaloCut) bars, we see that cutting off the matter power spectrum prematurely can increase as well as decrease the error. To check this, we computed the error on the Hubble constant h using the EHTF power spectrum without nonlinear corrections with an artificial varying cutoff k_{\max} . The result is shown in fig. 7.3. As we would expect, the error diverges as k_{\max} goes to zero, since virtually no information is left in that limit and h is unconstrained. Furthermore, the error approaches a constant if k_{\max} becomes sufficiently large, which also makes sense. However, the dip at around $k = 0.7h/\text{Mpc}$ is completely unexpected:

| | Linear | | | Halofit | | trgfast | | HaloCut | |
|------------|----------|--------|------|---------|------|---------|------|---------|------|
| | fiducial | abs. | rel. | abs. | rel. | abs. | rel. | abs. | rel. |
| Ω_b | 0.049 | 0.0069 | 14.% | 0.005 | 10.% | 0.0081 | 17.% | 0.0044 | 9.% |
| Ω_c | 0.2678 | 0.0088 | 3.3% | 0.0068 | 2.5% | 0.01 | 3.9% | 0.0071 | 2.6% |
| h | 0.6704 | 0.051 | 7.7% | 0.048 | 7.1% | 0.068 | 10.% | 0.013 | 1.9% |
| n_s | 0.9619 | 0.022 | 2.3% | 0.038 | 3.9% | 0.028 | 2.9% | 0.012 | 1.2% |
| σ_8 | 0.8347 | 0.011 | 1.3% | 0.0084 | 1.% | 0.017 | 2.1% | 0.0017 | 0.2% |

Table 7.3 Same as table 7.1, but now the linear power spectrum comes from CLASS.

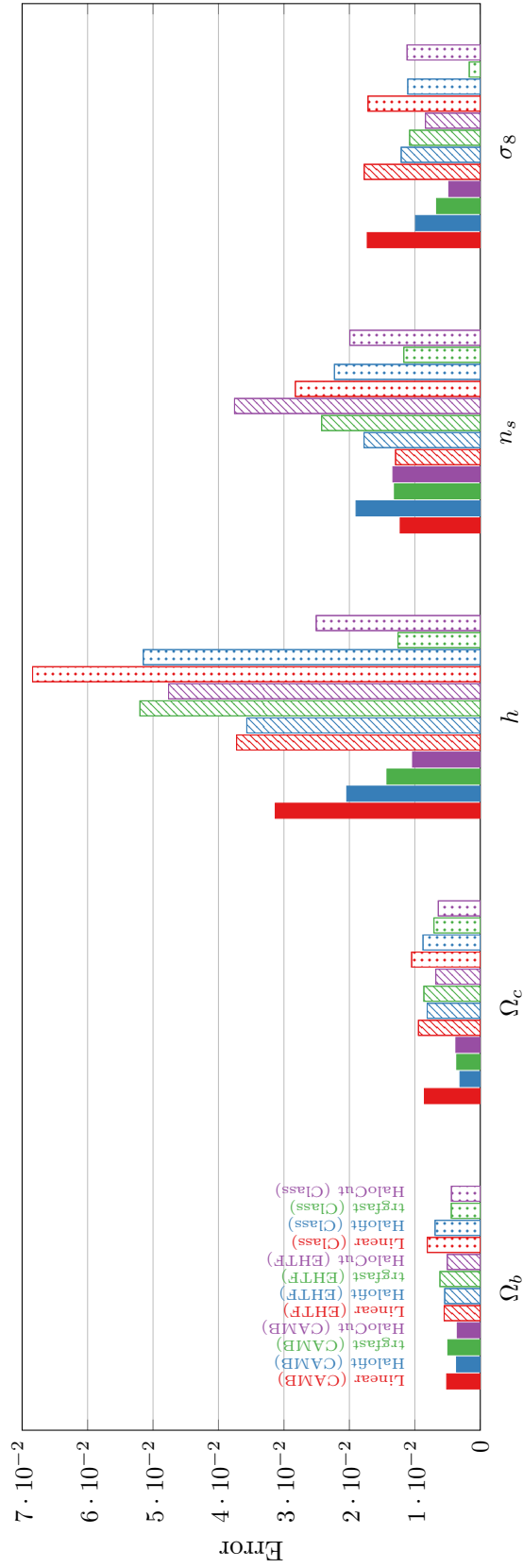


Figure 7.2 Comparison of the fully marginalized Fisher matrix errors on our parameters using different sources for the linear and nonlinear matter power spectrum. The solid, shaded, dotted bars represent the errors obtained by using CAMB, EHTF, CLASS, respectively. Red stands for no nonlinear corrections, blue for HaloFit, green for trgfast, purple for HaloCut.

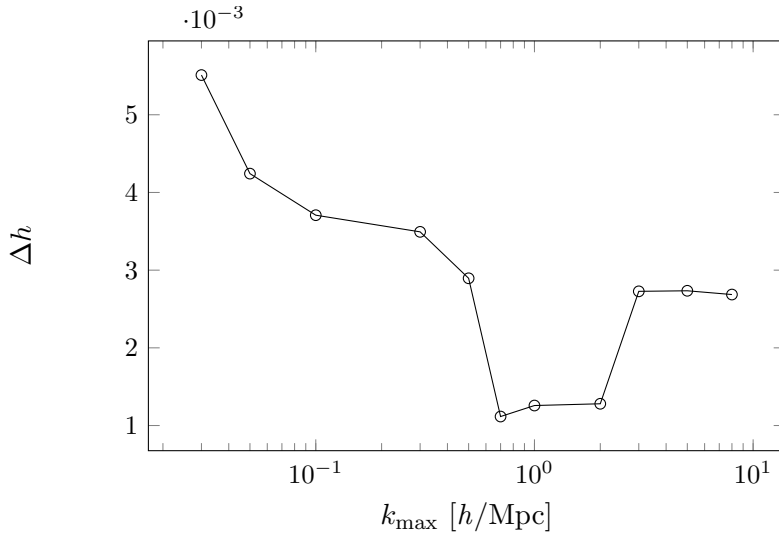


Figure 7.3 The error on the Hubble constant h with all other parameters fixed as a function of the power spectrum cutoff k_{\max} . It features a peculiar minimum at $k_{\max} \approx 0.7$.

If decreasing the cutoff more and more means removing information from the Fisher matrix, the errors should strictly increase.

Note that to save computation time, we only computed the *fixed* error on h , not the marginalized one. The fully marginalized error requires all elements of the Fisher matrix to be computed, while fixed error requires only the element on the diagonal corresponding to h . It is reasonable to assume that the dip would also be observed if we chose to compute the marginalized error. Indeed, using only the fixed error has the advantage of removing the unpredictable effects of a matrix inversion from the problem.

Still, even just one entry of the Fisher matrix is a rather complicated expression involving the trace of the product of four matrices which depend not just on the matrix inverse of the convergence power spectrum but also on its derivative. Thus, it is not immediately obvious how changing the cutoff affects the Fisher matrix. Of course, despite the best of our abilities, numerical noise may still be an explanation, albeit unlikely. Further studies are needed to explain this behavior.

Apart from this unexpected dip, we see in fig. 7.2 that even the choice of the source of the linear matter power spectrum matters. For instance, compare all red bars for Δh : No nonlinear corrections were used, but the resulting error on h can vary by a factor of two if one decides to use CLASS instead of CAMB. It also shows that using nonlinear corrections can sometimes increase the error (in the case of n_s) or decrease the error (in the case of Ω_c).

Summarizing, we can say that the choice of tools to compute the linear matter power spectrum and nonlinear corrections to it is far from trivial. However, these results should be considered preliminary and a more in depth analysis is required to convincingly rule out that this effect is merely a numerical artifact.

Chapter 8

Model-independent constraints on η

In this chapter, we compute the weak lensing Fisher matrix for the anisotropic stress η (not to be confused with the e-folding time in the previous part) in a completely model-independent way, where η is allowed to vary both with redshift and scale. The results presented in the following have been published in Amendola et al. (2014).

8.1 Observables

The long-heralded era of precision cosmology has arguably arrived on the day the results from the measurements by the Planck satellite at the percent level were published (Planck Collaboration et al., 2013). But until now, all measurements were merely of parameters of cosmological models, such as the matter density $\Omega_{\text{m}0}$, the equation of state w , and so on. Today, we almost find ourselves in the fortunate position where the precision of cosmological observations have reached such a high quality that model-independent observations may soon become feasible. Amendola et al. (2013) determined that besides the expansion rate $H(z)$, only these three physical quantities are actually observable:

$$A = Gb\delta_{\text{m}0}, \quad (8.1)$$

$$R = Gf\delta_{\text{m}0}, \quad (8.2)$$

$$L = \Omega_{\text{m}0}GY(1 + \eta)\delta_{\text{m}0}. \quad (8.3)$$

Here, b is the linear bias, defined by $\delta_{\text{gal}} = b(k, a)\delta_{\text{m}}$, G is the linear growth function, and $f = G'/G$ is the growth rate. Primes indicate a derivative with respect to $\log a$. The function $\eta(z, k)$ is the gravitational slip or anisotropic stress and defined as the ratio of the two gravitational potentials (see eq. (2.36)), i.e.

$$\eta(k, a) \equiv -\frac{\Phi}{\Psi}, \quad (8.4)$$

whereas Y is the effective dimensionless Newton's constant,

$$Y(k, a) \equiv -\frac{2k^2\Psi}{3\Omega_{\text{m}}\delta_{\text{m}}}. \quad (8.5)$$

The latter two quantities measure the influence of dark energy on the cosmological perturbations (Amendola et al., 2007).

We can remove the model-dependency on A , R , and L by forming ratios where δ_{m0} cancels out, i.e.

$$P_1 = R/A = f/b, \quad (8.6)$$

$$P_2 = L/R = \Omega_{m0} Y(1 + \eta)/f, \quad (8.7)$$

$$P_3 = R'/R = f + f'/f, \quad (8.8)$$

since it contains the transfer function which requires a parameterized model for dark energy. These ratios are also independent of the initial conditions or cosmic variance because the only stochastic source, the matter perturbations, cancels out. However, the overall normalization factor of $H(z)$ is also subject to cosmic variance (Marra et al., 2013), so we will use $E(z) = H(z)/H_0$ instead.

These quantities can be used to form an expression for the anisotropic stress (Motta et al., 2013),

$$1 + \eta = \frac{3P_2(1+z)^3}{2E^2(P_3 + 2 + \frac{E'}{E})}. \quad (8.9)$$

As we will see later, the uncertainty on P_3 can be so large that the numerator in eq. (8.9) can have an error range extending into the negative (see the appendix of Amendola et al. (2014) for details), causing unstable behavior. We will thus use a more useful but equivalent quantity defined as

$$\bar{\eta} \equiv \frac{2}{1 + \eta} = \frac{2\Psi}{\Psi - \Phi}. \quad (8.10)$$

To be as general as possible, we also want to investigate the Horndeski Lagrangian, which is the most general second-order scalar-tensor theory (Horndeski, 1974; Deffayet et al., 2011). It corresponds to the action (Felice and Tsujikawa, 2013)

$$\mathcal{S} = \int d^4x \sqrt{-\det g_{\mu\nu}} \left[\frac{M_{\text{p}}^2}{2} R + P(\phi, X) - G_3(\phi, X) \square\phi + \mathcal{L}_4 + \mathcal{L}_5 \right], \quad (8.11)$$

with

$$\mathcal{L}_4 = G_4(\phi, X) R + G_{4,X} [(\square\phi)^2 - (\nabla_\mu \nabla_\nu \phi) (\nabla^\mu \nabla^\nu \phi)], \quad (8.12)$$

$$\begin{aligned} \mathcal{L}_5 = & G_5(\phi, X) G_{\mu\nu} (\nabla^\mu \nabla^\nu \phi) - \frac{1}{6} G_{5,X} [(\square\phi)^3 - 3(\square\phi) (\nabla_\mu \nabla_\nu \phi) (\nabla^\mu \nabla^\nu \phi) \\ & + 2(\nabla^\mu \nabla_\alpha \phi) (\nabla^\alpha \nabla_\beta \phi) (\nabla^\beta \nabla_\mu \phi)]. \end{aligned} \quad (8.13)$$

where P and G_i 's ($i = 3, 4, 5$) are functions in terms of the scalar field ϕ and $X = -\partial^\mu \phi \partial_\mu \phi / 2$ with the partial derivatives $G_{i,X} \equiv \partial G_i / \partial X$. The Horndeski Lagrangian covers a wide range of models, in all of which the anisotropic stress (in the quasi-static limit) can be written as (Amendola et al., 2014)

$$\eta = h_2 \left(\frac{1 + k^2 h_4}{1 + k^2 h_5} \right). \quad (8.14)$$

The $h_{2,4,5}$ are in general time-dependent. Fortunately, we can restrict ourselves to studying the compact expression in eq. (8.14) and do not need to worry about the complicated Lagrangian in eq. (8.11).

