Title

Strong Lens Line-Of-Sight Analyses in the Era of Massive Galaxy Surveys

Author

Wells, Patrick Ryan

Publication Date

2024

Peer reviewed|Thesis/dissertation

Strong Lens Line-Of-Sight Analyses in the Era of Massive Galaxy Surveys

By

Patrick R. Wells
Dissertation

Submitted in partials satisfaction of the requirements for the degree of

Doctor of Philosophy

in

Physics

in the

Office of Graduate Studies

of the

University of California

Davis

Approved:

Christopher Fassnacht, Chair

Tucker Jones

David Wittman

Committee in Charge

2024

Contents

List of Figures

List of Tables

Abstract of the Dissertation

Strong Lens Line-Of-Sight Analyses in the Era of Massive Galaxy Surveys

One of the foundational goals of modern cosmology is the precise and accurate measurement of the expansion rate of the universe, which is denoted by the Hubble constant (H_0). In recent years, the results of the most mature methods available for making this measurement have diverged to such an extent that some have called it a "crisis in cosmology." The ultimate resolution of the crisis is not yet clear, but there is no doubt that additional methods for constraining H_0 can shed light on this underlying problem.

Time-delay cosmography is one such method for constraining the expansion rate. It involves careful measurements and modeling of strong lensing systems where the luminosity of the background object is variable. This variability appears at different times to the observer, and the length of this "time delay" depends on the underlying cosmology in addition to the geometry and mass distribution of the lens system itself. Chapter 1 of this dissertation provides an introduction to modern physical cosmology, and further introduction to the theory of gravitational lensing and time-delay cosmography.

The environment of the lensing system introduces perturbations that affect the results of the cosmographic analysis, generally at the percent level. Accounting for this bias is crucial for accurately measuring the underlying cosmological parameters. While some structures can be accounted for in the primary mass model itself, this requires high quality data and significant modeling time, and is generally not possible for the majority of objects in the field. The cumulative effect of the remaining perturbers is known as the "external convergence" ($\kappa_{\rm ext}$) and must be handled statistically.

This dissertation explores one technique for quantifying the collective impact of these remaining objects. The technique involves comparing a summary statistics from a line of sight of interest to summary statistics from large number of similar lines of sight in a suitable reference survey. This provides an empirical estimate of the mass density in the field as compared to the Universe as a whole, which can be compared to cosmological simulations to estimate the value of $\kappa_{\rm ext}$. This quantity is used directly in the final cosmological inference as a correction factor on the time-

delay distance, which is inversely proportional to H_0. Chapter 2 provides an introduction to these techniques and an application to a time-delay lens.

Currently, there are only around a dozen time-delay lenses that have been fully analyzed to put constraints on cosmological parameters. However the number of known systems is rising fast, and the Legacy Survey of Space and Time (LSST) is expected to increase this number by around a factor of 10. Working with these systems requires the development of tools that are capable of operating at larger scale, and statistical models that can combine information from many lens systems into a single estimate of cosmological parameters. Chapter 3 introduces a framework for estimating the population distribution of $\kappa_{\rm ext}$ for a sample of lenses. I use this framework to show that a sample of 25 lenses from the Strong Lensing Legacy Survey (SL2S) fall in preferentially overdense lines of sight. This finding is expected based on previous work, but this dissertation represents the first time this overdensity has been quantified in a way that is useful for time-delay cosmography. Additionally, I present a new statistical model which may shed light into underlying mass distribution in our lines of sight and provide a path forward for improving the measurement.

The analyses presented in this dissertation where accompanied by a significant amount of work developing high-quality software to perform the analyses and enable future ones. The software produced is capable of scaling to many more systems than are currently available for analysis and adapting to significant changes in the underlying techniques without being rewritten. This software, and some of the philosophy behind its development, is presented in detail in Chapter 4

Finally in Chapter 5 I draw some conclusions based on this work and look forward to the future of the $\kappa_{\rm ext}$ measurement and cosmological software.

Acknowledgments

I often joke that my parents feel that my PhD is as much their accomplishment as it is mine. Every joke has some truth to it, and there's no doubt they deserve a healthy portion of the credit. I have worked hard for this, but I believe deeply that the person I am today is largely the result of the influence of the people who have ended up in my life mostly by accident. This dissertation is dedicated to every single one of them, but there are some who deserve special recognition.

I dedicate this dissertation to my parents. To my father, who seeded the love for mathematics and computation that have carried me through this program and into my career. And to my mother, who placed everything on hold to educate myself and my siblings, and taught me about human beings.

I dedicate this dissertation to my wife Rebecca, who has scarified more than I had any right to ask for. She has been with me since a time when I thought I was going to be a doctor (the medical sort). She has stuck with me through the complexities of multiple moves, and challenges in her other close relationships that were brought on by ours. A few days from the writing of these acknowledgements, she will be following me on yet another adventure as we move across the country so I can take a new job. I cannot imagine living in this world without her.

I dedicate this thesis to my family, particularly my siblings Brian and Natalie. When I was much younger, I sometimes struggled to understand how they were so different from me. As an adult, I now recognize just how important thos differences are. I am proud of them both for what they have accomplished and what they will accomplish in the future. I also dedicate this thesis to my grandparents, Ken and Anne Evans and Russ and Bonnie Hensel. Each was a central figure in my life in their own way. After all, it does take a village.

I dedicate this dissertation to my advisor Chris, who picked me up when I was at a low point and has been nothing but supportive and encouraging throughout this journey. I will always marvel that he knew before I did what I would enjoy working on. More so than anyone else he deserves credit for helping me make the transition from student to scientist.

I dedicate this dissertation those who have contributed to my math and science education over the past 12 years, particularly John McDaniel and Kelly Molitor from Nevada Union High School, and Ken Hennessey, Gerard Fasel, Kassahun Betre and John Mann from Pepperdine University.

I dedicate this dissertation to those teachers and mentors in other disciplines, whose impact is all the more important for giving me a more rounded view of the world. They helped me see parts of this universe that I could never have seen by way of astronomy. In particular, I want to thank Meegan Toro, Rod Baggett and Molly Star from Nevada Union High school, as well Katie Frye and Ryan Board from Pepperdine University.

I dedicate this dissertation to the friends I made along the way. In particular Danielle Adair, Preston Hereford, Arsalan Adil, Rizwan Asghar, Yashwanth Bezawada, Patricia Bolan, Jake Cohen, Tyler Erjavec, Pratik Gandhi, Julie He, Ian Lim, Pritom Mozumdar, Karthik Prabhu Palimar, Sudheer Sreedhar and Siddarth Vadnerkar. Many of them went through the graduate school experience with me, including the traumas of COVID and losing one of our own only a month or so before graduation. But I have learned so much from each of them, and will miss them dearly as we go off in our separate directions.

I dedicate this dissertation to my therapists, particularly Blanca and Daniel, who have helped me understand how to live in this world and be a more authentic version of myself.

Lastly, I want to dedicate this thesis to the many students I had the privilege of working with over my time in graduate school. Teaching introductory physics help me learned how to talk about my work in an accessible and relatable way, and gave me people skills that I believe will be essential to my future career. Even though my teaching work doesn't appear explicitly in this dissertation, working with students was consistently the best parts of my week.

I also wish to acknowledge the support and mentorship that has been extended to me by the TDCOSMO collaboration, especially by Simon Birrer and Liz Buckley-Geer. Chapters 2 and 3 of this dissertation were published under the TDCOSMO umbrella, and their guidance through that process has been instrumental in helping me produce work I am proud of.

The world is a very different place than it was when I started my graduate school journey. Or maybe I am just a very different person. I've done a lot of growing up over the past six years. I am proud of this work, but I believe its just the beginning of what I will be able to do over a lifetime. Thank you to everyone, past and present, who is a part of that.

Chapter 1

ΛCDM, The Hubble Tension, and Time-Delay Cosmography

As with many cosmological analyses, the overarching goal of the work presented in this dissertation is the precise anbd accurate inference of cosmological parameters. Time-delay cosmography aims to provide a route to measuring parameters such as the Hubble constant. However in practice, the work in this dissertation is several steps removed from this overall result. A solid grounding in modern cosmology is therefore necessary to appreciate the importance of this work

In this chapter, I present an introduction to the core ideas that provide the foundation for my work in later chapters. I begin with an introduction to the standard ΛCDM model of cosmology, and discuss the current tension in measurements of the expansion rate. I then move to gravitational lensing and the role of time-delay cosmography in shedding light on the source of this tension. I briefly introduce the relationship of my work to time-delay cosmography, and discuss some of the software practices that I utilize to produce my results.

1.1 How we Got Here: Brief History of Physical Cosmology

Humans have demonstrated interest in the stars since antiquity. For the vast majority of our history, light pollution was virtually non-existent, and anyone spending time outside on a clear night could see the stars more vibrantly than most of us living in the 21st century ever will. Interest in the Universe and our place within it is embedded deep in the mythology of cultures around the world. Ancient astronomers and mystics constructed elaborate cosmologies based on a blend of direct observation and spiritual discovery.

At times, this mythology has clashed with the discoveries of the more empirically-minded. Among the most famous examples in the western canon is that of Galileo, who was put on trial by the Catholic church for championing the heliocentric model of the solar system developed by Copernicus. But barriers are not only put up by individuals outside the scientific community. Changes in understanding of the fundamental nature of reality has always been a uphill battle, and history is littered with knowledgeable and talented scientists who have ended up tragically wrong. This is a clear example of a common occurrence: the Universe will inevitably humble us just when we think we're starting to get a grasp on it.

However even as the heliocentric model fell out of fashion in the west, the idea of distant galaxies and indeed a whole Universe beyond the Milky Way was largely out of reach. Galaxies beyond the Milky Way have been observed for centuries, but were termed "spiral nebulae" and thought to exist as part of our own Galaxy. However there were those who thought there may be more to the story. Emmanuel Kant famously posited in 1755 that these "spiral nebulae" were in fact "island universes," but even he did not comprehend the true scale of the cosmos.

The observational foundations of modern cosmology were arguably laid in the early 19th century by Henrietta Leavitt, who first discovered and quantified the period-luminosity relationship of Cepheid variable stars (Leavitt & Pickering, 1912). This "standard candle" represented the first time astronomers had a tool to measure the distance to objects that were beyond the range of parallax. In 1924, Edwin Hubble used this relationship to show unequivocally that the "spiral nebulae" were in fact beyond the bounds of the Milky Way (Hubble, 1925), kicking off a fundamental shift in our scientific understanding of the cosmos.

At the same time, advances in theoretical understanding gave astronomers the tools to describe a more dynamic Universe. Einstein's General Relativity showed that spacetime (and by extension the Universe itself) would bend and warp based on the the mass and energy it contained (Einstein, 1915). In 1927, Belgian physicist and Catholic priest Georges Lemaître presented his theory of the "primeval atom," which posited that the modern Universe had originated from a much smaller, denser, and hotter Universe in the distant past (Lemaître, 1927). Though initially rejected by Einstein himself, this theoretical development was put on firm footing in 1929 when Hubble clearly demonstrated the redshift-distance relationship of distant galaxies, showing that the Universe was indeed expanding Hubble (1929). He additionally provided an empirical estimate of the costant

which now bears his name and while the measurement itself was only correct to an order of magnitude, the evidence was the beginning of the end for the steady-state model of the Universe favored by many at the time.

Though the remainder of the 20th century, cosmology flourished as it became more of a precision science. Penzias & Wilson (1965) first detected (accidentally) the cosmic microwave background, a fundamental prediction derived from Lemaître's primeval atom by Alpher & Herman (1948). Over the coming decades, successive ground and space-based missions painstakingly charted the small anisotropies in the CMB and used them to probe the foundational nature of the early Universe. Meanwhile, additional observations suggested that the majority of the mass in the Universe is invisible to even the most advanced telescopes. The term "dark matter" was coined by Fritz Zwicky based on a study of the Coma Cluster, where he demonstrated that the implied mass of the cluster based on the movement of its members was inconsistent with the amount of visible matter (Zwicky, 1933). Further observations throughout the 20th century confirmed his hypothesis. Among the most famous example is the measurement of galaxy rotation curves such as in Rubin & Ford (1970). However this imbalance between observed mass and gravitational effects is visible in many phenomena, including gravitational lensing and the previously mentioned CMB. Although it has never been directly observed, dark matter has clearly become the dominant paradigm to explain these observations.

Perhaps the most recent shakeup to the baseline cosmological model came in 1998, when evidence of the accelerated expansion of the Universe was presented simultaneously by teams led by Adam Riess and Saul Perlmutter based on measurements of Cepheid-calibrated Type Ia supernovae (Riess et al., 1998; Perlmutter et al., 1999). The accelerated expansion suggested the existence of an additional energy component in the Universe, which came to be known as dark energy. In the standard cosmological paradigm, this energy is thought to be intrinsic to spacetime itself, but its exact nature remains stubbornly hidden. Attempts to quantify this intrinsic energy with quantum field theory have resulted in what has been jokingly called the "worst prediction in physics" due to overshooting the amount of dark energy needed to reproduce observations by some 100 orders of magnitude.

The work of the past century has culminated in the modern ΛCDM model of cosmology. This model assumes the existence of a cosmological constant (Λ) represents the intrinsic energy of

space time and is responsible for the dark energy component discussed above. It also assumes the existence of cold dark matter (CDM), which is responsible for structure formation and makes up a significant majority of matter in the Universe. While imperfect, the model is remarkably successful in a number of regimes, particularly on large scales. It is without question an incredible achievement, and has stood up remarkably well to the high-precision measurements of the last two decades.

As presented here, physical cosmology has always been a data-oriented field. Although Einstein initially rejected Lemaître's (theoretical) proposal of an expanding Universe, he found Hubble's work to be far too convincing to be ignored. By modern standards, the quantity and quality of available data was quite poor. Hubble's discovery of universal expansion was made with only 24 objects, yet fundamentally undermined the position of those who believed in a steady-state Universe. Similarly Riess et al. made their inference of accelerated expansion with 50 supernovae. Cosmological data has historically been hard to come by, for the simple reason that telescopes are expensive and the amount of time required to obtain high-quality data scales with distance.

