

Modeling the Dark Sector with the Equation of State and the Sound Speed

Maxwell Aifer

Advisor: Daniel Grin

Collaborator: Tristan Smith

Haverford College Department of Physics

(Dated: May 30, 2019)

In the current Λ CDM model of cosmology, most of the energy content of the universe is stored in substances which are poorly understood, collectively known as the “dark sector”. New experiments promise to measure signals such as the Cosmic Microwave Background (CMB) with great precision, which can give us more information about the dark sector. Precision measurements of the CMB (e.g. Planck 2018) have been used to get error bounds on cosmological parameters for a specific Λ CDM dark sector model. We give a model-independent description of the dark sector using the generalized dark matter (GDM) formalism which is characterized by a time-dependent equation of state (w) the effective sound speed (c_s^2), which is both time-dependent and scale-dependent. We want to answer the question “how well can precision measurements of the CMB constrain equation of state and sound speed of the dark sector”. As of now, our implementation of the GDM cosmology reproduces the qualitative features of the time-dependent overall density perturbation, but the power spectrum is not reproduced.

CONTENTS

| | |
|---|----|
| I. Introduction | 4 |
| I.1. Energy Content of the Universe | 4 |
| I.1.1. The Dark Sector | 4 |
| I.1.2. Dark Matter Candidates | 7 |
| I.2. The Cosmic Microwave Background | 8 |
| I.2.1. Statistical description of the CMB | 8 |
| I.3. Goals of this project in relation to past research | 10 |
| I.4. Metric Expansion of Space | 11 |
| I.5. Cosmological Parameters | 12 |
| I.5.1. Recombination | 14 |
| I.6. Supernova data | 16 |
| I.7. Baryon Acoustic Oscillations | 16 |
| I.8. The Stress-Energy Tensor | 16 |
| I.8.1. Single Fluid Approximation | 18 |
| I.8.2. Generalized Dark Matter | 21 |
| I.8.3. Two Fluid Approximation | 22 |
| I.8.4. Plane wave expansion | 23 |
| I.9. Principle Component Analysis | 24 |
| I.9.1. The Fisher Matrix | 24 |
| I.9.2. Principal Components | 26 |
| I.10. This Thesis | 27 |
| II. Background Cosmology | 27 |
| III. Perturbations | 28 |
| III.1. Two Fluid Approximation | 30 |
| III.1.1. Nondimensionalization | 30 |
| III.2. Three Fluid Approximation | 31 |
| IV. perturbations and the CMB power spectrum | 32 |
| V. Generalized Dark Matter | 32 |

| | |
|--|----|
| V.1. Description | 32 |
| VI. Principal Component Analysis | 33 |
| VI.1. Luminosity-Redshift Data | 34 |
| VI.1.1. Constant equation of state parameter | 34 |
| VI.1.2. Variable equation of state parameter | 35 |
| VI.1.3. Parameterizing the equation of state parameter | 35 |
| VI.1.4. Obtaining principal components | 36 |
| VI.2. PCA with the CMB | 37 |
| VII. Implementation and Results | 38 |
| VII.1. Implementation | 38 |
| VII.1.1. Background Computations | 38 |
| VII.1.2. Computing the mode evolution | 39 |
| VII.1.3. Precomputing bessel functions and derivatives | 41 |
| VII.1.4. Computing the power spectrum | 42 |
| VII.1.5. GDM parameters | 42 |
| VII.2. Results | 43 |
| VII.3. Source Contributions | 44 |
| VII.4. The power spectrum | 46 |
| VII.5. GDM effective fluid quantities | 49 |
| VIII. Conclusion | 53 |
| VIII.1. Methodology | 53 |
| VIII.2. Future work | 53 |
| References | 54 |
| IX. Appendix | 55 |
| IX.1. Mathematica Code | 55 |
| IX.2. Python Code | 58 |
| IX.2.1. mode_evolution.ipynb | 58 |
| IX.2.2. get_eff_params.ipynb | 65 |
| IX.2.3. 1_fluid.ipynb | 74 |

I. INTRODUCTION

I.1. Energy Content of the Universe

The universe is believed to be homogeneous and isotropic on very large scales. That means that it looks the same at every point in space and looks the same no matter what direction one is looking in. Given these assumptions, on the largest scales the only interesting properties of the universe are the densities of the different forms of energy which are present. Throughout the history of the universe, the bulk of the energy content of the universe is believed to have been contained in five different components: Baryonic matter (atoms), photons, neutrinos, dark matter, and dark energy. The first two things in that list are understood very well, the third is understood not quite so well, the last two are barely understood at all. This is why they have the word dark in their names, and why they collectively form what is called the “dark sector”.

I.1.1. *The Dark Sector*

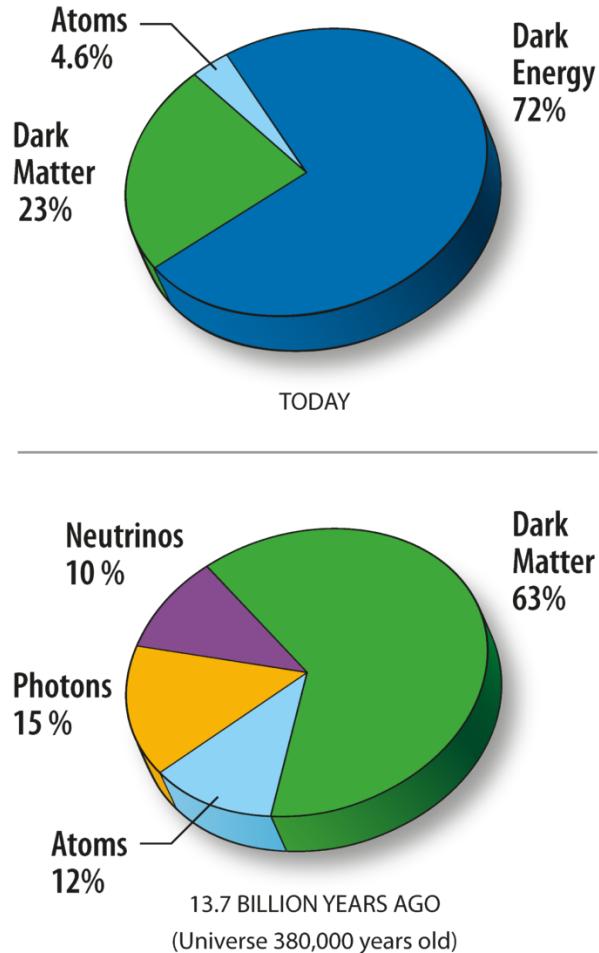


Figure 1. Proportions of energy density in the universe, 13.7 billion years ago and today. [1]

In today's universe, baryonic matter accounts for a mere 4.6% of the energy and photons take up a much smaller, almost negligible, portion. Baryonic matter is composed of the quarks and leptons of the standard model, and its dynamics can be modeled very well. Everything else is called "dark matter" or "dark energy". While the two names are similar, the concepts are vastly different, and shouldn't be confused. [2]

First, dark matter. Dark matter was invented to solve the problem of anomalous galaxy rotation curves, and might be defined as any substance whose existence solves this problem. There are other problems as well, but the biggest problem is the galaxy rotation curves. A galaxy rotation curve is a function which gives the mean tangential velocity of stars in a galaxy as a function of the distance from the galactic center (See Figure 2). An example of galaxy rotation curve findings can be seen in Ref. [3].

The laws of gravity dictate that as one gets outside of most of the mass of the galaxy,

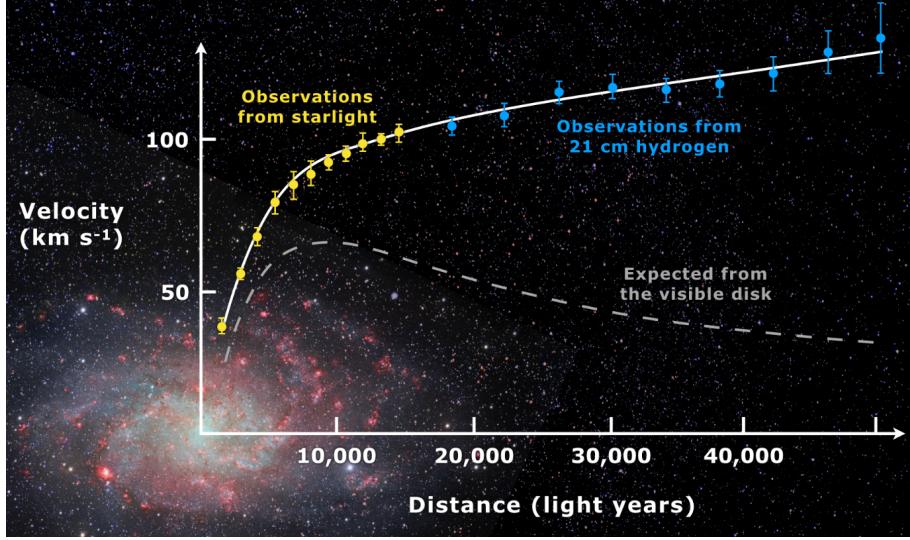


Figure 2. Galaxy rotation curves: expectation vs. reality.[4]

the angular velocity falls off, but this falloff is not observed. In fact, the angular velocity is flat, or continues to increase, far outside of the apparent mass center of the galaxy. This implies that there is some matter there that we cannot see, forming a “halo” around the visible portion of the galaxy, and extending much farther out than the visible part. The most popular cosmological model, called Λ CDM, assumes that the dark matter is “Cold Dark Matter” (CDM), meaning that it travels at sub-relativistic speeds. The details of this model can be found in Refs. [5], [6], and [7].

Dark energy is a different beast. While it is in some ways more mysterious than dark matter, it has a much more precise definition, mathematically speaking; dark energy is the energy associated with the cosmological constant Λ in the Einstein Equation (Eq. 1).

$$R_{\mu\nu} - \frac{1}{2}Rg_{\mu\nu} + \Lambda g_{\mu\nu} = \frac{8\pi G}{c^4}T_{\mu\nu} \quad (1)$$

Here $g_{\mu\nu}$ is the metric tensor, $R_{\mu\nu}$ is a tensor called the Ricci curvature tensor, and R is called the scalar curvature, defined as the contraction $g_{ij}R^{ij}$. Λ itself has units of m^{-2} , and the energy density associated with Λ is actually equal to $\frac{\Lambda c^2}{8\pi G}$. The cosmological constant was originally added by Einstein to his equation in order to permit a solution where the universe is stationary. Now it is known experimentally that the universe is expanding, and not stationary, so one would think that Λ would have become obsolete. And yet, the cosmological constant persists; it is the Λ in Λ CDM. The reason for its continued inclusion

has to do with measurements of the rate of expansion of space, which requires a fully relativistic description. Regardless, there are no known or imagined materials which have the properties of dark energy. It is better thought of as a geometric property of the universe than something substantial. Dark energy might be due to the energy which is present even in an ideal vacuum predicted by quantum mechanics or it might just be an irreducible law of the universe. Dark matter, on the other hand, behaves much more like classical matter. It interacts gravitationally and is presumably made of some kind of massive particles, like baryonic matter is. The only difference is that we can't see it. This has lead to a number of theories about what exactly the stuff is made of

1.1.2. Dark Matter Candidates

The most intuitive, and least complicated, is the idea that dark matter is made up of Massive Astrophysical Compact Halo Objects (MACHOs). These objects would be things like planets, gravitationally bound rock and dust which is not big enough to form a star. An issue with the MACHO theory is that in the galaxies with anomalous rotation curves, these objects would need to account for a large portion of the galactic mass, and there would be so many of them that there would be measureable variations in the apparent intensity of stars in the galaxy due to MACHOs transiting in between us and the stars. We do not observe this.

There is also the possibility that dark matter is made of Weakly Interacting Massive Particles (WIMPS). As the name suggests, these would be elementary particles which interact via the weak force and gravity, but do not interact electromagnetically (thus making them "dark"). These particles have been seen as a promising candidate for dark matter because their existence was also predicted by supersymmetry theories which would add particles to the standard model.

Another theory is that the dark matter is made up of a particle called an axion. The axion was first proposed by Roberto Peccei and Helen Quinn. It arises from an added term in the Lagrangian describing quantum chromodynamics. the theory of the strong force, which Peccei and Quinn proposed adding in order to solve the strong-CP problem. Essentially, the standard model QCD lagrangian is unchanged if all electric charge has its sign flipped and all spatial directions are reversed in sign as well. This symmetry is not observed by other

forces (such as electroweak interactions), and so Peccei and Quinn proposed adding an extra term which gets rid of this symmetry, the axion being a consequence of this extra term.

Without experimental data, it is difficult to see how we could decide which (if any) of these hypotheses is correct. While dark matter was originally discovered through galaxy rotation curves, one of the most useful tools for analyzing it today is Cosmic Microwave Background radiation, a truly ancient source of information in the form of the faint glow from the early universe.

I.2. The Cosmic Microwave Background

In 1964, astronomers Robert Wilson and Arno Penzias discovered that radiation can be detected from all directions with the properties of the radiation emitted by a black body with an extremely uniform temperature of $2.72548 \pm 0.00057\text{K}$. This phenomenon was dubbed the Cosmic Microwave Background (abbreviated CMB). The CMB is remarkably isotropic, meaning the temperature associated with the radiation is very similar, no matter which direction one measures the radiation in. Numerous important results in cosmology have been deduced from the CMB. For a good discussion of the CMB, see Ref. [7].

I.2.1. Statistical description of the CMB

While the temperature of the CMB is very uniform, most of what we can learn from the CMB is discovered by looking at the small fluctuations in temperature from place to place in the sky. This is done by analyzing the statistical properties of the function $T(\theta, \phi)$, which gives the temperature of the CMB in some direction. Fluctuations are generally defined as fractional deviations from the mean:

$$\delta_T(\theta, \phi) = \frac{T(\theta, \phi) - \bar{T}}{\bar{T}}$$

Where \bar{T} is the whole-sky CMB temperature average. The temperature fluctuation is generally taken to scale with the fluctuations in pressure and density so δ_T is indicative of the

fractional overdensity δ , given by Eq. (2).

$$\delta(\theta, \phi) = \frac{\delta\rho}{\bar{\rho}} = \frac{\rho(\theta, \phi) - \bar{\rho}}{\bar{\rho}} \quad (2)$$

Because we are interested in the characteristic length scales of temperature fluctuation, and not in the temperature at any particular point, it is useful to represent δ_T in a way which emphasizes the variability on different length scales. Furthermore, as our data on the CMB is spherical in nature, the spherical harmonics are a natural choice. The spherical harmonics, denoted $Y_\ell^m(\theta, \phi)$ comprise a set of functions (indexed by ℓ and m here) defined on a spherical domain which are pairwise orthogonal and are solutions to Laplace's equation:

$$\nabla^2 f = 0 \quad (3)$$

The coefficients a_{lm} in the spherical harmonic expansion of the temperature fluctuation are obtained from Eq. (4):

$$a_{lm} = \int_0^{2\pi} \int_0^\pi Y_{lm}^*(\theta, \phi) \sin(\theta) f(\theta, \phi) d\theta d\phi \quad (4)$$

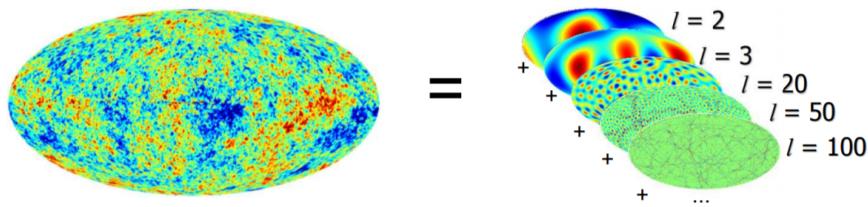


Figure 3. An illustration of how the CMB temperature map is deconstructed into spherical harmonic components with varying length scales. Reproduced from Eriksen's presentation [8].

The spherical harmonic functions can be viewed, simplistically, as akin to a Fourier representation of a function, but on a spherical domain. The parameter $|m|$ determines the number of "nodal lines" (circles of constant value) which are longitudes of a sphere, while $l - |m|$ determines the number of nodal lines which are latitudinal. In practice, the coefficients are generally averaged over m . The reason for this is that by definition of the spherical harmonics, $|m|$ cannot exceed l , so l sets the smallest length scale for fluctuations.

The averaged coefficients are indexed by l , and called C_l .

$$C_l = \langle |a_{lm}|^2 \rangle = \frac{1}{2l+1} \sum_{m=-l}^{+l} a_{lm}^* a_{lm}$$

This list of numbers—the values of C_l —is called the CMB power spectrum, and it is one of the most accurately measured signals in cosmology today, due to its power to tell us about the nature of the dark sector.

I.3. Goals of this project in relation to past research

This work is part of an ongoing, NASA-funded collaboration with the main goal of improving our characterization of the dark sector. The research proposal this work is based on is Ref. [9]. Past work (See, e.g. Ref. [10]) has used the CMB to obtain constraints on parameters in the Λ CDM model of cosmology (Specifically the energy densities of the dark matter and dark energy). In that body of work, the dark sector is assumed to have certain thermodynamic properties: the equation of state parameter w , which is the ratio of pressure to energy density, is set to -1 for the dark energy and to 0 for the cold dark matter. Because w is assumed to be constant for both dark matter and dark energy, the sound speed (c_s^2), defined as the derivative of pressure with respect to energy density, is also constant and is equal to w .

In this work we adopt the GDM formalism, where we use a single equation of state and a single sound speed to describe the entire energy content of the universe. We will characterize the entire ensemble of energy components as a single fluid, and first order perturbations to the fluid density will be expressed in spatial frequency-space (k-space). The quantities w and c_s^2 are allowed to be functions of time, and c_s^2 can also depend on wavenumber k .

The ultimate goal we are working towards is obtaining constraints (error bounds) on the functions $w(t)$ and $c_s^2(t, k)$ using data from various cosmological sources including the Cosmic Microwave Background. These constraints can be obtained by using the methods of principal component analysis and the Fisher Matrix, explained later in this introduction.

I.4. Metric Expansion of Space

The metric tensor $g_{\mu\nu}$ is a generalization of the Minkowski metric $\eta_{\mu\nu} = \text{diag}(-1, 1, 1, 1)$, which allows us to compute the spacetime interval between two spacetime events with coordinate vectors x and y :

$$\Delta s^2 = \eta_{\mu\nu}(y - x)^\mu(y - x)^\nu$$

It performs the role of an inner product on a vector space, but because it is not positive definite, we cannot call it an inner product. Technically, $\eta_{\mu\nu}$ is a *bilinear form*, but it is not hard to see why we favor the more suggestive term *metric*. Its purpose is to associate a spacetime interval with a displacement vector in some coordinate system. In general relativity, spacetime as a whole is not a single vector space, due to curvature. Instead, there is a separate vector space at each point (known as the tangent space) where we can express infinitesimal spacetime intervals as infinitesimal coordinate displacement vectors as follows

$$ds^2 = g_{\mu\nu}(x)dx^\mu dx^\nu$$

Note that in general the metric can be a function of the spacetime coordinates. A thorough discussion of the metric of general relativity can be found in Ref. [11]. From the assumptions of homogeneity and isotropy of space, it can be deduced (see [5] and [6]) that the following line element, called Robertson-Walker metric, is the most general possible metric for spacetime (on length scales large enough that homogeneity and isotropy are valid)

$$ds^2 = -c^2 dt^2 + a^2(t)[dr^2 + f_K^2(r)d\omega^2] \quad (5)$$

Where

$$f_K(r) = \begin{cases} K^{-1/2} \sin(K^{1/2}r) & K > 0 \\ r & K = 0 \\ |K|^{-1/2} \sinh(|K|^{1/2}r) & K < 0 \end{cases}$$

and $d\omega$ is a solid angle element, which can be expressed in terms of azimuthal angle θ and polar angle ϕ as follows:

$$d\omega = d\theta^2 + \sin^2 \theta d\phi^2$$

The dynamics of the scale factor a are then described by the Friedmann Equations (Eq. 6

and 7), which can also be derived from general relativity. These equations can be found in any cosmology text (see, e.g. [5], [6], and [7]).

$$\left(\frac{\dot{a}}{a}\right)^2 = \frac{8\pi G}{3}\rho - \frac{Kc^2}{a^2} + \frac{\Lambda}{3} \quad (6)$$

$$\dot{\rho} = -3H(\rho + p) \quad (7)$$

Where a is the scale factor, G the gravitational constant, ρ the energy density, K the curvature of space, Λ the cosmological constant, and H the Hubble constant. The scale factor a scales the spatial component of the spacetime metric. In general, coordinates of spacetime events will either be specified as “physical” coordinates (\mathbf{x}, t) or as comoving coordinates (\mathbf{r}, τ) , which are related (up to a choice of origin) by Equations (8) and (9).

$$\mathbf{x} = a\mathbf{r} \quad (8)$$

$$\tau = \int_0^t \frac{c}{a(t')} dt' \quad (9)$$

The significance of the conformal time is that it represents the farthest distance that a photon could have traveled during the universe’s existence. In Eq. (6) and (7), the dot denotes a derivative with respect to physical time t .