The goal now is to make forecasts for the observable η as measured by future supernova, galaxy cluster, and weak lensing observations, where we want to focus only on the latter. The other probes are covered in Amendola et al. (2014). As intermediate results we will also have errors on \bar{A} , \bar{R} , \bar{L} , and E . In particular, we will cover four different cases:

1. We let η vary with redshift and scale, i.e. $\eta = \eta(z, k)$.
2. We fix the scale and consider η to be a function of redshift only, i.e. $\eta = \eta(z)$.
3. We assume η to be a constant, corresponding to Λ CDM and other models with a non-clustering dark energy component.
4. We investigate the Horndeski parameterization.

In all cases the fiducial value for η (and $\bar{\eta}$) is unity at all redshifts and scales.

8.2 Binning

Before we proceed, let us define a clear convention on how we use indices and their notation in the following. There are many quantities with indices and it become confusing quickly as to what they are referring to.

We use α and β exclusively for indexing the Fisher matrix parameters, e.g. Θ_α . Redshift binning appears in two different contexts here. First, we define bins in which we assume the functions $L(z)$, $R(z)$, $A(z)$ and $E(z)$ to be constant. These we will link to an index with the letters a and b , i.e. $L_a = L(z_a)$. Second, we divide up the redshift space for weak lensing tomography (see section 6.3). This is arbitrary in principle and not necessarily dependent of the first binning, as they will be summed up in the final expression for the Fisher matrix in eq. (6.59), but we will use the same bins. Only to avoid confusion their indices will be labeled differently with i , j , l , or m .

When using the variables z , k , or ℓ , their regular symbol refers to the lower boundary of a bin while the barred symbol refers to the center of the bin, i.e. z_a would be the lower boundary of the a th bin and \bar{z}_a would be at its center. Let us now elaborate on the binning in redshift space. With a Euclid-like survey in mind, we consider redshift bins from $z = 0.5$ to $z = 2.1$ with a bin size of $\Delta z = 0.2$. However, since we encountered coincidental degeneracies due to low statistics in the high redshift bins, we merge the last three bins into one larger bin. Thus, we have $N_z = 6$ redshift bins. These will be used for the weak lensing tomography as well as the discretization of the observables.

Since we also want to obtain uncertainties on the observables under the assumption that they can vary with scale, we require the k -space to be binned as well. In the end we also want to constrain h_2 , h_4 , and h_5 , and if we allow these three parameters to vary with redshift, we have 18 parameters in total. However, since h_4 and h_5 are degenerate, we will later fix h_5 and the total number of parameters will be 12. Thus, in order to be able to project η on the Horndeski parameters, we require at least two k -bins, such that $\eta(k, z)$ is also determined by the same number of parameters. We decided to use three k -bins at each redshift bin. For k -bins we use the indices μ and ν .

For illustration, this notation enables us to write quantities such as $\bar{L}_{a\mu}$, which equals the value of $\bar{L}(\bar{z}_a, k_\mu)$, where a runs from 1 to 6 and μ runs from 1 to 3.

The k -bins were formed by splitting up the interval $[k_{\min}, k_{\max}]$ into three bins such that $\Delta \log k = \text{const}$. The maximum wavenumber k_{\max} was chosen to be well inside the linear regime for each redshift bin, i.e. by imposing $\sigma^2(\mathcal{R}) = 0.35$ with $\mathcal{R} = \pi/2k$ (Seo and Eisenstein, 2003), where

$$\sigma^2(\mathcal{R}) = \int_0^\infty \frac{dk}{k} \frac{k^3}{2\pi^2} P_{\delta_m}(k) |\tilde{W}_{\mathcal{R}}(k)|^2. \quad (8.15)$$

Here, $\tilde{W}_{\mathcal{R}}(k)$ is the Fourier transform of the spherical top hat function (Eisenstein and Hu, 1999), i.e.

$$\tilde{W}_{\mathcal{R}}(k) = \frac{3}{(k\mathcal{R})^3} (\sin k\mathcal{R} - k\mathcal{R} \cos k\mathcal{R}). \quad (8.16)$$

The minimum wavenumber was set to $0.007h/\text{Mpc}$. Tightly linked to the k -bins are the ℓ -bins, which are needed in the sum of the Fisher matrix. We will discuss how to derive them later.

8.3 Weak lensing Fisher matrix

We divide the observables A , R , and L by δ_{t_0} each and write them with a bar (i.e. $\bar{A} \equiv A/\delta_{t_0}$, and so on), because when we consider the case where η is independent of k , these quantities are not well defined. These new variables do not form observable quantities as described in Amendola et al. (2013), but the ratios P_1 , P_2 , and P_3 do.

The expression for the convergence power spectrum in eq. (6.52) can be approximated as

$$P_{ij}(\ell) = H_0 \int_0^\infty p_{ij}(z, \ell) dz \approx H_0 \sum_a \frac{\Delta z_a}{E_a} K_i K_j \bar{L}^2 \delta_{t,0}^2(\bar{z}_a, k(\ell, \bar{z}_a)). \quad (8.17)$$

Here we defined the integrand as

$$p_{ij}(z, \ell) \equiv \frac{K_i(z) K_j(z)}{E(z)} \bar{L}(z)^2 \delta_{t,0}^2(z, k(\ell, z)), \quad (8.18)$$

where the relevant scale depends on the multipole ℓ and the comoving distance at the given redshift $d_C(z)$. In particular we have

$$k(\ell, z) = \frac{\ell}{\pi d_C(z)} \quad \text{and} \quad K_i(z) = \frac{3}{2} H_0 (1+z) W_i(z). \quad (8.19)$$

The window functions $W_i(z)$ were defined in eq. (6.53). The Fisher matrix parameters are $\Theta = (\bar{L}(\bar{z}_1), E(\bar{z}_1), \dots)$. Here, ℓ is being summed from 5 to ℓ_{\max} with $\Delta \log \ell = 0.1$, where ℓ_{\max} corresponds to the value listed in table 8.2 for the redshift bin a or b — whichever is smaller. For the matter power spectrum we use the fitting function from Eisenstein and Hu (1999) with no nonlinear corrections.

The issue of how to choose ℓ_{\max} is not quite straight forward. We want ℓ_{\max} to be related to k_{\max} for each particular redshift bin, but the relationship between ℓ and k also depends on z , see eq. (8.19). We see that ℓ is not determined until we fix z . For

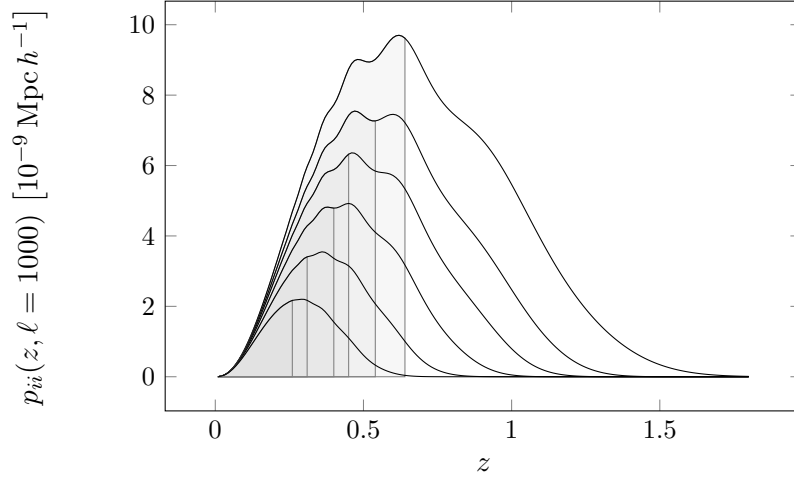


Figure 8.1 The integrands and their median.

this value we choose the median redshift with respect to the integrand $p_{ii}(z, \ell)$, which in turn depends on ℓ and on the redshift bin. Its defining equation is

$$\int_0^{z_{\text{med}}} p_{ii}(z, \ell) dz = \frac{1}{2} \int_0^{\infty} p_{ii}(z, \ell) dz. \quad (8.20)$$

Finally, k_{max} also depends on the redshift via the relation

$$\sigma^2(\mathcal{R}(z, k)) = 0.35. \quad (8.21)$$

Thus, we have the quantities z_{med} , k_{max} , and ℓ_{max} which shall simultaneously satisfy all three conditions above (replacing $z = z_{\text{med}}$, $k = k_{\text{max}}$, and $\ell = \ell_{\text{max}}$ where applicable). To find their values we use recursion.

We begin with $z_{\text{med}} = 1$, compute the k_{max} for this redshift as before by imposing $\sigma^2(\mathcal{R}) = 0.35$, solve eq. (8.19) for ℓ , and compute $z_{\text{med}}(\ell, i)$. We repeat this step until the value for z_{med} converges with an accuracy of approximately 1%. A list of the values for ℓ_{max} as well as z_{med} used in each redshift bin can be found in table 8.2. The integrands along with their median value are depicted in fig. 8.1.

Computing the Fisher matrix requires the derivative of the convergence power spectrum. To find them, we divide the integral in eq. (8.17) into N_z integrals that each cover one redshift bin. We will use the full expression for \bar{L} when computing the integral over that redshift bin, but when taking the derivative we assume the integral of bin a depends quadratically on $\bar{L}(z_a)$ (see eq. (8.17)). Thus, we obtain for the derivative a simple expression where we multiply the integral over the a th bin by $\frac{2}{\bar{L}(z_a)}$, i.e.

$$\frac{\partial P_{ij}(\ell)}{\partial \bar{L}(\bar{z}_a)} = \frac{2}{\bar{L}(\bar{z}_a)} \int_{z_a}^{z_{a+1}} p_{ij}(z, \ell) dz. \quad (8.22)$$

We found that this gives a more accurate result than simply assuming that $\bar{L}(z)$ is constant over one redshift bin.

Since E appears in the comoving distance, it is more complicated for the derivatives

| Parameter | Value | Description |
|------------------|-------|--|
| f_{sky} | 0.375 | Fraction of sky coverage |
| $\delta_z/(1+z)$ | 0.05 | Photometric redshift error |
| n_θ | 35 | Areal galaxy density per square arc minute |
| z_\star | 0.9 | Median redshift of the galaxy distribution |

Table 8.1 Survey parameters according to Euclid specifications (Laureijs et al., 2011).

| \bar{z} | ℓ_{max} | z_{med} | \bar{L} | $\Delta\bar{L}$ | $\Delta\bar{L}(\%)$ | E | ΔE | $\Delta E(\%)$ |
|-----------|---------------------|------------------|-----------|-----------------|---------------------|------|------------|----------------|
| 0.6 | 311 | 0.26 | 0.342 | 0.0044 | 1.3 | 1.37 | 0.0062 | 0.46 |
| 0.8 | 385 | 0.31 | 0.311 | 0.0044 | 1.4 | 1.53 | 0.0069 | 0.45 |
| 1.0 | 515 | 0.40 | 0.285 | 0.0059 | 2.1 | 1.72 | 0.017 | 0.96 |
| 1.2 | 609 | 0.45 | 0.262 | 0.0059 | 2.3 | 1.92 | 0.029 | 1.5 |
| 1.4 | 760 | 0.54 | 0.242 | 0.014 | 5.7 | 2.14 | 0.029 | 1.4 |
| 1.8 | 959 | 0.64 | 0.210 | 0.035 | 16 | 2.62 | 0.077 | 3.0 |

Table 8.2 Errors on E and \bar{L} from weak lensing only (with six redshift bins) and a list of the value ℓ_{max} used at each redshift together with the corresponding z_{med} value.

of P_{ij} with respect to $E(\bar{z}_a)$. We substitute the regular definition of E by an interpolating function that goes smoothly through all points $(\bar{z}_a, E(\bar{z}_a))$ and $(0, 1)$. Note that this step removes the last model dependency on ΛCDM : Instead of depending on Ω_m , E now depends on the values of all $E(\bar{z}_a)$. Consequently, so do all functions that depend on E , in particular the comoving distance and consequently the window functions $K_i(z)$. The derivatives are then obtained by varying the fiducial values of $E(\bar{z}_a)$ while keeping $L = \bar{L}\delta_{\text{m}0}/\sigma_8$ fixed so that we again do not include the derivative of $\delta_{\text{m}0}^2$ with respect to k .