However in the modern era, many sub-disciplines of cosmology are grappling with the opposite problem. Modern galaxy surveys produce data containing billions of extragalactic objects across thousands of square degrees of sky. Systems that were once wholly unique are now simply members of a large population of many known systems. While ΛCDM has largely taken this data in stride, there are tantalizing hints that there may be more to the story. In recent years one particular discrepancy, known as the Hubble tension, has widened to the point that it can no longer be ignored. This tension forms the background of much of the work presented in this dissertation, so we begin with a brief overview of ΛCDM and review some of its shortcomings.

1.2 Foundations of ΛCDM

The ΛCDM model assumes a background spacetime that is homogeneous and isotropic on large scales. This spacetime is described by a Friedmann–Lemaître–Robertson–Walker (FLRW) metric:

$$ds^2 = -c^2dt^2 + a(t)^2\left[\frac{dr^2}{1-kr^2} + d\Omega^2\right] \tag{1.1}$$

where $k = -1, 0, 1$ represents the intrinsic curvature and $a(t)$ denotes the value of the scale factor at time t. It is conventional to set $a = 1$ at the present day. We define the Hubble parameter as the

fractional rate of increase of the scale factor at a given time:[1]

$$H(t) = \frac{\dot{a}(t)}{a(t)}. \tag{1.2}$$

For relatively nearby objects, the Hubble parameter can be estimated simply by taking the ratio of their line-of-sight velocity to their absolute distance. However over the longer expansion history of the Universe, the density of the various components of the Universe change and the Hubble parameter evolves non-linearly according to the Friedmann equation:

$$H^2(a) = \frac{8\pi G}{3}\rho(a) - \frac{kc^2}{a^2} + \frac{\Lambda c^2}{3} \tag{1.3}$$

where G is the gravitational constant, ρ is the energy density of the Universe, and Λ is a cosmological constant.

The intrinsic spatial curvature of the Universe is determined by the combined energy density of all its components. Often, it is useful to talk about the energy density of a given component of the Universe as a fraction of the critical density of the Universe, which is the density of a universe that is spatially flat on large scales:

$$\Omega_{\rm i} = \frac{\rho_{\rm i}}{\rho_{\rm c}} \tag{1.4}$$

where

$$\rho_{\rm c} \equiv \frac{8\pi G}{H^2}. \tag{1.5}$$

It is conventional to recast equation 1.3 in terms of these densities and redshift (z), where $z = 0$ denotes the present day. The relative rate of expansion of the Universe at a given redshift z can therefore be computed by

$$H^2(z) = H_0^2\left[\Omega_r(1+z)^4 + \Omega_m(1+z)^3 + \Omega_k(1+z)^2 + \Omega_\Lambda\right] \tag{1.6}$$

where Ω_r, Ω_m, Ω_k and Ω_Λ represent the density of radiation, matter, curvature, and dark energy and the cosmological constant at the present day in units of the critical density, and H_0 represents the current value of the expansion rate.

[1] It is worth mentioning here that accelerated expansion actually does *not* mean that H is increasing, rather that $\ddot{a}$ is positive.

In standard ΛCDM, the Universe is flat ($\Omega_k = 0$). Radiation density falls off rapidly in the early Universe both due to the increasing volume of the Universe and cosmological redshift. For the systems studied in this dissertation, it is safe to approximate the above with a simplified relationship:

$$H^2(z) = H_0^2[\Omega_m(1+z)^3 + \Omega_\Lambda] \tag{1.7}$$

The scaling relationship above is set by the equation of state of the various components in the universe, which is parameterized by

$$w = \frac{p}{\rho} \tag{1.8}$$

where p and ρ represent the density and pressure of the component when modeled as a perfect fluid. Given w, the relationship between the density of the component ρ and the scale factor a is given by

$$\rho \propto a^{-3(1+w)}. \tag{1.9}$$

For example, matter is pressureless which leads to an equation of state $w = 0$ and a scaling relationship $\rho_m \propto \frac{1}{a^3}$. From a conceptual standpoint, matter dilutes in the universe as one might expect. If the physical volume of a given region of the universe doubles, the density of matter halves.

In classical ΛCDM, dark energy is modeled as a cosmological constant with equation of state $w = -1$ for all times. This leads to the behavior shown above, where the density of dark energy remains constant as the Universe expands. However, there are a number of extensions where the dark energy equation of state is allowed to either take on a different value or vary over time. Although the ΛCDM model is significantly more complicated than the relationship above would suggest, late-Universe analyses are primarily sensitive to H_0, Ω_m, and w.

1.2.1 Distance Measures in Cosmology

An expanding, dynamic spacetime complicates standard notions of distance. The Universe contains no road signs or distance markers, making it difficult to determine an absolute distance scale. There are a number of different measurements that are useful for different applications. For theoretical

work it is often helpful to use comoving coordinants, which remain constant for objects with no peculiar velocity. The comoving distance from an observer to a source at redshift z is given by:

$$D_C = \frac{c}{H_0} \int_0^z \frac{dz'}{\sqrt{\Omega_m (1+z)^3 + \Omega_\Lambda}}, \tag{1.10}$$

where I use the same simplification introduced in Eqn. 1.7. Note that this equation is only valid for a flat Universe.

A constant *comoving* distance will always result in a changing *physical* distance. This physical distance is ultimately what our techniques must measure directly. One such physical distance which will be useful in later sections is the *angular diameter distances*, which is given by

$$D_A = \frac{D_C}{1+z}. \tag{1.11}$$

D_A is equivalent to the ratio of an object's physical size to its angular size on the sky. A unique property of relativistic cosmology is that this distance does not increase monotonically. Past $z \approx 1.7$, the angular diameter distance decreases and objects appear larger on the sky.

However as one might expect, objects *do* appear dimmer with increasing distance according to the inverse square law. Given a distant object of known intrinsic luminosity, the distance inferred by comparing to its apparent luminosity is known as the *luminosity distance* and is given by

$$D_L = (1+z) D_C, \tag{1.12}$$

which does increase monotonically.

The difference between these distance measures is salient when inferring the value of H_0 with different probes. Any measure that can empirically determine the relationship between redshift and one of these distance scales implicitly probes the underlying cosmology. As such, developing techniques for determining this relationship is one of the primary preoccupations of modern cosmology.

1.2.2 Standard Techniques for Measuring H_0 and the Hubble Tension

In practice, the headlining result for any distance measure is often the inferred value of H_0. Although there are practical reasons for this in some cases, this is not always the case. When fitting a ΛCDM

cosmology to the Cosmic Microwave Background for example, the value of H_0 is not among the sampled parameters, and is instead derived after the fact.

Traditional techniques anchoring the distance scale can broadly be broken down into two primary categories. The first, known as the "distance ladder," involves the use of objects whose absolute luminosity can be determined *a priori*, such that their relative luminosity can be inverted to determine a luminosity distance. One of the foundational extragalactic rungs of this ladder is the Cepheid period-luminosity relationship, which can calibrated by measurements of these stars that are within range of parallax. Cepheids can then be used to calibrate the absolute luminosity of Type Ia supernovae, which are among the most luminous events in the Universe and can be seen at cosmological distances. Combined with redshift information of the supernovae hosts, this technique directly measures the expansion history of the Universe. For the most recent results produced with this technique, see Riess et al. (2022)

The second method involves observations of the cosmic microwave background and other cosmic structures. With these methods, the entire cosmlogical model must be fit to the observed data together. Broadly speaking, the distance scale can be determined by comparing the angular size of CMB anisotropies to their expected physical size at recombination. This result is much more model dependent than the distance latter but is extremely precise, with the most recent results from the Planck team reaching sub-percent precision (Aghanim et al., 2020a).

Historically these techniques have generally been in agreement about the rate of expansion, with the difference between their central values acceptable given the precision of the measurements. However in the past decade or so, increased precision has revealed a significant discrepancy between their inferred values. This so-called "Hubble Tension" may be the result of previously-unnoticed systematics in one or both of the measurements, or the breakdown of the ΛCDM cosmological model (see Knox & Millea, 2020, for an overview of some potential solutions). Work exploring both of these hypotheses is ongoing, and this dissertation will not make an argument for or against either. However regardless of the ultimate resolution, the tension highlights the need for independent measurements of H_0, particularly ones that do not depend (or depend only weakly) on the background cosmological model. Time-delay cosmography represents one such technique and is the subject of this dissertation. However a number of complementary techniques exist, such Standard Sirens (e.g. Bom et al., 2024). Other techniques such as the Tip of the Red Giant Branch

(e.g. Anand et al., 2022) can be used to replace or compliment rungs on the distance ladder. For an overview of the results a variety of techniques, see Figure 1.1.

1.2.3 Cold Dark Matter and Structure Formation

Foundational to modern cosmology is the so-called "Cosmological Principle," which posits that no observer should think of themselves as being in a "special" place in the Universe. This principle implies homogeneity and isotropy: the Universe should more or less look the same on large scales to any observer at any location looking in any direction. This idea is consistent with observations on a wide variety of scales. Yet structure *does* exist in the Universe on smaller scales. Although structure formation is not the focus of this dissertation, much of the work presented in later chapters is implicitly probing and attempting to quantify the structure in a small region of the Universe. As such, I will take a moment to qualitatively discuss its origins in the standard model.

In the standard paradigm, dark matter is collisionless and decouples from the primordial plasma at non-relativistic (cold) speeds. The seeds of cosmological structure are thought to be the result of density perturbations laid down by inflation in the very earliest moments of the Big Bang. These density perturbations have a nearly scale-invariant power spectrum, which result in the amplitude of these fluctuations decreasing with size. Regions of higher density naturally attract more material due to gravity, but baryonic matter cannot fall freely due to radiation pressure.

Dark matter on the other hand has no such constraint, and density perturbations continue to grow over time. This initial growth is known as the "linear" regime of structure formation. In this regime, dark matter density perturbations are small compared to the background density field, and perturbations at different scales evolve independently. This picture provides an excellent fit to observations of large-scale structure. However over time, perturbations grow to the point that they can no longer be treated in a linear approximation. This regime is important for the contents of this dissertation, because I aim to quantify mass densities in a small region of the Universe.

Eventually, galaxies form inside these dark matter structures. Over time, smaller dark matter structures combine to form larger structures, resulting in the Universe as we see it today. Simulations of structure formation based on this picture are consistent with data across a range of scales. On smaller scales a number of discrepancies exist, but some of these may be mitigated by higher-resolution simulations with more sophisticated galaxy formation models. (see Weinberg et al., 2015, for an overview)

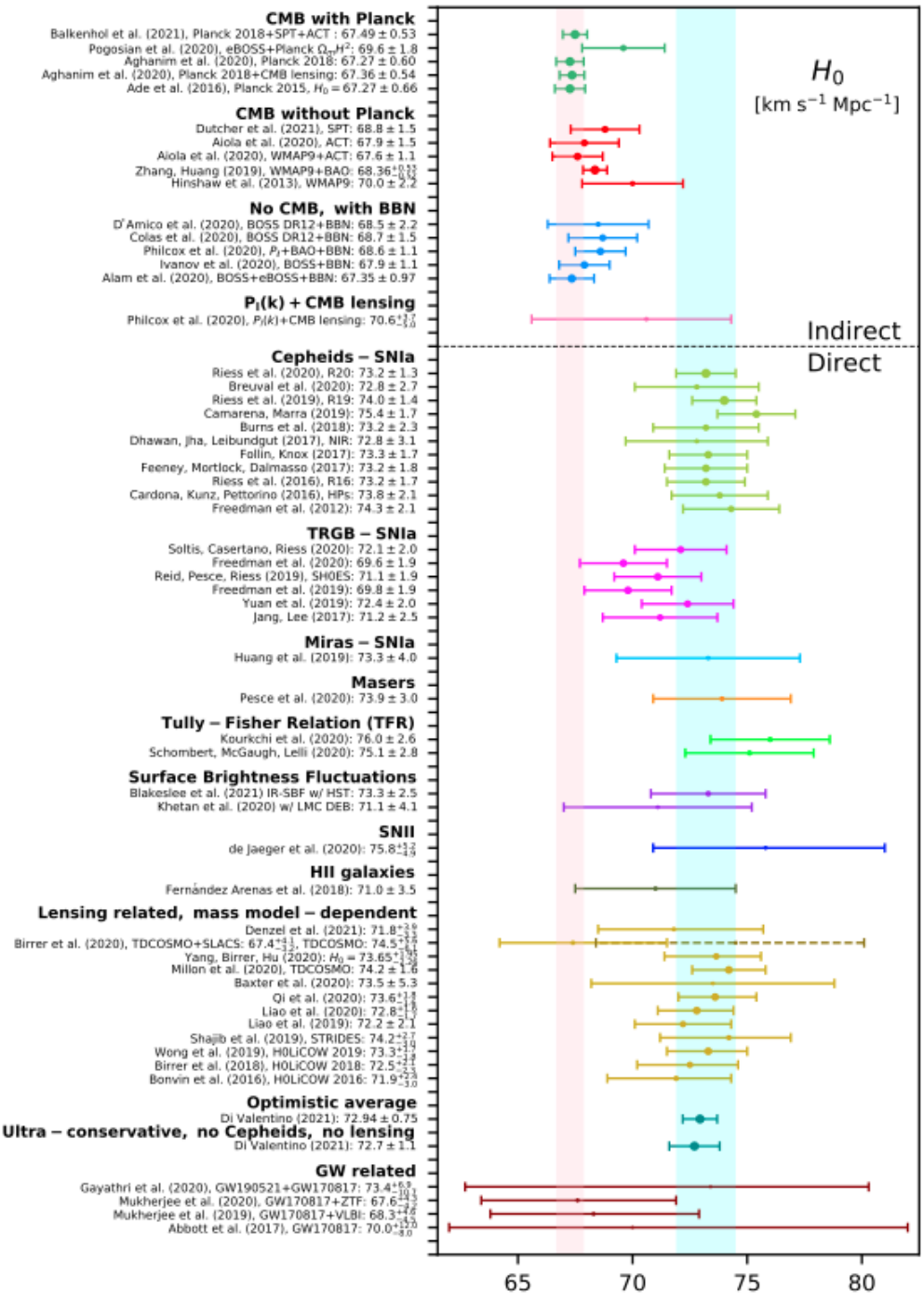

Figure 1.1 Overview of measurements of H_0 from a wide variety of methods as presented in Di Valentino et al. (2021). Measurements above the dotted line are model-dependent early-universe probes, while those below involve late universe probes and are generally less dependent on the underlying model. The red and blue regions represent the current best measurements based on the CMB (Planck Collaboration, 2020) and cephied-calibrated supernovae (Riess et al., 2022), which are the current highest-precision measurements in their respective regimes. There is a clear discrepancy between these results. Reducing the error bars on the plethora of additional techniques is a primary goal of the astronomy community.