I.5. Cosmological Parameters

The Hubble constant H is defined as the slope of the line which best predicts the relationship between the distance from us to an object and the velocity at which that object recedes from us, i.e.

$$v \sim Hr$$

where r is the distance and v the velocity. The Hubble constant is a function of time, but we often refer to its current value, which is called H_0 . Then we define the critical density as

$$\rho_c = \frac{3H_0^2}{8\pi G} \quad (10)$$

| Parameter | Value |
|-------------------|----------------------|
| $\Omega_b h^2$ | 0.2237 ± 0.00015 |
| $\Omega_c h^2$ | 0.1200 ± 0.0012 |
| Ω_Λ | 0.6847 ± 0.0073 |
| Ω_m | 0.3153 ± 0.0073 |
| H_0 [km/ Mpc s] | 67.36 ± 0.54 |

It turns out that if the overall energy density of the universe equals the critical density, then the universe is flat, meaning that on large scales the Minkowski metric $\eta_{\mu\nu}$ accurately describes the geometry of spacetime. We will assume that this is true for the remainder of this paper, meaning that we treat the curvature parameter K as zero in Eq. (6). It is often convenient to talk about the densities of various energy-carriers in relation to the critical density, leading to the definition of the density parameters. For a substance with density ρ_a at present time, the density parameter Ω_a is defined as

$$\Omega_a = \frac{\rho_a}{\rho_c}$$

In the standard Λ CDM cosmology there are four of them: Ω_b , Ω_c , Ω_γ , Ω_Λ . Because we are assuming a flat spacetime, all of the density parameters must sum to 1. For models with nonzero curvature the density parameters are still required to sum to 1, but there is an additional curvature density parameter Ω_K . Sometimes Ω_Γ is taken to include both photons and neutrinos, but in this work we will include a separate neutrino density parameter Ω_ν . The parameters and their values, as determined by Planck [10], are shown in Table (I.5). Note that the dimensionless Hubble parameter h is defined such that $H_0 = (100 \text{ km s}^{-1} \text{Mpc}^{-1})h$. Some of the best constraints on these parameters come from measurements of the cosmic microwave background, which will now be described. The density parameters represent the densities today, and are constants, but the actual densities change as a function of time, and their time dependence can be obtained from the Friedmann Equations, as shown later in this thesis. Because the scale factor is a monotone increasing function of time, it is often used as a time coordinate instead, which is especially useful for talking about the time-dependence of the densities. In this scheme, for example, matter and radiation densities are proportional to a^3 and a^4 respectively. Because initially there is more radiation than matter, there therefore must be a time where the densities of radiation and matter are equal. The value of the scale factor at the time of matter-radiation equality (a_{eq}) is an important quantity

which can be computed from the density parameters, as shown later in this thesis.

I.5.1. Recombination

Information on recombination can be found in Ref. [7]. The isotropy of the CMB implies that the set of spacetime events from which the radiation originates is causally connected. This can be explained by the theory of cosmic inflation, which says that the universe expanded exponentially in a period of about 10^{-32} seconds, with distances being multiplied by a factor of roughly 10^{22} so the particles which emitted the CMB photons we see today were very close together at the time of emission. Together, the set of spacetime events where the present-day CMB originated from is known as the surface of last scattering.

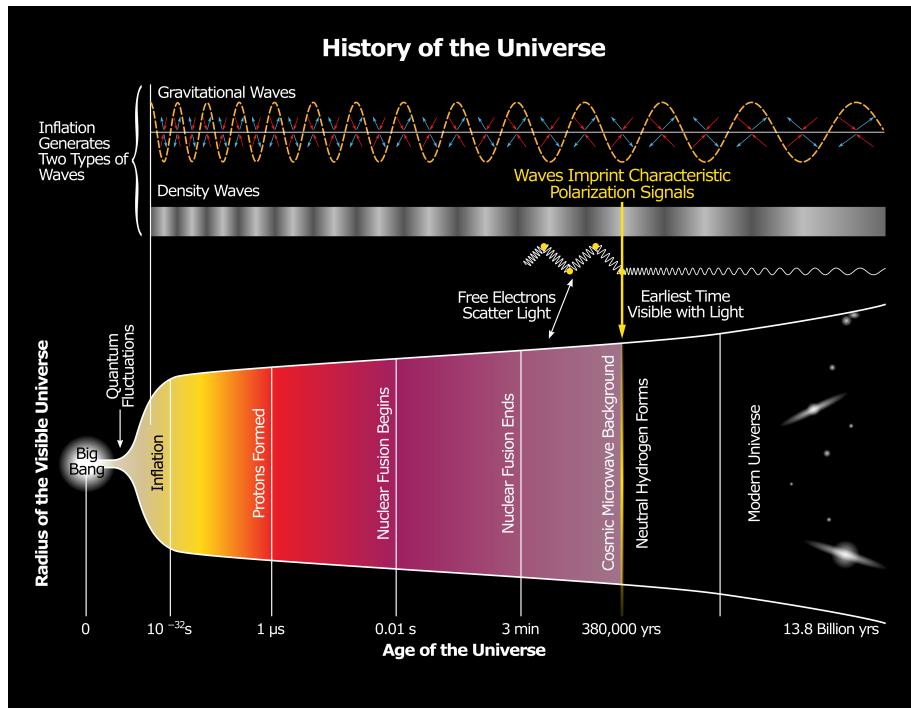


Figure 4. An illustration of the history of the universe. Reproduced from ref. [12].

In order for us to see black-body radiation emitted by particles in the early universe it is necessary that there is an unobstructed path between us and those particles. Because the temperatures were very high and the universe was very dense shortly after the big bang, light and matter were coupled together in a plasma; photons could not travel very far unobstructed because they would scatter off electrons. Therefore we do not see radiation from the early universe until after the temperatures had decreased to about 3000 K, at which

point the electrons and protons are low enough in energy to form hydrogen atoms. This is called the recombination epoch. The change in the transmission properties of the early universe can be quantified using the Thomson scattering rate.

The rate at which electrons and photons scatter is determined by the mean free path λ :

$$\lambda = \frac{1}{x_e n_e \sigma_T} \quad (11)$$

Where n_e is the total number density of electrons, x_e is the fraction of electrons which are not bound to Hydrogen atoms, and σ_T is the Thomson cross section (i.e. the effective cross-sectional area for electron-photon interactions). Recall that the current conformal time (denoted τ_0) is the distance that photons have traveled from the big bang to reach us now. The fraction of photons which were not deflected in the interval between their time of origin and the present is roughly equal to the difference in conformal times divided by the mean free path.

$$\frac{\text{photons received}}{\text{photons sent}} \approx \frac{\tau_0 - \tau}{\lambda} = (\tau_0 - \tau) x_e n_e \sigma_T$$

The Thomson opacity μ is defined as the inverse of the above expression, the ratio of photons sent to photons received. Accounting for the fact that the free electron fraction is also a function of time, and that the mean free path is scaled by the scale factor, we arrive at Eq. (12) for μ .

$$\mu(\tau) = \int_{\tau}^{\tau_0} \frac{d\mu}{d\tau} d\tau \quad (12)$$

$$= \int_{\tau}^{\tau_0} a x_e n_e \sigma_T d\tau \quad (13)$$

This equation can be found in Ref. [13]. In the time period before recombination, the free electron fraction is near 1, and Thomson scattering interactions determine the behavior. The combination of these fluid interactions and the force of gravity allows for the formation of standing waves in the density, which are called Baryon Acoustic Oscillations (BAO).

I.6. Supernova data

Another source cosmological data comes from type 1a supernovae. For the purpose of this thesis, the relevant thing to know about 1a supernovae is that there are a lot of them, they have predictable luminosity, and a predictable emission spectrum. Because the luminosity is known, these events are called “standard candles”, and we can measure their distances based on how bright they appear to yield the luminosity distance, D_L , which is given by Eq (14).

$$D_L = 10^{\frac{m-M}{5}+1} \quad (14)$$

Where M is the absolute magnitude and m the apparent magnitude. Because the wavelength of the emitted light is known, we can also calculate the redshift fairly accurately, which is defined by Eq. (15).

$$z = \frac{\lambda_{\text{observed}} - \lambda_{\text{emitted}}}{\lambda_{\text{emitted}}} \quad (15)$$

The redshift tells us how much the scale factor has changed since the light was emitted. Therefore the relationship between luminosity distance and redshift can tell us about the expansion history of the universe. Finally, if we assume that the expansion is described by the Friedmann Equations and the Λ CDM model, then we can use these results to constrain cosmological parameters like Ω_c , Ω_b , etc. In this paper though, we will mainly explore the constraints which can be obtained by using the CMB to study Baryon Acoustic Oscillations.

I.7. Baryon Acoustic Oscillations

The imprint of early-universe oscillations on the CMB can be predicted by treating the various energy components of the universe as fluids which only interact with each other gravitationally. Unfortunately, some of these fluids are moving at relativistic speeds, and must be treated with special relativity. Therefore, instead of considering densities and pressures to be invariant across reference frames, we must group them together into an object called the Stress-Energy tensor, which allows for an interchange between density and pressure when changing to a moving reference frame.

I.8. The Stress-Energy Tensor

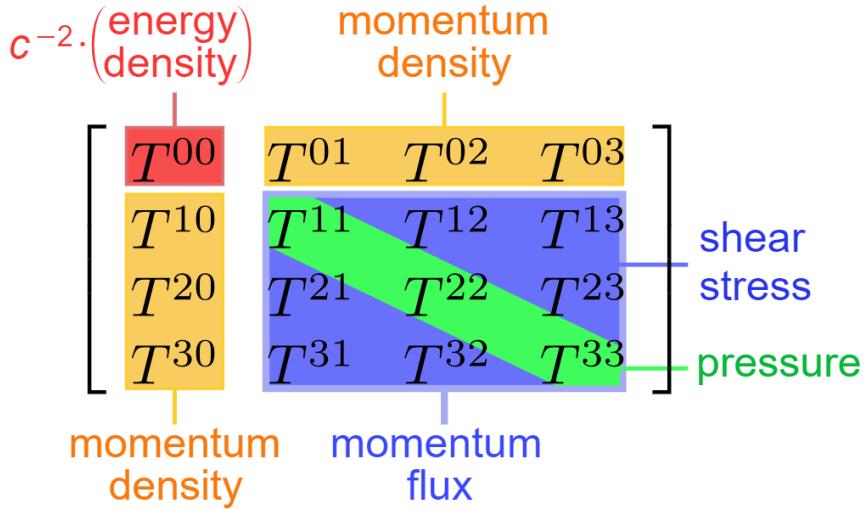


Figure 5. The 16 components of the Stress-Energy Tensor. There are 9 independent components. [14]

The stress-energy tensor $T^{\mu\nu}$ can be thought of as a matrix where μ and ν index the four rows and four columns respectively. It varies as a function of time and space, and describes the state of a continuous distribution of matter. It is called a tensor because under a differentiable coordinate transformation $x^\mu \mapsto x'^\mu$ it transforms as follows [11]:

$$T^{\mu'\nu'}(x') = T^{\mu\nu}(x) \frac{\partial x'^\mu}{\partial x^\mu} \frac{\partial x'^\nu}{\partial x^\nu}$$

$T^{\mu\nu}$ is called a symmetric tensor because $T^{\mu\nu} = T^{\nu\mu}$ for any μ and ν . Note that T^{00} is also not a Lorentz-invariant quantity; simply executing a Lorenz boost to a frame moving in the $+x^1$ direction gives

$$\begin{aligned} x^0 &\rightarrow \gamma(x^0 - vx^1) \\ x^1 &\rightarrow \gamma(x^1 - vx^0) \end{aligned}$$

and therefore

$$T^{00} \rightarrow T^{00}\gamma^2 + T^{11}\gamma^2 + 2T^{01}\gamma^2v^2$$

It is true, however, that T^{00} corresponds to a conserved quantity, which is what we call energy. That is a consequence of the fact that each row of the stress-energy tensor (and

therefore each column), when considered as a vector field $\mathbb{R}^4 \rightarrow \mathbb{R}^4$, is divergenceless:

$$\partial_\nu T^{\mu\nu} = 0 \quad (16)$$

Eq. (16) is called the relativistic continuity equation. Equation (16) is not enough to fully specify the dynamics. We also need to apply the other relativistic Euler equation, Eq. (17), which can be found in Daniel Baumann's cosmology notes [7].

$$T^{\mu\nu} = (\rho + p)u^\mu u^\nu + p\eta^{\mu\nu} \quad (17)$$

Where ρ is the energy density in the rest frame, p is the pressure, $u = (\gamma c, \gamma \mathbf{v})$ the relativistic four-velocity and $\eta = \text{diag}(-1, 1, 1, 1)$ is the Minkowski metric. For a system of non-interacting fluids, each fluid has its own stress-energy tensor $T_i^{\mu\nu}$. We will also speak of the individual energy densities and pressures, ρ_i and p_i respectively.

In reality we suspect that there are at least six different components in the universe at the time of Recombination: atoms, photons, neutrinos, dark matter, and dark energy. However, the density of dark energy at this time is negligible compared to the other components, so it is ignored. Neutrinos and photons are both relativistic, and so these might be treated as a single fluid.

Here we give an account of linear perturbation theory in a non-relativistic inviscid fluid with no thermal conductivity. The equations used later governing perturbations will include corrections from general relativity which will not be derived, but they still resemble the equation which can be obtained in the Newtonian limit, so it is a useful exercise to build intuition.

I.8.1. Single Fluid Approximation

A detailed derivation of the non-relativistic equations governing the evolution of oscillations in the primordial fluid can be found in the section on perturbations in Bartelmann's cosmology [6]. After inflation and before recombination, there must be macroscopic variation in density because quantum fluctuations have been magnified without time for equilibrium to be attained. In this period dynamics are dominated by short range interactions and we can approximate the plasma as a single fluid with density field ρ , pressure field p , and ve-

locity field v . If we assume that this fluid has zero viscosity and zero thermal conductivity (i.e. is a perfect fluid) then we can use the Euler fluid equations:

$$\frac{\partial \rho}{\partial t} = -\nabla \cdot (v\rho) \quad (18)$$

$$\frac{\partial v}{\partial t} + v \cdot \nabla v = -\nabla \Phi - \frac{1}{\rho} \nabla p \quad (19)$$

Where Φ is the gravitational potential field. The Newtonian gravitational potential can be described by Poisson's Equation:

$$\nabla^2 \Phi = 4\pi G \rho \quad (20)$$

For a perfect fluid, the the ratio of pressure and density is fixed, its value is equal to the square of the sound speed c_s :

$$p = c_s^2 \rho \quad (21)$$

Just as the density can be decomposed into the average density and the local fluctuations ($\rho = \bar{\rho} + \delta\rho$), the velocity field can be separated into the motion due to the uniform hubble expansion and the “peculiar velocity”, or the velocity in comoving coordinates. That is, if \mathbf{x} is the comoving position of a fluid element and \mathbf{r} the physical position, we have

$$\dot{\mathbf{r}} = H\mathbf{r} + a\dot{\mathbf{x}} = \mathbf{v}_0 + \delta\mathbf{v}$$

Where $\mathbf{v}_0 = H\mathbf{r}$ and $\delta\mathbf{v} = a\dot{\mathbf{x}}$. The homogeneous quantities \mathbf{v}_0 and $\bar{\rho}$ are required to solve the fluid equations on their own, as are the total velocity and density fields. Also, second order terms (e.g. $\delta\rho\delta\mathbf{v}$) are dropped. We then define the comoving velocity perturbation $\mathbf{u} = \delta\mathbf{v}/a$ and the potential perturbation $\delta\Phi$ such that $\nabla^2\delta\Phi = 4\pi G \bar{\rho}\delta$. Using these quantities and the density contrast δ , Equations (18), (19) and (20) can be transformed into the following set of perturbation equations.

$$\dot{\delta} + \nabla \cdot \mathbf{u} = 0 \quad (22)$$

$$\dot{\mathbf{u}} + 2H\mathbf{u} = -\frac{\nabla\delta p}{a^2\rho_0} + \frac{\nabla\delta\Phi}{a^2} \quad (23)$$

$$\nabla^2\delta\Phi = 4\pi G \bar{\rho} a^2 \delta \quad (24)$$

Which yield the second order differential equation

$$\ddot{\delta} + 2H\dot{\delta} = 4\pi G\bar{\rho}\delta + \frac{c_s^2\nabla^2\delta}{a^2} \quad (25)$$

We'll let $\tilde{\delta}$ denote the fourier transform of δ :

$$\begin{aligned} \tilde{\delta}(k, t) &= \int \delta(x, t)e^{-ik\cdot x}d^3x \\ \delta(x, t) &= \frac{1}{(2\pi)^3} \int \tilde{\delta}(k, t)e^{ik\cdot x}d^3k \end{aligned}$$

Taking the Fourier transform of Eq. (25) yields

$$\ddot{\tilde{\delta}} + 2H\dot{\tilde{\delta}} = \tilde{\delta} \left(4\pi G\bar{\rho} - \frac{c_s^2 k^2}{a^2} \right) \quad (26)$$

Where we have used the fact that the Fourier transform and differentiation are both linear operations, as well as the property that the Fourier transform of $\nabla^2\delta$ is equal to $k^2\tilde{\delta}$. Eq. 26 tells us that a mode with wavenumber k will oscillate with frequency given by:

$$\omega = \sqrt{wk^2 - 4\pi G\bar{\rho}} \quad (27)$$

Note that if $wk^2 < 4\pi G\bar{\rho}$, the solution is not oscillatory, either grows or decays. For this reason the Jeans wavenumber is defined as the minimum wavenumber for which oscillations can develop:

$$k_J = \sqrt{\frac{4\pi G\bar{\rho}}{w}} \quad (28)$$

For wavelengths larger than the Jeans wavelength, the behavior is determined by whether the wavelength is larger or smaller than the hubble radius $R_H = \frac{c}{H}$, and whether the universe is radiation-dominated or matter-dominated. If $\lambda > R_H$ then $\tilde{\delta}$ will scale with a^2 if during a radiation-dominated era and with a during a matter-dominated era. If $\lambda < R_H$ then $\tilde{\delta}$ will be relatively constant during a radiation-dominated era and will scale with a during a matter-dominated era [6]. The solution for the single fluid case is particularly simple, but unfortunately the universe is more complicated, and must be modeled using multiple fluid components. We will eventually attempt to use the Generalized Dark Matter (GDM) formalism to create a slightly more complicated single-fluid description which still exhibits

the behavior of the more complete treatment. Below we outline the basic concept of GDM and see how it may help us.

I.8.2. Generalized Dark Matter

Quite generally, a system of coupled differential equations can be transformed into a single differential equation of higher order. Alternatively, a system can be transformed into a differential equation of the same order, but with some constant replaced by a time dependent function. Consider, for example, a pair of coupled oscillators described by the system:

$$\begin{aligned} m_1 \ddot{x}_1 &= -k_1 x_1 - k_2(x_1 - x_2) \\ m_2 \ddot{x}_2 &= -k_2(x_2 - x_1) \end{aligned}$$

These can be transformed into the single fourth order equation for x_2 :

$$m_1 m_2 x_2^{(4)} + (m_2 k_1 + k_2(m_1 + m_2)) \ddot{x}_2 + k_1 k_2 x_2 = 0$$

But if we have access to an experiment or simulation of the system, we can take another approach: We can observe the motion and record the function $x_2(t)$. Then we define a time varying spring constant $\tilde{k}_2(t)$ as:

$$\tilde{k}_2(t) \equiv -\frac{m_2 \ddot{x}_2(t)}{x_2(t)}$$

This ensures that the motion of the mass m_2 is described by the differential equation:

$$\ddot{x}_2 = -\tilde{k}_2(t)x_2$$

If we can only see the motion of the second mass, there is no way of knowing for sure whether the underlying mechanism is a system of two coupled oscillators or a single spring with a time varying spring constant. Generally, we might assume that the coupled oscillator explanation is more satisfactory (because it requires less information to describe), unless the time-varying k model results in the function function $\tilde{k}_2(t)$ having some theoretically

favorable form.