This concludes the construction of the weak lensing Fisher matrix. The errors from weak lensing only are listed in table 8.2 and visualized in fig. 8.2. To see how well the individual probes constrain E , see table 8.3. In the next step, the Fisher matrix will be combined with corresponding matrices for galaxy clustering and supernova surveys, which provide additional constraints on \bar{A} , \bar{R} , and E (for details, see Amendola et al. (2014)). The parameters of the total Fisher matrix written out explicitly are

$$\Theta = (A_{11}, R_{11}, L_{11}, A_{12}, R_{12}, L_{12}, A_{13}, R_{13}, L_{13}, E_1, \dots, A_{63}, R_{63}, L_{63}, E_6). \quad (8.23)$$

Here it is important that we properly keep track of the indices, as we are effectively flattening a three index quantity (one for the observable, one for the redshift bin, and one for the k -bin) into a one index quantity.

8.4 Projection on η

All is left to do now is to project the Fisher matrix for \bar{A} , \bar{R} , \bar{L} and E onto $\bar{\eta}$. We will do so by including some interesting intermediate steps, in particular we will project onto (P_1, P_2, P_3, E) as well as $(P_1, P_2, P_3, E'/E)$ before moving to $\bar{\eta}$.

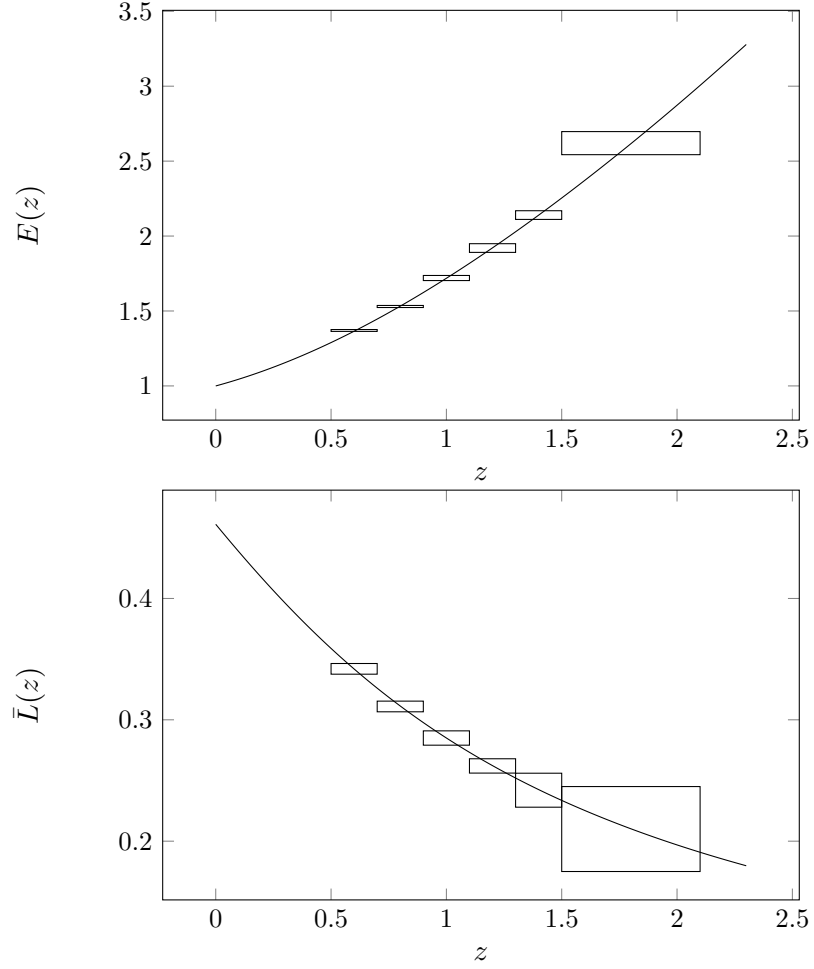


Figure 8.2 Errors on $E(z)$ and $\bar{L}(z)$ from the weak lensing Fisher matrix only.

| z | E | WL | | GC | | SN | | WL+GC | | WL+GC+SN | |
|-----|------|------------|----------------|------------|----------------|------------|----------------|------------|----------------|------------|----------------|
| | | ΔE | $\Delta E(\%)$ | ΔE | $\Delta E(\%)$ | ΔE | $\Delta E(\%)$ | ΔE | $\Delta E(\%)$ | ΔE | $\Delta E(\%)$ |
| 0.6 | 1.37 | 0.0062 | 0.46 | 0.12 | 8.5 | 0.0026 | 0.19 | 0.0062 | 0.45 | 0.0023 | 0.16 |
| 0.8 | 1.53 | 0.0069 | 0.45 | 0.073 | 4.8 | 0.0041 | 0.27 | 0.0068 | 0.44 | 0.0029 | 0.19 |
| 1.0 | 1.72 | 0.017 | 0.96 | 0.058 | 3.4 | 0.0086 | 0.50 | 0.016 | 0.91 | 0.0067 | 0.39 |
| 1.2 | 1.92 | 0.029 | 1.5 | 0.050 | 2.6 | 0.016 | 0.83 | 0.024 | 1.2 | 0.012 | 0.65 |
| 1.4 | 2.14 | 0.029 | 1.4 | 0.051 | 2.4 | 0.028 | 1.3 | 0.022 | 1.0 | 0.017 | 0.78 |
| 1.8 | 2.62 | 0.077 | 3.0 | 0.061 | 2.3 | - | - | 0.046 | 1.8 | 0.043 | 1.7 |

Table 8.3 Errors on E from the three probes.

When moving from one set of parameters Θ to a different, but equivalent set $\check{\Theta}$ that describes the same model, the new Fisher matrix turns out to be (Amendola and Tsujikawa, 2010)

$$\check{F}_{\alpha\beta} = F_{\gamma\delta} \frac{\partial\Theta_\gamma}{\partial\check{\Theta}_\alpha} \frac{\partial\Theta_\delta}{\partial\check{\Theta}_\beta}, \quad (8.24)$$

or simply

$$\check{F} = J^T F J \quad (8.25)$$

with the Jacobian

$$J_{\alpha\beta} = \frac{\partial\Theta_\alpha}{\partial\check{\Theta}_\beta}. \quad (8.26)$$

One variable needs special attention: P_2 depends on the derivative of R with respect to $\log a$ (see eq. (C.11)), i.e.

$$R'(z) = -(1+z)\partial R/\partial z. \quad (8.27)$$

This expression can be approximated by

$$R'(z) = -(1+z)(R(z+\Delta z) - R(z))/\Delta z \quad (8.28)$$

and similarly for $E'(z)$ later when we project onto $\bar{\eta}$. We use the redshift bin width for Δz such that we can write

$$R'(z_a) = -1(1+z_a)(R(z_{a+1}) - R(z_a))/(z_{a+1} - z_a), \quad (8.29)$$

making it possible to project from the old set of parameters $(\bar{A}, \bar{R}, \bar{L}, E)$ onto the one (P_1, P_2, P_3, E) . However, now we cannot use the last redshift bin anymore, since every value of R' at one redshift bin depends on its value at the following bin. As an interesting intermediate step, we also project onto $(P_1, P_2, P_3, E'/E)$. For the final result, we project onto $(P_1, P_2, \bar{\eta}, E)$ and marginalize over the remaining parameters to obtain the uncertainties on $\bar{\eta}$.

After carefully computing the derivatives and constructing the Jacobian, we obtain the error estimates on the new set of parameters which can be seen in table 8.4.

For the case of a constant $\bar{\eta}$, we have $\bar{\eta}_a = \bar{\eta}$ and the Jacobian is simply

$$\frac{\partial\bar{\eta}}{\partial\bar{\eta}_a} = 1. \quad (8.30)$$

In this case, we determined the error on $\bar{\eta}$ to be $\Delta\bar{\eta} = 0.010$.

For the Horndeski case, we project $\bar{\eta}$ onto (h_2, h_4) which were defined in eq. (8.14) with the fiducial values being $h_2 = 1$ and $h_4 = 0$. The reason why we cannot include h_5 in the Fisher matrix analysis is that h_4 and h_5 are degenerate at the fiducial model where $h_4 = h_5 = 0$ and the Fisher matrix would become singular. Also, for very large k , the expression for η reduces to

$$\eta = h_2 \frac{h_4}{h_5}, \quad (8.31)$$

where it is obvious that all parameters are fully degenerate. Note that we are measuring k in units of $0.1h/\text{Mpc}$ (we can think of it as dividing k by $k_\star = 0.1h/\text{Mpc}$ in eq. (8.14)), such that h_2 and h_4 are dimensionless. The results can be found in table 8.9.

| \bar{z} | P_1 | ΔP_1 | $\Delta P_1(\%)$ | P_2 | ΔP_2 | $\Delta P_2(\%)$ | P_3 | ΔP_3 | $\Delta P_3(\%)$ |
|-----------|-------|--------------|------------------|-------|--------------|------------------|-------|--------------|------------------|
| 0.6 | 0.766 | 0.012 | 1.6 | 0.729 | 0.013 | 1.8 | 0.134 | 0.13 | 99 |
| 0.8 | 0.819 | 0.010 | 1.2 | 0.682 | 0.011 | 1.6 | 0.317 | 0.12 | 38 |
| 1.0 | 0.859 | 0.0093 | 1.1 | 0.650 | 0.011 | 1.7 | 0.460 | 0.12 | 26 |
| 1.2 | 0.888 | 0.0092 | 1.0 | 0.628 | 0.014 | 2.3 | 0.569 | 0.13 | 23 |
| 1.4 | 0.911 | 0.010 | 1.1 | 0.613 | 0.020 | 3.3 | 0.654 | 0.11 | 16 |

Table 8.4 Fiducial values and errors for the parameters P_1 , P_2 , P_3 for every bin. The last bin has been omitted since R' is not defined there.

| \bar{z} | P_1 | ΔP_1 | $\Delta P_1(\%)$ | P_2 | ΔP_2 | $\Delta P_2(\%)$ | P_3 | ΔP_3 | $\Delta P_3(\%)$ |
|-----------|-------|--------------|------------------|-------|--------------|------------------|-------|--------------|------------------|
| 0.7 | 0.794 | 0.0079 | 0.99 | 0.703 | 0.0074 | 1.0 | 0.231 | 0.042 | 18 |
| 1.1 | 0.875 | 0.0067 | 0.77 | 0.638 | 0.0072 | 1.1 | 0.518 | 0.050 | 9.7 |
| 1.5 | 0.920 | 0.0099 | 1.1 | 0.607 | 0.010 | 1.7 | 0.688 | 0.048 | 7.0 |

Table 8.5 Same as table 8.4, but with four redshift bins. The last bin has again been omitted.

| \bar{z} | (E'/E) | $\Delta E'/E$ | $\Delta E'/E(\%)$ | $\bar{\eta}$ | $\Delta \bar{\eta}$ | $\Delta \bar{\eta}(\%)$ |
|-----------|----------|---------------|-------------------|--------------|---------------------|-------------------------|
| 0.6 | -0.920 | 0.022 | 2.4 | 1 | 0.11 | 11 |
| 0.8 | -1.04 | 0.046 | 4.4 | 1 | 0.091 | 9.1 |
| 1.0 | -1.13 | 0.099 | 8.7 | 1 | 0.090 | 9.0 |
| 1.2 | -1.21 | 0.12 | 10 | 1 | 0.097 | 9.7 |
| 1.4 | -1.26 | 0.09 | 7.1 | 1 | 0.073 | 7.3 |

Table 8.6 Fiducial values and errors for the parameters E'/E and $\bar{\eta}$ (the z -varying case) for every bin. The last bin has again been omitted.

| \bar{z} | (E'/E) | $\Delta E'/E$ | $\Delta E'/E(\%)$ | $\bar{\eta}$ | $\Delta \bar{\eta}$ | $\Delta \bar{\eta}(\%)$ |
|-----------|----------|---------------|-------------------|--------------|---------------------|-------------------------|
| 0.7 | -0.983 | 0.023 | 2.3 | 1 | 0.031 | 3.1 |
| 1.1 | -1.17 | 0.044 | 3.7 | 1 | 0.037 | 3.7 |
| 1.5 | -1.29 | 0.060 | 4.6 | 1 | 0.032 | 3.2 |

Table 8.7 Same as table 8.6, but with four redshift bins. The last bin has again been omitted.