The most important fact for the purposes of this work is that the majority of mass in any large region of space (that is, containing at least a few galaxies) is made up of dark matter. This presents challenges for any astrophysical analysis, as the only way to infer the quantity of dark matter is through its gravitational effects. Galaxy rotation is one such effect, but much more interesting for the purposes of this dissertation is gravitaional lensing, which refers to the bending of light by spacetime curvature as it passes by a massive object. Gravitational lensing studies require significantly more mass to explain observations than can be seen with visible light.

1.3 Gravitational Lensing Fundamentals

In this section, I review gravitational lensing and some of the theoretical tools that can be utilized to describe lens systems. Gravitational lensing refers to the apparent[2] bending of light observed by an outside observer when that light passes by a massive object. This effect is a result of the bending of spacetime as predicted in Einstein's General Relativity(Einstein, 1915)[3], and first confirmed observationally by the Eddington expedition in (Dyson et al., 1920).

1.3.1 Basic Formalism

a

According to General Relativity, a ray of light passing by a spherical mass M will be deflected by an angle $\hat{\alpha}$ given by

$$\hat{\alpha} = \frac{4GM}{c^2 r} \tag{1.13}$$

where r is the impact parameter, or distance of closest approach. However many objects in the Universe are not spheres, and many lensing systems involve configurations where the light passes too close to the mass for it to be approximated as such. In such cases, we must treat the mass as distributed in space and integrate over this distribution to determine the lensing effect. When the distances in the system are much larger than the scale of the mass itself (as with galaxy-scale lenses), we can treat the primary lens as being infinitesimally thin:

[2]Everything is "apparent" in relativity. Light follows null geodesics in spacetime, which can be thought of as the generalization of straight lines in a curved spacetime.

[3]Extensions to Newtonian mechanics also predict lensing, but by a different amount

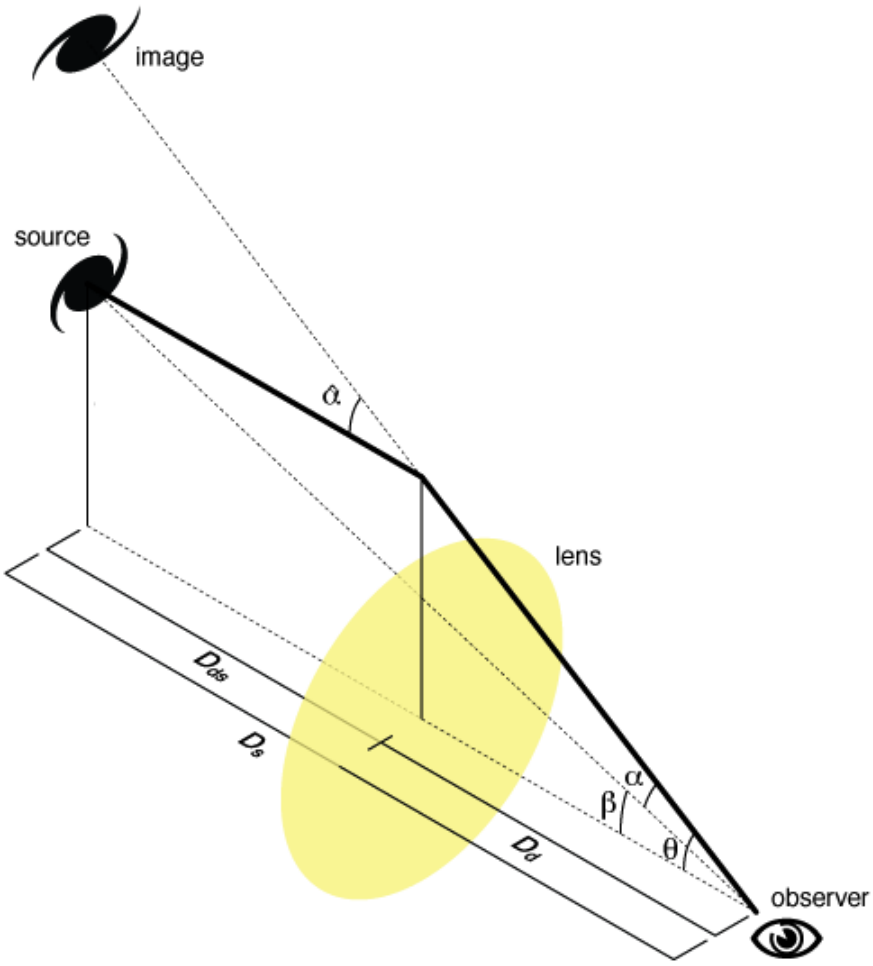

Figure 1.2 Geometry of a generic lens system with exaggerated angles. Typical angles are in the arcsecond range. Figure adapted under CC BY-SA 4.0 license from original by Michael Sachs

$$\vec{\theta} - \vec{\beta} = \vec{\alpha}(\vec{\theta}) = \frac{4G}{c^2} \int \frac{(\vec{\theta} - \vec{\theta}')\Sigma(\vec{\theta}')}{|\vec{\theta} - \vec{\theta}'|^2} \tag{1.14}$$

where Σ represents the projected surface mass density of the lens at some angular position θ' and $\vec{\beta}$ is the (unobservable) angular position of the source on the sky. This is known as the "thin lens approximation." It is conventional to redefine the surface mass density in terms of the convergence κ

$$\kappa(\vec{\theta}) = \frac{\Sigma(\vec{\theta})}{\Sigma_{cr}} \tag{1.15}$$

where Σ_{cr} is the *lensing critical density* given by

$$\Sigma_{cr} \equiv \frac{c^2 D_s}{4\pi G D_d D_{ds}} \tag{1.16}$$

where D_s, D_d and D_{ds} represent the angular diameter distances from the observer to the source, observer to the deflector, and deflector to the source respectively. The critical density denotes the minimum surface density required to produce strong lensing, which I discuss in more detail in the next section. It is often convenient to define the deflector mass in terms of the *lensing potential*

$$\psi(\vec{\theta}) = \frac{1}{\pi} \int d^2\theta' \kappa(\vec{\theta'}) ln|\vec{(}\theta) - \vec{(}\theta')| \tag{1.17}$$

In this formalism the deflection angle and convergence can easily be extracted from the lensing potential by

$$\vec{\alpha}(\vec{\theta}) = \vec{\nabla}\psi(\vec{\theta}) \tag{1.18}$$

$$\kappa(\vec{\theta}) = \frac{1}{2}\nabla^2\psi(\vec{\theta}). \tag{1.19}$$

The formalism presented here is generally sufficient for purposes of time-delay cosmography, where the dominant source of lensing is a single galaxy at a cosmological distance from both the observer and the background source. However there are cases where this formalism is inadequate, such as weak lensing due to cosmic shear and in cases where a single background source is lensed by multiple objects. In such cases it is sometimes possible to treat the system in the multiple-plane formalism, where each lens is approximated as a thin lens. In practice this approach is challenging both due to its computational complexity and the fact that most mass along any line of sight is unobservable.

1.3.2 Strong and Weak Lensing

Generally speaking, gravitational lensing is divided into two primary regimes. Strong lensing involves a source being lensed into multiple images or an Einstein ring. This occurs when the path taken by the light from a background source passes within a region where $\kappa > 1$. Massive galaxies are the only type of strong lens considered in this thesis, but galaxy groups and clusters also commonly produce this effect.

Weak lensing refers to distortions of the apparent shapes of background objects as they pass through the Universe. Formally, this occurs in cases where $\kappa < 1$. In practice, most "weak lensing"

analyses involve mass concentrations significantly less than one. Virtually every visible galaxy is lensed, but these perturbations are generally much too small to be observed on individual objects. Instead, weak lensing is usually measured with correlation statistics of many objects in some region. This is an important cosmological tool as it can be used to trace the underlying large-scale distribution of mass in the universe. Whether or not an object appears lensed (and by how much) depends on the location of the observer with relation to the source and deflector. Strong lensing is a relatively rare effect that requires precise alignment of an observer, source, and the lensing structure.

1.4 Time-Delay Cosmography

Time-delay cosmography relies on the inherent asymmetry of real strong lens systems and the cosmological distance scales intrinsic to their geometry to place constraints on cosmological parameters, particularly H_0. When an object is lensed into multiple images, the light from these images will take different paths to the observer over different amounts of time. When the brightness of the background source varies, this "time delay" will be directly observable since the variations will appear in different images at different times. The background source in this analysis is usually a quasar. However recently this analysis has been performed with observations of the supernova Refsdal (Kelly et al., 2023), which bears the name of the Norwegian astrophysicist who first proposed this technique in Refsdal (1964).

The time delay for a given lens system is the result of two effects. The first is the intrinsic difference in path length between the various routes taken by light from the background object as it passes by the lens and (ultimately) to the observer. The second is the Shapiro delay, which is an inherent time delay observed in any physical process happening inside a gravitational potential. In general these are competing effects. A ray of light passing near the center of the lensing potential will travel less additional distance while experiencing a greater Shapiro delay than a ray of light passing farther from the center. Time delays induced by these effects range in length from hours to years, but are typically of order a few days to a few weeks for galaxy-scale lenses.

The amount of the delay depends on the geometry of the system and the primary mass itself. Given a single lens and a background source, the total time delay between two images located at angular positions θ_A and θ_B is given by

$$\Delta t_{AB} = \frac{D_{\Delta t}}{c}[\tau(\vec{\theta}_A, \vec{\beta} - \tau(\vec{\theta}_B, \vec{\beta})] \tag{1.20}$$

where τ is the Fermat potential

$$\tau(\vec{\theta}, \vec{\beta}) = \frac{|\vec{\theta} - \vec{\beta}|^2}{2} - \psi(\vec{\theta}), \tag{1.21}$$

and $D_{\Delta t}$ is the *time delay distance* given by

$$D_{\Delta t} = (1 + z_d)\frac{D_s D_d}{D_{ds}} \tag{1.22}$$

where z_d is the redshift of the deflector. This quantity is inversely proportional to H_0 (see Eqn. 1.10). Given a well-measured time delay and model of the lensing galaxy, it is therefore possible to invert this relationship to determine H_0.[4]

As with nearly every other scientific analysis, this story is complicated when data are actually gathered. Measuring time delays, for example, is challenging in practice because it requires monitoring the system over many months or even years to catch the relatively small changes in brightness. Similarly lens modeling is limited by the quality of data that is available for a given lens. Even with excellent data, lens modeling requires a number of assumptions about mass distributions in the lensing galaxies, though future surveys may be able to provide more empirical constraints. The work presented here focuses on one of these uncertainties, which derives not from the lens itself but from its environment.

1.4.1 The Mass Sheet Degeneracy and $\kappa_{\rm ext}$

There is an important challenge in lens modeling that significantly complicates the picture presented above. Suppose we have a model of a lens system mass distribution $\kappa(\vec{\theta})$ and source position $\vec{\beta}$ that successfully reproduces the available observables for the system such as image positions, magnification ratios, and time delays. We then introduce a family of models defined by

$$\kappa_\lambda(\vec{\theta}) = \lambda\kappa(\vec{\theta}) + (1 - \lambda) \tag{1.23}$$

$$\vec{\beta}_\lambda = \lambda\vec{\beta}. \tag{1.24}$$

[4]My advisor likes to remind me not to editorialize too much, but I must admit I find it mind-blowing that astronomers have managed to both figure this out theoretically *and* execute it in practice.

This family of models will reproduce the observables equally well, but will yield different values of the product $\Delta t \times H_0$:

$$H_0 \Delta t \rightarrow \lambda H_0 \Delta t. \tag{1.25}$$

This transformation is known as a "mass sheet transformation," and the associated degeneracy as the "mass sheet degeneracy." Conceptually this transformation can be thought of as the scaling of the given mass distribution, combined with the addition of a infinite mass sheet coplanar with the lens. An infinite mass sheet is clearly non-physical, but it is possible to build models with truncated mass sheets that still reproduce this degeneracy. Although it is not specific to time-delay cosmography, breaking this degeneracy is crucial to obtaining reliable constraints on H_0 from strong lensing.

The mass sheet degeneracy can be broken by measuring stellar kinematics of the lensing galaxy. In particular, the angular diameter distance to the deflector, D_d can be independently inferred by

$$D_d = \frac{1}{1+z_d} \frac{c\Delta t_{AB}}{\Delta \phi_{AB}} \frac{c^2 J(\xi_{lens}, \xi_{light}, \beta_{ani})}{(\sigma^P)^2} \tag{1.26}$$

where σ^P is the line-of-sight projected stellar velocity dispersion of the lensing galaxy, and J is a form of the Jeans equation which captures information about the mass and light distributions of the lensing galaxy. This independent inference of D_d can be used to mitigate the effects of the mass sheet degeneracy.

The mass sheet transformation can be broken down into an external and internal component. It is the external mass sheet transformation, denoted $\kappa_{\rm ext}$, which is the primary focus of this dissertation. There is a clear physical origin of such a transformation. As the light passes through the Universe, it will pass by many different mass structures which will affect its trajectory. These perturbations are not directly observable, but can bias the resulting cosmological inference on the order of a few percent. Specifically, if the value of $\kappa_{\rm ext}$ is known, it can be used to correct the inferred value of the time-delay distance by

$$D_{\Delta t} = \frac{D'_{\Delta t}}{1 - \kappa_{\rm ext}} \tag{1.27}$$

where $D'_{\Delta t}$ is the uncorrected value of the time-delay distance. This directly propagates to inferred

value of the Hubble constant by

$$H_0 = (1 - \kappa_{\rm ext}) H_0' \tag{1.28}$$

where H_0' is the value of the Hubble constant inferred from the primary lens mass model, time delays, and stellar kinematics.