The idea of generalized dark matter is basically to carry out the above process on a system of coupled fluids in a cosmological model, where the parameters w and c_s^2 basically fill the role of the spring constant. This approach is useful for the following reason: Though we suspect that the unseen energy in the universe is made up of multiple different components with different constant properties, we don't know for sure because the data we have access to is the signature of the time evolution of the *total* energy density. We can simulate data such as the perturbation history using our current hypothesis of multiple fluid components, and then produce a model based on a single fluid whose parameters change in time in order to perfectly reproduce the behavior of the fiducial model. We may find that these parameters have to depend on time in drastic and unlikely ways in order for this to happen. Or we may find that the time-dependence of the single-fluid model is relatively stable (in which case our multi-fluid model might be “over-fitting”), or even suggestive of a different physical mechanism than we currently suspect. Keeping this GDM formalism in mind, we now move on to describe a somewhat more complicated fluid model of the universe, this time with two fluids interacting only gravitationally. This model includes relativistic effects which we won't derive, but otherwise bears a lot of resemblance to the equations of motion for the single fluid.

I.8.3. Two Fluid Approximation

Seljak [13] gives the corresponding perturbation equations for a “two fluid” model. The system includes radiation, baryonic matter, and cold dark matter, but baryonic matter and radiation are treated as one fluid with dependence $\delta_b = \frac{3}{4}\delta_\gamma$ and $v_b = v_\gamma$. The quantities on the left hand sides of Equations 40 are all coefficients of Fourier modes for the dimensionless wave-vector $\kappa = k\tau_r$, where τ_r is conformal time at recombination. δ_c and v_c are the dimensionless perturbations to density and velocity for cold dark matter respectively. δ_γ and v_γ are the dimensionless perturbations to density and velocity for the coupled radiation and baryonic matter fluid. ϕ is the gravitational potential perturbation and δ and v are the

overall density and velocity perturbations. Also, $y = \frac{a}{a_{eq}}$, $a_{eq} = \frac{\Omega_\gamma}{\Omega_m}$, $y_b = \frac{\bar{\rho}_b}{\bar{\rho}_\gamma}$, and $y_c = \frac{\Omega_c}{\Omega_m}y$.

$$\begin{aligned}
\dot{\delta}_c &= -\kappa v_c + 3\dot{\phi} & \dot{v}_c &= -\eta v_c + \kappa\phi \\
\dot{\delta}_\gamma &= -\frac{4}{3}\kappa v_\gamma + 4\dot{\phi} & \dot{v}_\gamma &= \left(\frac{4}{3} + y_b\right)^{-1} \left[-\eta y_b v_\gamma + \frac{\kappa\delta_\gamma}{3} + \kappa\phi \left(\frac{4}{3} + y_b\right) \right] \\
\phi &= -\frac{3}{2}(\eta/\kappa)^2(\delta + 3\eta v/\kappa) & \dot{\phi} &= -\eta\phi + \frac{3\eta^2 v}{2\kappa} \\
\delta &= \frac{\delta_\gamma \left[1 + \frac{3}{4}(y - y_c) \right] + y_c \delta_c}{1 + y} & v &= \frac{v_\gamma \left(\frac{4}{3} + y - y_c \right) + y_c v_c}{1 + y}
\end{aligned} \tag{29}$$

These equations can be evolved until x_{rec} for different values of κ in order to determine the Fourier power spectrum (different from the multipole power spectrum) at the time of recombination.

I.8.4. Plane wave expansion

Notice that Equations (40) describe the evolution of perturbation modes in terms of a scalar wavenumber rather than a three-dimensional wave-vector. These can be taken as valid equations governing the development of plane waves, which would hold good in a universe where all quantities depend only on one spacial coordinate. The bad news is that we do not live in such a universe. The good news is that all perturbations can be expressed as superpositions of plane wave solutions oriented on various axes. When we look at the CMB we are seeing a spherical section of this superposition of plane waves. If we want to predict the multipole expansion of the CMB anisotropies we need to know what a spherical section of a single plane wave looks like. Fortunately, this is described by Rayleigh's plane wave expansion. Consider a plane wave whose wave vector \mathbf{k} has spherical coordinates (k, θ_k, ϕ_k) . The value of this function at a point \mathbf{r} with spherical coordinates (r, θ, ϕ) can be expressed by the plane wave expansion, given in Eq. (30).

$$\exp(i\mathbf{k} \cdot \mathbf{r}) = 4\pi \sum_{\ell=0}^{\infty} \sum_{m=-\ell}^{\ell} i^\ell j_\ell(kr) Y_{\ell m}^*(\theta_k, \phi_k) Y_{\ell m}(\theta, \phi) \tag{30}$$

Where j_ℓ is the ℓ th spherical Bessel function. This is the reason that spherical Bessel functions will be present in the final expressions for the averaged multipole coefficients.

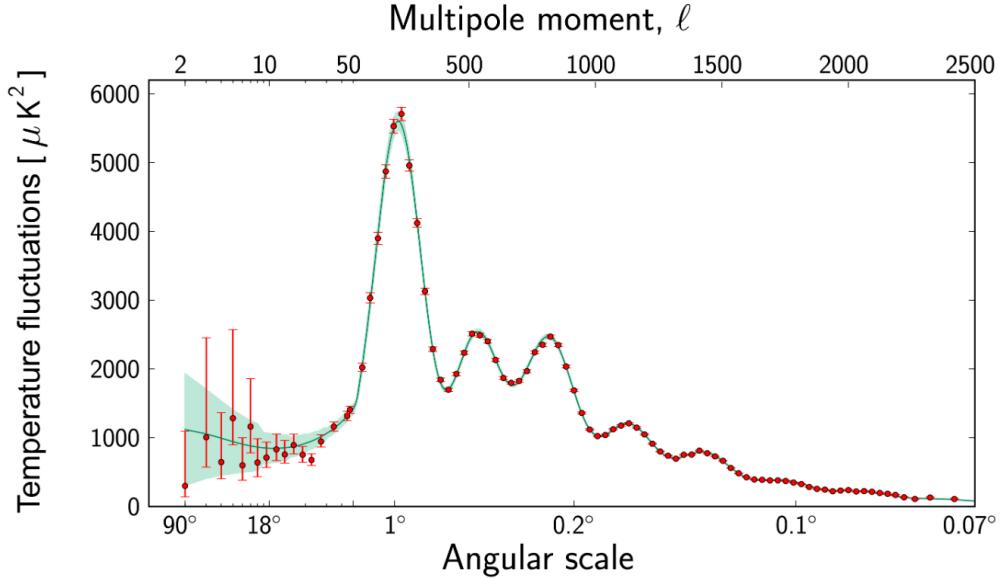


Figure 6. Theoretical power spectrum of the CMB (solid line) and experimental constraints (vertical bars). Reproduced from ref. [?]

I.9. Principle Component Analysis

Principal Component Analysis (PCA)¹ is a technique used to reduce the dimensionality of a data set by identifying the largest modes of variation. Intuitively, the idea is that sometimes the data is described by a collection of variables which is redundant, or there are strong correlations in the data which make a certain linear transformation of the coordinates advantageous.

I.9.1. The Fisher Matrix

Say we have designed an experiment to measure some set of numbers $X_1 \dots X_N$, which will be stored as components of the vector $X \in \mathbb{R}^N$. For instance these numbers might be values of C_ℓ for the CMB temperature fluctuations where $\ell = 2, 3, 4, \dots N + 1$. Further

¹ See refs. [?] and [15] for a detailed explanation of PCA. See ref. [16] for an application of PCA to cosmological constraints.

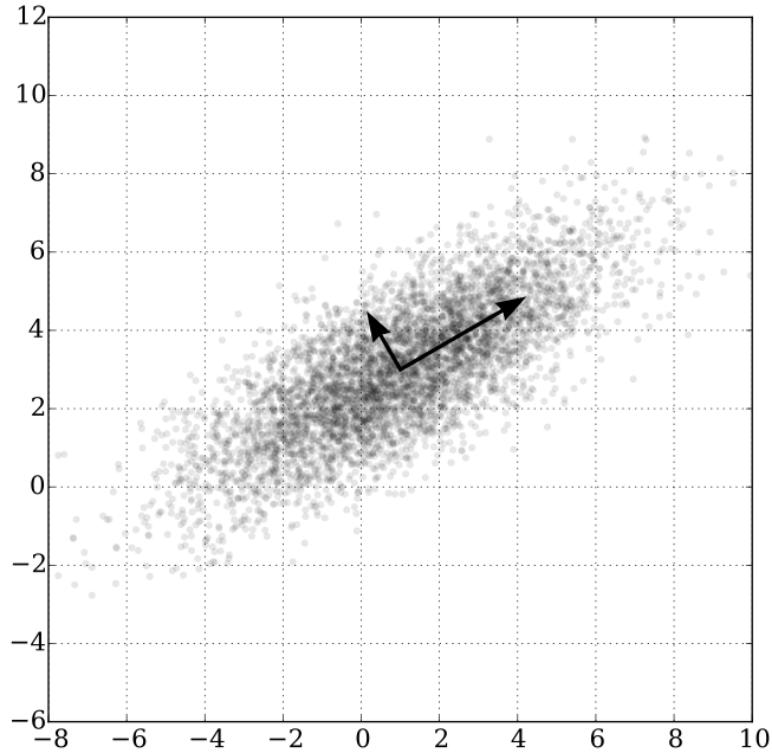


Figure 7. A bivariate data set (scattered dots) and the principal component vectors (arrows). The largest principal component captures most of the variation in the data. Reproduced from ref. [17].

suppose that we have access to a model which predicts the outcome of the experiment based on the values of a set of parameters $\theta_1 \dots \theta_M$, comprising the vector $\theta \in \mathbb{R}^M$. The parameters θ_j might include the present-day density parameters (Ω_m , Ω_r , etc.) or the current Hubble constant H_0 , for example. Generally there some parameter values will be considered more likely than others to be the true values, based on prior knowledge. The set of most likely, or “fiducial”, values for the parameters should be determined and stored in the vector $\hat{\theta}$.

In this context the model can be viewed as a function $f : \mathbb{R}^M \rightarrow \mathbb{R}^N$. Assuming this function is differentiable at the point $\hat{\theta}$, we can form the Jacobian matrix at $\hat{\theta}$:

$$J_{ij} = \left. \frac{\partial X_i}{\partial \theta_j} \right|_{\hat{\theta}}$$

Since we haven’t actually done the experiment yet, X can be considered a random vector. Usually we can assume that the components of X are drawn from a multivariate normal

distribution with mean μ and covariance matrix Σ :

$$P(X; \theta) = \frac{1}{(\sqrt{2\pi})^N |\Sigma|} \exp \left[-\frac{(X - \mu)^T \Sigma^{-1} (X - \mu)}{2} \right] \quad (31)$$

The entries of Σ will be determined by the experimental methods, but in any case the covariance matrix is a real and symmetric $N \times N$ matrix. Frequently Σ will be a diagonal matrix where every diagonal entry is nonzero, meaning that the measured quantities are uncorrelated and have nonzero variance. The Fisher Information Matrix (FIM) for the parameters θ_j with respect to observables X_i is defined in Eq. (32).

$$F = J^T \Sigma^{-1} J \quad (32)$$

Note that Σ must be invertible in order for Eq. (32) to be useful. In fact, if Σ is singular then the probability density given by Eq. (31) is undefined, so there was never a valid probabilistic model anyway, meaning the parameter sets need to be adjusted. The Fisher matrix is a real and symmetric $M \times M$ matrix.

1.9.2. Principal Components

Once the Fisher matrix has been obtained, principal components can be found by using some numerical method to find an eigendecomposition of F . An eigendecomposition of a matrix A consists of a diagonal matrix D and an invertible matrix P satisfying the equation

$$A = P D P^{-1}$$

The columns of the matrix P in the eigendecomposition of F are eigenvectors of F , and the entries on the diagonal of D are the associated eigenvalues; each eigenvalue is associated with one eigenvector, and the eigenvector and associated eigenvalue are found in the same columns of their respective matrices. When the eigenvectors are listed in order of decreasing magnitude of associated eigenvalue, there will generally be a small number of eigenvectors with large associated eigenvalues, and a steep drop-off after the first few. The first few eigenvectors (with large eigenvalues) are then called the Principal Components (PCs) of the Fisher Matrix.

It should always be possible to find such an eigendecomposition for the matrix F , but sometimes there will be eigenvalues that are equal to zero, meaning that the rows/columns of F are not pairwise linearly independent. This means that the linearized model has a degeneracy with respect to some combination of parameters. More specifically, there is some linear combination of parameters which can be varied in the neighborhood of $\hat{\theta}$ without affecting the output of the model. Generally, for each zero eigenvalue, the number of parameters θ_j can be reduced by one.

I.10. This Thesis

In this work we will develop model based on Seljak’s two fluid approximation [13], but which includes neutrinos explicitly as a third fluid component. The model will be used to generate a Fisher matrix with respect to certain parameters we would like to constrain (for example, the neutrino effective sound speed or the expansion history of the universe). We will then use the techniques of principal component analysis to obtain the best-constrained components, linear combinations of the parameters which we would be able to get the tightest bounds on using measurements of the CMB.

II. BACKGROUND COSMOLOGY

Here we derive the relations which are used to evolve the homogeneous “background” quantities. The time-dependent behavior of the background cosmology is captured by the scale factor as a function of time, $a(t)$. We can solve for $a(t)$ using the Friedmann Equations (6 and 7). Eq. (7) can be solved by Eq. (33).

$$\rho = \rho_0 a^{-3(1+w)} \quad (33)$$

Where $w = \frac{p}{\rho}$ is the equation of state. We change variables to differentiate with respect to conformal time τ , which we denote with a prime.

$$\left(\frac{a'}{a}\right)^2 = \frac{8\pi G a^2}{3} \bar{\rho}$$

Photons and neutrinos have $w = -1/3$ while baryonic matter and cold dark matter have $w = 0$. Substituting in these values, we have:

$$\left(\frac{a'}{a}\right)^2 = \frac{8\pi G a^2}{3} [(\bar{\rho}_{\gamma,0} + \bar{\rho}_{\nu,0})a^{-4} + (\bar{\rho}_{b,0} + \bar{\rho}_{c,0})a^{-3}] \quad (34)$$

$$(a')^2 = \frac{8\pi G}{3} [(\bar{\rho}_{\gamma,0} + \bar{\rho}_{\nu,0}) + (\bar{\rho}_{b,0} + \bar{\rho}_{c,0})a]$$

We define a_{eq} to be the scale factor for which the density of radiation equals the density of matter, calculated as:

$$\begin{aligned} (\bar{\rho}_{\gamma,0} + \bar{\rho}_{\nu,0})a_{eq}^{-4} &= (\bar{\rho}_{b,0} + \bar{\rho}_{c,0})a_{eq}^{-3} \\ a_{eq} &= \frac{\bar{\rho}_{\gamma,0} + \bar{\rho}_{\nu,0}}{\bar{\rho}_{b,0} + \bar{\rho}_{c,0}} \end{aligned}$$

We can substitute a_{eq} into the previous expression to get the following:

$$(a')^2 = \frac{8\pi G}{3} (\bar{\rho}_{b,0} + \bar{\rho}_{c,0})(a + a_{eq}) = H_0^2 \Omega_m (a + a_{eq})$$

Following Seljak [13], we define the variables $y = \frac{a}{a_{eq}}$, $x = \sqrt{\frac{\Omega_m}{a_r}} H_0 \tau$, and $\alpha = \sqrt{\frac{a_r}{a_{eq}}}$, with $a_r \sim \frac{1}{1100}$.

$$\left(\frac{dy}{dx}\right)^2 = \frac{a_r}{\Omega_m H_0^2 a_{eq}^2} \left(\frac{da}{d\tau}\right)^2 = \frac{a_r}{a_{eq}} (y+1) = \alpha^2 (y+1)$$

The solution to the differential equation is:

$$y = (\alpha x)^2 + 2\alpha x \quad (35)$$

Eq. 35 describes the evolution of the scale factor as a function of conformal time.

III. PERTURBATIONS

We now discuss the evolution of the inhomogeneous metric perturbations. The full derivation is produced by considering a first-order perturbation to the spacetime metric g in the Einstein Equation. In the conformal Newtonian gauge, Ma and Bertschinger (1995) [18] give the equations for the perturbation evolution as the following system of two first order

ODEs:

$$\dot{\delta} = -(1+w)(\theta - 3\dot{\phi}) - 3\frac{\dot{a}}{a} \left(\frac{\delta P}{\delta \rho} - w \right) \delta \quad (36)$$

$$\dot{\theta} = \frac{\dot{a}}{a}(1-3w)\theta - \frac{\dot{w}}{1+w}\theta + \frac{\delta P/\delta \rho}{1+w}k^2\delta - k^2\sigma^2 + k^2\psi \quad (37)$$

Where k is wavenumber, $\delta(k) = \mathcal{F}\{\frac{\delta\rho}{\rho}\}$ is the fourier transform of the density perturbation, $\theta(k)$ is the Fourier transform of the scalar velocity potential, and $\phi(k)$, $\psi(k)$, and $\sigma(k)$ are the Fourier transforms of the three components of the metric perturbation. We then make the following assumptions:

1. $\sigma = 0$.
2. $\psi = \phi$.
3. $\delta P/\delta \rho = w$. This quantity is equal to the square of the sound speed c_s . The assumption will be relaxed when we consider the GDM model.
4. For baryonic matter and cold dark matter $w = 0$, while for radiation and neutrinos $w = 1/3$. $\dot{w} = 0$

For cold dark matter then, for example, Equations 36 and 37 reduce to the following equations

$$\begin{aligned} \dot{\delta}_c &= -kv_c + 3\dot{\phi} \\ \dot{v}_c &= -\frac{\dot{a}}{a}v_c + k\phi \end{aligned}$$

Equations 36 and 37 are valid for non-interacting fluids, but Ma and Bertschinger [18] show that when baryonic matter and radiation are coupled, a new term enters the expression for the radiation velocity potential evolution which is equal to $an_e\sigma_T(\theta_b - \theta_\gamma)$, where σ_T is the Thomson cross-section and n_e is the electron number density. Similarly, a term enters the expression for the evolution of the baryonic matter potential velocity which is equal to $\frac{4\bar{\rho}_\gamma}{\bar{\rho}_b}\dot{\mu}(v_\gamma - v_b)$. Seljak defines the quantity $\dot{\mu} = an_e\sigma_T$. Therefore, in Seljak's notation, and subject to the assumptions above, the equations governing evolutions of photon, cold dark

matter, baryonic matter, and neutrino perturbations take the following form:

$$\begin{aligned}
\dot{\delta}_b &= -kv_b + 3\dot{\phi} & \dot{v}_b &= -\frac{\dot{a}}{a}v_b + \frac{4\bar{\rho}_\gamma}{3\bar{\rho}_b}\dot{\mu}(v_\gamma - v_b) + k\phi \\
\dot{\delta}_c &= -kv_c + 3\dot{\phi} & \dot{v}_c &= -\frac{\dot{a}}{a}v_c + k\phi \\
\dot{\delta}_\gamma &= -\frac{4}{3}kv_\gamma + 4\dot{\phi} & \dot{v}_\gamma &= \frac{k\delta_\gamma}{4} + \dot{\mu}(v_b - v_\gamma) + k\phi \\
\dot{\delta}_\nu &= -\frac{4}{3}kv_\nu + 4\dot{\phi} & \dot{v}_\nu &= \frac{k\delta_\nu}{4} + k\phi
\end{aligned} \tag{38}$$

We also must solve for the metric perturbation ϕ as a function of time. The conformal time derivative of the gravitational potential evolves in accordance with Eq. (39).

$$\phi' = -\frac{a'}{a}\phi + \frac{4\pi Ga^2 f}{k} \tag{39}$$

Where $f = \frac{4}{3}v_\gamma\bar{\rho}_\gamma + \frac{4}{3}v_\nu\bar{\rho}_\nu + v_b\bar{\rho}_b + v_c\bar{\rho}_c$. Substituting in the densities as a function of the scale factor gives

$$f = \frac{4}{3}(v_\gamma\bar{\rho}_{\gamma,0} + v_\nu\bar{\rho}_{\nu,0})a^{-4} + (v_b\bar{\rho}_{b,0} + v_c\bar{\rho}_{c,0})a^{-3}$$

At this point we will introduce the “two fluid approximation” of Seljak’s paper as a baseline model.