| \bar{z} | μ | P_1 | ΔP_1 | $\Delta P_1(\%)$ | P_2 | ΔP_2 | $\Delta P_2(\%)$ | P_3 | ΔP_3 | $\Delta P_3(\%)$ | $\bar{\eta}$ | $\Delta \bar{\eta}$ | $\Delta \bar{\eta}(\%)$ |
|-----------|-------|-------|--------------|------------------|-------|--------------|------------------|-------|--------------|------------------|--------------|---------------------|-------------------------|
| 0.6 | 1 | | 0.14 | 18 | | 0.12 | 17 | | 1.4 | 1100 | 1 | 1.1 | 120 |
| | 2 | 0.766 | 0.032 | 4.1 | 0.729 | 0.030 | 4.1 | 0.134 | 0.33 | 240 | | 0.26 | 26 |
| | 3 | | 0.013 | 1.7 | | 0.015 | 2.0 | | 0.15 | 110 | | 0.12 | 12 |
| 0.8 | 1 | | 0.11 | 13 | | 0.092 | 13 | | 1.2 | 380 | | 0.93 | 93 |
| | 2 | 0.819 | 0.024 | 2.9 | 0.682 | 0.021 | 3.1 | 0.317 | 0.26 | 83 | 1 | 0.2 | 20 |
| | 3 | | 0.011 | 1.4 | | 0.013 | 1.9 | | 0.14 | 43 | | 0.1 | 10 |
| 1.0 | 1 | | 0.093 | 11 | | 0.076 | 12 | | 1.1 | 240 | | 0.82 | 82 |
| | 2 | 0.859 | 0.020 | 2.3 | 0.65 | 0.019 | 2.9 | 0.46 | 0.23 | 51 | 1 | 0.17 | 17 |
| | 3 | | 0.011 | 1.2 | | 0.012 | 1.8 | | 0.14 | 31 | | 0.1 | 11 |
| 1.2 | 1 | | 0.084 | 9.4 | | 0.074 | 12 | | 1.1 | 190 | | 0.78 | 78 |
| | 2 | 0.888 | 0.017 | 2.0 | 0.628 | 0.021 | 3.2 | 0.569 | 0.23 | 40 | 1 | 0.16 | 16 |
| | 3 | | 0.011 | 1.2 | | 0.017 | 2.7 | | 0.17 | 29 | | 0.12 | 12 |
| 1.4 | 1 | | 0.079 | 8.7 | | 0.084 | 14 | | 0.79 | 120 | | 0.55 | 55 |
| | 2 | 0.911 | 0.017 | 1.9 | 0.613 | 0.027 | 4.4 | 0.654 | 0.17 | 26 | 1 | 0.12 | 12 |
| | 3 | | 0.013 | 1.4 | | 0.023 | 3.8 | | 0.14 | 21 | | 0.094 | 9.4 |

Table 8.8 Here, the errors on P_1 , P_2 , P_3 and η are listed for the z, k -varying case at every redshift and every k -bin (using the index μ).

8.5 Discussion of the results

Summarizing, we find that the precision with which future Euclid-like surveys together with supernova data can constrain the observables P_1 , P_2 , P_3 , and E without assuming an underlying model can be at the sub-percent level (tables 8.4 and 8.5). Especially the errors on the dimensionless Hubble parameter E only exceed 1% at redshifts higher than 1.5, and even for a redshift of 2 the error is less than 2% (see table 8.3). As we mentioned in section 8.1, the error estimate on η is divided into four cases, which we now discuss individually.

Case 1: constant η

When we assume that η does not vary with redshift or scale, the combined constraints from all probes amount to 1%. This is, in fact, the best case scenario, as all the gathered information goes to constraining one single quantity.

Case 2: z -varying η

For P_1 and P_2 in the z -dependent case the order of magnitude of the uncertainties is similar to the ones on E , and they go down even more when moving from six redshift bins to four bins (covering the same redshift interval in total). Only for P_3 the errors are between 16% and 99% (tables 8.4 and 8.5), most likely due to the fact that it includes an explicit derivative. The increase of the error by an order of magnitude applies to $\bar{\eta}$

| \bar{z} | Δh_2 | Δh_4 |
|-----------|--------------|--------------|
| 0.6 | 0.58 | 0.56 |
| 0.8 | 0.44 | 0.32 |
| 1. | 0.37 | 0.22 |
| 1.2 | 0.35 | 0.18 |
| 1.4 | 0.25 | 0.1 |

Table 8.9 Absolute errors on h_2 and h_4 from all probes. Because of the degeneracy between h_5 and h_4 , h_5 has been fixed. The fiducial values are $h_2 = 1$ and $h_4 = 0$.

as well, which now has constraints between 7% and 11%.

Case 3: z, k -varying η

In the case where we also allow the observables to vary with k , all errors increase (see table 8.8), and are now typically between 1% and 20% for P_1 and P_2 . As for P_3 , it is barely constrained at all, having errors ranging from 20% to over 1000%. Similarly, $\bar{\eta}$ now has errors between 9% and 120%. This is to be expected, since trying to constrain more parameters from the same amount of information cannot yield smaller errors. In particular, the smallest error for one redshift bin in the z, k -varying case is always larger than the error on the same quantity in the same redshift bin in the z -varying case.

Case 4: Horndeski parameterization

In the Horndeski case we do have a model dependence, but it is a very general dark energy model. Here, $\bar{\eta}$ depends on k by construction, and we let the parameters h_2 and h_4 vary with redshift as well. The errors are displayed in table 8.9, and they are in the range 0.2-0.6.

8.6 Exclusion plot for the Horndeski parameters

We can go one step further in our analysis and exclude regions of the parameter space¹. For instance, we can fix h_5 to some value (not necessarily zero), project the Fisher matrix for $\bar{\eta}(k, z)$ from section 8.4 onto the parameter space spanned by h_2 and h_4 (now not time dependent), marginalize over h_4 and thus find the error on h_2 depending on h_5 . This way we plotted $\Delta h_2(h_4)$, $\Delta h_2(h_5)$, and $\Delta h_4(h_5)$ (see figs. 8.3 to 8.5). The area above the curve will be excluded by future Euclid-like surveys at the 1σ level. To find the 2σ level exclusion, simply multiply by 2.

In the next step, we will reinterpret the Horndeski parameterization as a Yukawa-like potential, where h_4 and h_5 are related to the coupling strength Q and range λ by

$$h_4 = (1 + Q)\lambda^2, \quad h_5 = \lambda^2 \quad (8.32)$$

such that the expression for η becomes

$$\eta = h_2 \frac{1 + (1 + Q)\lambda^2 k^2}{1 + \lambda^2 k^2}. \quad (8.33)$$

This makes sense when we consider that the scalar field in the Horndeski model induces a force which manifests itself as a correction to the Newton potential (Amendola and Tsujikawa, 2010), i.e. at the linear level the potential can be written as

$$\Psi(r) = -\frac{GM}{r} \left(1 + Qe^{-r/\lambda} \right). \quad (8.34)$$

Similarly to before, we can create a plot (fig. 8.6) that shows what regions in the Q - λ -plane can be excluded by future surveys by fixing λ to some fiducial value and marginalizing over h_2 . An interesting feature in the Q - λ exclusion plot is the minimum

¹The results of this section are to be published as part of a separate article, Guarnizo et al. (2014).

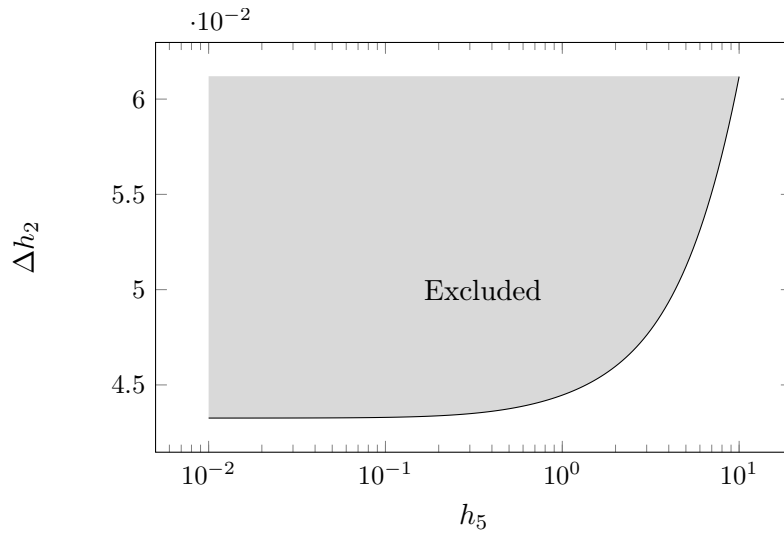


Figure 8.3 Exclusion plot for the Horndeski parameters h_2 and h_5 .

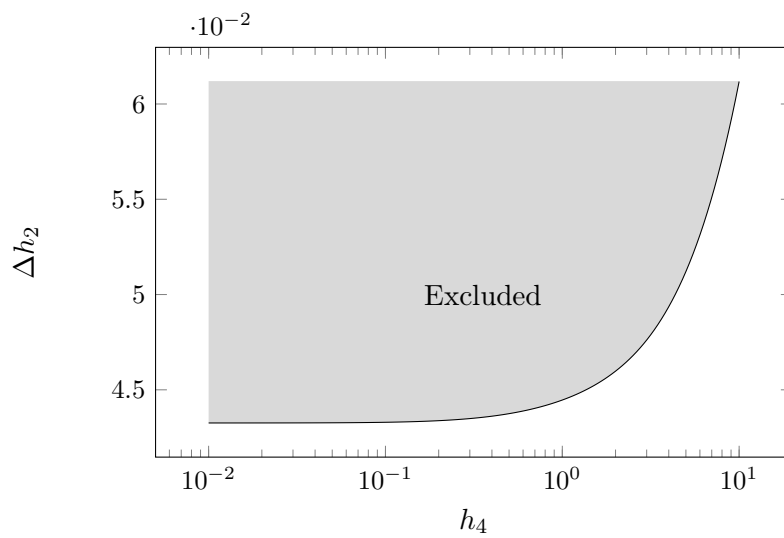


Figure 8.4 Exclusion plot for the Horndeski parameter h_2 and h_4 .

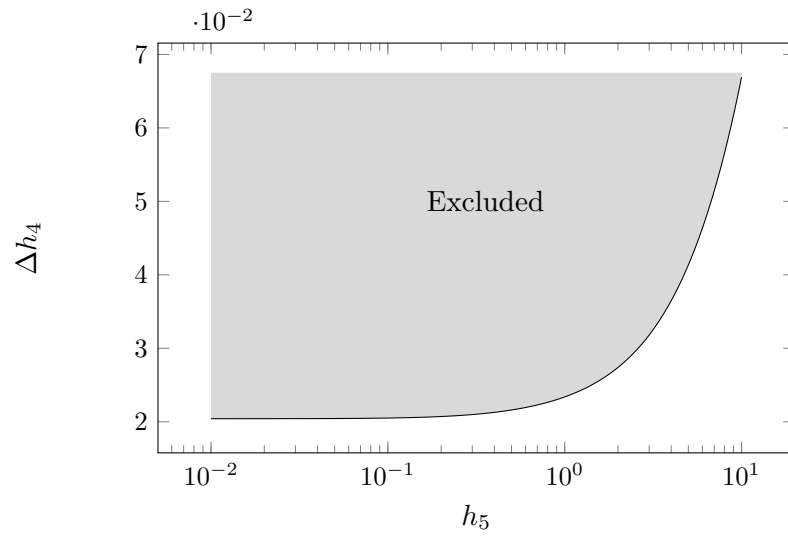


Figure 8.5 Exclusion plot for the Horndeski parameters h_4 and h_5 .

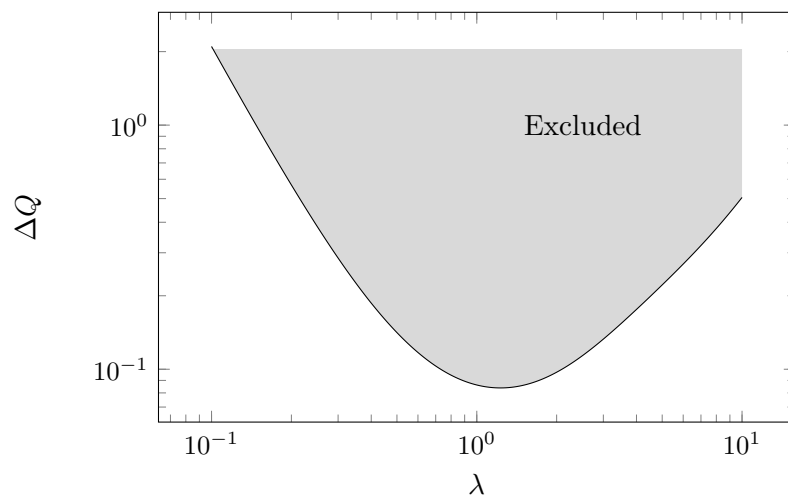


Figure 8.6 Exclusion plot for the Yukawa coupling strength Q and range λ . Note that k is measured in units of $0.1h/\text{Mpc}$ such that λ is dimensionless.

at $\lambda \approx 1.3$, which can be explained by the fact that $\eta \rightarrow h_2$ for $\lambda \ll 1$, so Q is not constrained at all. Also, for $\lambda \gg 1$, we have $\eta \rightarrow h_2(1 + Q)$, such that Q and h_2 are degenerate, which also causes the error on Q to diverge.