Estimating $\kappa_{\rm ext}$ is difficult for a number of reasons, particularly the fact that the majority of mass along the line of sight is in dark matter. Two techniques have been applied to constrain this mass sheet in practice. The first uses galaxy number counts as tracers of the underlying mass distribution, and compares them to similar lines of sight in simulations to estimate $\kappa_{\rm ext}$ for a given lens system (e.g. Rusu et al., 2017; Buckley-Geer et al., 2020). This technique is the primary focus of this dissertation, and procedures for using this technique to estimate $\kappa_{\rm ext}$ will be introduced at length in Chapter 2

The second technique for estimating $\kappa_{\rm ext}$ is with weak lensing (e.g. Tihhonova et al., 2018). Given high-resolution imaging data of the field, it is possible to estimate the shear field by cross-correlating the shapes of background galaxies. This shear field can be used to reconstruct a convergence field of the system, which is then compared to simulations to estimate the external convergence. This technique makes good physical sense, as it attempts to model the actual distribution of mass in the lens in field. However in practice it does not achieve meaningfully better constraints than the first method, while requiring higher-quality data and more astronomer hours. This technique was not explored during the preparation of this dissertation, but may be worth revisiting in the future.

1.5 Astronomy at Scale: Challenges and Principles

Although this dissertation is about astronomy, it is also about software. The $\kappa_{\rm ext}$ analysis discussed throughout this dissertation relies heavily on large galaxy surveys spread out over large areas of sky. Developing software that can efficiently interact with these survey datasets is a significant challenge in its own right.

The amount of data available to the astronomy community has expanded by orders of magnitude in the last decade or two, and will continue to expand as next-generation surveys come online. This presents a number of scientific challenges. By and large, the community is responding well to these

challenges. Machine learning models, which are capable of crunching through data at tremendous speed, have become a foundational tool in the discipline. In time delay cosmography, the results from many different systems have been combined into hierarchical Bayesian models, which use all information available from many systems to produce a single inference. These sorts of techniques are being adopted elsewhere too, as systems that were once unique are now just a single member of a large population.

But these analysis techniques implicitly rely on robust underlying data infrastructure that is capable of serving the needs of many different astronomers. This data infrastructure receives much less attention in the community. Operations teams for individual surveys must and do build sophisticated infrastructure for managing and processing data as it arrives from the observatory. However the tools necessary for the *usage* of the produced data at its full scale by downstream astronomers is meaningfully lacking.

Modern astronomical datasets are large by any standard, but they are by no means unprecedented. The LSST team estimates that the total cumulative volume of processed data delivered over the 10-year survey period will reach approximately 500 PB. However nearly a decade ago Amazon Web Services debuted a service for moving *exabyte* scale datasets betweeen data centers, and as far back as 2011 engineers at CERN were working on systems to store and process exabyte-scale data produced by the LHC. These facts are not meant to downplay the significance of the challenge of working at this scale, but only to point out that many of the challenges of working with these datasets are well understood by systems engineers in the scientific community and the private sector.

Throughout my work, I have focused on what I call the "software-engineering approach to astronomy." This approach involves clearly separating scientific problems from engineering ones, and developing the skills necessary to tackle these engineering problems effectively. It focuses on producing reusable, extensible software that *assumes* an analysis will be performed many times over a large data volume, but does not place unnecessarily tight constraints on what form that analysis may take. This approach does not attempt to downplay the importance of exploratory analysis or suggest that all astronomers should focus their time on becoming highly-competent software engineers. Instead, it recognizes that the long-term sustainability of computational science is an important goal that requires real expertise in topics *outside* astronomy.

The scientific work presented in Chapter 3 of this dissertation demonstrates the power of this

approach. In that chapter, I build a population model of a large number of systems. Core to this analysis is my ability to model the individual systems in this sample in a consistent and reliable way. This does not eliminate the possibility of unaccounted-for biases in the measurement, or an issue in the population model itself. But it does mean that these biases likely apply across the entire sample, and changes made to mitigate these biases can be applied across the entire quickly and reliably.

The tools used to perform these analyses were developed using the exact approach discussed above. I recognized the set of computational needs that would be consistent across my work (and likely others), and built tools designed to meet these needs while being flexible regarding the specifics. The first tool I produced was `heinlein`, a data management library that provides a single unified Python interface to data from some underlying survey. It is built in particular to accelerate cone searches, where the dataset is queried for all data that falls within some distance of a given point. This core ability is useful in a wide variety of contexts. I followed this with `cosmap`, which is built on top of `heinlein` and provides a consistent interface for defining analyses that repeatedly sample data from a large sky survey. `cosmap` handles parameter validation, I/O, logging, and the flow of data through the analysis. `cosmap` makes no assumptions about what computations are performed on the data, allowing for a wide variety of use cases.

While the *motivation* for building these tools was scientific, the *approach* taken to actually producing them (and the decision about where one ends and the other begins) was based on software engineering. I discuss some of these decisions more in Chapter 4.

1.6 This Dissertation

The work presented in this dissertation has two main aims. First, to develop software infrastructure for estimating κ_{ext} rapidly and reliably. And second, to develop the techniques necessary to model populations of lines of sight, which is useful for time-delay cosmography and may provide insight into the strong lens selection function.

The software developed in the preparation of this dissertation has undergone and continues to undergo updates and improvements. The `lenskappa` package discussed in Chapter 2 has been almost entirely replaced by the `cosmap` package discussed in Chapter 3. This is a good thing, as my understanding of the problem and the tools needed to solve it have improved significantly over the

past several years, and incorporating that understanding into software has significant benefits. As such, the discussion of software in Chapter 2 should be viewed primarily as a historical account. Secondly, it is worth noting that the basic techniques for estimating $\kappa_{\rm ext}$ for a single lens system existed well before I started working on this dissertation, and much of the work in Chapter 2 is aimed at reproducing this analysis on a new system. As such, it also does not represent our most up-to date understanding of our scientific techniques. However the core of the analysis remains unchangedin Chapter 3. Inventing a new set of techniques is not the goal of this work. Instead, the goal is to push our techniques into the era of big-data astronomy by developing next-generation software tools and population analysis techniques. Additionally, certain aspects

Chapter 2 introduces the techniques I use to estimate $\kappa_{\rm ext}$ in depth and applies them to the SDSSJ0924 lens system. As of the time of this writing, work on a full cosmographic analysis of SDSSJ0924 incorporating the result from this chapter is ongoing. This chapter also includes a discussion of the basic computational techniques used in the analysis, and some more general principles of computation and its role in astronomy in the coming decades.

Chapter 3 presents a population model of the lines of sight to 25 strong lenses in the Strong Lensing Legacy Survey (SL2S). I introduce a new statistical method for modeling these lines of sight that has some implications for our understanding of the underlying physics. This analysis uncovered a number of shortcomings in the software developed for Chapter 2 which are discussed. We find that the lines of sight in the sample are drawn from a sample that is overdense as compared to the population of all lines of sight in the Universe. Although this result is expected from previous work, this chapter represents the first time observational data has been used to estimate $\kappa_{\rm ext}$ at a population level.

In Chapter 4, I discuss in depth the software design principles I have used to build the tools used to perform the analyses from the previous two chapters. I demonstrate how these implementations allow users to iterate on their techniques more rapidly than would otherwise be possible, and be more confident in their results do to the reliability and consistency of the underlying system. I discuss the importance of these techniques to the long-term sustainability of astronomy in the era of big data, and argue that the community should invest in developing individuals and teams who are technically focused to work alongside their science-oriented peers.

Chapter 5 presents conclusions drawn from the preceding chapters, and looks forward to the

future of time-delay cosmography in general and the line of sight analysis in particular. I also discuss the future of my software and some of my longer-term vision for the kinds of analyses it may be able to enable.

Chapter 2

Practical Techniques for Estimating External Convergence of Strong Gravitational Lens Systems and Applications to the SDSS J0924+0219 System

This chapter has been published in Astronomy & Astrophysics as a TDCOSMO Collaboration paper.

2.1 Abstract

Time-delay cosmography uses strong gravitational lensing of a time-variable source to infer the Hubble constant. The measurement is independent from both traditional distance ladder and CMB measurements. An accurate measurement with this technique requires considering the effects of objects along the line of sight outside the primary lens, which is quantified by the external convergence ($\kappa_{\rm ext}$). In absence of such corrections, H_0 will be biased towards higher values in overdense fields and lower values in underdense fields. We discuss the current state of the methods used to account for environment effects. We present a new software package built for this kind of analysis and others that can leverage large astronomical survey datasets. We apply these techniques to the SDSS J0924+0219 strong lens field. We infer the relative density of the SDSS J0924+0219 field by computing weighted number counts for all galaxies in the field, and comparing to weighted

number counts computed for a large number of fields in a reference survey. We then compute weighted number counts in the Millennium Simulation and compare these results to infer the external convergence of the lens field. Our results show the SDSS J0924+0219 field is a fairly typical line of sight, with median $\kappa_{\rm ext} = -0.012$ and standard deviation $\sigma_\kappa = 0.028$.

2.2 Introduction

One of the most important problems in modern cosmology is the so-called Hubble Tension: the name given to an apparent discrepancy between the value of the Hubble Constant (H_0) inferred from the Cosmic Microwave Background assuming the standard Λ-CDM cosmological model (e.g. Aghanim et al., 2020b), and the value inferred from Cepheid-calibrated Type Ia supernovae. (e.g. Riess et al., 2021). The solution to this discrepancy may involve unknown systematics, new physics, or some combination of the two. Work on the resolution is ongoing, but having independent methods for inferring the value of the Hubble constant is critical for solving the problem. Such methods include using the tip of the red giant branch to calibrate supernovae,(Freedman et al., 2019) and Baryon Acoustic Oscillations and Big Bang Nucleosynthesis (BAO+BBN) (Cuceu et al., 2019), among others. Here we consider Time Delay Cosmography, which uses strong gravitational lensing of time-variable sources (usually quasars) to infer the value of H_0.

TDCOSMO is an international collaboration which aims to use strong gravitational lensing to infer the value of H_0 with sub-percent precision. A typical lens in the TDCOSMO sample involves a source quasar being strongly lensed by a foreground galaxy, producing four images. When the quasar's luminosity varies, this variation appears in each of th eimages at different times. The "time delay" between two images depends on the mass distribution of the lens, and the *time-delay distance*:

$$D_{\Delta t} = (1 + z_{\rm d}) \frac{D_{\rm d} D_{\rm s}}{D_{\rm ds}} \propto \frac{1}{H_0} \tag{2.1}$$

where $z_{\rm d}$ is the redshift of the lensing galaxy (or "deflector"), and $D_{\rm d}$, $D_{\rm s}$, and $D_{\rm ds}$ are the angular diameter distances to the deflector, source, and from the deflector to the source, respectively. Given a model of the lens and a measurement of the time delays, it is possible to infer the time-delay distance and therefore H_0.

When analysing a strong lens, there are a number of sources of uncertainty that propagate to

the final inferred value of H_0 (see Millon et al., 2020, for a more complete overview). Among these are so-called environmental effects: gravitational bodies besides the primary lens that affect the lensing observables and, by proxy, the inferred value of H_0. Accounting for these effects is crucial in improving the precision of the inferred measurement. Broadly speaking we treat weak perturbers differently from strong ones, with the difference being determined with the "flexion shift" formalism (see for example Buckley-Geer et al., 2020; Sluse et al., 2019; McCully et al., 2017).

In this paper, we focus on the techniques for inferring the cumulative effect of all weak perturbers along the line of sight to the source quasar. In principle, an "ideal" analysis would include these perturbers in a complete mass model of the system, rendering the analysis we do here unnecessary. While this may be possible for a few systems where high-quality spectroscopic data of the perturbers is available, it does not scale well to the large number of strong lensing systems we aim to analyze in the future. Instead of building a complete model, we compare the field of interest to some suitably large reference field, with the goal of estimating the relative density of the field as compared to the universe at large. This is a purely statistical analysis. In this paper, we will discuss the current state of this technique, apply it to the field around the SDSS J0924+0219 lens system (hereafter J0924, see section 2.5.1), and discuss how it might be iterated upon in the future. The cumulative effect of all weak perturbers is parameterized by the *external convergence*, denoted $\kappa_{\rm ext}$.

While controlling uncertainties on individual lens systems is crucial, the precision of the H_0 measurement can be improved by including many lenses in the final inference. Of course, this is much easier said than done as the amount of work required to fully analyze a single lens is significant. However in the modern era of data-driven astronomy, the amount of available data is increasing by orders of magnitude and developing tools for efficiently analyzing these datasets is a top priority. To that end, we introduce `lenskappa`, a new package designed specifically for the types of environment analysis discussed in this paper, but with broader applications. `lenskappa` is built on top of `heinlein`, a data management library designed for use with large astronomical survey datasets. At present `heinlein` and `lenskappa` support Subaru Hyper Suprime Cam Strategic Survey Program (HSC-SSP; Aihara et al., 2019), the Dark Energy Survey (Dark Energy Survey Collaboration, 2005), and the CFHT Legacy Survey (Gwyn, 2012).

This paper is organized as follows: In Section 2, we outline the fundamentals of Time Delay

Cosmography and the basic process used to compute a value of $\kappa_{\rm ext}$ for a generic lens system. In Section 3 we discuss the datasets used in our analysis, and introduce `lenskappa`. In Section 4, we report the results of our analysis of the J0924 field obtained using `lenskappa`. In Section 5 we discuss our results and look forward to later work.

2.3 Summary of the technique

At a high level, gravitational lensing is the result of the underlying mass distributions of the universe, with greater concentrations of mass resulting in more significant lensing. In general, it is difficult to assess the impact of any one mass structure on the image of some background object. Strong lenses are naturally an exception, which occurs when a single, high-mass object falls on (or nearly on) the axis drawn between an observer and some background source. However we are seeking to understand the total lensing effect of many additional objects, each producing a tiny effect on the lensing observables. As light passes through the universe, the amount of lensing it undergoes will be determined by the relative concentration of mass along its full path of travel. Our analysis therefore seeks to infer the effects of the mass distributions in our line of sight by comparing it to many lines of sight in the universe. By determining how close this line of sight is to the average line of sight in the universe, we can place constraints on its impact on our primary lens observables. In this section, we introduce the basics of time delay cosmography and describe the technique we use to estimate $\kappa_{\rm ext}$ for a generic lens system. For a more complete overview, we refer the reader to Treu & Marshall (2016) and Birrer et al. (2022).