III.1. Two Fluid Approximation

This approximation is obtained by setting $\delta_\nu = \delta_\gamma = \frac{4}{3}\delta_b$ and $v_\nu = v_\gamma = v_b$. This is basically justified by the idea that baryons and photons are coupled into a single fluid, and that photons and neutrinos act very similarly.

III.1.1. Nondimensionalization

Recall that the evolution of the scale factor was expressed in terms of dimensionless quantities x , y , and α . In the dimensionless perturbation equations, dotted variables will be differentiated with respect to the dimensionless time coordinate x . In order to remove dimensions from the full perturbation equations, we define $y_c \equiv \frac{\bar{\rho}_{c,0}}{\bar{\rho}_{m,0}}y$, $y_c \equiv \frac{\bar{\rho}_{c,0}}{\bar{\rho}_{m,0}}y$, $\tau_r \equiv$

$\frac{2}{H_0} \sqrt{\frac{a_r}{\Omega_m}}$, $\kappa \equiv k\tau_r$, and

$$\eta \equiv \frac{2\alpha(\alpha x + 1)}{\alpha^2 x^2 + 2\alpha x}$$

In terms of these unitless parameters, the evolution equation becomes:

$$\frac{d\phi}{dx} = \frac{\sqrt{a_r}}{\sqrt{\Omega_m} H_0} \phi' = \frac{\sqrt{a_r}}{\sqrt{\Omega_m} H_0} \left(-\frac{y'}{y} \phi + \frac{4\pi G a^2 f}{k} \right)$$

The second term becomes

$$\frac{4\pi G a^2 f}{k} = 4\pi G \bar{\rho}_{m,0} \frac{a_{eq}(v_\gamma(\frac{4}{3} + y - y_c) + v_c y_c)}{a^2 k}$$

So the equations governing the evolution of perturbations in the two-fluid approximation are given by 40:

$$\begin{aligned} \dot{\delta}_c &= -\kappa v_c + 3\dot{\phi} & \dot{v}_c &= -\eta v_c + \kappa\phi \\ \dot{\delta}_\gamma &= -\frac{4}{3}\kappa v_\gamma + 4\dot{\phi} & \dot{v}_\gamma &= \left(\frac{4}{3} + y_b\right)^{-1} \left[-\eta y_b v_\gamma + \frac{\kappa\delta_\gamma}{3} + \kappa\phi \left(\frac{4}{3} + y_b\right) \right] \\ \dot{\phi} &= -\eta\phi + \frac{3\eta^2 v}{2\kappa} & v &= \frac{v_\gamma \left(\frac{4}{3} + y - y_c\right) + y_c v_c}{1 + y} \end{aligned} \quad (40)$$

III.2. Three Fluid Approximation

Now we include v_ν separately while still keeping $v_b = v_\gamma$. The equations governing the evolution of perturbations in the three fluid approximation are given by 41.

$$\begin{aligned} \dot{\delta}_c &= -\kappa v_c + 3\dot{\phi} & \dot{v}_c &= -\eta v_c + \kappa\phi \\ \dot{\delta}_\gamma &= -\frac{4}{3}\kappa v_\gamma + 4\dot{\phi} & \dot{v}_\gamma &= \left(\frac{4}{3} + y_b\right)^{-1} \left[-\eta y_b v_\gamma + \frac{\kappa\delta_\gamma}{3} + \kappa\phi \left(\frac{4}{3} + y_b\right) \right] \\ \dot{\delta}_\nu &= -\frac{4}{3}\kappa v_\nu + 4\dot{\phi} & \dot{v}_\nu &= \frac{\kappa\delta_\nu}{4} + \kappa\phi \\ \dot{\phi} &= -\eta\phi + \frac{3\eta^2 v}{2\kappa} & v &= \frac{v_\gamma \left[\frac{4}{3} + y - y_c\right] + (v_\nu - v_\gamma)y_\nu + v_c y_c}{1 + y} \end{aligned} \quad (41)$$

where $y_\nu \equiv \frac{\bar{\rho}_{\nu,0}}{\bar{\rho}_{m,0}} y$

IV. PERTURBATIONS AND THE CMB POWER SPECTRUM

We have described the model for simulating fluid perturbations, but how can we observe evidence of these ripples? The signature of these perturbations shows up in the CMB power spectrum. The expression for the ensemble-averaged multipole expansion coefficients of the CMB (C_l) can be found in Seljak [13] and Kodama and Sasaki 1984 [19] and is given below:

$$C_l = 4\pi \int_0^\infty k^2 P(k) T(k) D_l^2 dk \quad (42)$$

$$D_l = (\phi + \frac{\delta_\gamma}{4}) j_l(k\tau_0 - k\tau_{rec}) + v_b j'_l(k\tau_0 - k\tau_{rec}) + 2 \int_{\tau_{rec}}^{\tau_0} j_l(k\tau_0 - k\tau) \dot{F}(\tau) \quad (43)$$

$$F(\tau) = \frac{\phi(\tau)}{\phi(\tau_0)} \quad (44)$$

Where $P(k)$ is a function called the primordial power spectrum for the potential ϕ , and is generally taken to be k raised to some power, $T(k)$ is introduced to model damping effects (it is not clarified in those papers what the explicit form of $T(k)$ is or where the damping comes from), and j_l are spherical bessel functions. As far as fluid dynamics is concerned, C_l only depends on the quantities v_b , $(\phi + \frac{\delta_\gamma}{4})$, and \dot{F} , and for this reason these quantities are called “source contributions”.

V. GENERALIZED DARK MATTER

V.1. Description

The Generalized Dark Matter (GDM) formalism was first described in Wayne Hu’s 1998 paper [20], and aims to describe the entire “dark sector” as a single fluid with time-dependent properties. Here we describe the application of the GDM model to the problem at hand. Consider a collection of fluids with equations of motion given by 7, 36, and 37, as well as a suitable interaction term where necessary. Assume that for each fluid individually,

$$\frac{P_i}{\rho_i} = \frac{\delta P_i}{\delta \rho_i} = w_i$$

and $\dot{w}_i = 0$. First we define $\rho_T = \sum \rho_i$, $P_T = \sum P_i$. Then it is straightforward to compute an overall density contrast δ_T :

$$\delta_T = \frac{\sum \delta_i \rho_i}{\sum \rho_i} = \frac{(\delta_\gamma \Omega_\gamma + \delta_\nu \Omega_\nu) a^{-4} + (\delta_\gamma \Omega_b + \delta_c \Omega_c) a^{-3}}{\Omega_\Lambda + (\Omega_\gamma + \Omega_\nu) a^{-4} + (\Omega_b + \Omega_c) a^{-3}} \quad (45)$$

We also define the effective equation of state for the combined fluids:

$$w_T = \frac{P_T}{\rho_T} = \frac{\sum w_i \rho_i}{\rho_T} = \frac{\frac{1}{3}(\Omega_\gamma + \Omega_\nu) a^{-4} - \Omega_\Lambda}{\Omega_\Lambda + (\Omega_b + \Omega_c) a^{-3} + (\Omega_\gamma + \Omega_\nu) a^{-4}} \quad (46)$$

Now Eq. (7) can be rewritten as:

$$\begin{aligned} \dot{\bar{\rho}}_{tot} &= -3\mathcal{H} \sum \bar{\rho}_i (1 + w_i) \\ &= -3\mathcal{H} \bar{\rho}_T \frac{\sum \bar{\rho}_i (1 + w_i)}{\bar{\rho}_T} \\ &= -3\bar{\rho}_T \mathcal{H} (1 + w_T) \end{aligned}$$

The total velocity perturbation is defined as

$$v_T = \frac{\sum \rho_i v_i (1 + w_i)}{w_T \sum \rho_i} \quad (47)$$

Finally we define the total sound speed:

$$c_{sT}^2 = \frac{\delta P_T}{\delta \rho_T} = \frac{\sum \delta_i \rho_i w_i}{\sum \delta_i \rho_i} \quad (48)$$

By construction, if Equations 36 and 37 hold for each fluid individually, then they should hold for the aggregate fluid quantities δ_T , v_T , w_T and c_{sT}^2 . If we let w_T and c_{sT}^2 be functions of time then we can capture the behavior of a collection of fluids using a single fluid with time-dependent properties. The functions $c_{sT}^2(x)$ and $w_T(x)$ will serve as the parameters in principal component analysis.

VI. PRINCIPAL COMPONENT ANALYSIS

In this thesis we are exploring the use of the CMB to obtain principal components of w and c_s^2 , but there are other ways to obtain principal components of functions like this.

Here, as an example, we demonstrate that supernova data can be used to obtain principal components for w . We use this as a more basic example to introduce the technique in the context of cosmological parameters.

VI.1. Luminosity-Redshift Data

VI.1.1. Constant equation of state parameter

Type II Supernovae are a useful tool in cosmology because we can estimate both their distance and velocity indirectly. The velocity is measured using the redshift, and the distance is measured using the Luminosity, and is therefore called Luminosity distance. Eq. 49 predicts a relationship between redshift and luminosity distance measurements in the case where the curvature is zero. 49 [21]:

$$D_L(z) = (1+z)D_H \int_0^z \frac{dz'}{E(z')} \quad (49)$$

Where z is the redshift, D_L the Luminosity distance, $D_H = \frac{c}{H_0}$ is the Hubble distance and $E(z) = H(z)/H_0$ is the ratio of the Hubble parameter at redshift z to the Hubble parameter at redshift zero. “Primed” variables (e.g. z') represent dummy variables for integration, rather than derivatives. When the equation of state parameter for dark energy (w_Λ) is identically equal to negative one, the function $E(z)$ is given by:

$$E(z) = \sqrt{\Omega_R(1+z)^4 + \Omega_M(1+z)^3 + \Omega_k(1+z)^2 + \Omega_\Lambda}$$

Where $\Omega_R, \Omega_M, \Omega_\Lambda$ are the density parameters as defined in the 2015 Planck results [?], and Ω_k is defined as:

$$\Omega_k = 1 - (\Omega_R + \Omega_M + \Omega_\Lambda)$$

If $\Omega_k \neq 0$, then Eq. 49 instead gives the comoving distance D_C , but is still a good approximation to the luminosity distance for small Ω_k . To save us some clutter, we’ll define the polynomial P (Eq. 50).

$$P(1+z) \equiv \Omega_R(1+z)^4 + \Omega_M(1+z)^3 + \Omega_k(1+z)^2 \quad (50)$$

With this substitution, Equation(1) becomes:

$$\frac{D_L(z)}{D_H} = (1+z) \int_0^z (P(1+z') + \Omega_\Lambda)^{-1/2} dz'$$

VI.1.2. Variable equation of state parameter

We are really interested in the case where the dark energy equation of state parameter can be some function of z . Then the function $E(z)$ would generalize to:

$$E(z) = \sqrt{P(1+z) + \Omega_\Lambda \frac{\rho_\Lambda(z)}{\rho_{\Lambda,0}}}$$

Given an arbitrary function for the equation of state $w_\Lambda(z)$, we can express the ratio $\frac{\rho_\Lambda(z)}{\rho_{\Lambda,0}}$ via Eq. 51

$$\frac{\rho_\Lambda}{\rho_{\Lambda,0}} = \exp \int_0^z \frac{3(w_\Lambda(\frac{1}{1+z}) + 1)}{1+z} dz' \quad (51)$$

At this point we can substitute the generalized function $E(z)$ into Eq. (1) and evaluate numerically.

VI.1.3. Parameterizing the equation of state parameter

Numerical methods are well-suited to constraining a finite, countable set of parameters in a model which predicts observables such as luminosity distance. In light of this fact, we can study the constraints on w_Λ by expressing it as an expansion over some functional basis. If $\varphi_j(a)$ are some basis functions with coefficients θ_j then the expansion takes the form of Eq. (52), where we are expanding around the function $w_\Lambda(a) = -1$.

$$w_\Lambda(a) = -1 + \sum_j \theta_j \varphi_j(a) \quad (52)$$

The basis functions should be defined on the interval

$$[a_{\min}, 1] \rightarrow [0, z_{\max}]$$

In this expansion, Eq. (51) becomes:

$$\frac{\rho_\Lambda}{\rho_{\Lambda,0}} = \exp \int_0^z \frac{3 \sum_j \theta_j \varphi_j \left(\frac{1}{1+z'} \right)}{1+z'} dz'$$

VI.1.4. Obtaining principal components

To obtain the principal components, first we take the partial derivative of the experimental observables (luminosity distances) with respect to the parameters ϕ_j to construct the matrix J :

$$J_{ij} = \frac{\partial D_{Li}}{\partial \phi_j}$$

Then we obtain the covariance matrix Σ for the observables and compute the Fisher matrix $J\Sigma^{-1}J$, as described in the introduction. The eigenvectors of the Fisher matrix are then rank-ordered (by magnitude of eigenvalue, greatest to lowest), and functions corresponding to eigenvectors of the Fisher matrix are identified as principal components. We have replicated a method used by Huterer and Starkman (2003) [16] to obtain the principal components of $w(z)$ using Supernovae data, and the first four of which are shown below.

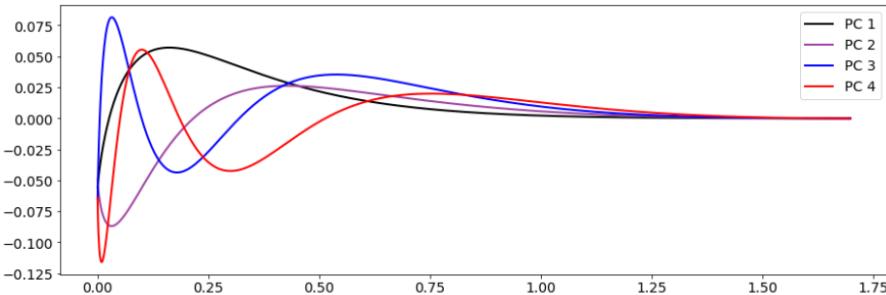


Figure 8. First four (best constrained) principal components of the function $w(z) + 1$.

We can also identify the worst-constrained functions in the principal component basis. These are shown below

These plots may seem pretty uninformative at first, but they carry very practical information. The principal components (PCs) form a basis for the function $w(z)$. The plots reveal that the best constrained basis elements have more variability at low redshift and the worst constrained basis elements have more variability at high redshift. That means we can better constrain the behavior of the equation of state at low redshift using supernovae data.

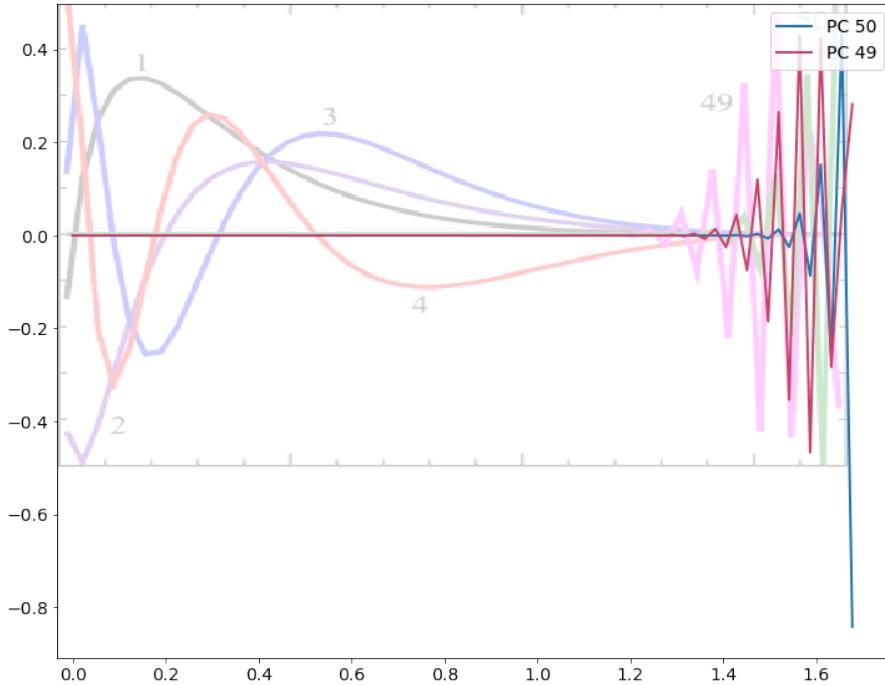


Figure 9. Last two (worst constrained) principal components of the function $w(z) + 1$.

VI.2. PCA with the CMB

Here we have the same basic situation, but there are now two functions being input into the model, and the outputs are the C_l values in the CMB power spectrum. First we identify the fiducial (maximal likelihood) functions $\hat{w}(x)$ and $\hat{c}_s^2(k, x)$. These are obtained by setting all parameters to their fiducial values and then calculating these functions from their definitions (given above). Then we write general variations of these functions:

$$c_s^2(k, x) = \hat{c}_s^2(k, x) + \sum_j^N A_j \psi_j(k, x)$$

$$w(x) = \hat{w}(x) + \sum_j^N B_j \varphi_j(x)$$

Where A and B are N -component vectors storing the expansion coefficients which we will vary, φ_j is a set of basis functions for a one-dimensional domain, and ψ_j is a set of basis functions for a two-dimensional domain. This is necessary because c_s^2 is a function of two variables. Then we define the vector V , which is obtained by listing first the components of

A and then the components of B :

$$V = (A_1 \dots A_N, B_1 \dots B_N)$$

Then, as before, we obtain the partial derivatives $J_{ln} = \partial C_l / \partial V_n$, and then get the Fisher Matrix

$$F = J^T \Sigma^{-1} J$$

where Σ is the covariance matrix for the C_l values. The procedure for obtaining principal components is then exactly the same as before. The principal components are more difficult to interpret in this case however, because they contain information about both functions. They should not be viewed as having physical meaning, but more as tools to tell us the best constraints we can hope to get on w and c_s^2 from experiment, as well as allowing us to define new quantities which can be better constrained.

VII. IMPLEMENTATION AND RESULTS

VII.1. Implementation

The fluid simulation itself was based on a simulation written in Mathematica by collaborator Tristan Smith at Swarthmore. The code, in its original state, computed the power spectrum produced by Seljak’s “two fluid” model. It was modified to simulate neutrinos as a separate fluid, using the equations of the “three fluid” model. The system of ordinary differential equations was solved using the `scipy.integrate` python library. The computations are broken down in the following sections:

VII.1.1. Background Computations

The first step is to simulate the evolution of the background quantities. This involves entering the cosmological parameters as dimensionless quantities and then integrating the Friedmann Equation numerically using `scipy.integrate` in python. It is integrated with respect to the scale factor in order to determine the dimensionless conformal time at recombination, at the present time, and at the time of matter-radiation equality. These calculations

are shown in the following python code block:

```
def eta_itgd(a):
    """Conformal time integrand"""
    return 1/(a**2. * H0 *
               np.sqrt( OmegaM/(a**3) + OmegaRad/(a**4) + OmegaLambda)
               )

# compute conformal time today
(eta_today,_) = integrate.quad(eta_itgd, 0, 1)

# compute conformal time at recombination
(eta_rec,_) = integrate.quad(eta_itgd, 0, 10.**(-3.))

# compute conformal time at matter-radiation equality
(eta_eq,_) = integrate.quad(eta_itgd, 0, aeq)
```

VII.1.2. Computing the mode evolution

The next step is to numerically evolve Equations 41 until the time of recombination for each value of κ in some range. This involves setting the initial conditions for the perturbations, then defining the function which returns the derivatives of all quantities, and using `scipy.integrate.solve_ivp` to evolve the equations. The max step size for the ODE solver was found by trial and error, and eventually chosen to be 0.001. All of that is accomplished with the following code

```
for i, k in enumerate(k_list):
    xi = np.min([10**-4 / k, xeq/10000])
    phi_i = 1
    delta_gamma_i = -2*phi_i*(1 + 3*y(xi)/16)
    delta_c_i = .75 * delta_gamma_i
    v_gamma_i = -k/eta(xi) * (
        delta_gamma_i/4 + (2*k**2 * (1 + y(xi))*phi_i)/
```

```

(9*eta(xi)**2 * (4./3. +y(xi))))
a0 = a(x_list[0])
vc_i = v_gamma_i

#neutrino initial conditions
v_nu_i = v_gamma_i
delta_nu_i = delta_gamma_i
#v_nu_i = v_gamma_i
#delta_nu_i = 0
v_0[i] = (((4/3)*(v_gamma_i+v_nu_i)*OmegaRad)*a0**-4. +((vc_i+v_gamma_i)*OmegaRad)*a0**-4 + (OmegaM*a0**-3)+ OmegaLambda)

# solve the ODE
def d_func(x, Y):
    """return function derivatives"""
    delta_c = Y[0]
    vc = Y[1]
    delta_gamma = Y[2]
    v_gamma = Y[3]
    delta_nu = Y[4]
    v_nu = Y[5]
    phi = Y[6]

    DY = np.zeros(7)
    # potential derivative
    DY[6] = (-eta(x) * phi + (3*eta(x)**2*
        ((v_gamma)*(4./3. +y(x)-yc(x))+ynu*(v_nu-v_gamma)+yc(x)*vc))/(
        (2*(1+y(x))*k)))
    # dark matter derivatives
    DY[0] = -k * vc + 3*DY[6]
    DY[1] = -eta(x)*vc+k*phi

```

```

#radiation derivatives

DY[2] = -4./3. * k *v_gamma + 4*DY[6]
DY[3] = (-eta(x)*yb(x)*v_gamma + k*delta_gamma/3)/(
    4./3. + yb(x)) + k*phi

#neutrino derivatives

DY[4] = -4./3. * k *v_nu + 4*DY[6]
DY[5] = k*delta_nu/4 + k * phi
return DY

x_span = [xi, xrec]
Y0 = [delta_c_i, v_gamma_i, delta_gamma_i, v_gamma_i, delta_nu_i, v_nu_i, phi_i]
out = integrate.solve_ivp(d_func, x_span, Y0, max_step=.001)

```

Once the mode evolution is done, the source terms can be evaluated for each wavenumber at the time of recombination.