These plots can be helpful when trying to rule out certain dark energy models. For instance, in $f(R)$ models, the anisotropic stress takes the form eq. (8.14) (De Felice et al., 2011) and some sub-models correspond to one point in the h_4 - h_5 plane (fig. 8.5).

Chapter 9

Conclusion and outlook

In summary, we presented a new implementation of the TRG technique to compute nonlinear corrections to the matter power spectrum. We validated the results using not only similar implementations from independent groups but also data from N -body simulations for the standard Λ CDM model as well as coupled quintessence. We achieve an improvement over existing implementations that exceed a decrease of the runtime by a factor of 50. The code realizing this novel method has been made publicly available and was developed as a library to make it as simple as possible to integrate with existing and future code bases.

Regarding future applications of the code developed in part I, the natural progression would be to incorporate this in existing projects such as COSMOMC¹ (Lewis and Bridle, 2002), which provides a framework to perform Markov Chain Monte-Carlo explorations of the cosmological parameter space.

The code should be applicable as is to massive neutrino cosmologies. It is necessary to obtain matter perturbations and neutrino perturbations from CAMB at every time step (Lesgourgues et al., 2009), but these only appear in the background functions, such that TRGFAST should be able to handle this (possibly with the help of Cosmomathica) without modifying the code itself.

Another interesting project could be to check if the method presented here is also applicable using a more general approach. For instance, Saracco et al. (2010) extended the TRG formalism even further to include multiple matter components, such as cold and baryonic matter. If possible, a generalization of our code would be far from straightforward, but the techniques we developed may be of use. In particular, the expressions for the integrand and the TRG equations (see appendix F) would have to be generated again with taking the new matter component into account.

An investigation to extend the TRG formalism to the trispectrum level may also be worthwhile, however, it is far from clear if the resulting equations are manageable and whether a similar technique can be applied to make the reduction of the number of dimensions in the integrals feasible.

As part of the first application of TRGFAST to compute the weak lensing Fisher matrix, we found strong hints that the choice of what code to use for the linear or nonlinear matter power spectrum may be crucial to the accuracy with which we can estimate the errors from future surveys. Since the analysis was not completely conclusive, it is advisable to do further research in this direction. It may be of great interest

¹<http://cosmologist.info/cosmomc/>

to the cosmological community.

Besides the work on TRG, we found Fisher matrix constraints on the anisotropic stress η (or the closely related quantity $\bar{\eta}$) by future Euclid-like surveys in an entirely model-independent way. Even when avoiding the issue of initial conditions and after removing the model-dependence from the expansion history, we find that the constraints placed on the anisotropic stress can be quite low. The error on $\bar{\eta}$ is as low as 1% when assuming it does not depend on time or scale. When $\bar{\eta}$ is allowed to vary with redshift, the constraints range from 7% to 11% or are around 3%, depending on the number of redshift bins. When also allowing $\bar{\eta}$ to vary with scale, the constraints loosen to between 10% and 120%. In the Horndeski case, the errors on two of the Horndeski parameters, h_2 and h_4 , are between 10% and 60%.

Now it is interesting to see where certain $f(R)$ models lie in the exclusion plot for h_4 and h_5 to find out with what confidence level they could be ruled by Euclid.

Appendix A

Cosmomathica

To carry out the research, several third party software packages or algorithms had to be recruited: CLASS, CAMB, EHTF, Copter, Coyote interpolator, and Halofit. Another heavily used tool was MATHEMATICA, where the data produced by the aforementioned codes typically was imported and processed. Therefore, it made sense to use the MATH-LINK technology and develop a MATHEMATICA package that provides a convenient way to call these tools in the form of MATHEMATICA statements. Most of these packages have different conventions when passing input parameters and/or outputting the results.

Since some members of the cosmology community may benefit from having such a unifying package, we release it alongside TRGFAST on GitHub¹ under the name *Cosmomathica*. It may be merged with TRGFAST in a future version. A demonstration on how to set up Cosmomathica and how to use each external software package has been included in form of a MATHEMATICA notebook.

¹<https://github.com/User0815/cosmomathica>

Appendix B

Splines

Since splines take such a central role in our algorithm used for solving the TRG equations, we want to include a short treatment here. For a comprehensive text on splines, see Knott (2000).

Splines are needed when we want to smoothly interpolate a function given in the form of tabulated values (x_i, y_i) with $1 \leq i \leq n$. In the context of splines, these value pairs are often called “knots” (since they do not have to correspond to a function), but we will refer to them as “sampling points.” The idea is to define n polynomials $p_i(x)$ of degree D with

$$p_i(x) = \sum_{k=0}^D a_{i,k} x^k \quad (\text{B.1})$$

for each interval which make up a piecewise function, called *splines of order D* , such that

$$\begin{aligned} p_i(x_i) &= y_i && \text{for } 1 \leq i \leq n-1, \\ p_i(x_{i+1}) &= y_{i+1} && \text{for } 1 \leq i \leq n-1, \text{ and} \\ p_i^{(k)}(x_{i+1}) &= p_{i+1}^{(k)}(x_{i+1}) && \text{for } 1 \leq k \leq D-1, 1 \leq i \leq n-2. \end{aligned} \quad (\text{B.2})$$

The first two equations simply impose continuity, while the third equation ensures that the spline is $D-1$ times differentiable at each sampling point. Note that we have $(D+1)(n-1)$ degrees of freedom in total, but that these conditions only fix $2(n-1) + (D-1)(n-2)$ of them.

For the case $D=1$ the number of degrees of freedom matches the number of variables and we have linear interpolation, which is straight-forward. (Even though continuity cannot be guaranteed, the case $D=0$ is sometimes useful but trivial.) For $D=2$ the above conditions are already not sufficient to uniquely determine all coefficients $a_{i,k}$. In general, the discrepancy is $D-1$.

In order to get a fully determined system of equations, we need to impose $D-1$ more conditions, such as choosing the slope or the curvature of the splines in either the first or the last sampling point, or both. Then eq. (B.2) forms a solvable system of linear equations.

Splines of higher order tend to oscillate, while splines of low order might not be flexible enough to interpolate the given data. As it turns out, there is a sweet spot in the trade off between oscillation control and flexibility for cubic splines, where we

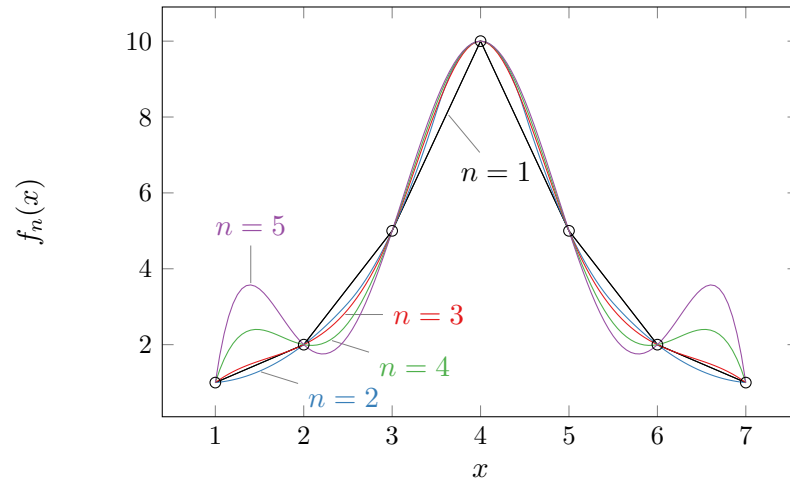


Figure B.1 Splines of different degree n interpolating an arbitrarily chosen set of sampling points. While there is no objective measure to determine the “best” degree, the cubic splines appear remarkably smooth.

impose the two additional conditions of vanishing curvature at both end points. This frequently used configuration allegedly corresponds to a flexible ruler (a spline) that is clamped through a set of nails on a board of wood (Thompson and Soni, 1998), which is why this special case is also called *natural splines*. For an example of splines interpolating a given set of sampling points, see fig. B.1.

Appendix C

Time variables and derivatives

An overview of some useful identities:

$$a(z) = \frac{1}{1+z} \quad (\text{C.1})$$

$$\eta(a) = \log\left(\frac{a}{a_{\text{ini}}}\right) \quad (\text{C.2})$$

$$a(\eta) = a_{\text{ini}} \exp(\eta) \quad (\text{C.3})$$

$$\eta(z) = \log\left(\frac{1+z_{\text{ini}}}{1+z}\right) \quad (\text{C.4})$$

$$z(\eta) = \frac{1+z_{\text{ini}}}{\exp(\eta)} - 1 \quad (\text{C.5})$$

$$H = \frac{d \log a}{dt} \quad (\text{C.6})$$

$$\mathcal{H} = \frac{d \log a}{d\tau} = aH \quad (\text{C.7})$$

$$\frac{\partial}{\partial \eta} = \mathcal{H}^{-1} \frac{\partial}{\partial \tau} \quad (\text{C.8})$$

$$\frac{\partial}{\partial \tau} = \frac{1}{a} \frac{\partial}{\partial t} \quad (\text{C.9})$$

$$\frac{\partial}{\partial a} = -\frac{1}{a^2} \frac{\partial}{\partial z} \quad (\text{C.10})$$

$$\frac{\partial}{\partial \eta} = a \frac{\partial}{\partial a} = -\frac{1}{a} \frac{\partial}{\partial z} = -(1+z) \frac{\partial}{\partial z} \quad (\text{C.11})$$

$$\frac{\mathcal{H}'}{\mathcal{H}} = 1 + \frac{H'}{H} \quad (\text{C.12})$$

Appendix D

Numerical antiderivatives

Very often, we want to compute functions that are defined as an integral. In cosmology, a prominent and simple example is the comoving distance:

$$d_C(z) = \int_0^z \frac{dz'}{H(z')}. \quad (\text{D.1})$$

Here, the Hubble parameter $H(z)$ is not necessarily a simple expression. The Hubble parameter could be model-independent and be given by an interpolating function as in chapter 8. Or we could have dynamical dark energy with a time variable equation of state, leading to a nonstandard time dependence and complicating the Hubble parameter. Another example where the integrand is usually an interpolating function is the weak lensing window function (see eq. (6.53)). In either case, no closed expression for the antiderivative exists. To use these functions within other functions in a computer program, we need to find a numerical antiderivative. We will demonstrate the problems associated with this by using MATHEMATICA (version 9.0.1.0).

Finding an antiderivative of an expensive function naively like this

```
1 F[x_] := NIntegrate[f[t], {t, x0, x}];  
2 iF = Interpolation[Table[{x, F[x]}, {x, x0, x1, dx}]]];
```

is not efficient. Instead, it is better to rephrase the problem as an ODE using `NDSolve`, as in

```
1 iF = NDSolveValue[{y'[x] == f[x], f[x0] == 0}, y,  
2 {x, x0, x1}];
```

because the definition in eq. (D.1) is equivalent to

$$d'_C(z) = \frac{1}{H(z)}, \quad d_C(0) = 0. \quad (\text{D.2})$$

In the first case (the “naive method”), `f` has to be sampled in most of the same region for every step, leading to many superfluous evaluations. In the second case (the “ODE method”), an efficient algorithm with a dynamical, optimized step size is used, reducing the number of evaluations of `f` significantly. Such an efficient algorithm could be the commonly used fourth-order Runge-Kutte algorithm, but many others exist (see Press et al. (2007) for a reference).

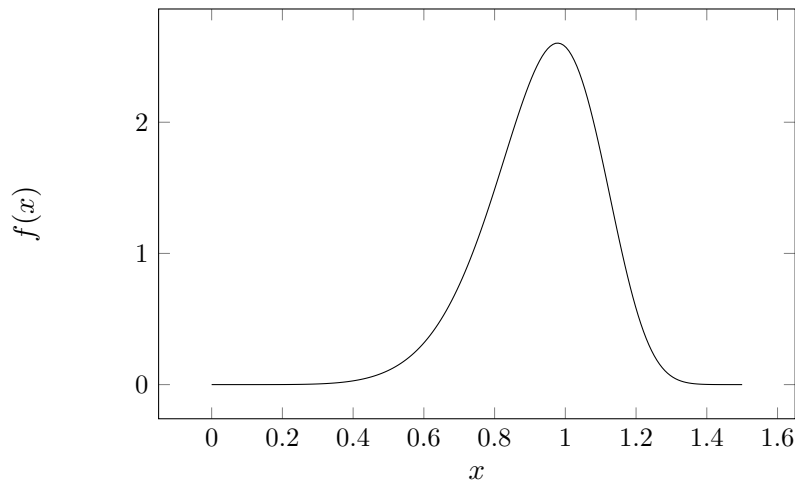


Figure D.1 A plot of the function used in this example, $f(x) = 7x^6 e^{-x^7}$.