2.3.1 Fundamentals of Time Delay Cosmography, and Strong vs. Weak Lensing

Time delay cosmography focuses on strong lensing of time-variable sources, usually a quasar. As the luminosity of the source varies, this variation appears in each of the several images of the source, but not at the same time. The time delay between any two images can be written as follows:

$$\Delta t_{ab} = \frac{D_{\Delta t}}{c}\left[\frac{(\overrightarrow{\theta_a} - \overrightarrow{\beta})^2}{2} - \frac{(\overrightarrow{\theta_b} - \overrightarrow{\beta})^2}{2} - \psi(\overrightarrow{\theta_a}) + \psi(\overrightarrow{\theta_b})\right], \tag{2.2}$$

where $\overrightarrow{\theta}$ represents the angular position of an image on the sky, $\overrightarrow{\beta}$ represents the actual (unobservable) angular position of the source on the sky, and ψ is the scaled lensing potential. The first

two terms are a result of the different distances traveled by the light for the two images, while the later two terms are the difference in the Shapiro time delay. The goal of a complete cosmographic analysis is to infer the *time-delay distance* (see eq. 2.1). Using measurements of the time delay combined with a robust model of the lensing galaxy it is possible to measure the time delay distance and, therefore, the Hubble constant.

The multiple images observed in such a system are en example of *strong gravitational lensing*. Qualitatively, strong lensing is any lensing which produces multiple images of some background object. Quantitatively, strong lensing occurs whenever the local density of the lens is greater than the *lensing critical density:*

$$\Sigma_{cr} = \frac{c^2 D_s}{4\pi G D_{ds} D_d} \tag{2.3}$$

For a given perturber, the *convergence* at a given location is defined as the local density in units of the critical density:

$$\kappa(\overrightarrow{\theta}) = \frac{\Sigma(\overrightarrow{\theta})}{\Sigma_{cr}} \tag{2.4}$$

For $\kappa < 1$, strong lensing does not occur. Instead, mass distributions with $\kappa < 1$ result in magnification or demagnification of the image of the background object. This is the situation for the perturbers along the line of sight in our system that are not included in the primary lens model, though we note that some perturbers that meet this criterion *are* included in the mass model based on their flexion shift (see section 2.3.2), and therefore excluded from our statistical analysis. However, the effect of the pertubers we *do* include in the estimate discussed here is not directly observable because the actual angular size of the background object is not known.

We quantify the cumulative effect of all perturbers (not including those incorporated into the primary lens model) and voids along the line of sight with the external convergence, $\kappa_{\rm ext}$. Conceptually, the value of $\kappa_{\rm ext}$ is the convergence of a mass sheet which, if placed coplanar to the primary lens, would produce the same magnification or demagnification as the perturbers do collectively. The external convergence is defined relative to that of a line of sight where the mass distribution is smoothly distributed with a density equal to the global mass density of the Universe. Therefore positive values of $\kappa_{\rm ext}$ represent lines of sight that are overdense with respect to the overall density of the Universe, while underdense lines of sight have negative values.

These perturbers can also produce shear, denoted $\gamma_{\rm ext}$, which results in stretching and distorting of the image of the source. However, this effect can be estimated in the primary lens model, as it affects the location of the the images and the shape of the Einstein ring (if present). We use the $\gamma_{\rm ext}$ constraint from the lens model in our analysis as an additional constraint (see section 2.3.6)

Assuming $\kappa_{\rm ext}$ can be measured, it serves as a correction factor to the computed value of the time delay distance:

$$D_{\Delta t} = \frac{D'_{\Delta t}}{1 - \kappa_{\rm ext}}, \tag{2.5}$$

where $D'_{\Delta t}$ represents the uncorrected value of the time delay distance. This propagates directly to the inferred value of the Hubble constant by

$$H_0 = (1 - \kappa_{\rm ext}) H'_0 \tag{2.6}$$

where H'_0 is the value of the Hubble Constant inferred before correcting for the environment. We see therefore that in the absence of the appropriate corrections, the inferred value of Hubble constant would be biased towards higher values in an overdense field, and lower values in an underdense field.

We now review the techniques we use to compute the value of $\kappa_{\rm ext}$ for a generic lens system.

2.3.2 Relevant Perturbers in the Line of Sight

To start, we identify objects along the line of sight that contribute to $\kappa_{\rm ext}$. We separate objects into *strong* and *weak* perturbers, using an operational definition discussed below. Strong perturbers are generally close to the center of the field or a galaxy group (see for example Sluse et al., 2019; Fassnacht et al., 2006). Strong perturbers are included explicitly in the mass model of the lens, while weak perturbers are treated statistically. To separate these, we use the *flexion shift* formalism, first proposed in McCully et al. (2017). The flexion shift of an object is given by

$$\Delta_3 x = f(\beta) \times \frac{(\theta_E \theta_{E,p})^2}{\theta^3} \tag{2.7}$$

where θ_E and $\theta_{E,p}$ are the Einstein radii of the main lens and perturber respectively, and θ is their angular separation. The quantity $f(\beta)$ is given by

$$f(\beta) = \left\{ \begin{array}{ll} (1-\beta)^2, & \text{if } z > z_{\rm d} \\ 1, & \text{if } z < z_{\rm d} \end{array} \right\} \tag{2.8}$$

where

$$\beta = \frac{D_{\rm dp} D_{\rm s}}{D_{\rm p} D_{\rm ds}} \tag{2.9}$$

and where $D_{\rm dp}$, $D_{\rm s}$, $D_{\rm p}$, and $D_{\rm ds}$ are the angular diameter distances from the deflector to the perturber, to the source/perturber, and from the deflector to the source. The flexion shift roughly measures the perturbations to the images of the source due to 3rd order terms from the perturber. Ultimately, what constitutes a "weak" or "strong" perturber is somewhat arbitrary, but there this is a clear trade-off between the improvement from including a particular perturber in the mass model and the amount of work required to do so. McCully et al. (2017) recommends using $\Delta_3 x = 10^{-4}$ arcsec as the cutoff between strong and weak perturbers to ensure a $< 1\%$ bias on H_0.

2.3.3 Comparison Datasets

The first step in determining the value of $\kappa_{\rm ext}$ is comparing the field of interest to a large number of lines of sight from some large reference field. This comparison gives us an empirical estimate of the relative matter density of the lens field as compared to all lines of sight in the universe. The reference field should be large enough to avoid sampling bias. For a small comparison field (on the order of a few deg^2) statistical overdensities or underdensities may occur (see for example Fassnacht et al., 2010). However modern survey datasets are available which cover hundreds to thousands of square degrees, allowing us to use a sufficiently large comparison field to avoid sampling bias (see section 2.5.2 for a discussion of our choices for this analysis). Ideally the data for the reference field and lens field are taken by the same instrument and processed by the same analysis pipeline. This turns out to be the case for the analysis of J0924 discussed later in this paper, but will not be true in general. At a very minimum we seek data with at least one band in common, with deep enough observations to produce meaningful results. When comparing the survey dataset to our line of sight, we set a magnitude cut and remove any objects from both datasets fainter than this cut. Previous work (see Collett et al., 2013) has suggested that $i < 24$ represents a good limit, as setting a fainter limit does not appear to meaningfully impact the results of this style of analysis. This limit

is also bright enough to be well above the detection limits of modern sky surveys, ensuring reliable photometry.

Once the appropriate data are in hand, it is important to consider which objects should be used in the comparison. It is important to set a magnitude cut that will remove objects too close to the detection limit of the instrument for photometry to be reliable, while still leaving enough data to make robust estimates of the relative density. However setting too bright of a limit will result in having too few objects to compare to.

Additionally, we cut out all objects with a redshift greater than the redshift of the source quasar, as these objects will not affect the path of the light as it travels from the quasar to our telescopes.

2.3.4 Weighted Number Counts of Lens Field

Because the value of $\kappa_{\rm ext}$ cannot be directly measured, we first define tracer quantities that can be computed directly from the available data. By comparing the value of these quantities in the lens field to the value of the identical quantity computed for a large number of reference fields, we obtain an empirical estimate of the relative density of the line of sight of interest. As a first approximation, we expect the greatest contribution to $\kappa_{\rm ext}$ from massive objects close to the center of the line of sight. The primary mass contribution in any line of sight will be dark matter halos, but these are not directly observable. To quantify the contribution from these weak perturbers, we compute weighted number counts of the visible structures (i.e. galaxies) in the lens field as compared to a large number of reference fields. This technique has been used extensively in previous work (Fassnacht et al., 2006; Suyu et al., 2010; Greene et al., 2013; Buckley-Geer et al., 2020) . To do this, we select a region of interest around the lens and compare it to a large number of identically-shaped fields selected at random from the reference survey. At each step, we compute the ratio of the weighted number counts for the galaxies in the lens field to the identical statistic computed in the given reference field. For the given step, the value of the weight is therefore:

$$W_i = \frac{\sum_j w_{j,\rm lens}}{\sum_j w_{j,i}} \tag{2.10}$$

Where i indexes the reference fields, j indexes the galaxies in a given field, and w is the value of the weighted statistic for the given galaxy.

Following Rusu et al. (2017) we also consider a second style of weighting that improves our

Name	Value	Symbol
Number Count	$w_j = 1$	w_n
Inverse Distance	$w_j = 1/r_j$	$w_{1/r}$
Potential	$w_j = m_j/r_j$	w_p
Redshift	$w_j = z_s \cdot z_j - z_j^2$	w_z
z/r	$w_j = w_{z,j}/r_j$	$w_{z/r}$

Table 2.1 Weighted number counts considered in this work

results. Instead of summing the value of weights for all objects in the field, we instead compute the weight for a given field as $w_{i,\mathrm{meds}} = n_i \overline{w_j}$ where n_i is the number of galaxies in the given reference field and $\overline{w_j}$ is the median value of the weight for all galaxies in the reference field. Doing this helps avoid situations where single objects dominate the sum in a particular line of sight. This is especially important for weights involving stellar mass and the inverse separation. In this scheme, the value of the weight for the given reference field is therefore

$$W_{i,\mathrm{meds}} \equiv \frac{n_{\mathrm{lens}} \overline{w_j}_{,\mathrm{lens}}}{n_i \overline{w_{j,i}}} \tag{2.11}$$

There has been discussion in the literature about which are the best weights to consider for the most robust determination of κ_{ext} (Greene et al., 2013; Rusu et al., 2017, 2019). In this work, we only consider a subset of the weights considered in previous works (see Table 2.1).

2.3.5 Weighted Number Counts in Simulated Data

In order to determine the posterior distribution of κ_{ext} for the given system, it is necessary to compare the weighted number counts to similar counts obtained from a reference field for which κ_{ext} is known. We use a simulated dataset for this purpose. The simulation must contain several components in order to be suitable for this analysis. First, it must contain catalogs of galaxies with known luminosity and redshift. Second, values of the external convergence must be measured at a suitably large number of points to be representative of the universe at large. Rusu et al. (2017) examined possible biases from inferring κ_{ext} using the number counts method in the Millennium Simulation, and found these methods produced a good estimate of κ_{ext}. We discuss our choice of simulation further in section 2.5.3. Because this technique involves ratios, much of the dependence

on the simulation's underlying cosmological parameters should cancel out. However, ensuring this would require a second simulation with the attributes described. An exciting development in this space are the initial results from the MillenniumTNG (Hernández-Aguayo et al., 2022). Full weak-lensing convergence maps are planned, but not yet available.

2.3.6 From Weighted Number Counts to $\kappa_{\rm ext}$

We now have weighted number counts for the lens field itself and for a large number of fields in a simulated dataset, each of which is associated with a value of $\kappa_{\rm ext}$. We seek to compute $p(\kappa_{\rm ext}|\mathbf{d})$: the probability distribution of $\kappa_{\rm ext}$ given the data. We can replace this with the joint probability distribution of $\kappa_{\rm ext}$ *and* the data as follows:

$$p(\kappa_{\rm ext}|\mathbf{d}) = \frac{p(\kappa_{\rm ext}, \mathbf{d})}{p(\mathbf{d})} = \int dW_q \frac{p(\kappa_{\rm ext}, W_q, \mathbf{d})}{p(\mathbf{d})} \tag{2.12}$$

After some work it can be shown (see Rusu et al., 2017)

$$p(\kappa_{\rm ext}|\mathbf{d}) = \int p_{\rm sim}(\kappa_{\rm ext}|W) p(W|\mathbf{d}) \prod_i dW_i \tag{2.13}$$

Where $p(W|\mathbf{d})$ is the probability distribution of the given weighting scheme given the data, and $p_{\rm sim}(\kappa_{\rm ext}|W)$ is the probability distribution of $\kappa_{\rm ext}$ in the simulated dataset, given a particular value of the weight. Here, it is implicitly assumed that the simulated dataset is the correct prior for the observable universe. In particular, we assume that it accurately maps values of weights to values of $\kappa_{\rm ext}$

Additionally, a constraint on $\kappa_{\rm ext}$ based on $\gamma_{\rm ext}$ can be included, which accounts for the expected correlation between these two values. In general, γ is a two-dimensional vector on the plane of the sky, but we use only the overall magnitude in our analysis. $\gamma_{\rm ext}$ is a parameter that can be fitted in the primary mass model of the lens. We use $\gamma_{\rm ext}$ just as we would a weighted number count distribution. For any range of values of $\gamma_{\rm ext}$, we can construct a histogram of the value of $\kappa_{\rm ext}$ for all lines of sight with values of $\gamma_{\rm ext}$ that fall in this range. We then weight the contribution from these lines of sight by $p(\gamma|\mathbf{d})$, which is the posterior on γ inferred from the mass model.This provides a prior on $\kappa_{\rm ext}$ that can meaningfully affect the final result. In section 2.6, we present results both with and without using $\gamma_{\rm ext}$ as a constraint.