VII.1.3. Precomputing bessel functions and derivatives

Scipy has an implementation of the spherical bessel functions (scipy.special.spherical_jn) which is pretty quick. However there is an optimization possible due to the fact that the bessel function and its first derivative must be evaluated repeatedly for the same l values. That is simply to precompute the bessel functions values for each combination of k and ℓ , and then use a forward-difference calculation to estimate the first derivative. The following code precomputes these arrays:

```

# precompute all bessels and derivatives:
J_lk = np.zeros((len(ell_list),len(k_list)))
DJ_lk = np.zeros((len(ell_list),len(k_list)))
k_scaled = k_list*(eta_today-eta_star)/tau_r
for i in range(len(k_list)):
    J_lk[:,i] = spherical_jn(ell_list,k_scaled[i])

```

```
DJ_lk[:,i] = .5*(spherical_jn(ell_list + 1, k_scaled[i])\
- spherical_jn(ell_list - 1, k_scaled[i]))
```

VII.1.4. Computing the power spectrum

Finally the values C_ℓ can be calculated. This involves integrating Eqs. (43) and (42). Because we are integrating over an array of fixed k values, an efficient implementation is the scipy cumulative trapezoidal rule (scipy.cumtrapz). The power spectrum values are obtained by the following code block:

```
Cl_intgd =(np.exp(-2*k_list**2. * xs**2)*
((SW_terp(k_list) + ISW_terp(k_list))*  

 J_lk + DOP_terp(k_list)*  

(-(J_lk/(2*k_scaled))+  

 DJ_lk))**2)  

Cl_list = integrate.cumtrapz(Cl_intgd/k_list, axis=1)[:, -1]
```

The values then must be interpolated and properly scaled.

VII.1.5. GDM parameters

Now we calculate the effective single-fluid parameters of the GDM formalism. This is done simply using the definitions of w_T and c_{sT}^2 in Eqs. (46) and (48). It is accomplished by the following code:

```
#Get delta_T:  

delta_T =(a_list**(-3)*(OmegaB*delta_gamma + OmegaCDM*delta_c) +  

→a_list**(-4) * (OmegaGamma*delta_gamma + OmegaNu*delta_nu))\  

/(OmegaLambda + a_list**(-3) * OmegaM + a_list**(-4) * OmegaRad)  

#get w_eff:  

w_eff =((1./3.)*a_list**(-4) * (OmegaGamma + OmegaNu) - OmegaLambda)\  

/(OmegaLambda + a_list**(-3) * OmegaM + a_list**(-4) * OmegaRad)
```

```
#get cs2_eff:
cs2_eff =((1./3.)*a_list**(-4) * (OmegaGamma*delta_gamma +
→OmegaNu*delta_nu))\
/(a_list**(-3)*(OmegaB*delta_gamma + OmegaCDM*delta_c) + a_list**(-4) *
→(OmegaGamma*delta_gamma + OmegaNu*delta_nu))
```

VII.2. Results

Figures (10), (11), and (12) show the time dependence of the density perturbations δ_c , δ_γ , and δ_ν respectively, for different values of κ . We see that inhomogeneities in the cold dark matter density field essentially grow linearly, while perturbations of the coupled photon/baryon fluid and the neutrino fluid oscillate.

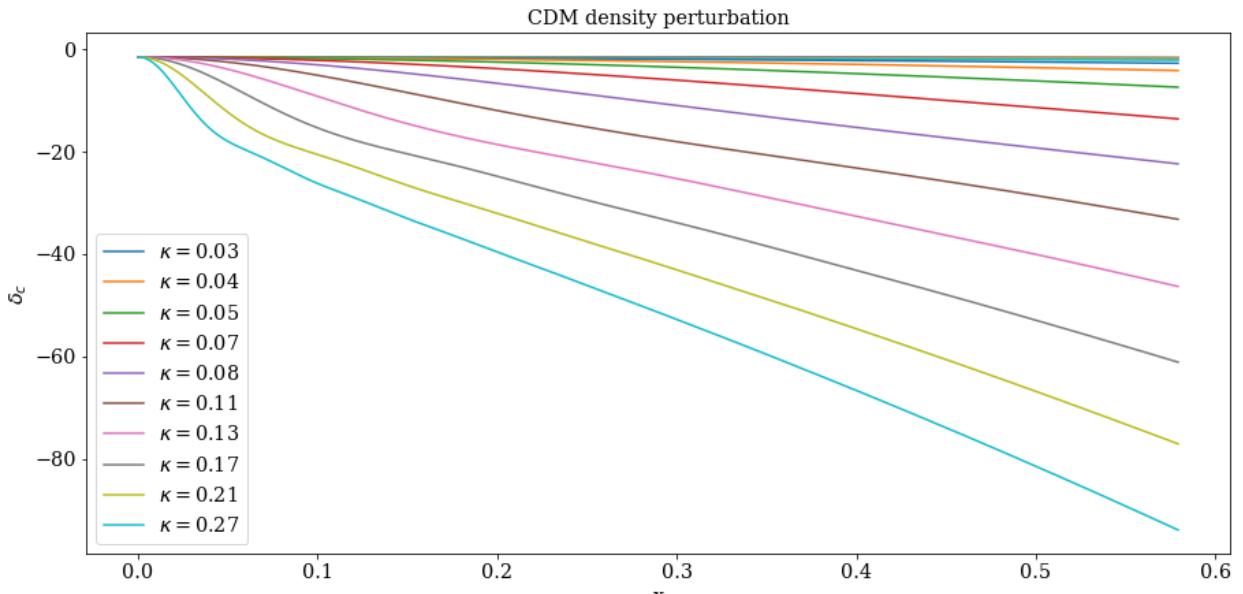


Figure 10. Time evolution of the density perturbation for cold dark matter.

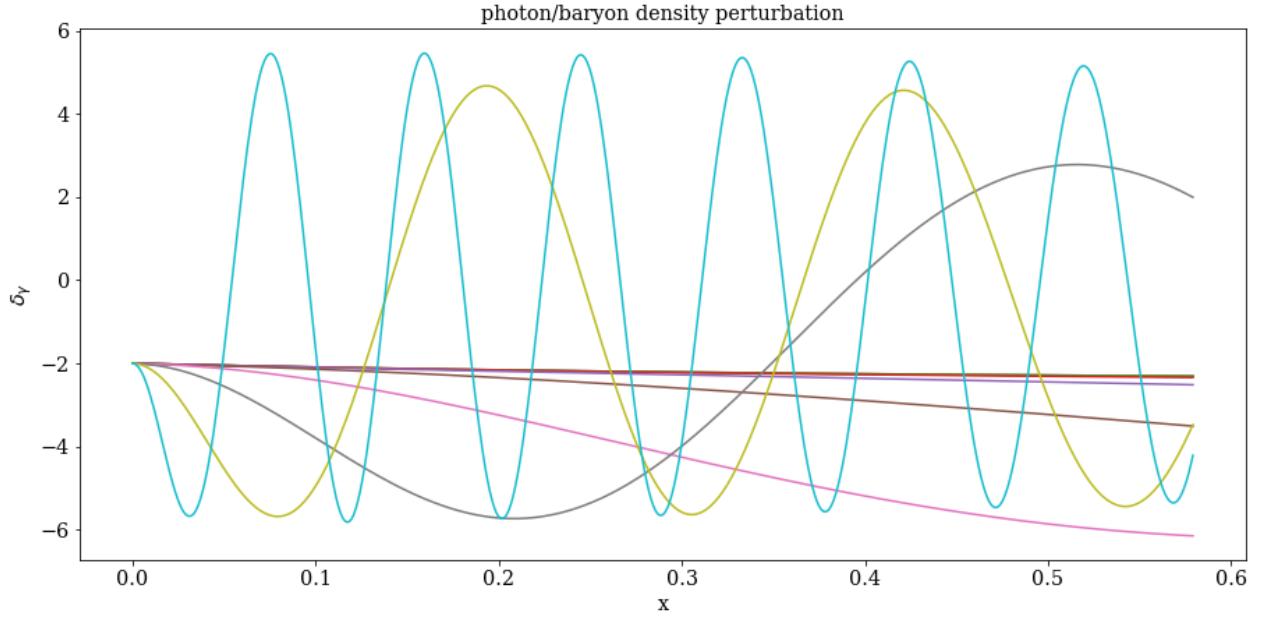


Figure 11. Time evolution of the density perturbation for the combined photon/baryonic matter fluid.

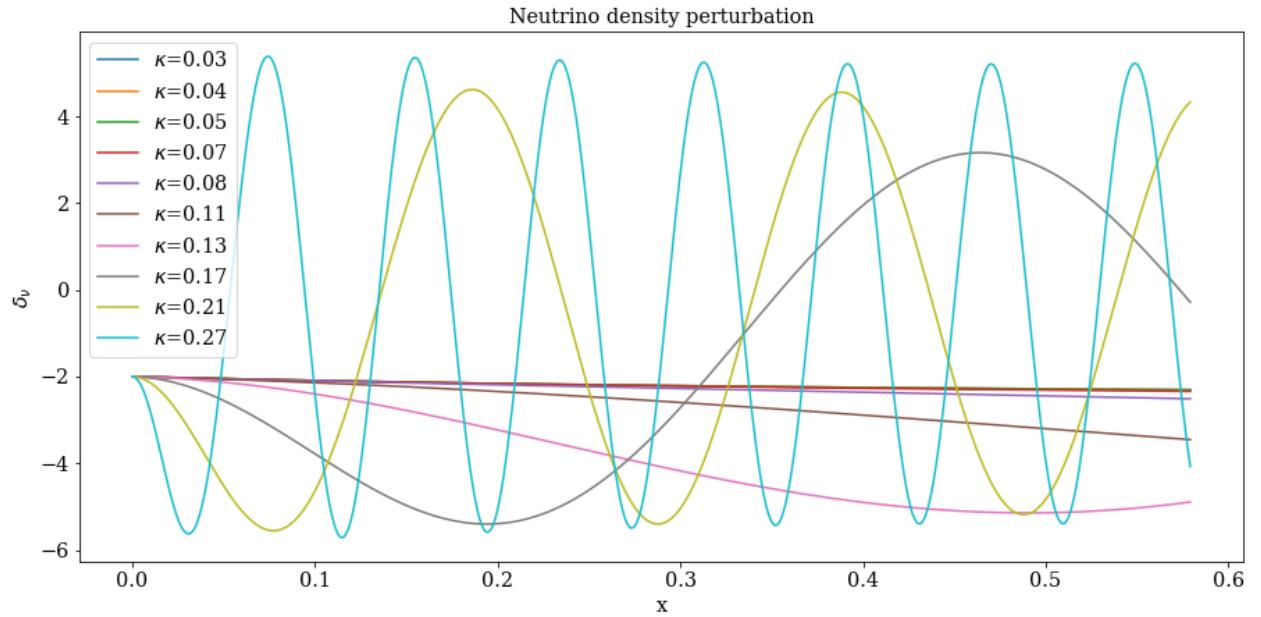


Figure 12. Time evolution of the density perturbation for neutrinos.

VII.3. Source Contributions

Figure 13 is a graph from Seljak's paper [13] showing the quantities v_γ and $\phi + \frac{\delta_\gamma}{4}$ at the time of recombination, as a function of κ . We produced a similar figure (Figure 14). For

each value of κ , the equations are evolved until recombination, at which point the relevant values are recorded. The qualitative behavior is in agreement with Seljak's, although the actual curves are slightly different.

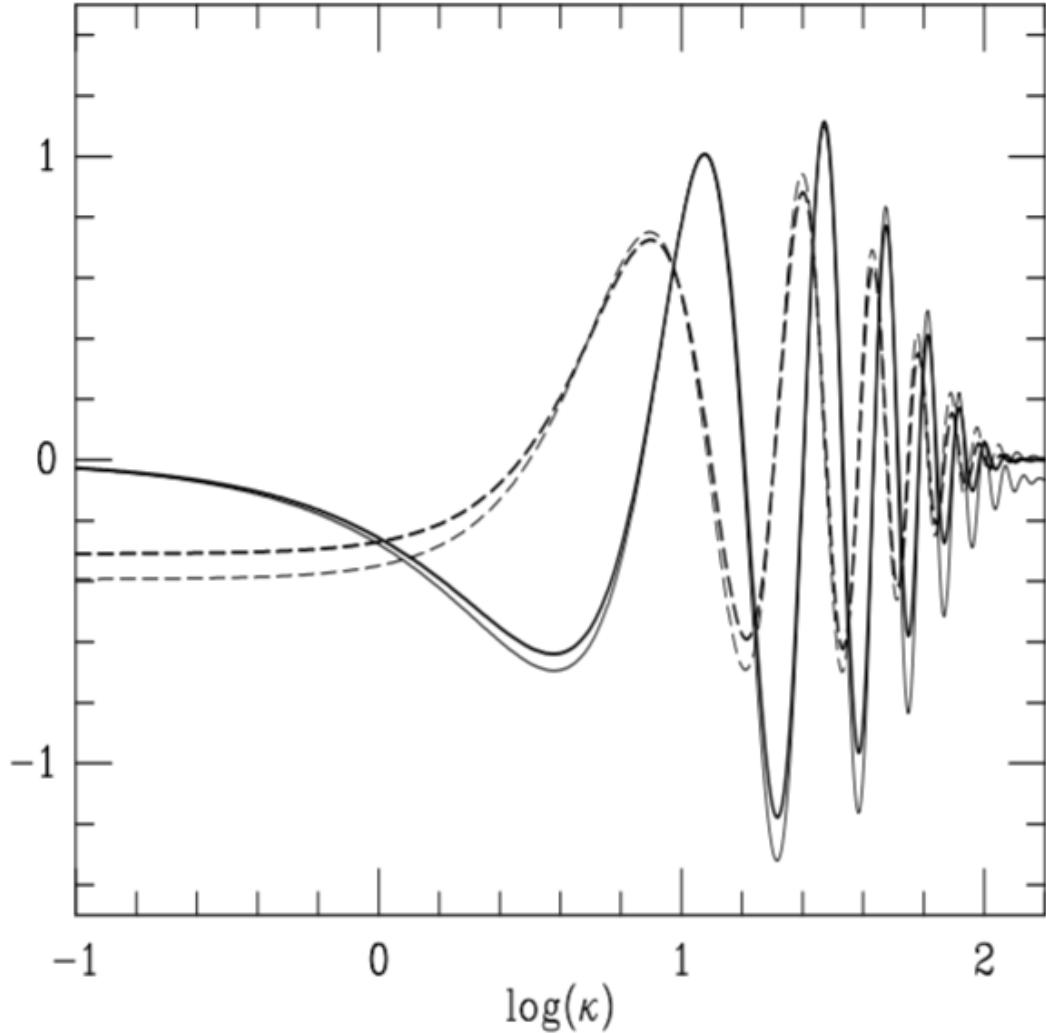


Figure 13. Source terms v_γ (dashed line) and $\phi + \frac{\delta_\gamma}{4}$ (solid line) as a function κ (log scale), from Seljak's paper [13]

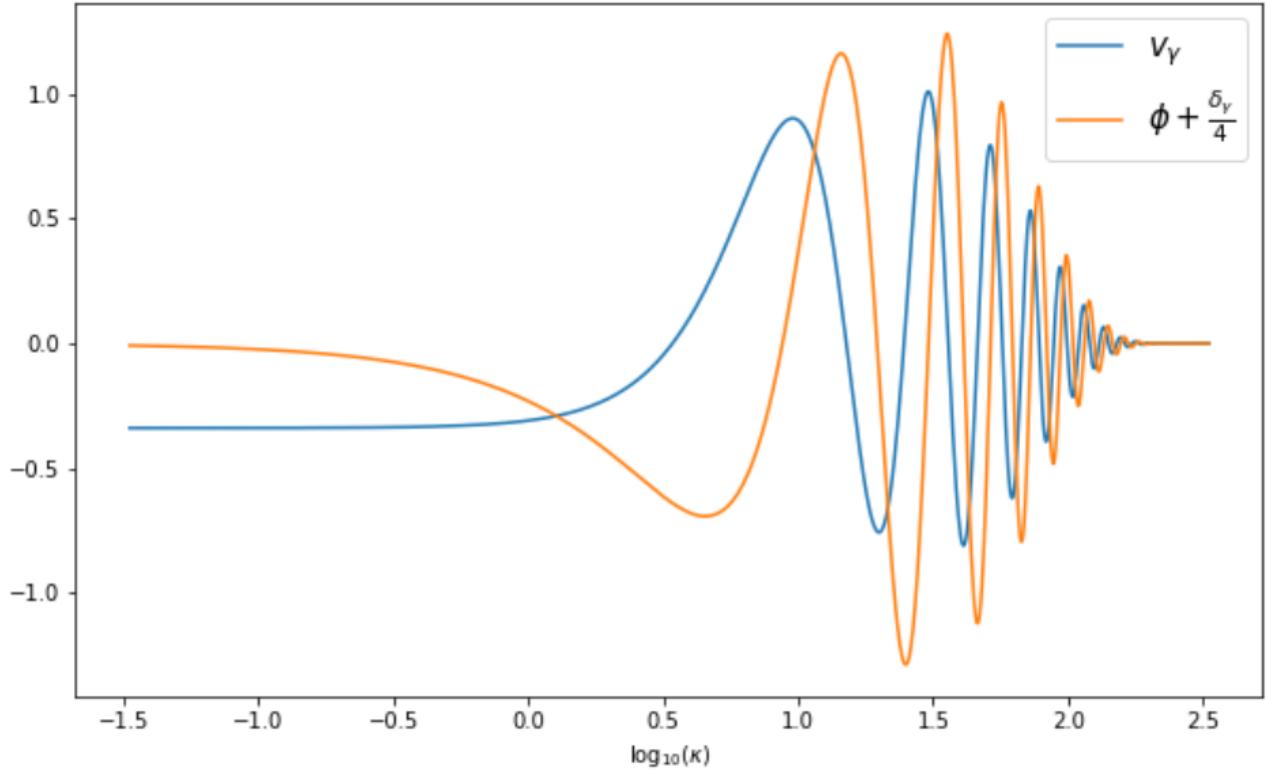


Figure 14. Source terms v_γ and $\phi + \frac{\delta_\gamma}{4}$ as a function κ (log scale), from our python simulation

VII.4. The power spectrum

Here we show the actual power spectra generated using the source contributions shown above. Our fiducial power spectrum (Figure 16) roughly agrees with experiment (See e.g. figure 6).

We also experimented with changing the neutrino density parameter while adjusting the photon density parameter in order to keep the total radiation density parameter constant. Increasing the neutrino density (and therefore decreasing photon density) had the effect of suppressing the overall intensity and shifting the peaks to the right.