The issue can be demonstrated by choosing a simple example, numerically finding the antiderivative $F(x)$ of $f(x) = 7x^6 e^{-x^7}$ (plotted in fig. D.1) with $F(0) = 0$, which is $F(x) = 1 - e^{-x^7}$. Consider the following code snippet in MATHEMATICA:

```

1 faux[x_] := x^6 Exp[-x^7];
2 f[x_?NumericQ] := faux[Sow[x]];
3 Samplingpoints1 = Reap[
4   F1 = Interpolation[Table[{x, NIntegrate[f[t],
5     {t, 0, x}]}, {x, 0, 1.6, .1}]]][[2, 1]];
6 Samplingpoints2 = Reap[
7   F2 = NDSolveValue[{y'[x] == f[x], y[0] == 0}, y,
8     {x, 0, 1.5}]]][[2, 1]];

```

This approximates the antiderivative numerically in two distinct ways. First, the naive method is to create a table of all values at equidistant points and evaluating the integral numerically at each point. The second method solves the problem as an ODE. The MATHEMATICA expressions `Sow` and `Reap` enable us to keep track of where the function has been evaluated.

The naive method ends up sampling the function 627 times while the ODE method takes only 206 samples. This difference can be even more drastic with less well-behaved functions. Figure D.2 shows where exactly the function gets sampled and fig. D.3 shows a histogram of the sampling points. While the naive method samples $f(x)$ over and over again for small x , the ODE method traverses the interval almost linearly after an initial attempt to find the optimal step size.

When comparing the two functions that are supposed to approximate $F(x)$, we use the supremum norm, which is defined for real functions $f(x) : X \subset \mathbb{R} \rightarrow \mathbb{R}$ as

$$\|\cdot\|_{\infty} \equiv \sup_{x \in X} |f(x)|. \quad (\text{D.3})$$

Using F_0 to denote the actual antiderivative, F_{Naive} for the interpolated antiderivative

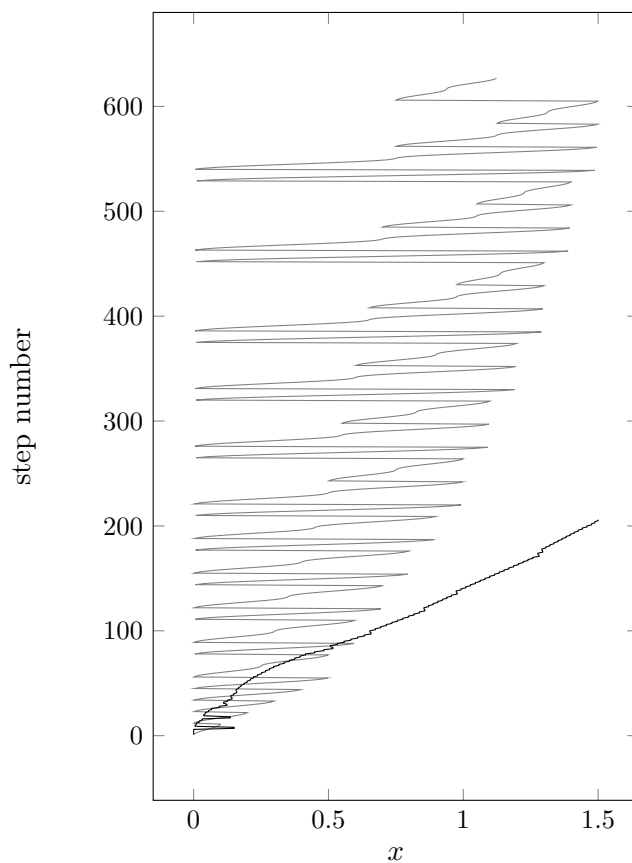


Figure D.2 A visualization of how the interval is traversed for the naive method (gray) and the ODE method (black) starting at the top. While the naive method goes back and forth multiple times to find $F(x)$ at a single value $x = x_0$ due to the integral, the ODE method samples the function in an almost linear fashion after briefly estimating the optimal step size at the beginning.

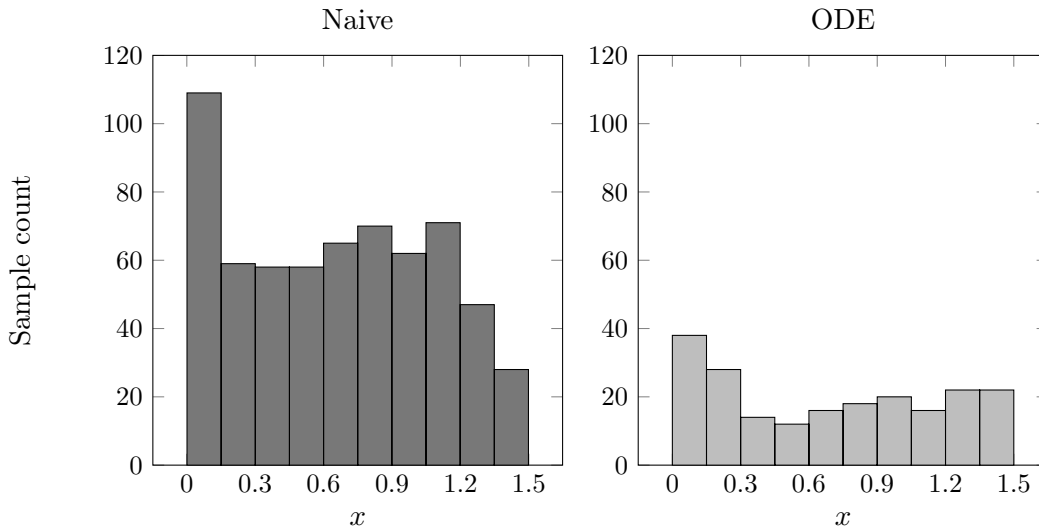


Figure D.3 Histogram for the sampling points. The naive method (left) samples the function a lot more often for small x than the ODE method (right), and needs many more evaluations in general. The sample count is not strictly decreasing most likely due to the curvature of the sampled function in the middle of the interval.

via `Interpolation`, and F_{ODE} for the interpolated antiderivative via `NDSolve`, we find that using `NDSolve` improves the result significantly:

$$\|F_0 - F_{\text{Naive}}\|_{\infty} = 2.7 \times 10^{-3} \quad (\text{D.4})$$

$$\|F_0 - F_{\text{ODE}}\|_{\infty} = 2.1 \times 10^{-7} \quad (\text{D.5})$$

A plot of the absolute error for both cases can be seen in fig. D.4. So while the ODE method evaluates $f(x)$ only a fraction of times when compared to the naive method, it does orders of magnitudes better at approximating the real solution. Thus, any code that has to evaluate a numerical antiderivative a lot should resort to treating the integral as an ODE to maximize efficiency.

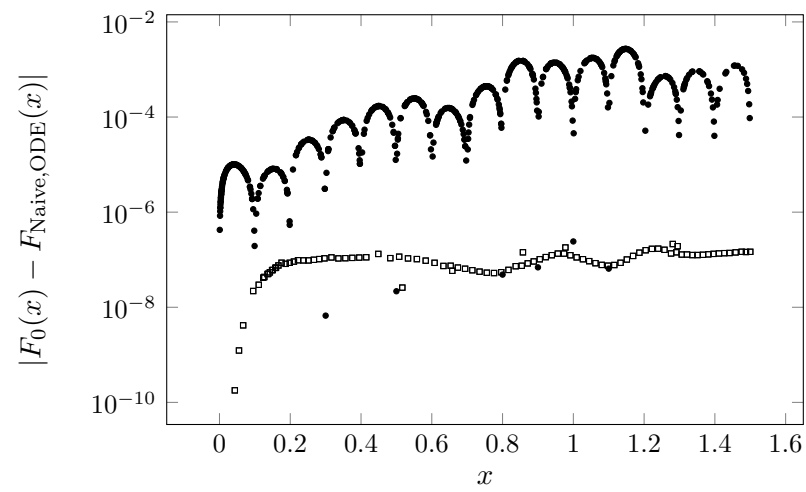


Figure D.4 Comparing the absolute errors of the interpolating functions when using the naive method (black dots) and the ODE method (blank boxes).

Appendix E

Differential equations in full form

Just to get an idea of what the equations we are trying to solve (the TRG equations) look like explicitly, we display them here in their full form. Dependence on η and k has been suppressed. Here, we applied the following mapping:

$$y_1 = I_{112,111} \quad (\text{E.1})$$

$$\vdots \quad (\text{E.2})$$

$$y_{14} = I_{222,222} \quad (\text{E.3})$$

$$y_{15} = P_{11} \quad (\text{E.4})$$

$$y_{16} = P_{12} \quad (\text{E.5})$$

$$y_{17} = P_{22}. \quad (\text{E.6})$$

Similarly, we simplified the indices for A . The TRG equations in full form then become

$$y'_1 = 2A_1e^\eta - 3y_1 + y_2 + y_3 + y_5 \quad (\text{E.7})$$

$$y'_2 = 2A_2e^\eta - \Omega_{21}y_1 - \Omega_{22}y_2 - 2y_2 + y_4 + y_6 \quad (\text{E.8})$$

$$y'_3 = 2A_3e^\eta - \Omega_{21}y_1 - \Omega_{22}y_3 - 2y_3 + y_4 + y_7 \quad (\text{E.9})$$

$$y'_4 = 2A_4e^\eta - \Omega_{21}y_2 - \Omega_{21}y_3 - 2\Omega_{22}y_4 - y_4 + y_8 \quad (\text{E.10})$$

$$y'_5 = 2A_5e^\eta - \Omega_{21}y_1 - \Omega_{22}y_5 - 2y_5 + y_6 + y_7 \quad (\text{E.11})$$

$$y'_6 = 2A_6e^\eta - \Omega_{21}y_2 - \Omega_{21}y_5 - 2\Omega_{22}y_6 - y_6 + y_8 \quad (\text{E.12})$$

$$y'_7 = 2A_7e^\eta - \Omega_{21}y_3 - \Omega_{21}y_5 - 2\Omega_{22}y_7 - y_7 + y_8 \quad (\text{E.13})$$

$$y'_8 = 2A_8e^\eta - \Omega_{21}y_4 - \Omega_{21}y_6 - \Omega_{21}y_7 - 3\Omega_{22}y_8 \quad (\text{E.14})$$

$$y'_9 = 2A_9e^\eta - 3y_9 + 2y_{10} + y_{12} \quad (\text{E.15})$$

$$y'_{10} = 2A_{10}e^\eta - \Omega_{21}y_9 - \Omega_{22}y_{10} - 2y_{10} + y_{11} + y_{13} \quad (\text{E.16})$$

$$y'_{11} = 2A_{11}e^\eta - 2\Omega_{21}y_{10} - 2\Omega_{22}y_{11} - y_{11} + y_{14} \quad (\text{E.17})$$

$$y'_{12} = 2A_{12}e^\eta - \Omega_{21}y_9 - \Omega_{22}y_{12} - 2y_{12} + 2y_{13} \quad (\text{E.18})$$

$$y'_{13} = 2A_{13}e^\eta - \Omega_{21}y_{10} - \Omega_{21}y_{12} - 2\Omega_{22}y_{13} - y_{13} + y_{14} \quad (\text{E.19})$$

$$y'_{14} = 2A_{14}e^\eta - \Omega_{21}y_{11} - 2\Omega_{21}y_{13} - 3\Omega_{22}y_{14} \quad (\text{E.20})$$

$$y'_{15} = \frac{16\pi e^\eta y_2}{k} - 2y_{15} + 2y_{16} \quad (\text{E.21})$$

$$y'_{16} = \frac{4\pi e^\eta(2y_6 + y_{11})}{k} - \Omega_{21}y_{15} - \Omega_{22}y_{16} - y_{16} + y_{17} \quad (\text{E.22})$$

$$y'_{17} = \frac{8\pi e^\eta y_{14}}{k} - 2\Omega_{21}y_{16} - 2\Omega_{22}y_{17} \quad (\text{E.23})$$

Note that all A_i depend quadratically on y_{15}, y_{16}, y_{17} , making the system nonlinear.