In previous work (e.g. Rusu et al., 2017), this full probability distribution for each weight $p(W|\mathbf{d})$

was replaced by a normal distribution centered on the median of the full weight distribution, with a width determined by examining measurement uncertainties. In practice, this width was much smaller than the width of the actual distribution. This is cheaper computationally, but ignores covariance between the various types of the weights. While the increase in computation time is significant, the majority of the important decisions in the analysis are made when we compute weighted number count ratios, and doing an integration of the full distribution does not meaningfully increase our time-to-result.

Since all the individual weights are measured at the same sequence of randomly-drawn fields, we can construct a full m-dimensional probability distribution $p(W_s|\mathbf{d}) = p(W_n, W_{1/r}, ...|\mathbf{d})$ where m is the number of weights being considered. We then explore this probability distribution when implementing the formalism described above. Formally, the posterior on $\kappa_{\rm ext}$ becomes:

$$p(\kappa_{\rm ext}|\mathbf{d}) = \int p_{\rm sim}(\kappa_{\rm ext}|W_s)p(W_s|\mathbf{d})d^m W_s p \tag{2.14}$$

When computing this quantity, we split the m-dimensional probability distribution into 200^m m-dimensional bins. We have also tested this procedure 100^m bins and see consistent results for the J0924 field. With significantly more bins, the computational time balloons and the number of fields in each bin drops significantly, even near the center of the distribution. The value of $p(W_s|\mathbf{d})$ is simply the number of lines of sight in the reference survey that fall in this bin divided by the total number of lines of sight being considered. Given the large numbers of lines of sight we consider, the distributions are smooth and it may be possible to model them explicitly and explore the distribution with an MCMC, but we save this for a future analysis. The exception to this is the pure unweighted number counts in a $45''$ aperture, due to the fact that the "weight" is integer valued and the number of galaxies in the aperture is relatively modest.

For $p_{\rm sim}(\kappa_{\rm ext}|W_s)$ we construct a histogram of the measured values of $\kappa_{\rm ext}$ for all lines of sight from the simulated dataset that fall within the given bin, normalized by the number of lines of sight in the bin. Without this normalization, $\kappa_{\rm ext}$ values near the mean of the simulation will always be weighted more heavily because there are comparatively more of them. Our choice of simulation is discussed in section 2.5.3

The inputs to our analysis code include the full weight distributions for the lens field, the weights computed from the simulated dataset, and the $\kappa_{\rm ext}$ maps for the simulated dataset for the source

redshift. We iterate over the probability distribution discussed above, computing a histogram of the values of $\kappa_{\rm ext}$ in each m-dimensional bin. The overall histogram is the sum of these histograms, weighted by the value of the weight distribution in that bin.

2.3.7 Comparison to Other Methods

There are other methods for estimating $\kappa_{\rm ext}$ which have been explored as discussed below. Ultimately many of these techniques involve a trade-off between speed of analysis and precision of the final result. Our goal is explicitly to design and implement techniques that allow us to analyze hundreds or even thousands of lens systems in a reasonable amount of time, with the goal of combining full results (including lens modeling, and time delay measurements) from many lenses to make some final statement about the value of H_0. Because of this, a technique with slightly less constraining power for a single lens is tolerable if it can be performed and iterated on rapidly. A few other techniques for estimating $\kappa_{\rm ext}$ are discussed below. While all show promise, and are interesting for their own sake, none show a significant enough improvement to justify the increased time and complexity of analysis, at least in the context of our stated large-scale goals.

2.3.7.1 Weak Lensing Analysis

While the value of κ at any given point on the sky is not observable, γ *can* be measured by looking a distortions in the shapes of galaxies in the line of sight. Assuming γ can be measured, techniques such as those presented in Kaiser et al. (1995) can be employed to reconstruct the underlying mass distribution. However this analysis requires extremely high quality data about the morphology of galaxies in the line of sight, as γ is measured from the extremely small distortions that are present in the images of the galaxies. Obtaining such high quality data, as well as the overhead of analyzing it, represents a significant bottleneck. This analysis was performed in Tihhonova et al. (2018) on the same line of sight analyzed in Rusu et al. (2017). Results between the two methods are consistent, with only modest improvement to precision for the weak lensing analysis.

2.3.7.2 Explicit Modeling

Building an explicit mass model of all perturbers (or at least, some larger sample) is attractive from a pure astrophysics perspective. This approach was explored in McCully et al. (2017) and demonstrated some success. However, the analysis assumed lines of sight with extensive spectroscopic coverage. While this may be true for some lines of sight, spectroscopy is expensive

in time and resources and it is not obvious that the improvement is significant enough to justify this. Furthermore, explicitly modeling a line of sight requires making assumptions about the host halos of the galaxies, a potential source of additional bias.

2.3.7.3 Machine Learning Methods

An additional interesting approach using Bayesian graph neural networks (BGNNs) is presented in Park et al. (2023). This approach was compared to a toy version of the analysis discussed here, using only a single summary statistic instead of a combination of several. The BGNN technique demonstrated greater precision and accuracy over using a single summary statistic on simulated data, however we estimate the difference would be much less significant if the comparison was done against the full line of sight analysis discussed in this work. That said, a more detailed comparison between these techniques would be welcome in the future. Park et al. (2023) also demonstrated a meaningful bias for both techniques in more extreme fields ($\kappa < -0.05$ and $\kappa > 0.06$) due to a lack of similarly extreme fields in simulated datasets to compare to. This is worth examining further, but does not immediately suggest significant improvements from using the BGNN techniques, especially for extreme fields.

2.4 Datasets, automation, and `lenskappa`

As with many areas of astronomy, time-delay cosmography is grappling with datasets that are growing at unprecedented rates. There are dozens of known quad lenses which may be suitable for cosmographic analysis, but a full analysis has only been completed on a small fraction of them. Techniques for increasing the rate of analysis are therefore crucial to the continued success of the technique.

While there are many open science questions in Time Delay Cosmography, much of the solution to the rate problem lies in software engineering rather than astronomy. When building for this kind of of analysis, we keep three key questions in mind:

- What is the minimum amount of data required to complete a particular analysis step with sufficient precision?
- How much of the analysis can realistically be automated?
- Do we expect the analysis techniques to change significantly in the future?

The answer to these questions may not be independent. For example, a pipeline that uses less data at the expense of increasing uncertainties on individual systems may be tolerable if it significantly increases the rate at which these systems can be analyzed. The third question is also important. Writing flexible software packages that can easily be updated as analysis techniques evolve usually increase the time to first result, but substantially decrease *average* time to result in the long run.

Among the various steps required to fully analyze a lens system, the number counts technique discussed in this paper is likely the most straightforward to automate. It is mostly statistical, and the most difficult computational challenge is efficiently filtering a large dataset by location. Furthermore the survey datasets used in this analysis are quite robust, and many of the lens systems fall within one or more survey footprints. This makes the first question a non-issue, at least for a significant fraction of the lenses.

To that end, we introduce `lenskappa`[1] and `heinlein`[2]. The goal of `lenskappa` is to build a tool capable of automating environment analysis to the greatest extent possible, while still providing sufficient flexibility to allow us to iterate on our current methods. `lenskappa` is in turn built on top of `heinlein`, which serves as a high-level interface to locally stored astronomical datasets. When computing weighted number counts in `lenskappa`, the core weighting loop consists of:

1. Select a region of interest from a large survey.
2. Retrieve object catalog and auxiliary data for the region.
3. Compute interesting quantities, using the data retrieved for the region.

A single iteration of this loop is represented in Figure 2.1

`heinlein` handles the middle step of this loop. It provides high-level routines for storage, retrieval, and filtering of large survey datasets, as well as intuitive interfaces for interaction between data types (for example, applying a bright star mask to a catalog). `heinlein` can perform a 120" cone search in the HSC dataset in around 1.5 seconds, without requiring the data be pre-loaded into memory. For later queries of nearby locations, the speed is improved by more than an order

[1] https://github.com/PatrickRWells/lenskappa
[2] https://github.com/PatrickRWells/heinlein

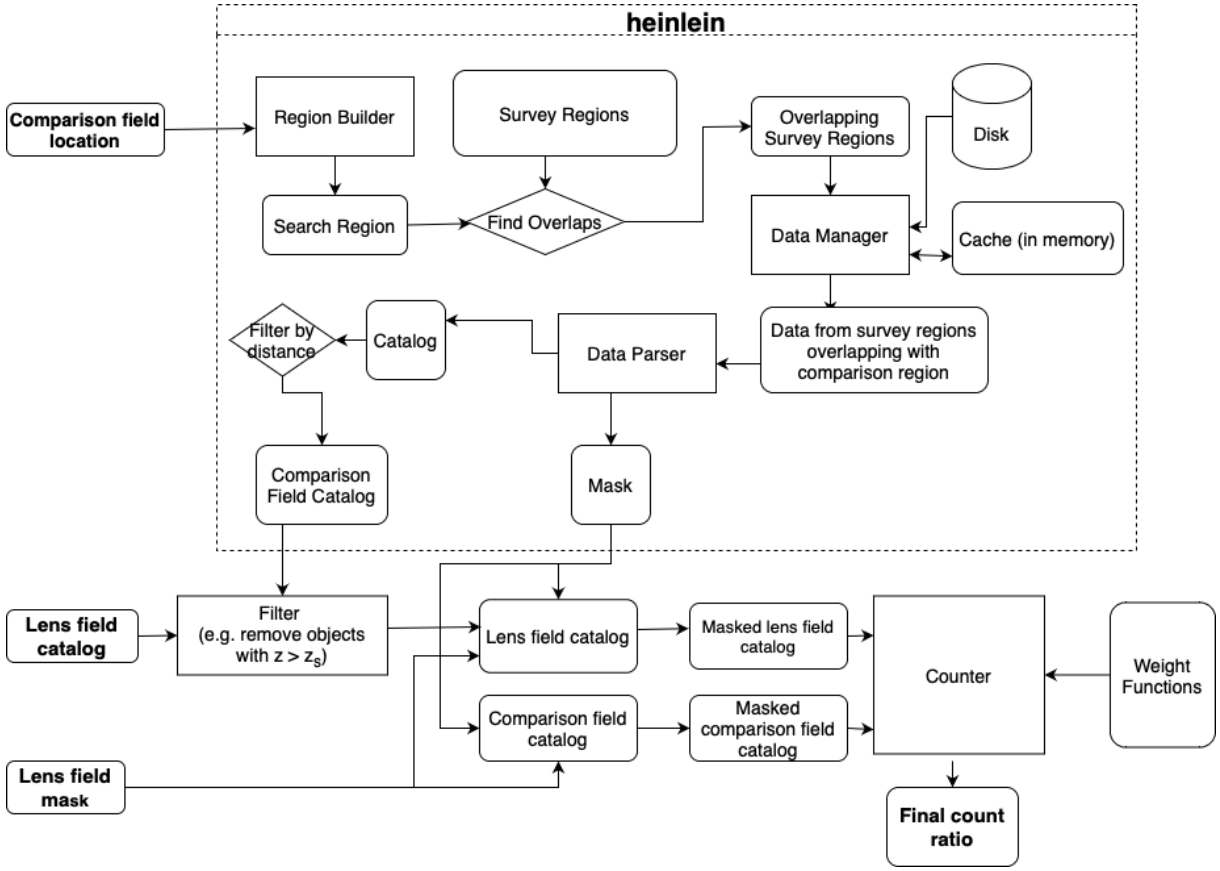

Figure 2.1 Flowchart representing the process of computing a weighted count ratio given a location in a reference survey. Work done in `heinlein` is contained within the dotted box.

of magnitude through caching. This makes `heinlein` suitable both for interactive use and for the kind of analysis done in `lenskappa`.

With data retrieval optimized, `lenskappa` focuses on allowing users to design and implement analyses that operate on large swathes of the sky. The techniques described in this paper, for example, could easily be adapted to build mass maps of the universe as seen in these surveys. However the goal is to work towards a tool which allows for much more flexibility, enabling *any* analysis that involves calculations done on many small regions within a large astronomical survey. Other applications could include lens finding, though we note that image data is not yet supported in `heinlein`.

`lenskappa` includes several features to facilitate this including:

- High-level API for defining analyses
- Plugin architecture for adding new capabilities without modifying the core code.

- Automatic support for any dataset supported by `heinlein`

2.4.1 Analyzing modern astronomical survey datasets at scale

One of the big challenges in doing analyses on these kinds of datasets is the need for the computing environments to be close to the data whenever possible. Querying over the internet is useful for assembling datasets, but is not a particularly good solution when analyzing a dataset at scale. Many researchers will not have access to sufficient storage to store these datsets, and it is impractical to expect individual survey teams to provide computing resource for general use. The size of these datasets will enable next-generation analyses, but only with the development of next-generation tools running at scale, which will require computing infrastructure that may not be readily available to many researcher.

We support using cloud computing services to fill this gap. Commercial providers have expanded and matured by a significant margin over the last decade, and routinely handle storage and analysis tasks on datasets orders of magnitude larger than the ones being discussed here. Additionally, cloud computing technologies are significantly more accessible than on-premise technology: they require far lower startup costs and can be quickly scaled (to accommodate more users, or bigger jobs) without the bottlenecks that slow down the expansion of on-premise infrastructure. It is also possible to use cloud solutions as a supplement to already-existing on premise solutions. Holzman et al. (2017) demonstrated this by analyzing data from the Compact Muon Solenoid experiment at massive scale.

In the future, we plan to develop `lenskappa` and `heinlein` tools that could be easily deployed onto services like these, providing quick and easy access to large survey datasets in addition to techniques for processing and analyzing that data. The datasets would be stored in the cloud, allowing users to deploy their analyses without worrying about the connection to the underlying dataset. Such an approach has been demonstrated by Kusnierz et al. (2022), which enabled serverless access to `ROOT` for quick analysis tasks on high energy datasets without the user having to manually retrieve the data. With some work, `heinlein` will enable this type of analysis by serving as a bridge between computing infrastructure (which could be managed by individual researchers) and the underlying data lake (which could be managed by the survey team). Abstracting away data retrieval will allow researchers to focus on what matters most: designing the analyses they wish to perform and interpreting results.