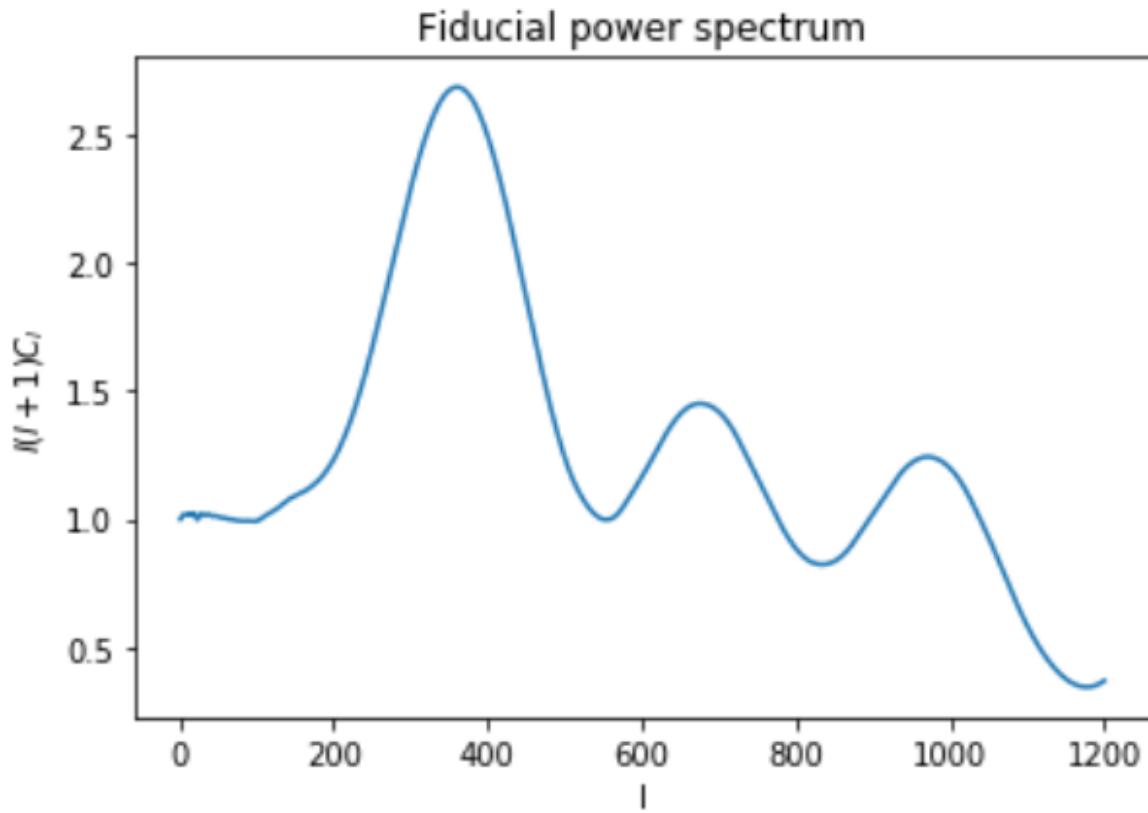


Figure 15. Fiducial power spectrum calculated with maximum likelihood parameters.

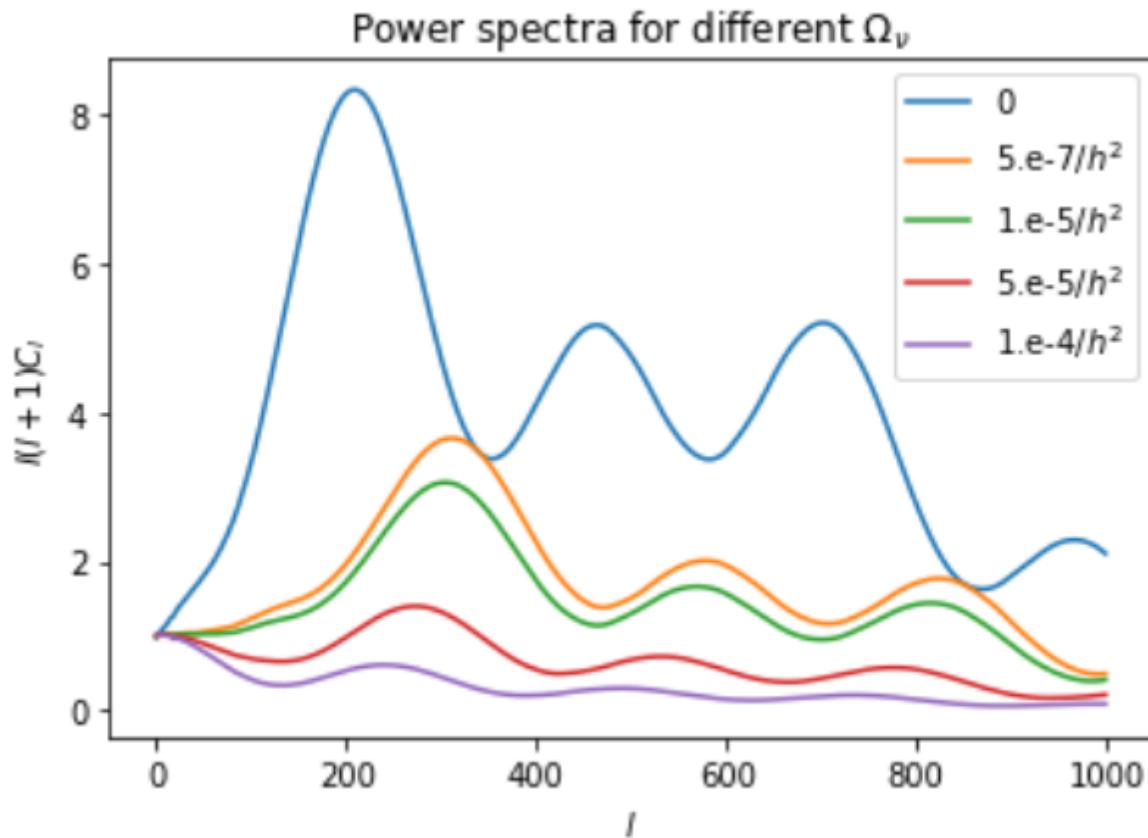


Figure 16. Power spectra calculated with different values of Ω_ν while Ω_{rad} is held constant.

VII.5. GDM effective fluid quantities

Now we deal with the problem of extracting the GDM functions $w_T(x)$ and $c_{sT}^2(x)$. For the equation of state, we can actually obtain $w_T(x)$ analytically without running the simulation, because it only depends on the background (homogeneous) fluid quantities and not on perturbations.

$$w_T = \frac{\frac{1}{3}(\Omega_\gamma + \Omega_\nu)a^{-4} - \Omega_\Lambda}{\Omega_\Lambda + (\Omega_b + \Omega_c)a^{-3} + (\Omega_\gamma + \Omega_\nu)a^{-4}}$$

Similarly and analytic expression can be derived for the derivative of w_T with respect to x . Both w_T and $\frac{\partial w_T}{\partial x}$ are shown in the graph below.

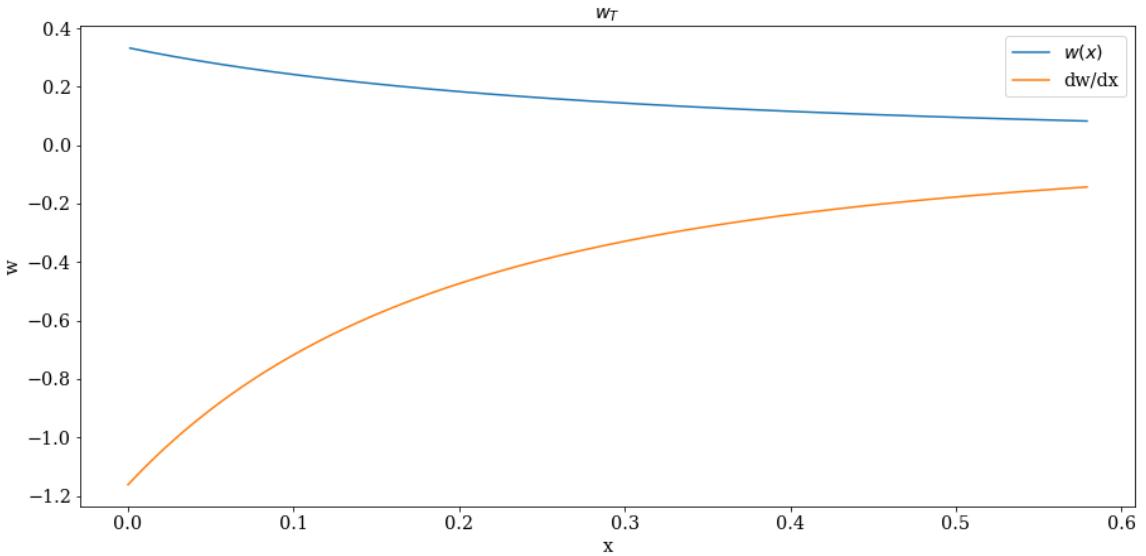


Figure 17. w_T and its derivative

This is what we should expect. At early times radiation dominates so the effective equation of state is very near to $1/3$, which is the equation of state for photons. At later times as matter begins to dominate the effective equation of state decreases towards zero, which is the equation of state for baryonic matter.

Unfortunately, c_{sT}^2 poses more of a challenge. First off, it depends on the perturbations and not just the background quantities, and so we have to actually run the model and compute it from the output.

$$c_{sT}^2 = \frac{\sum \delta_i \rho_i w_i}{\sum \delta_i \rho_i}$$

There is another issue with the above expression, which is that the oscillating density perturbations are frequently equal to zero. In principle, the numerator and denominator should both go to zero, possibly leading to good behavior near the points where the function is undefined. In practice, however, because of machine imprecision the function blows up near these points. See figure 18 for plots of c_{sT} extracted from the simulation for different values of κ :

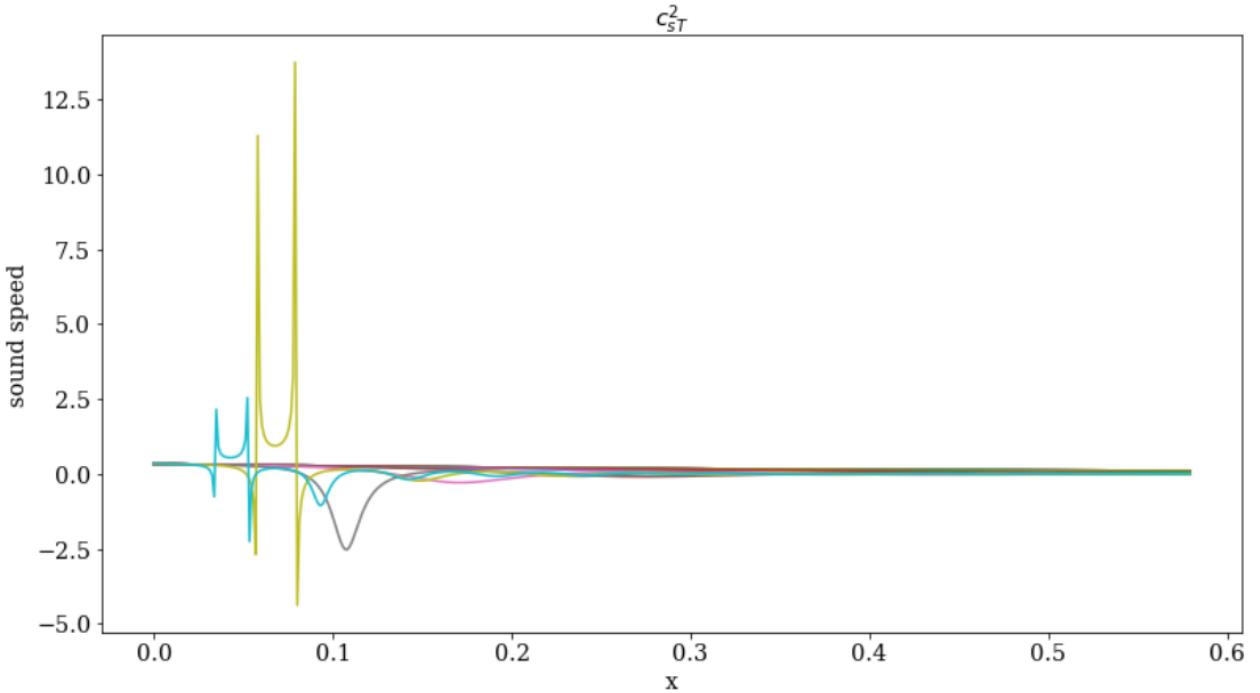


Figure 18. sound speed computed from fluid simulation, plotted for different values of κ .

The sound speed behaves relatively well for low wavenumbers. Figure VII.5 shows the evolution for low wavenumber $\kappa < 200$.

Here too, we see very reasonable behavior: the sound speed matches the equation of state closely at low wavenumbers where the perturbations oscillate slowly or not at all. For the wavenumbers with higher oscillation the density perturbations go to zero more often so the sound speed is more sensitive to tiny fluctuations, and so in this regime it does not match the equation of state as closely.

After computing the equation of state and sound speed for the aggregated fluids, we plug

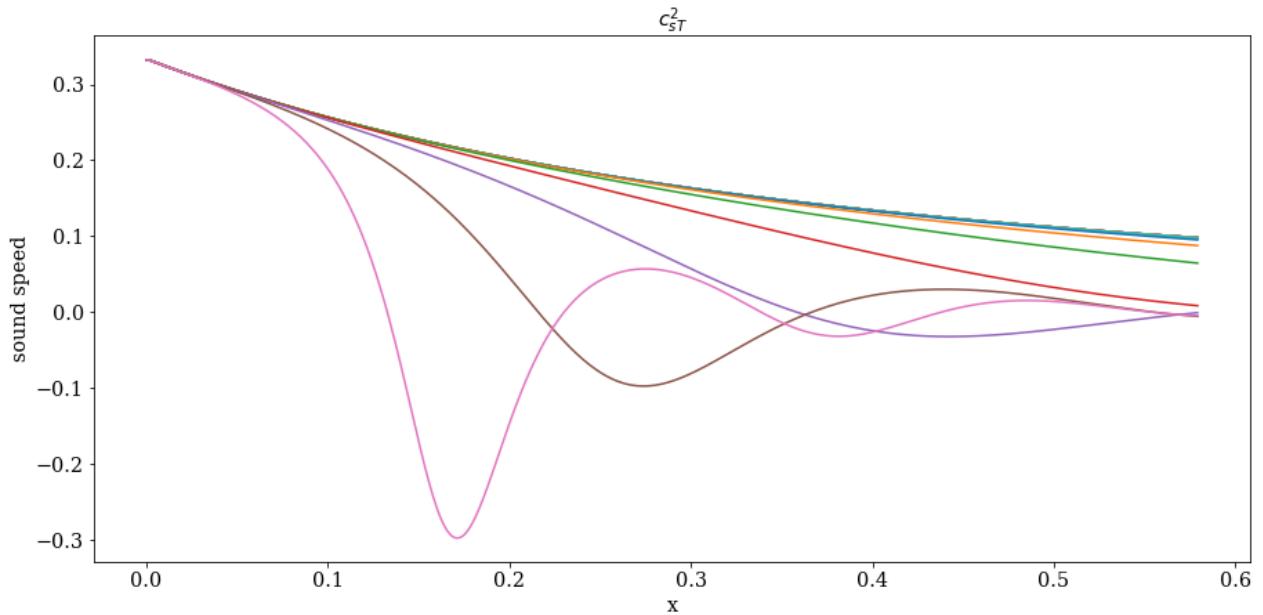


Figure 19. sound speed computed from fluid simulation, plotted for different values of κ with $\kappa < 200$.

these in to the time-dependent single-fluid model. Figure 20 shows the total density perturbation computed for the three-fluid model, and figure 21 shows the density perturbation for the one-fluid model with variable w and c_{sT}^2 . We can see that they are basically in agreement qualitatively, except the single fluid model blows up to very large density perturbations, to the point where the linear theory is probably no longer valid. It is currently unclear why this happens. Unfortunately, the power spectrum also blows up, which probably has to do with the instability of the c_{sT}^2 computation.

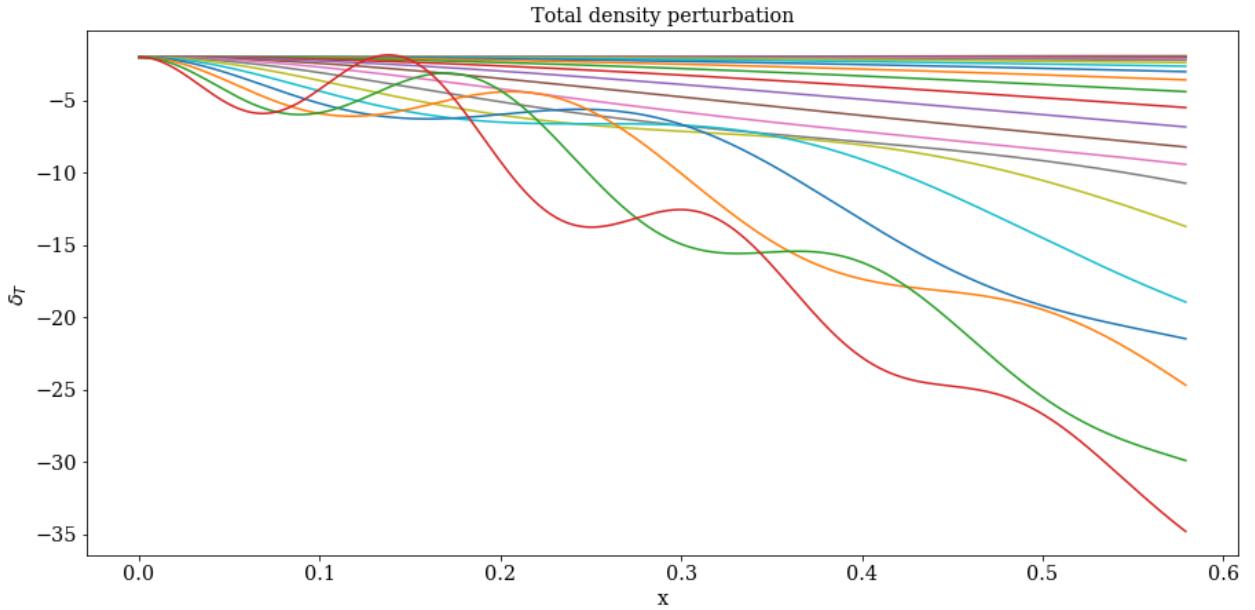


Figure 20. Total density perturbation δ_T for three-fluid system, plotted for different values of κ .

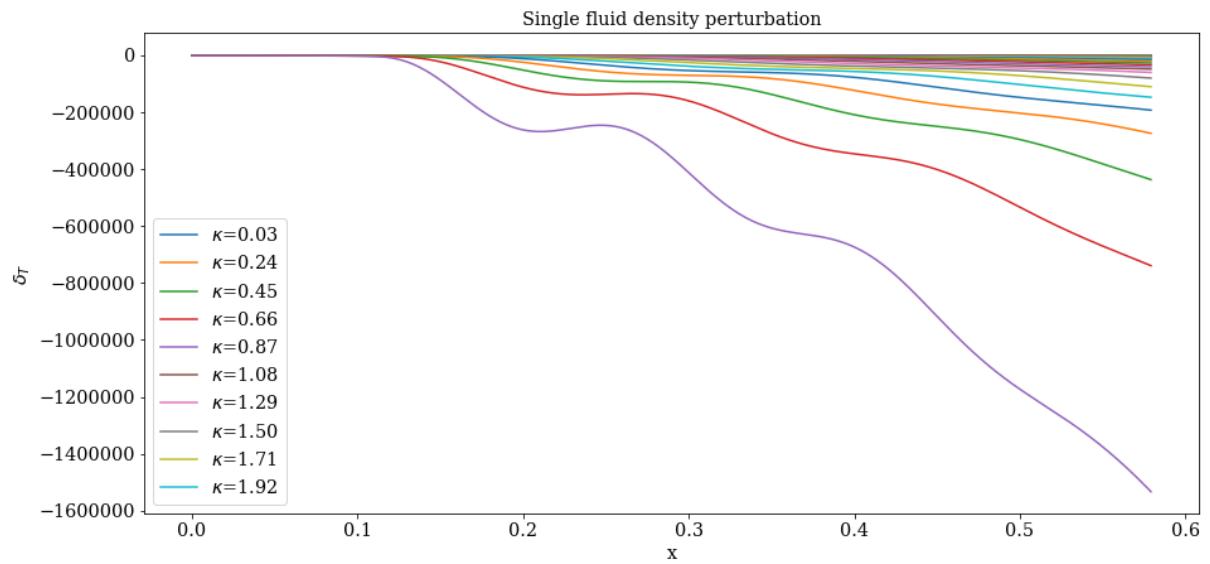


Figure 21. density perturbation for single-fluid, variable w and c_{sT}^2 model, plotted for different values of κ .