Appendix F

Excerpts from select source files

F.1 Definition of the integrand $K_i(k, q)$

Since the full equations for the integrand in eq. (3.16) can become rather lengthy when written out explicitly, we are generating them with a MATHEMATICA program. It creates a file filled with Fortran code that is then used in the code base via the `include` statement. Because the 14 components of the integrand $K_i(k, q)$ depend on up to seventh powers of k and q , they have been brought into the Horner form to minimize the number of additions and subtractions, which can be numerically unstable. All variables like `diff_(1)` ($= M_1(|q - k|, k)$) or `P22q` ($= P_{22}(q)$) have been defined appropriately in the Fortran program.

Here is an excerpt from that file with 206 lines:

```
1 ! This file has been generated by Mathematica
2 Kdq(1) = (k2*(12*diff_(4)*P12q + k2*(-24*diff_(3)*P12q +&
3   k2*(12*diff_(2)*P12q + 2*k2*P11k*P12q))) +&
4   q**2*(12*diff_(4)*P12k + k2*(-12*diff_(8)*P11k -&
5   12*diff_(8)*P11q - 24*diff_(3)*P12k - 36*diff_(3)*P12q +&
6   k2*(24*diff_(7)*P11k + 12*diff_(2)*P12k + 24*diff_(2)*P12q +&
7   k2*(-12*diff_(6)*P11k + 12*diff_(6)*P11q - 4*P11q*P12k +&
8   12*diff_(1)*P12q - 9*P11k*P12q - 12*logqk*P11k*P12q))) +&
9   q*(-4*k5*P11k*P12q + q*(-36*diff_(3)*P12k +&
10  k2*(12*diff_(7)*P11k + 36*diff_(7)*P11q + 24*diff_(2)*P12k +&
11  36*diff_(2)*P12q + k2*(24*diff_(6)*P11q +&
12  k2*(-12*diff_(5)*P11k + 12*diff_(5)*P11q) + 12*diff_(1)*P12k&
13  + 9*P11q*P12k + 12*logqk*P11q*P12k + 6*P11k*P12q +&
14  24*logqk*P11k*P12q)) + q*(k3*(20*P11q*P12k + 12*P11k*P12q) +&
15  q*(36*diff_(2)*P12k + q*(-12*k*P11q*P12k +&
16  q*(k2*(-12*diff_(5)*P11k + 12*diff_(5)*P11q) -&
17  12*diff_(1)*P12k + 12*logqk*P11q*P12k)) +&
18  k2*(12*diff_(6)*P11k - 36*diff_(6)*P11q +&
19  k2*(24*diff_(5)*P11k - 24*diff_(5)*P11q) - 6*P11q*P12k -&
20  24*logqk*P11q*P12k - 12*diff_(1)*P12q -&
21  12*logqk*P11k*P12q)))))))/(384*k2*q**3)
22
23 Kdq(2) = (6*diff_(4)*P22q + k2*(-12*diff_(3)*P22q +&
```

```

24     k2*(6*diff_(2)*P22q + k2*P11k*P22q)) +&
25     q**2*(-12*diff_(8)*P12q - 12*diff_(3)*P22q +&
26     k2*(12*diff_(2)*P22q + k2*(12*diff_(6)*P12q - 3*P11k*P22q))&
27     + q*(-8*k3*P11k*P22q + q*(6*diff_(12)*P11k +&
28     6*diff_(12)*P11q + 24*diff_(7)*P12q +&
29     q**2*(-12*diff_(11)*P11q + k2*(12*diff_(10)*P11k -&
30     12*diff_(10)*P11q) + q**2*(-6*diff_(10)*P11k +&
31     6*diff_(10)*P11q) - 12*diff_(6)*P12q) + 6*diff_(2)*P22q +&
32     k2*(12*diff_(11)*P11q + k2*(-6*diff_(10)*P11k +&
33     6*diff_(10)*P11q) + 12*P11k*P22q))))/(192*q**3)
34     ...

```

F.2 The explicit TRG equations

For convenience, the system of coupled differential equations (see eqs. (3.11) and (3.12)) are being generated by MATHEMATICA as well. They are of the simple form

$$y' = f(y), \quad (\text{F.1})$$

where y is a vector with 17 components. Again, variables with a trailing underscore are defined in the Fortran program.

```

1  ! This file has been generated by Mathematica
2  yprime_(1) = 2*expeta*A_(1) - 3*y_(1) + y_(2) + y_(3) +&
3  y_(5)
4  yprime_(2) = 2*expeta*A_(2) - Omega21*y_(1) - 2*y_(2) - Omega22
5  *y_(2)&
6  + y_(4) + y_(6)
7  yprime_(3) = 2*expeta*A_(3) - Omega21*y_(1) - 2*y_(3) - Omega22
8  *y_(3)&
9  + y_(4) + y_(7)
10 yprime_(4) = 2*expeta*A_(4) - Omega21*y_(2) - Omega21*y_(3) -
11 y_(4) -&
12 2*Omega22*y_(4) + y_(8)
13 yprime_(5) = 2*expeta*A_(5) - Omega21*y_(1) - 2*y_(5) - Omega22
14 *y_(5)&
15 + y_(6) + y_(7)
16 ...

```

Bibliography

- Amara, A. and Réfrégier, A. “Optimal Surveys for Weak-Lensing Tomography.” *Monthly Notices of the Royal Astronomical Society*, **381** (2007)(3), 1018. [arXiv:0610127v2](#).
- Amendola, L. “Coupled Quintessence.” *Physical Review D*, **62** (1999)(4), 43511. [arXiv:9908023](#).
- Amendola, L. “Perturbations in a coupled scalar field cosmology.” *Monthly Notices of the Royal Astronomical Society*, **312** (2000)(3), 521. [arXiv:9906073](#).
- Amendola, L. “Linear and Nonlinear Perturbations in Dark Energy Models.” *Physical Review D*, **69** (2004)(10), 103524. [arXiv:0311175v2](#).
- Amendola, L. and Tsujikawa, S. *Dark Energy: Theory and Observations*. Cambridge University Press (2010). ISBN 0521516005.
- Amendola, L., Kunz, M., and Sapone, D. “Measuring the Dark Side (with Weak Lensing).” *Journal of Cosmology and Astroparticle Physics*, **2008** (2007)(04), 013. [arXiv:0704.2421v1](#).
- Amendola, L., Kunz, M., et al. “Observables and unobservables in dark energy cosmologies.” *Physical Review D*, **87** (2013)(2), 023501. [arXiv:1210.0439](#).
- Amendola, L., Fogli, S., et al. “Model-independent constraints on the cosmological anisotropic stress.” *Physical Review D*, **89** (2014)(6), 063538. [arXiv:1311.4765](#).
- Anselmi, S., Ballesteros, G., and Pietroni, M. “Non-linear dark energy clustering.” *Journal of Cosmology and Astroparticle Physics*, **2011** (2011)(11), 014.
- Audren, B. and Lesgourgues, J. “Non-linear matter power spectrum from Time Renormalisation Group: efficient computation and comparison with one-loop.” *Journal of Cosmology and Astroparticle Physics*, **2011** (2011)(10), 037. [arXiv:1106.2607](#).
- Baldi, M. “Time-dependent couplings in the dark sector: from background evolution to non-linear structure formation.” *Monthly Notices of the Royal Astronomical Society*, **411** (2011)(2), 1077.
- Baldi, M., Pettorino, V., et al. “Hydrodynamical N -body simulations of coupled dark energy cosmologies.” *Monthly Notices of the Royal Astronomical Society*, **403** (2010)(4), 1684. [arXiv:0812.3901](#).

-
- Bartelmann, M. and Schneider, P. “Weak Gravitational Lensing.” *Physics Reports*, **340** (1999)(4-5), 223. [arXiv:9912508](#).
- Bernardeau, F. “The evolution of the large-scale structure of the universe: beyond the linear regime.” Technical report, Institut de Physique Théorique, Gif-sur-Yvette (2013). [arXiv:1311.2724](#).
- Bernardeau, F., Colombi, S., et al. “Large-scale structure of the Universe and cosmological perturbation theory.” *Physics Reports*, **367** (2002)(1-3), 1. [arXiv:0112551v1](#).
- Bernardeau, F., Van de Rijt, N., and Vernizzi, F. “Resummed propagators in multi-component cosmic fluids with the eikonal approximation.” *Physical Review D*, **85** (2012)(6), 063509. [arXiv:1109.3400](#).
- Bezanson, J., Karpinski, S., et al. “Julia: A Fast Dynamic Language for Technical Computing.” *arXiv*, (2012). [arXiv:1209.5145](#).
- Blanchard, A., Douspis, M., et al. “Large-scale galaxy correlations as a test for dark energy.” *Astronomy and Astrophysics*, **449** (2006)(3), 925. [arXiv:0512085](#).
- Blas, D., Lesgourgues, J., and Tram, T. “The Cosmic Linear Anisotropy Solving System (CLASS). Part II: Approximation schemes.” *Journal of Cosmology and Astroparticle Physics*, **2011** (2011)(07), 034. [arXiv:1104.2933](#).
- Bond, J. R. and Efstathiou, G. “Cosmic background radiation anisotropies in universes dominated by nonbaryonic dark matter.” *The Astrophysical Journal*, **285** (1984), L45.
- Bouchet, F. R., Juszkiewicz, R., et al. “Weakly nonlinear gravitational instability for arbitrary Omega.” *The Astrophysical Journal*, **394** (1992), L5.
- Brans, C. and Dicke, R. H. “Mach’s Principle and a Relativistic Theory of Gravitation.” *Physical Review*, **124** (1961)(3), 925.
- Buchdahl, H. “Non-linear Lagrangians and cosmological theory.” *Monthly Notices of the Royal Astronomical Society*, **150** (1970), 1.
- Caldwell, R., Dave, R., and Steinhardt, P. “Cosmological Imprint of an Energy Component with General Equation of State.” *Physical Review Letters*, **80** (1998)(8), 1582. [arXiv:9708069](#).
- Caldwell, R., Kamionkowski, M., and Weinberg, N. “Phantom Energy: Dark Energy with $w < -1$ Causes a Cosmic Doomsday.” *Physical Review Letters*, **91** (2003)(7). [arXiv:0302506v1](#).
- Carlson, J., White, M., and Padmanabhan, N. “Critical look at cosmological perturbation theory techniques.” *Physical Review D*, **80** (2009)(4), 043531. [arXiv:0905.0479](#).
- Carroll, S. *Spacetime and Geometry: An Introduction to General Relativity*. Benjamin Cummings (2003). ISBN 0805387323.
- Carroll, S., Press, W. H., and Turner, E. L. “The Cosmological Constant.” *Annual Review of Astronomy and Astrophysics*, **30** (1992)(1), 499.

- Carroll, S. M. “The Cosmological Constant.” *Living Reviews in Relativity*, **4** (2001). [arXiv:0004075](#).
- Crocce, M. and Scoccimarro, R. “Renormalized cosmological perturbation theory.” *Physical Review D*, **73** (2006a)(6), 063519. [arXiv:0509418v2](#).
- Crocce, M. and Scoccimarro, R. “Memory of initial conditions in gravitational clustering.” *Physical Review D*, **73** (2006b)(6), 063520. [arXiv:0509419](#).
- Crocce, M. and Scoccimarro, R. “Nonlinear evolution of baryon acoustic oscillations.” *Physical Review D*, **77** (2008)(2), 023533. [arXiv:0704.2783](#).
- De Felice, A. and Tsujikawa, S. “f(R) Theories.” *Living Reviews in Relativity*, **13** (2010)(3), 1. [arXiv:1002.4928v2](#).
- De Felice, A., Kobayashi, T., and Tsujikawa, S. “Effective gravitational couplings for cosmological perturbations in the most general scalar–tensor theories with second-order field equations.” *Physics Letters B*, **706** (2011)(2-3), 123. [arXiv:1108.4242v2](#).
- Deffayet, C., Gao, X., et al. “From k-essence to generalized Galileons.” *Physical Review D*, **84** (2011)(6), 064039. [arXiv:1103.3260](#).
- Eisenstein, D. J. and Hu, W. “Power Spectra for Cold Dark Matter and Its Variants.” *The Astrophysical Journal*, **511** (1999)(1), 5. [arXiv:9710252v1](#).
- Eisenstein, D. J., Hu, W., and Tegmark, M. “Cosmic Complementarity: Joint Parameter Estimation from Cosmic Microwave Background Experiments and Redshift Surveys.” *The Astrophysical Journal*, **518** (1999)(1), 2. [arXiv:9807130](#).
- Enqvist, K. “Lemaitre–Tolman–Bondi model and accelerating expansion.” *General Relativity and Gravitation*, **40** (2007)(2-3), 451.
- Felice, A. D. and Tsujikawa, S. “Shapes of primordial non-Gaussianities in the Horndeski’s most general scalar-tensor theories.” *Journal of Cosmology and Astroparticle Physics*, **2013** (2013)(03), 030. [arXiv:1301.5721v2](#).
- Goroff, M. H., Grinstein, B., et al. “Coupling of modes of cosmological mass density fluctuations.” *The Astrophysical Journal*, **311** (1986), 6.
- Graham, S., Kessler, P., and McKusick, M. “gprof: A Call Graph Execution Profiler.” *Proceedings of the SIGPLAN ’82 Symposium on Compiler Construction, SIGPLAN Notices*, **17** (1982)(6), 120.
- Guarnizo, A., Vollmer, A., and Amendola, L. “A cosmological exclusion plot for modified gravity.” *To be published*, (2014).
- Hahn, T. “The High-Energy Physicist’s Guide to MathLink.” Technical report, Max-Planck-Institut für Physik (2011). [arXiv:1107.4379](#).
- Harrison, E. “Fluctuations at the Threshold of Classical Cosmology.” *Physical Review D*, **1** (1970)(10), 2726.