2.5 Analysis of SDSS J0924+0219

In this section, we discuss the analysis of the J0924 system, as performed by `lenskappa`.

2.5.1 The Lens Field

SDSS J0924+0219 is a quadruply lensed quasar first reported in Inada et al. (2003). The quasar itself is at redshift $z = 1.523$, while the lensing galaxy is located at $z = 0.384$ (Eigenbrod, A. et al., 2006). Quadruply lensed quasars are particularly valuable for cosmographic analysis because it is in principle possible to measure 12 different time delays, though only three of these are independent. An image of the field can be seen in Figure 2.2. This system was modeled in Chen et al. (2022)

One particularly important feature of the lens field is the bright star located in the lower left. The star covers a reasonably large fraction of the overall area in the 120" aperture, undoubtedly covering several background objects and making photometry for objects very near it unreliable. We apply techniques that have previously been used in this kind of analysis to correct for this. This technique is discussed in more detail in section 2.5.2.

The lens field falls within the Subaru Hyper Surpime-Cam Strategic Survey Program footprint, and full color information is available for all objects in the relevant catalogs. We use these catalogs, including photometric redshifts, for all objects inside the field. Additionally we use the bright star masks and photometric redshift PDFs provided by the HSC team.

Our analysis follows the same outline discussed in section 2.3. We discuss the details particular to this lens field below.

2.5.2 The HSC Survey and Weighted Number Count Ratios

The Hyper Suprime-Cam Subaru Strategic Survey is a large survey program, aiming to cover roughly 1400 deg^2 of sky in five photometric bands (*grizy*) down to $i \sim 26$, with deeper coverage expected in smaller regions of the sky (Aihara et al., 2017). We base our analysis on the roughly 400 deg^2 that had coverage in all five bands as of the second data release (Aihara et al., 2019). Data release 3 was made available while this paper was in preparation, but we do not consider it here.

The HSC Survey is a natural choice for a comparison dataset for this system because the J0924 field itself falls within the survey footprint. We therefore have both robust and comparable photometry. When computing weighted number count ratios, the basic approach is identical to the one outlined in the section 2.3.4, with some specific adjustments:

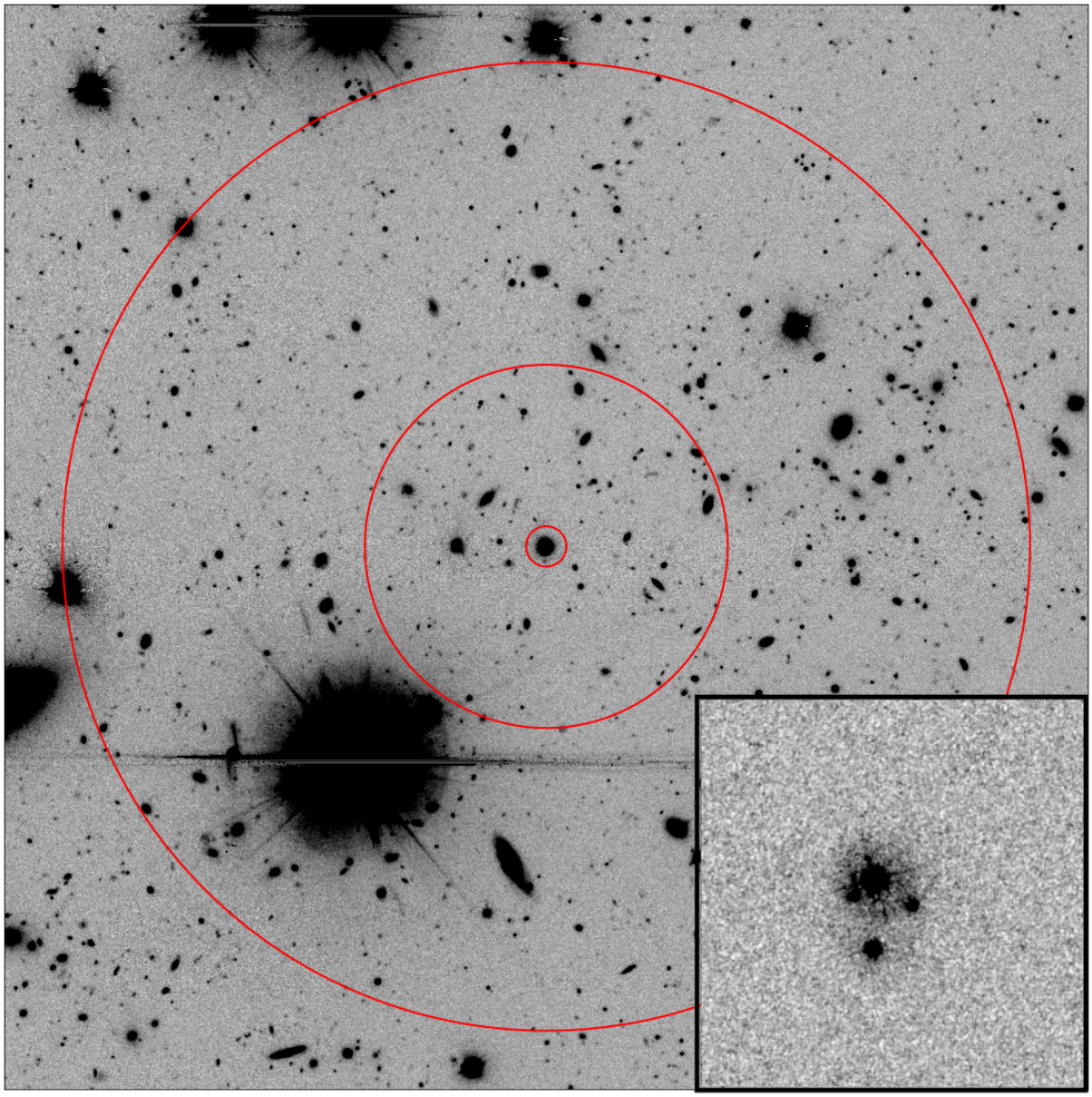

Figure 2.2 Field around SDSS J0924, shown in HSC i-band. The red rings represent 5”, 45” and 120” apertures respectively. Bottom right: 10” cutout of the lens system from HST imaging.

We use objects with `r_extendness_value` = 1.0, which selects galaxies. Bosch et al. (2017) demonstrates that their algorithm for computing this quantity does a reasonably good job of selecting galaxies, though it may incorrectly classify some galaxies as stars, and vice-versa. However we note that the HSC-wide survey was intentionally selected to enable cosmological analyses by being in regions that are away from the galactic plane and low on dust extinction (Aihara et al., 2017). This, combined with robust bright-star masks and our large sample size ensures unmasked stars do not significantly impact the final weighted number count ratios.

The full photometric redshift PDFs for all objects in the data release have been made available by the HSC team (Nishizawa et al., 2020). They use two separate fitting algorithms, and results from both algorithms are included in the catalog. We compute weighted number count ratios using both sets of redshifts, and do not find a meaningful difference between the resultant distributions. As such, we use the "DEmP" redshift and stellar masses for our analysis (Hsieh & Yee, 2014; Tanaka et al., 2017).

When computing weighted number counts, we remove all objects closer than 5 arcseconds from the center of the field following Rusu et al. (2017). Objects this close to the center of the field are typically explicitly included in the mass model of the lens, and so we also remove them from the comparison fields to avoid biasing results.

The HSC survey team makes available masks that represent areas of the sky where photometry may be unreliable or lacking due to the presence of bright stars (Coupon et al., 2017). This is particularly important in our field due to the presence of the bright star that can be seen in Figure 2.2 When iterating over the reference survey, we retrieve the bright star masks for each region being considered. We apply both these masks and the masks for the lens field itself to both catalogs at each weighting step. Doing this ensures that results are not biased if a given field in the reference survey has significantly more or less of its area covered by bright star masks. This procedure was first used in Rusu et al. (2017)

The 400 deg^2 of sky we consider here is separated into seven disconnected regions. Initially, we computed weighted number count ratios in five of these regions. This produces nearly-identical distributions, with the difference between the lowest and highest median being 0.1σ the standard deviation of the distribution, suggesting our fields are large enough to to avoid sampling bias. Based on this, we restrict our subsequent analysis to 135 deg^2 of sky located in the region $332^\circ <$

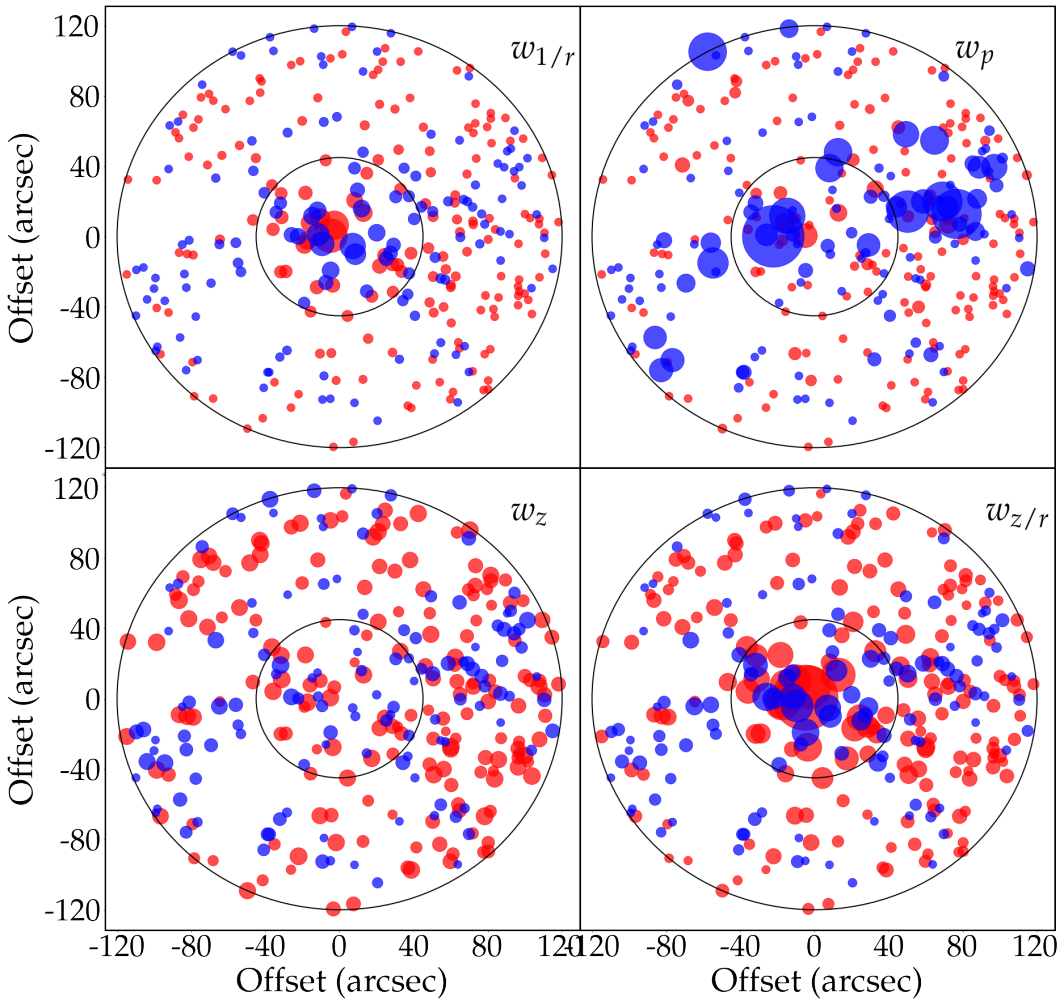

Figure 2.3 Location and relative weights of all galaxies with $z < z_s$. The black circles represent the 45" and 120" apertures respectively. Blue objects are those with $i < 23$, while red objects are those with $23 < i < 24$. Relative object weights of the object are represented by the size of the dot.

RA $< 359°$ and $-1.5° <$ Dec $< 6°$. For each combination of aperture and limiting magnitude, we compute weighted count ratios at 100,000 randomly selected fields.

2.5.3 Millennium Simulation

The Millennium Simulation (Springel et al., 2005) is a dark matter only simulation split into 64 $4 \times 4\,\mathrm{deg}^2$ fields. After the original run was completed, synthetic galaxy catalogs were painted into the resultant halos by several teams. Following (Rusu et al., 2017) we use the semi-analytic catalogues of De Lucia & Blaizot (2007). Additionally, Hilbert et al. (2009) split each $4 \times 4\,\mathrm{deg}^2$ field into a grid of 4096x4096 points and used ray tracing to compute convergence and shear at each of these points in 63 redshift planes. These, combined with its large size, makes it an excellent

choice for our analysis. In our analysis, we use redshift plane 36 with $z = 1.504$.

First, we compute the weighted number counts at a large number (order 10^6) of equally spaced grid points in the Millennium simulation. For the 45" aperture, we place the fields 90" apart (snapped to the nearest grid point), while for the 120" aperture we place fields 60" apart. Both cases result in over 1.5 million lines of sight across the simulation, each of which has an associated value of $\kappa_{\rm ext}$ and $\gamma_{\rm ext}$ This differs from the same calculation for the lens field itself in a key way: the values reported are the total value of the weights at every point considered, rather than a ratio of values. To normalize, we divide weighted number count in each field by the median value for all lines of sight in our sample. The median value of the resultant distribution is therefore unity.

A key difference between the synthetic catalogs and the real data from the HSC survey is that in the Millennium Simulation catalog redshifts are exact. In previous work (eg. Rusu et al., 2017) this difference was accounted for by computing photometric redshifts for all objects in the Millennium Simulation, using the same pipeline that was used to compute the photometric redshifts in the comparison dataset. We are unable to use this technique in this case because the HSC survey photo-z pipelines are not publicly available as of the preparation of this paper. Instead, we download the full catalog of training data used by the HSC. These are galaxies for which spectroscopic redshifts are available. We divide the objects in the test dataset into redshift and magnitude bins. For each galaxy in a bin, we compute the offset in the central value of the redshift ($z_{phot} - z_{spec}$), and take the median of these as our estimate of the redshift bias in that bin. Additionally we take the median value of σ_z (as reported by the HSC photometric redshift pipeline) for all photometric redshifts in that bin. We take the the median value of σ_z as our estimate of photometric redshift uncertainty. For each object in the Millennium Simulation catalog, we construct an artificial photemetric redshift PDF. The center of the distribution is the "actual" redshift of the object, offset by the the amount computed previously for the appropriate bin. The width of the distribution is the value of σ_z computed for the same bin.