At this point, the output of our power spectrum generator is too unstable and the computation time too long for us to obtain principal components for w and c_{sT}^2 from the power spectrum.

VIII. CONCLUSION

VIII.1. Methodology

The results above support the conclusion that the main features of the CMB power spectrum can be produced from a model based on a system of perfect fluids obeying the relativistic Euler Equations. The results also support the conclusion that a single fluid model with variable equation of state and sound speed can capture the qualitative behavior of the dark sector. We also found that neutrinos could be incorporated separately from photons and still produce a well-behaved power spectrum. Varying the relative level of neutrinos while keeping the overall radiation density fixed had an observable effect on the power spectrum, supporting the conclusion that the coupling to baryonic matter has an important effect on the photon perturbation evolution, in contrast to the neutrinos which have no coupling to matter. On the other hand, we found that the effective sound speed computed from the multiple-fluid model was unstable due to zero-crossings of the density perturbation. We also found that the single-fluid model with variable equation of state and sound speed failed to produce a well-behaved power spectrum. This seems to be partly due to the zero-crossing problem and partly due to an algorithmic error, but it is also possible that the single fluid model is not capable of capturing the true behavior of the universe. We did not obtain any direct evidence for or against the conclusion that principal component analysis will be successful on this system. However, comparison with other similar analysis (e.g. [16]) imply that evaluation time will need to be sped up considerably in order to profitably use PCA on the model.

VIII.2. Future work

The first objective of future work should be producing an actual power spectrum using the single fluid model with variable equation of state and sound speed. It may be necessary to first resolve the issue with zero-crossings of the density perturbation causing blowup of the sound speed. One possibility is simply to use the pressure as a time-dependent *GDM* parameter instead of the sound speed. Then we would not have an issue because the density would not enter the GDM parameters, and in particular would not enter the denominator of anything. We need to demonstrate convergence of this power spectrum when the differential

equations are solved with smaller step sizes and when the GDM parameters are sampled more densely through interpolation. Then this power spectrum must be compared to the full Λ CDM model result, and both of these should be compared to the results of a code like CLASS.

If this is successful, then the next step is algorithmic optimization. There is a lot of room for improvement here, as CLASS generates more detailed and more accurate power spectrums in a fraction of the time that our code does. In fact, the best course of action might be to adapt CLASS to our own purposes by modifying the code where necessary. Assuming that the code can be optimized to the point where PCA is feasible, we should first attempt to do functional PCA using localized basis functions (e.g. wavelets, box functions, triangles, etc.) because these seem to be quicker to converge than modal basis functions and therefore will give us a better picture of the principal components with fewer evalutatons of the code. Once success has been demonstrated with local basis functions, we should move on to other orthogonal function families such as Legendre polyonomials, etc.

Once principal components of w and c_s (or p) have been obtained, we can compare these best-constrained components to the fiducial functions which are obtaiend for different particle models of the dark sector (i.e. for axions, WIMPS, etc.) This will hopefully allow us to make statements about whether CMB measurements could help rule out (or support) some particle models of dark matter.

- [1] “Content of the Universe - WMAP 9yr Pie Chart,” .
- [2] V. Trimble, **25**, 425 (1987).
- [3] V. C. Rubin, W. K. Ford, and N. Thonnard, *The Astrophysical Journal* **238**, 471 (1980).
- [4] M. D. Leo, “English: Rotation curve of spiral galaxy Messier 33 (yellow and blue points with error bars), and a predicted one from distribution of the visible matter (gray line). The discrepancy between the two curves can be accounted for by adding a dark matter halo surrounding the galaxy.” (2018).
- [5] B. Ryden, *Introduction to Cosmology* (Cambridge University Press, 2016) google-Books-ID: 07WSDQAAQBAJ.
- [6] M. Bartelmann, *Cosmology* (University of Heidelberg).

- [7] D. Baumann, “Cosmology,” .
- [8] H. K. Eriksen, “An introduction to the CMB power spectrum,” .
- [9] T. Smith and D. Grin, *NASA ATP Proposal* , 17 (2017).
- [10] P. Collaboration and N. e. a. Aghanim, arXiv:1807.06209 [astro-ph] (2018), arXiv: 1807.06209.
- [11] S. M. Carroll, arXiv:gr-qc/9712019 (1997), arXiv: gr-qc/9712019.
- [12] Drbogdan and Yinweichen, “Vectorised version of File:HistoryOfUniverse-BICEP2-20140317.png,” (2014).
- [13] U. Seljak, *The Astrophysical Journal* **435**, L87 (1994), arXiv: astro-ph/9406050.
- [14] M. Bamse, “Contravariant components of the Stress-Energy Tensor.” (2013).
- [15] I. Jolliffe, in *International encyclopedia of statistical science* (Springer, 2011) pp. 1094–1096.
- [16] D. Huterer and G. Starkman, *Physical Review Letters* **90** (2003), 10.1103/PhysRevLett.90.031301, arXiv: astro-ph/0207517.
- [17] Nicoguaro, “GaussianScatterPCA,” (2016).
- [18] C.-P. Ma and E. Bertschinger, *The Astrophysical Journal* **455**, 7 (1995), arXiv: astro-ph/9506072.
- [19] H. Kodama and M. Sasaki, *Progress of Theoretical Physics Supplement* **78**, 1 (1984).
- [20] W. Hu, *The Astrophysical Journal* **506**, 485 (1998), arXiv: astro-ph/9801234.
- [21] D. W. Hogg, arXiv:astro-ph/9905116 (1999), arXiv: astro-ph/9905116.

IX. APPENDIX

IX.1. Mathematica Code

Here is the Mathematica code that was provided by Tristan Smith:

(* Define all of background quantities *)

h0 = 0.67;

H0 = 100 * h0;

Ωγ = 2.4710⁻⁵/h0²;

Ων = 1.6810⁻⁵/h0²;

Ωrad = Ωγ + Ων;

Ωcdm = 0.1201 /h0² ;

```

 $\Omega b = 0.0223 / h^2 ;$ 
 $\Omega M = \Omega_{\text{cdm}} + \Omega b ;$ 
 $\Omega \Lambda = 1 - (\Omega_{\text{cdm}} + \Omega b + \Omega \gamma) ;$ 
 $\text{etatoday} = \text{NIntegrate} \left[ \frac{1}{a^2 * H_0 * \sqrt{\frac{\Omega M}{a^3} + \frac{\Omega_{\text{rad}}}{a^4} + \Omega \Lambda}}, \{a, 0, 1\} \right] ;$ 
 $\text{etarec} = \text{NIntegrate} \left[ \frac{1}{a^2 * H_0 * \sqrt{\frac{\Omega M}{a^3} + \frac{\Omega_{\text{rad}}}{a^4} + \Omega \Lambda}}, \{a, 0, 10^{-3}\} \right] ;$ 
 $a_{\text{eq}} = \Omega \gamma / \Omega M ;$ 
 $\text{etaEQ} = \text{NIntegrate} \left[ \frac{1}{a^2 * H_0 * \sqrt{\frac{\Omega M}{a^3} + \frac{\Omega_{\text{rad}}}{a^4} + \Omega \Lambda}}, \{a, 0, a_{\text{eq}}\} \right] ;$ 
 $\tau r = \frac{1}{(\Omega M / a_{\text{rec}})^{1/2} H_0 / 2} ;$ 
 $z_{\text{eq}} = \frac{1}{a_{\text{eq}}} - 1 ;$ 
 $z_{\text{rec}}[\Omega b] := 1000 \Omega b^{-0.027/(1+0.11 \text{Log}[\Omega b])}$ 
 $a_{\text{rec}} = \frac{1}{1+z_{\text{rec}}[\Omega b]} ;$ 
 $\alpha = \sqrt{\frac{a_{\text{rec}}}{a_{\text{eq}}}} ;$ 
 $x_{\text{rec}} = ((\alpha^2 + 1)^{1/2} - 1) / \alpha ;$ 
 $x_{\text{eq}} = \text{etaEQ} / \tau r ;$ 
 $y[x] := (\alpha x)^2 + 2\alpha x$ 
 $a[x] := a_{\text{eq}} y[x]$ 
 $y_b[x] := 1.68 y[x] \Omega b / \Omega M$ 
 $y_c[x] := y[x] \Omega_{\text{cdm}} / \Omega M$ 
 $\eta[x] := \frac{2\alpha(\alpha x + 1)}{\alpha^2 x^2 + 2\alpha x}$ 
 $\Delta \phi[x] := (2 - 8/y[x_{\text{rec}}] + 16 x_{\text{rec}} / y[x_{\text{rec}}]^3) / (10 y[x_{\text{rec}}])$ 
 $x_s = 0.6 \Omega M^{1/4} \Omega b^{-1/2} a_{\text{rec}}^{3/4} h_0^{-1/2} ;$ 

Clear[SWTable, ISWTable, DOPTable, it]

SWTable = Table[0, {i, 0, 4, 4/1000}];

ISWTable = Table[0, {i, 0, 4, 4/1000}];

DOPTable = Table[0, {i, 0, 4, 4/1000}];

it = 1;

Do[Clear[q, κ, δγ, δc, vγ, vc, φ];

κ = τr 10^i / etatoday;

```

```

xi = Min [10-4/κ,  $\frac{x_{\text{eq}}}{10000}$ ] ;
φi = 1;
δγi = -2φi  $\left(1 + \frac{3y[\text{xi}]}{16}\right)$  ;
δci =  $\frac{3}{4}$  δγi;
vγi =  $\frac{-\kappa}{\eta[\text{xi}]} + \frac{2\kappa^2(1+y[\text{xi}])\phi_i}{9\eta[\text{xi}]^2(4/3+y[\text{xi}])}$  ;
q = NDSolve [{ {δc'[x] == -κvc[x] + 3φ'[x], vc'[x] == -η[x]vc[x] + κφ[x], δγ'[x] ==  $\frac{-4}{3}\kappa v\gamma[x] + 4\phi'[x]$ , vγ'[x] ==  $\frac{-\eta[x]yb[x]\gamma[x]+\kappa\delta\gamma[x]/3}{4/3+yb[x]} + \kappa\phi[x]$ , φ'[x] ==  $-\eta[x]\phi[x] + \frac{3\eta[x]^2(\gamma[x](\frac{4}{3}+y[x]-yc[x])+yc[x]vc[x])}{2(1+y[x])\kappa}$ , δγ[xi] == δγi, vγ[xi] == vγi, vc[xi] == vγi, φ[xi] == φi, δc[xi] == δci}, {δγ, δc, vγ, vc, φ}, {x, xi, xrec}]};
{δγ, δc, vγ, vc, φ} = Flatten[{δγ, δc, vγ, vc, φ}/.q];
SWTable[[it]] = {κ,  $\left(\phi[\text{xrec}] + \frac{\delta\gamma[\text{xrec}]}{4}\right)$ } ;
ISWTable[[it]] = {κ, 2Δφ[xrec]};
DOPTable[[it]] = {κ, vγ[xrec]};
it = it + 1,
{i, 0, 4, 4/1000}]

SWInterp = Interpolation[SWTable];
ISWInterp = Interpolation[ISWTable];
DOPInterp = Interpolation[DOPTable];

(*CompareswelltoFig.1ainSeljak*)

LogLinearPlot [{-SWInterp[κ]Exp[-κ2xs2], -ISWInterp[κ]Exp[-κ2xs2], -DOPInterp[κ]Exp[-κ2xs2]}, {κ, τr/etatoday,  $\frac{\tau r 10^4}{\text{etatoday}}$ }, PlotRange → All]

(*Generatetheκlistnecessarytodotheκintegral;
alsochoosealistoffellvaluesatwhichtoevaluatethepowerspectrum*)

κList = Table [κ, {κ, τr/etatoday,  $\frac{\tau r 10^4}{\text{etatoday}}$ ,  $\frac{\tau r 2\pi}{10\text{etatoday}}$ }];
ellList = Union[{2, 3, 4, 5, 6, 7, 8, 10, 12, 15, 20}, Table[ell, {ell, 30, 100, 10}], Table[ell, {ell, 120, 500, 20}], Table[ell, {ell, 550, 1200, 50}]];
CLList = Table[0, {i, 1, Length[ellList]}];
IntList =

```

```

Table [Exp [-2κList[[i]]2xs2]

((SWInterp[κList[[i]]] + ISWInterp[κList[[i]]])SphericalBesselJ [ellList[[j]], κList[[i]] ( $\frac{\text{etatoday}-\text{etastar}}{\tau r}$ )] +
DOPInterp[κList[[i]]]  $\left( -\frac{\text{SphericalBesselJ}[\text{ellList}[[j]], \kappa \text{List}[[i]] (\frac{\text{etatoday}-\text{etastar}}{\tau r})]}{2 \kappa \text{List}[[i]] (\frac{\text{etatoday}-\text{etastar}}{\tau r})} \right) +$ 
 $\frac{1}{2} (\text{SphericalBesselJ} [\text{ellList}[[j]] - 1, \kappa \text{List}[[i]] (\frac{\text{etatoday}-\text{etastar}}{\tau r})] -$ 
 $\text{SphericalBesselJ} [\text{ellList}[[j]] + 1, \kappa \text{List}[[i]] (\frac{\text{etatoday}-\text{etastar}}{\tau r})]))^2, \{i, 1, \text{Length}[\kappa \text{List}]\},$ 
{j, 1, Length[ellList]};

(*Do the numerical integral using the trapezoidal rule – much faster than a pre – fabed routine*)

Do[
CLList[[j]] = {ellList[[j]], Sum [ $\frac{1}{2} \left( \frac{\text{IntList}[[i,j]]}{\kappa \text{List}[[i]]} + \frac{\text{IntList}[[i+1,j]]}{\kappa \text{List}[[i+1]]} \right) (\kappa \text{List}[[i+1]] - \kappa \text{List}[[i]])$ ,
{i, 1, Length[κList] - 1}]};
```

, {j, 1, Length[ellList]}]

CIInterp = Interpolation[CLList];

Plot [ell(ell + 1) $\frac{\text{CIInterp}[\text{ell}]}{2(2+1)\text{CIInterp}[2]}$, {ell, 2, 1200}]

IX.2. Python Code

IX.2.1. mode_evolution.ipynb

```
# coding: utf-8

# In[54]:


from scipy import integrate
from scipy import interpolate
from scipy.special import spherical_jn
import numpy as np
import matplotlib.pyplot as plt
from numpy import genfromtxt
```

```
# In[55]:
```

```
h0 = 0.67
H0 = 100.*h0
OmegaGamma = 2.47e-5/(h0**2.)
OmegaNu = 1.68e-5/(h0**2.)
OmegaRad = OmegaGamma + OmegaNu
OmegaCDM = 0.1201/(h0**2.)
OmegaB = .03
OmegaM = OmegaCDM + OmegaB
OmegaLambda = 1-(OmegaCDM + OmegaB + OmegaGamma)
aeq = OmegaRad/OmegaM
```

```
# In[56]:
```

```
def eta_itgd(a):
    """Conformal time integrand"""
    return 1/(a**2. * H0 *
              np.sqrt( OmegaM/(a**3) + OmegaRad/(a**4) + OmegaLambda))
# compute conformal time today
(eta_today,_) = integrate.quad(eta_itgd, 0, 1)
# compute conformal time at recombination
(eta_rec,_) = integrate.quad(eta_itgd, 0, 10.**(-3.))
```

```
# compute eta_eq
(eta_eq,_) = integrate.quad(eta_itgd, 0, aeq)
```

In[57]:

```
# where does the expression for zrec come from?
zrec = lambda OmB: 1000*OmB**(-0.027/(1 + 0.11 * np.log(OmB)))
arec = 1/(1 + zrec(OmegaB))
tau_r = 1/(np.sqrt(OmegaM/arec)*H0/2)
zeq = 1/aeq - 1
alpha = np.sqrt(arec/aeq)
xrec = (np.sqrt((alpha**2 + 1)) - 1)/alpha
xeq = eta_eq/tau_r
```

In[58]:

```
# x is time coordinate
# y is scale factor relative to equality
def y(x): return (alpha*x)**2 + 2*alpha*x
def a(x): return aeq*y(x)
def yb(x): return 1.68*y(x)*OmegaB/OmegaM
def yc(x): return y(x)*OmegaCDM/OmegaM
ynu = OmegaNu/OmegaRad
def eta(x):
    return 2*alpha*(alpha*x + 1)/(alpha**2 * x**2 + 2*alpha*x)
def delta_phi():
    return (2-8/y(xrec) + 16*xrec/y(xrec)**3)/(10*y(xrec))
xs = 0.6*OmegaM**(.25) * OmegaB**(-.5)*arec**(.75)*h0**(-.5)
```

```
# In[59]:
```

```
# initialize arrays
K = 200
N = 500

SW = np.zeros(K)
ISW = np.zeros(K)
DOP = np.zeros(K)

x_list = np.linspace(0,xrec, N)
delta_gamma = np.zeros((K,N))
delta_c = np.zeros((K,N))
delta_nu = np.zeros((K,N))

k_list = tau_r * 10**np.linspace(0,4,K)/ eta_today

for i, k in enumerate(k_list):
    xi = np.min([10**-4 / k, xeq/10000])
    phi_i = 1
    delta_gamma_i = -2*phi_i*(1 + 3*y(xi)/16)
    delta_c_i = .75 * delta_gamma_i
    v_gamma_i = -k/eta(xi) * (
        delta_gamma_i/4. + (2*k**2 * (1 + y(xi))*phi_i)/
        (9.*eta(xi)**2 * (4./3. +y(xi)))))
    #neutrino initial conditions
    v_nu_i = v_gamma_i
    delta_nu_i = delta_gamma_i
```

```

v_nu_i = v_gamma_i
delta_nu_i = 0

# solve the ODE

def d_func(x, Y):
    """return function derivatives"""

    delta_c = Y[0]
    vc = Y[1]
    delta_gamma = Y[2]
    v_gamma = Y[3]
    delta_nu = Y[4]
    v_nu = Y[5]
    phi = Y[6]

    DY = np.zeros(7)
    DY[6] = (-eta(x) * phi + (3*eta(x)**2 *
        ((v_gamma)*(4./3. + y(x)-yc(x))+ynu*(v_nu-v_gamma)+yc(x)*vc))/(
        (2*(1+y(x))*k)))

# no neutrinos:
#DY[6] = -eta(x) * phi + (3*eta(x)**2*((v_gamma)*
#           (4./3. + y(x)-yc(x))+yc(x)*vc))/(2*(1+y(x))*k)

    DY[0] = -k * vc + 3*DY[6]
    DY[1] = -eta(x)*vc+k*phi
    DY[2] = -4./3. * k *v_gamma + 4*DY[6]
    DY[3] = (-eta(x)*yb(x)*v_gamma + k*delta_gamma/3)/(
        4./3. + yb(x)) + k*phi

#neutrino derivatives
    DY[4] = -4./3. * k *v_nu + 4.*DY[6]
    DY[5] = (k*delta_nu/4.) + k*phi

```

```

    return DY

x_span = [xi, xrec]
Y0 = [delta_c_i, v_gamma_i, delta_gamma_i, v_gamma_i, delta_nu_i, v_nu_i, phi_i]
out = integrate.solve_ivp(d_func, x_span, Y0, max_step=.001)
sol = out['y']

delta_c[i,:] = np.interp(x_list, out['t'], sol[0])
delta_gamma[i,:] = np.interp(x_list, out['t'], sol[2])
delta_nu[i,:] = np.interp(x_list, out['t'], sol[4])

# fill arrays
SW[i] = sol[6, -1] + sol[2, -1]/4.
ISW[i] = 2*delta_phi(xrec)
DOP[i] = sol[3, -1]

SW_terp = interpolate.interp1d(k_list, -SW * np.exp(-k_list**2 * xs**2))
ISW_terp = interpolate.interp1d(k_list, -ISW * np.exp(-k_list**2 * xs**2))
DOP_terp = interpolate.interp1d(k_list, -DOP * np.exp(-k_list**2 * xs**2))

# In[60]:

plt.plot(x_list, np.transpose(delta_c[0:200:10,:]))
LEG = [r"\$\\kappa=% .2f" % k_list[5*i] for i in range(10)]
plt.legend(LEG)
plt.title(r"CDM density perturbation")
plt.ylabel(r'\$\\delta_c\$')
plt.xlabel('x')
plt.show()

```

In[61]:

```
plt.plot(x_list, np.transpose(delta_gamma[0:200:20,:]))
LEG = [r"\kappa=% .2f" % k_list[5*i] for i in range(10)]
plt.title(r"photon/baryon density perturbation")
plt.xlabel('x')
plt.ylabel(r'\delta_\gamma')
plt.legend(LEG)
plt.show()
```