-
- Heitmann, K., Higdon, D., et al. “The Coyote Universe. II. Cosmological Models and Precision Emulation of the Nonlinear Matter Power Spectrum.” *The Astrophysical Journal*, **705** (2009)(1), 156. [arXiv:0902.0429](#).
- Heitmann, K., White, M., et al. “The Coyote Universe. I. Precision Determination of the Nonlinear Power Spectrum.” *The Astrophysical Journal*, **715** (2010)(1), 104. [arXiv:0812.1052](#).
- Heitmann, K., Lawrence, E., et al. “The Coyote Universe Extended: Precision Emulation of the Matter Power Spectrum.” *arXiv*, (2013), 17. [arXiv:1304.7849](#).
- Hobson, M. P., Jaffe, A. H., et al. *Bayesian Methods in Cosmology*. Cambridge University Press (2009). ISBN 9780521887946.
- Hogg, D. W. “Distance measures in cosmology.” *arXiv*, (1999). [arXiv:9905116](#).
- Horndeski, G. W. “Second-order scalar-tensor field equations in a four-dimensional space.” *International Journal of Theoretical Physics*, **10** (1974)(6), 363.
- Hu, W. “Power Spectrum Tomography with Weak Lensing.” *The Astrophysical Journal*, **522** (1999)(1), L21. [arXiv:9904153](#).
- Hu, W. and Tegmark, M. “Weak Lensing: Prospects for Measuring Cosmological Parameters.” *The Astrophysical Journal*, **514** (1999)(2), L65. [arXiv:9811168](#).
- Jain, B. and Bertschinger, E. “Second-order power spectrum and nonlinear evolution at high redshift.” *The Astrophysical Journal*, **431** (1994), 495. [arXiv:9311070](#).
- Kendall, M. G. and Stuart, A. *The Advanced Theory of Statistics, Vol. II*. London: Griffin, 2nd edition (1967). ISBN 0852640110.
- Khintchine, A. “Korrelationstheorie der stationären stochastischen Prozesse.” *Mathematische Annalen*, **109** (1934)(1), 604.
- Knott, G. D. *Interpolating Cubic Splines*. Springer Science & Business Media (2000). ISBN 0817641009.
- Komatsu, E., Smith, K. M., et al. “Seven-Year Wilkinson Microwave Anisotropy Probe (WMAP) Observations: Cosmological Interpretation.” *The Astrophysical Journal Supplement Series*, **192** (2010)(2), 57. [arXiv:1001.4538v3](#).
- Laureijs, R., Amiaux, J., et al. “Euclid Definition Study Report.” Technical report, Euclid collaboration (2011). [arXiv:1110.3193](#).
- Lawrence, E., Heitmann, K., et al. “The Coyote Universe. III. Simulation Suite and Precision Emulator for the Nonlinear Matter Power Spectrum.” *The Astrophysical Journal*, **713** (2010)(2), 1322. [arXiv:0912.4490](#).
- Lesgourgues, J. and Pastor, S. “Massive neutrinos and cosmology.” *Physics Reports*, **429** (2006)(6), 307. [arXiv:0603494](#).
- Lesgourgues, J., Matarrese, S., et al. “Non-linear power spectrum including massive neutrinos: the time-RG flow approach.” *Journal of Cosmology and Astroparticle Physics*, **2009** (2009)(06), 017. [arXiv:0901.4550](#).

- Lewis, A. and Bridle, S. “Cosmological parameters from CMB and other data: A Monte Carlo approach.” *Physical Review D*, **66** (2002)(10), 103511. [arXiv:0205436](#).
- Lewis, A., Challinor, A., and Lasenby, A. “Efficient Computation of Cosmic Microwave Background Anisotropies in Closed Friedmann-Robertson-Walker Models.” *The Astrophysical Journal*, **538** (2000)(2), 473. [arXiv:9911177](#).
- Lubin, M. and Dunning, I. “Computing in Operations Research using Julia.” *arXiv*, (2013). [arXiv:1312.1431](#).
- Ma, C.-P. and Bertschinger, E. “Cosmological Perturbation Theory in the Synchronous and Conformal Newtonian Gauges.” *The Astrophysical Journal*, **455** (1995), 7. [arXiv:9506072](#).
- Ma, Z., Hu, W., and Huterer, D. “Effects of Photometric Redshift Uncertainties on Weak-Lensing Tomography.” *The Astrophysical Journal*, **636** (2006)(1), 21. [arXiv:0506614v2](#).
- Makino, N., Sasaki, M., and Suto, Y. “Analytic approach to the perturbative expansion of nonlinear gravitational fluctuations in cosmological density and velocity fields.” *Physical Review D*, **46** (1992)(2), 585.
- Marra, V., Amendola, L., et al. “Cosmic Variance and the Measurement of the Local Hubble Parameter.” *Physical Review Letters*, **110** (2013)(24), 241305. [arXiv:1303.3121](#).
- Matarrese, S. and Pietroni, M. “Resumming cosmic perturbations.” *Journal of Cosmology and Astroparticle Physics*, (2007)(06), 026. [arXiv:0703563](#).
- Misner, C. W., Thorne, K. S., and Wheeler, J. A. *Gravitation*. Number Teil 3 in Gravitation. W. H. Freeman (1973). ISBN 9780716703440.
- Motta, M., Sawicki, I., et al. “Probing dark energy through scale dependence.” *Physical Review D*, **88** (2013)(12), 124035. [arXiv:1305.0008](#).
- Oloff, R. *Geometrie der Raumzeit: Eine mathematische Einführung in die Relativitätstheorie*. Vieweg+Teubner Verlag, 3rd edition (2004). ISBN 383481007X.
- Panofsky, W. K. H. and Phillips, M. *Classical Electricity and Magnetism: Second Edition*. Dover Books on Physics. Dover Publications (2012). ISBN 9780486132259.
- Pascual-Sánchez, J.-F. “Cosmic Acceleration: Inhomogeneity versus Vacuum Energy.” *Modern Physics Letters A*, **14** (1999)(23), 1539.
- Perlmutter, S., Aldering, G., et al. “Measurements of Ω and Λ from 42 High-Redshift Supernovae.” *The Astrophysical Journal*, **517** (1999)(2), 565. [arXiv:9812133](#).
- Perrotta, F., Matarrese, S., et al. “Nonlinear perturbations in scalar-tensor cosmologies.” *Physical Review D*, **69** (2004)(8), 084004. [arXiv:0310359](#).
- Pietroni, M. “Flowing with time: a new approach to non-linear cosmological perturbations.” *Journal of Cosmology and Astroparticle Physics*, **2008** (2008)(10), 036. [arXiv:0806.0971v3](#).

-
- Planck Collaboration, Ade, P. A. R., et al. “Planck 2013 results. XVI. Cosmological parameters.” *arXiv*, (2013), 67. [arXiv:1303.5076](#).
- Press, W. H., Teukolsky, S. A., et al. *Numerical Recipes*. Cambridge University Press, 3rd edition (2007). ISBN 0521880688.
- Pueblas, S. and Scoccimarro, R. “Generation of vorticity and velocity dispersion by orbit crossing.” *Physical Review D*, **80** (2009)(4), 043504. [arXiv:0809.4606](#).
- Ratra, B. and Peebles, P. “Cosmological consequences of a rolling homogeneous scalar field.” *Physical Review D*, **37** (1988)(12), 3406.
- Riess, A. G., Filippenko, A. V., et al. “Observational Evidence from Supernovae for an Accelerating Universe and a Cosmological Constant.” *The Astronomical Journal*, **116** (1998)(3), 1009. [arXiv:9805201v1](#).
- Saracco, F., Pietroni, M., et al. “Nonlinear matter spectra in coupled quintessence.” *Physical Review D*, **82** (2010)(2), 023528. [arXiv:0911.5396](#).
- Sato, M. and Matsubara, T. “Nonlinear biasing and redshift-space distortions in Lagrangian resummation theory and N-body simulations.” *Physical Review D*, **84** (2011)(4), 1. [arXiv:1105.5007v4](#).
- Scoccimarro, R., Colombi, S., et al. “Nonlinear Evolution of the Bispectrum of Cosmological Perturbations.” *The Astrophysical Journal*, **496** (1998)(2), 586. [arXiv:9704075](#).
- Seo, H. and Eisenstein, D. J. “Probing Dark Energy with Baryonic Acoustic Oscillations from Future Large Galaxy Redshift Surveys.” *The Astrophysical Journal*, **598** (2003)(2), 720. [arXiv:0307460](#).
- Seward, J. and Nethercote, N. “Using Valgrind to detect undefined value errors with bit-precision.” In “Proceedings of the USENIX’05 Annual Technical Conference,” (2005).
- Smith, R. E., Peacock, J. A., et al. “Stable Clustering, the Halo Model and Non-linear Cosmological Power Spectra.” *Monthly Notices of the Royal Astronomical Society*, **341** (2003)(4), 1311. [arXiv:0207664v2](#).
- Smoot, G. F., Bennett, C. L., et al. “Structure in the COBE differential microwave radiometer first-year maps.” *The Astrophysical Journal*, **396** (1992), L1.
- Spivak, M. *A Comprehensive Introduction to Differential Geometry*. Publish or Perish, 3rd edition (1999). ISBN 0914098705.
- Springel, V. “The cosmological simulation code GADGET-2.” *Monthly Notices of the Royal Astronomical Society*, **364** (2005)(4), 1105. [arXiv:0505010](#).
- Takahashi, R., Sato, M., et al. “Revising the Halofit Model for the Nonlinear Matter Power Spectrum.” *The Astrophysical Journal*, **761** (2012)(2), 152. [arXiv:1208.2701](#).
- Tegmark, M., Taylor, A. N., and Heavens, A. F. “Karhunen-Loeve Eigenvalue Problems in Cosmology: How Should We Tackle Large Data Sets?” *The Astrophysical Journal*, **480** (1997)(1), 22. [arXiv:9603021](#).

- Thompson, J. F. and Soni, B. *Handbook of Grid Generation*. CRC Press (1998). ISBN 0849326877.
- Tsujikawa, S. "Quintessence: a review." *Classical and Quantum Gravity*, **30** (2013)(21), 214003. [arXiv:1304.1961](#).
- Valageas, P. "Impact of shell crossing and scope of perturbative approaches, in real and redshift space." *Astronomy & Astrophysics*, **526** (2010), A67. [arXiv:1009.0106](#).
- Verde, L. and Heavens, A. F. "On the Trispectrum as a Gaussian Test for Cosmology." *The Astrophysical Journal*, **553** (2001)(1), 14. [arXiv:0101143](#).
- Wang, L. and Steinhardt, P. J. "Cluster Abundance Constraints for Cosmological Models with a Time-varying, Spatially Inhomogeneous Energy Component with Negative Pressure." *The Astrophysical Journal*, **508** (1998)(2), 483. [arXiv:9804015](#).
- Weinberg, S. "The Cosmological Constant Problem." *Reviews of Modern Physics*, **61** (1989)(1), 1.
- Wetterich, C. "Cosmology and the fate of dilatation symmetry." *Nuclear Physics B*, **302** (1988)(4), 668.
- Wetterich, C. "An asymptotically vanishing time-dependent cosmological "constant"." *Astronomy and Astrophysics*, **301** (1994), 321. [arXiv:9408025](#).