In[62]:

```
plt.plot(x_list, np.transpose(delta_nu[0:200:20,:]))
LEG = [r"\kappa=% .2f" % k_list[5*i] for i in range(10)]
plt.title(r"Neutrino density perturbation")
plt.xlabel('x')
plt.ylabel(r'\delta_\nu')
plt.legend(LEG)
plt.show()
```

In[63]:

```
#Get delta_T:
a = np.vectorize(a)
a_list = a(x_list)
```

```
# In[64]:
```

```
"""Compare to Seljak"""
im = plt.imread('data/seljak_1.png')
implot = plt.imshow(im, extent=[-1,2.2,-1.5,1.5], aspect='auto', zorder=0)

plt.plot(np.log10(k_list), -SW * np.exp(-k_list**2 * xs**2))
plt.plot(np.log10(k_list), -ISW * np.exp(-k_list**2 * xs**2))
plt.plot(np.log10(k_list), -DOP * np.exp(-k_list**2 * xs**2))
plt.show()
```

IX.2.2. get_eff_params.ipynb

```
# coding: utf-8
```

```
# In[7]:
```

```
from scipy import integrate
from scipy import interpolate
from scipy.special import spherical_jn
import numpy as np
import matplotlib.pyplot as plt
from numpy import genfromtxt
import pickle
```

```
# In[8]:
```

```

h0 = 0.67
H0 = 100.*h0
OmegaGamma = 2.47e-5/(h0**2.)
OmegaNu = 1.68e-5/(h0**2.)
OmegaRad = OmegaGamma + OmegaNu
OmegaCDM = 0.1201/(h0**2.)
OmegaB = 0.0223/(h0**2.)
OmegaM = OmegaCDM + OmegaB
OmegaLambda = 1-(OmegaCDM + OmegaB + OmegaGamma)
aeq = OmegaRad/OmegaM

```

In[9]:

```

def eta_itgd(a):
    """Conformal time integrand"""
    return 1/(a**2. * H0 *
               np.sqrt( OmegaM/(a**3) + OmegaRad/(a**4) + OmegaLambda))
)

# compute conformal time today
(eta_today,_) = integrate.quad(eta_itgd, 0, 1)

# compute conformal time at recombination
(eta_rec,_) = integrate.quad(eta_itgd, 0, 10.**(-3.))

# compute conformal time at matter-radiation equality
(eta_eq,_) = integrate.quad(eta_itgd, 0, aeq)

```

In[10]:

```

zrec = lambda OmB: 1000*OmB**(-0.027/(1 + 0.11 * np.log(OmB)))
arec = 1/(1 + zrec(OmegaB))
tau_r = 1/(np.sqrt(OmegaM/arec)*H0/2)
zeq = 1/aeq - 1
alpha = np.sqrt(arec/aeq)
xrec = (np.sqrt((alpha**2 + 1)) - 1)/alpha
xeq = eta_eq/tau_r

```

In[11]:

```

# x is time coordinate
# y is scale factor relative to equality
def y(x): return (alpha*x)**2 + 2*alpha*x
def a(x): return aeq*y(x)
def yb(x): return 1.68*y(x)*OmegaB/OmegaM
def yc(x): return y(x)*OmegaCDM/OmegaM
ynu = OmegaNu/OmegaRad
def eta(x):
    return 2*alpha*(alpha*x + 1)/(alpha**2 * x**2 + 2*alpha*x)
def delta_phi(x):
    return (2-8/y(xrec) + 16*xrec/y(xrec)**3)/(10*y(xrec))
xs = 0.6*OmegaM**(.25) * OmegaB**(-.5)*arec**(.75)*h0**(-.5)

```

In[12]:

```

# initialize arrays

K = 200
N = 500

SW = np.zeros(K)
ISW = np.zeros(K)
DOP = np.zeros(K)

x_list = np.linspace(1.e-5,xrec, N)
delta_gamma = np.zeros((K,N))
delta_c = np.zeros((K,N))
delta_nu = np.zeros((K,N))
v_0 = np.zeros(K)

k_list = tau_r * 10**np.linspace(0,4,K)/ eta_today

for i, k in enumerate(k_list):
    xi = np.min([10**-4 / k, xeq/10000])
    phi_i = 1
    delta_gamma_i = -2*phi_i*(1 + 3*y(xi)/16)
    delta_c_i = .75 * delta_gamma_i
    v_gamma_i = -k/eta(xi) * (
        delta_gamma_i/4 + (2*k**2 * (1 + y(xi))*phi_i) /
        (9*eta(xi)**2 * (4./3. +y(xi)))))
    a0 = a(x_list[0])
    vc_i = v_gamma_i

#neutrino initial conditions

v_nu_i = v_gamma_i
delta_nu_i = delta_gamma_i

```

```

#v_nu_i = v_gamma_i
#delta_nu_i = 0

v_0[i] = (((4/3)*(v_gamma_i+v_nu_i)*OmegaRad)*a0**-4. + \
((vc_i+v_gamma_i)*OmegaRad)*a0**-3)/( \
OmegaRad*a0**-4 + (OmegaM*a0**-3)+ OmegaLambda)

# solve the ODE
def d_func(x, Y):
    """return function derivatives"""

    delta_c = Y[0]
    vc = Y[1]
    delta_gamma = Y[2]
    v_gamma = Y[3]
    delta_nu = Y[4]
    v_nu = Y[5]
    phi = Y[6]

    DY = np.zeros(7)
    # potential derivative
    DY[6] = (-eta(x) * phi + (3*eta(x)**2* \
        ((v_gamma)*(4./3. \
            +y(x)-yc(x))+ynu*(v_nu-v_gamma)+yc(x)*vc))/ \
        (2*(1+y(x))*k))

    # dark matter derivatives
    DY[0] = -k * vc + 3*DY[6]
    DY[1] = -eta(x)*vc+k*phi

    #radiation derivatives
    DY[2] = -4./3. * k *v_gamma + 4*DY[6]
    DY[3] = (-eta(x)*yb(x)*v_gamma + k*delta_gamma/3)/(

```

```

4./3. + yb(x)) + k*phi

#neutrino derivatives
DY[4] = -4./3. * k *v_nu + 4*DY[6]
DY[5] = k*delta_nu/4 + k * phi
return DY

x_span = [xi, xrec]
Y0 = [delta_c_i, v_gamma_i, delta_gamma_i, v_gamma_i, delta_nu_i,
      →v_nu_i, phi_i]
out = integrate.solve_ivp(d_func, x_span, Y0, max_step=.001)
sol = out['y']

delta_c[i,:] = np.interp(x_list, out['t'],sol[0])
delta_gamma[i,:] = np.interp(x_list, out['t'],sol[2])
delta_nu[i,:] = np.interp(x_list, out['t'],sol[4])

# fill arrays
SW[i] = sol[6, -1] + sol[2, -1]/4.
ISW[i] = 2*delta_phi(xrec)
DOP[i] = sol[3, -1]

SW_terp = interpolate.interp1d(k_list, -SW * np.exp(-k_list**2 * xs**2))
ISW_terp = interpolate.interp1d(k_list, -ISW * np.exp(-k_list**2 * xs**2))
DOP_terp = interpolate.interp1d(k_list, -DOP * np.exp(-k_list**2 * xs**2))

# In[13]:

plt.plot(x_list, np.transpose(delta_c[0:200:10,:]))

```

```
LEG = [r"\$\\kappa=% .2f" % k_list[5*i] for i in range(10)]  
plt.legend(LEG)  
plt.title(r"CDM density perturbation")  
plt.ylabel(r'$\\delta_c$')  
plt.xlabel('x')  
plt.show()
```

In[14]:

```
plt.plot(x_list, np.transpose(delta_gamma[0:200:20,:]))  
LEG = [r"\$\\kappa=% .2f" % k_list[5*i] for i in range(10)]  
plt.title(r"photon/baryon density perturbation")  
plt.xlabel('x')  
plt.ylabel(r'$\\delta_{\\gamma}$')  
plt.legend(LEG)  
plt.show()
```

In[15]:

```
plt.plot(x_list, np.transpose(delta_nu[0:200:20,:]))  
LEG = [r"\$\\kappa=% .2f" % k_list[5*i] for i in range(10)]  
plt.title(r"Neutrino density perturbation")  
plt.xlabel('x')  
plt.ylabel(r'$\\delta_{\\nu}$')  
plt.legend(LEG)  
plt.show()
```

```
# In[16]:
```

```
"""Calculate single fluid effective parameters"""

a = np.vectorize(a)
a_list = a(x_list)
a_list[0] = a_list[1] #deal with divide-by-zero

#Get delta_T:
delta_T =(a_list**(-3)*(OmegaB*delta_gamma + OmegaCDM*delta_c) +
           ↵a_list**(-4) * (OmegaGamma*delta_gamma +
           ↵OmegaNu*delta_nu))/(OmegaLambda + a_list**(-3) * OmegaM + a_list**(-4) *
           ↵* OmegaRad)

#get w_eff:
w_eff =((1./3.)*a_list**(-4) * (OmegaGamma + OmegaNu) -
           ↵OmegaLambda)/(OmegaLambda + a_list**(-3) * OmegaM + a_list**(-4) *
           ↵OmegaRad)

#get cs2_eff:
cs2_eff =((1./3.)*a_list**(-4) * (OmegaGamma*delta_gamma +
           ↵OmegaNu*delta_nu))/(a_list**(-3)*(OmegaB*delta_gamma +
           ↵OmegaCDM*delta_c) + a_list**(-4) * (OmegaGamma*delta_gamma +
           ↵OmegaNu*delta_nu))

# In[20]:
```

```

plt.plot(x_list, np.transpose(delta_T[0:170:5,:]))
plt.title("Total density perturbation")
plt.xlabel('x')
plt.ylabel(r"$\delta_{\text{T}}$")
plt.show()
plt.show()

plt.plot(x_list, np.transpose(cs2_eff[1:300:10,:]))
plt.title(r"$c_{\text{sT}}^2$")
plt.xlabel('x')
plt.ylabel('sound speed')
plt.show()

plt.plot(x_list, np.transpose(w_eff))
plt.title(r"$w_{\text{T}}$")
plt.xlabel('x')
plt.ylabel('w')
plt.show()

```

In[18]:

```

"""Now save the effective variables to a pickle"""
pickle.dump(delta_T, open( "data/delta_T.p", "wb" ) )
pickle.dump(w_eff, open( "data/w_eff.p", "wb" ) )
pickle.dump(cs2_eff, open( "data/cs2_eff.p", "wb" ) )
pickle.dump(x_list, open( "data/x_list.p", "wb" ) )
pickle.dump(k_list, open( "data/k_list.p", "wb" ) )
pickle.dump(v_0, open( "data/v_0.p", "wb" ) )

```

IX.2.3. 1-fluid.ipynb

```
# coding: utf-8
```

```
# In[79]:
```

```
from scipy import integrate
from scipy import interpolate
from scipy.special import spherical_jn
import numpy as np
import matplotlib.pyplot as plt
from numpy import genfromtxt
import pickle
```

```
# In[80]:
```

```
h0 = 0.67
H0 = 100.*h0
OmegaGamma = 2.47e-5/(h0**2.)
OmegaNu = 1.68e-5/(h0**2.)
OmegaRad = OmegaGamma + OmegaNu
OmegaCDM = 0.1201/(h0**2.)
OmegaB = 0.0223/(h0**2.)
OmegaM = OmegaCDM + OmegaB
OmegaLambda = 1-(OmegaCDM + OmegaB + OmegaGamma)
aeq = OmegaRad/OmegaM
```

```
# In[81]:
```

```

def eta_itgd(a):
    """Conformal time integrand"""
    return 1/(a**2. * H0 *
              np.sqrt( OmegaM/(a**3) + OmegaRad/(a**4) + OmegaLambda)
              )
# compute conformal time today
(eta_today,_) = integrate.quad(eta_itgd, 0, 1)

# compute conformal time at recombination
(eta_rec,_) = integrate.quad(eta_itgd, 0, 10.**(-3.))

# compute eta_eq
(eta_eq,_) = integrate.quad(eta_itgd, 0, aeq)

# In[82]:

```

```

zrec = lambda OmB: 1000*OmB**(-0.027/(1 + 0.11 * np.log(OmB)))
arec = 1/(1 + zrec(OmegaB))
#what is tau_r?
tau_r = 1/(np.sqrt(OmegaM/arec)*H0/2)
zeq = 1/aeq - 1
alpha = np.sqrt(arec/aeq)
xrec = (np.sqrt((alpha**2 + 1)) - 1)/alpha
xeq = eta_eq/tau_r

```

```
# In[83]:
```

```

# x is time coordinate
# y is scale factor relative to equality
def y(x): return (alpha*x)**2 + 2*alpha*x
def A(x): return aeq*y(x)
def yb(x): return 1.68*y(x)*OmegaB/OmegaM
def yc(x): return y(x)*OmegaCDM/OmegaM
ynu = OmegaNu/OmegaRad
def eta(x):
    return 2*alpha*(alpha*x + 1)/(alpha**2 * x**2 + 2*alpha*x)
def delta_phi(x):
    return (2-8/y(xrec) + 16*xrec/y(xrec)**3)/(10*y(xrec))
xs = 0.6*OmegaM**(0.25) * OmegaB**(-0.5)*arec**0.75*h0**(-0.5)

```

In[84]:

```

"""Load the pickles"""
x_list = pickle.load( open( "data/x_list.p", "rb" ) )
k_list = pickle.load( open( "data/k_list.p", "rb" ) )
delta_T = pickle.load( open( "data/delta_T.p", "rb" ) )
w_eff = pickle.load( open( "data/w_eff.p", "rb" ) )
cs2_eff = pickle.load( open( "data/cs2_eff.p", "rb" ) )
v_0 = pickle.load( open( "data/v_0.p", "rb" ) )

```

In[85]:

```
def W(x):
```

```

a = A(x)

w = ((1./3.)*OmegaRad*a**-4. - OmegaLambda)/\
     (OmegaRad*a**-4.+OmegaM*a**-3.+OmegaLambda)

return w

def DW(x):
    a = A(x)

    da_dx = aeq*(2.*alpha + 2.*x*alpha**2.)

    dw_da = -((9 * a**4. * OmegaLambda*OmegaM + 16. * a**3. \
               *OmegaLambda * OmegaRad + OmegaM * OmegaRad)/
               (3. * (a**4. * OmegaLambda + a*OmegaM + OmegaRad)**2.))

    dw_dx = dw_da*da_dx

    return dw_dx

CS2 = interpolate.interp1d(x_list, cs2_eff,kind='quadratic',fill_value="extrapolate")

```

In[86]:

```

# initialize arrays

K = 200
N = 500

SW = np.zeros(K)
ISW = np.zeros(K)
DOP = np.zeros(K)

x_list = np.linspace(0,xrec, N)
delta_gamma = np.zeros((K,N))
delta = np.zeros((K,N))
v = np.zeros((K,N))

k_list = tau_r * 10**np.linspace(0,4,K) / eta_today

```

```

for i, k in enumerate(k_list):
    xi = np.min([10**-4. / k, xeq/10000.])
    phi_i = 1
    delta_i = -2.*phi_i*(1 + 3.*y(xi)/16.)
    v_i = -k/eta(xi) * (
        delta_i/4 + (2*k**2. * (1. + y(xi))*phi_i) /
        (9*eta(xi)**2. * (4./3. +y(xi))))
    v_i = v_0[i]

    x_old = 0.

    # solve the ODE
    def d_func(x, Y):
        global x_old
        """return function derivatives"""

        delta = Y[0]
        v = Y[1]
        phi_old = Y[2]

        DY = np.zeros(3)
        w = W(x)
        cs2 = CS2(x)[i]
        ETA = eta(x)

        #gdm version
        phi = -(3./2.)*(ETA/k)**2.*((delta + 3.*ETA*v/k))
        DY[2] = (phi - phi_old)/(x-x_old)
        DY[0] = -(1. + w)*(k * v - 3.*DY[2]) - 3.*ETA*(cs2 - w)*delta
        DY[1] = ETA*(1-3*w)*v + (cs2/(1+w))*k*delta + k * phi - v*Dw(x)/(1.+w)

        #DY[2] = -eta(x) * phi + 3*eta(x)**2*v/ (2*(1+y(x))*k)
        #DY[0] = -4./3. * k *v + 4*DY[2]
        #DY[1] = k*delta/4 + k * phi
        x_old = x

```

```

    return DY

x_span = [xi, xrec]
Y0 = [delta_i, v_i, phi_i]
out = integrate.solve_ivp(d_func, x_span, Y0, max_step=.001)
sol = out['y']

delta[i,:] = np.interp(x_list, out['t'], sol[0])
v[i,:] = np.interp(x_list, out['t'], sol[1])

# fill arrays
SW[i] = sol[2, -1] + sol[1, -1]/4.
ISW[i] = 2*delta_phi(xrec)
DOP[i] = sol[1, -1]

SW_terp = interpolate.interp1d(k_list, -SW * np.exp(-k_list**2 * xs**2))
ISW_terp = interpolate.interp1d(k_list, -ISW * np.exp(-k_list**2 * xs**2))
DOP_terp = interpolate.interp1d(k_list, -DOP * np.exp(-k_list**2 * xs**2))

# In[87]:

plt.plot(x_list, np.transpose(delta[1:170:2,:]))
LEG = [r"\kappa=% .2f" % k_list[10*i] for i in range(10)]
plt.title(r"Single fluid density perturbation")
plt.xlabel('x')
plt.ylabel(r'\delta_\nu')
plt.legend(LEG)
plt.show()

```

In[88]:

```
"""Compare to Seljak"""
if True:
    im = plt.imread('data/seljak_1.png')
    implot = plt.imshow(im, extent=[-1,2.2,-1.5,1.5], aspect='auto', zorder=0)

    plt.plot(np.log10(k_list), -SW * np.exp(-k_list**2 * xs**2))
    plt.plot(np.log10(k_list), -ISW * np.exp(-k_list**2 * xs**2))
    plt.plot(np.log10(k_list), -DOP * np.exp(-k_list**2 * xs**2))
    plt.ylim(-3000,3000)
    plt.show()
```

In[89]:

```
plt.title(r"$w_T$")
plt.xlabel('x')
plt.ylabel('w')
plt.plot(x_list,W(x_list))
plt.plot(x_list,DW(x_list))
plt.legend([r"$w(x)$", "dw/dx"])
plt.show()
```

In[90]:

#compute Cl

```

eta_star = eta_rec

k_list = np.arange(15914)*tau_r*2 * np.pi/(10 * eta_today) + tau_r/eta_today
ell_list = np.array([2,3,4,5,6,7,8,10,12,15]+range(20,100,10)+
                    range(120,500,20) + range(550,1250,50))

```

precompute all bessels:

```

J_lk = np.zeros((len(ell_list),len(k_list)))
DJ_lk = np.zeros((len(ell_list),len(k_list)))
k_scaled = k_list*(eta_today-eta_star)/tau_r
for i in range(len(k_list)):
    J_lk[:,i] = spherical_jn(ell_list,k_scaled[i])
    DJ_lk[:,i] = .5*(spherical_jn(ell_list + 1, k_scaled[i]) -\
                      spherical_jn(ell_list - 1, k_scaled[i]))

```

In[91]:

```

Cl_intgd =(np.exp(-2*k_list**2. * xs**2)*
            ((SW_terp(k_list) + ISW_terp(k_list))*_
             J_lk + DOP_terp(k_list)*_
             (-(J_lk/(2*k_scaled))+_
              DJ_lk))**2)
Cl_list = integrate.cumtrapz(Cl_intgd/k_list, axis=1)[:, -1]

```

In[92]:

```

Cl_terp = interpolate.interp1d(ell_list, Cl_list)
Cl_terp_quad = interpolate.interp1d(ell_list, Cl_list, kind='quadratic')
l_range = np.linspace(2, 1200, 1000)

```

```
plot_vals = Cl_terp(l_range)*l_range * (l_range+1) / (2 * (2+1) * Cl_terp(2))
plot_vals_2 = Cl_terp_quad(l_range)*l_range * (l_range+1) /\n
(2 * (2+1) * Cl_terp_quad(2))
```

```
# In[93]:
```

```
"""Compare C_l to mathematica"""
mat_data = genfromtxt('data/mat_data.csv', delimiter=',')
plt.plot(mat_data[:,0], mat_data[:,1], color='orange')
plt.plot(l_range, plot_vals_2, color='blue', linestyle=':')
plt.legend(['mathematica', 'python'])
```