At each weighing step in the Millennium Simulation, we then sample from these "photometric redshift" PDFs when computing weighted number counts. For each line of sight, we sample from each of these PDFs 50 times. This produces 50 separate catalogs (all with slightly different values for object "redshifts"), and we compute weighted number counts for each one. We find that this process produces no meaningful change to our final inference for $\kappa_{\rm ext}$, even when a weight that

depends on the redshift is taken into account. However this process massively increases the amount of data output by our code, and therefore significantly increases the amount of time required to do the final inference on $\kappa_{\rm ext}$. This does suggest that photometric redshift uncertainties do not have a significant impact on the inferred value of $\kappa_{\rm ext}$, but we plan to explore this more completely in the future.

2.6 Results and Discussion

Our weighted number counts for the lens field include a total of five weights (w_n, $w_{1/r}$, w_p, w_z, $w_{z/r}$, see Table 2.1), two apertures (45" and 120"), two limiting magnitudes (23 and 24), and two summing techniques (pure sum and medians). For brevity, we report only the medians of these distributions in Table 2.2. The full distributions, along with the analysis code used to produce the posterior distributions for $\kappa_{\rm ext}$ are available on github. [3]. A visual catalog of the objects in the field and their relative weights can be seen in Figure 2.3.

The weighted number count ratios suggest the SDSS J0924 field is mildly overdense as compared to the universe as a whole. This overdensity is significantly more obvious when considering the 45" aperture. This is quite reasonable; as the size of the aperture increases, the density of field will approach the density of the universe as a whole. Figure 2.4 shows the distributions of of the weights considered in this work.

It is however less obvious why the value of the weights seem to depend on the limiting magnitude, with the median values for $i < 23$ being significantly higher than for $i < 24$. This may suggest that the quality of the lens field catalog is poor below magnitude 23. We remind the reader that the 5" region around the lens itself is masked when computing weighted number counts.

Based on the values of our summary statistics, we would expect our final value of $\kappa_{\rm ext}$ to be somewhat positive. However the strength of the external shear of the field, reported in (Chen et al., 2022) is $0.017^{+0.001}_{-0.003}$. This places it significantly below the median (and, in fact, mean) for all lines of sight in the Millennium Simulation. For each combination of limiting magnitude and aperture we compute $\kappa_{\rm ext}$ for a number of different combinations of weights:

- Pure (unweighted) number counts, combined with each of the remaining weights individually.

[3] https://github.com/PatrickRWells/J0924-analysis

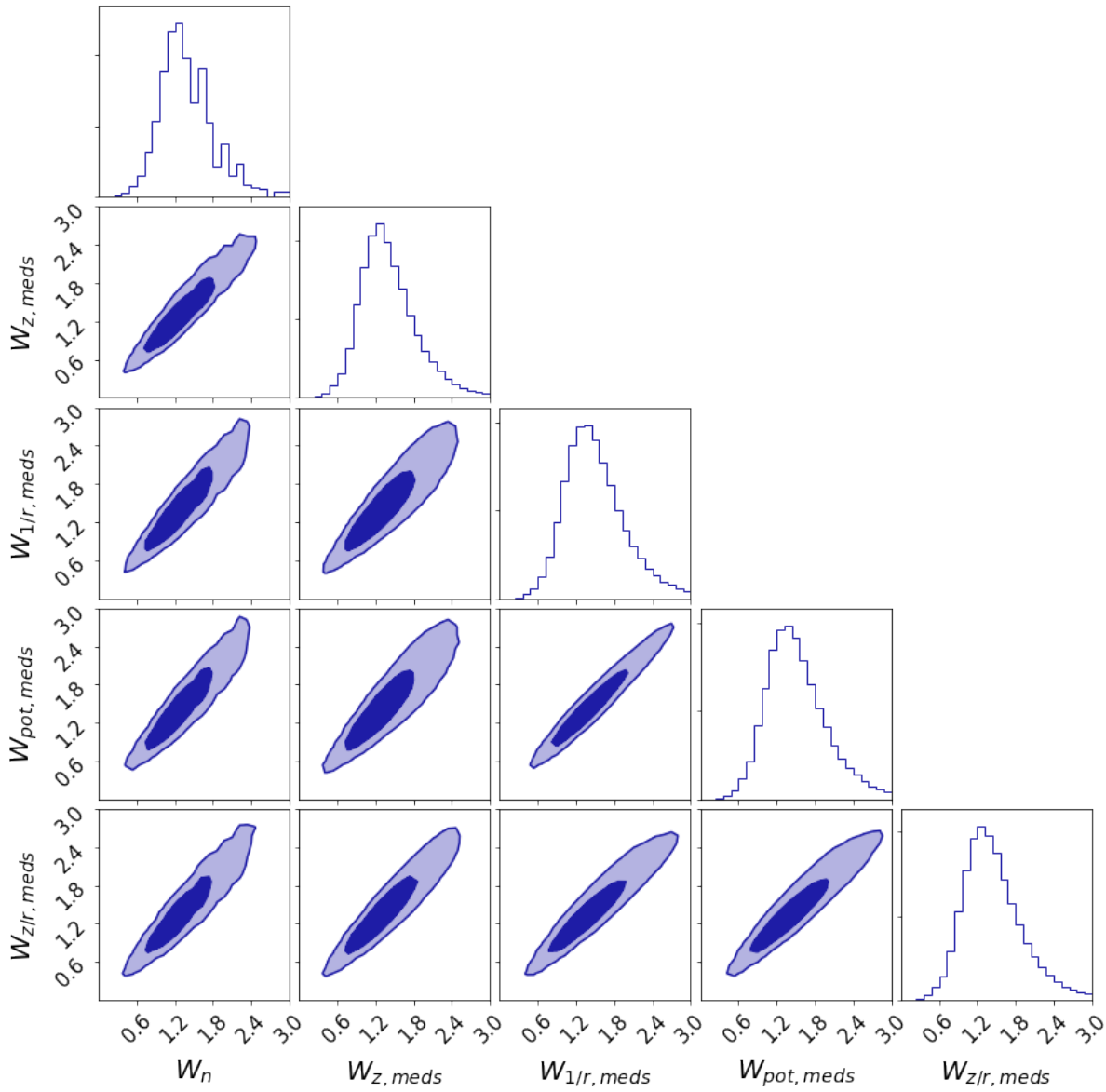

Figure 2.4 2D histograms of for each possible pair of the weight number count ratios considered in this work, computed with a 45" aperture and limiting magnitude $i < 24$. The inner and outer contours represent 68 and 95% confidence intervals, respectively.

Weight	$i < 23$, 120"	$i < 24$, 120"	$i < 23$, 45"	$i < 24$, 45"
w_n	1.04	1.04	1.42	1.35
$w_{1/r}$	1.17	1.09	1.57	1.35
w_p	1.19	1.10	1.56	1.36
w_z	1.03	0.99	1.56	1.37
$w_{z/r}$	1.21	1.07	1.74	1.37
$w_{1/r,meds}$	1.08	1.01	1.59	1.45
$w_{p,meds}$	1.12	1.01	1.50	1.47
$w_{z,meds}$	1.03	1.00	1.50	1.36
$w_{z/r,meds}$	1.10	1.04	1.63	1.40

Table 2.2 Median of weighted number counts for SDSS J0924 Field. "med" represents the second weighting scheme discussed, where $W_i = n_i \bar{w}_j$ See Table 2.1 for definitions of the various weights.

- Pure number counts and inverse distance weights, combined with each of the remaining weights individually.

- For each of the above cases, we run the kappa inference both with and without the constraint from γ.

The number count ratios used for this analysis are listed in Table 2.2. We do not mix distributions obtained using our two different weighting techniques.

We also repeat this for each combination listed above while using the median weighting scheme rather than the sum scheme. All together, this leaves us with 112 individual histograms for the value of kappa.

We find that the choice of specific weights is less important than the number of weights being considered. In all cases, inferring $\kappa_{\rm ext}$ with three weights instead of two tightens the resultant distribution, but does not significantly affect the central value. Rusu et al. (2017) found best results when combining w_n and w_r with one additional weight and the constraint from $\gamma_{\rm ext}$. We find the same here. Specifically, the central value of the distribution does not change meaningfully based on the choice of the third weight, but we find slightly tighter constraints when using w_p.

We also find that results are consistent between apertures, but find that a brighter value of the

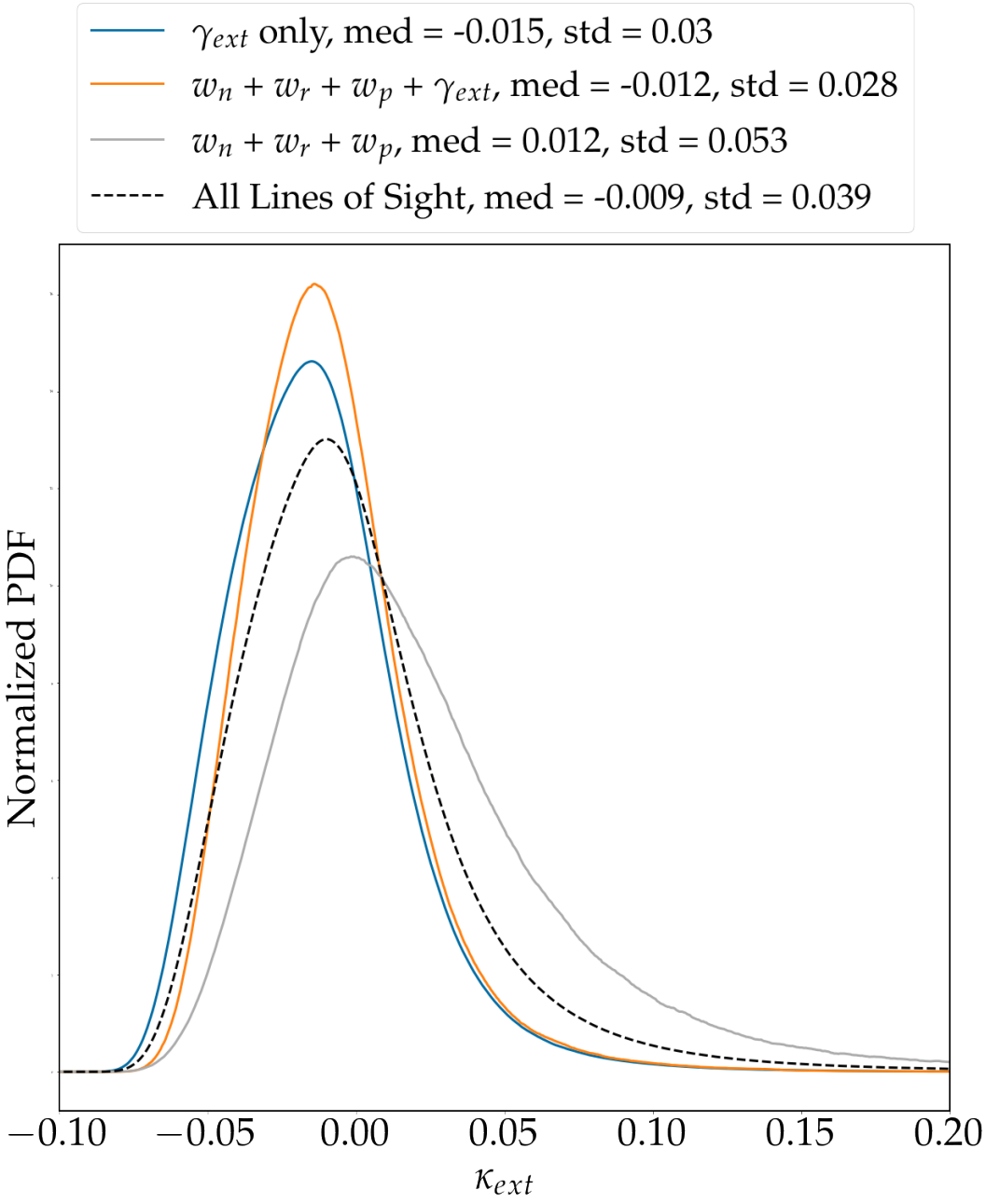

Figure 2.5 Comparison of smoothed posterior on $\kappa_{\rm ext}$ for $w_n + w_r + w_p$ without $\gamma_{\rm ext}$, $w_n + w_r + w_p$ with $\gamma_{\rm ext}$, and $\gamma_{\rm ext}$ alone.

limiting magnitude results in a noisier posterior on $\kappa_{\rm ext}$. Because so many objects fall between magnitude 23 and 24, removing these objects results in significantly noisier weighted number counts which translate to the final inference on κ. Considering only γ as a constraint leads to a median value on $\kappa_{\rm ext}$ of -0.015. As a result, including $\gamma_{\rm ext}$ as a constraint significantly lowers the central value of our distributions, though it also tightens the distribution. However we note this

shift is fairly modest as compared to the width of the distribution. This is similar to the result seen in Rusu et al. (2017), though in that case the inferred value of the shear was significantly closer to the median in the Milennium Simulation of 0.028, and the shift of the central value was not as significant. We find our tightest constraints on $\kappa_{\rm ext}$ through a combination of w_n, $w_{1/r}$ and w_p combined with constraints from $\gamma_{\rm ext}$. This leads us to a final value of $\kappa_{\rm ext}$ of -0.012 with a width $\sigma_\kappa = 0.028$. Without the constraint from $\gamma_{\rm ext}$, we obtain a median value of 0.012 with $\sigma_\kappa = 0.053$. Full posteriors for these combinations can be found in Figure 2.5. We use a 45" aperture and limiting magnitude of $i < 24$ for these results.

2.7 Conclusions and Future Work

In this paper, we have discussed the current state of the line of sight number counts technique for environment analyses in time delay cosmography. We have introduced two main improvements to previous iterations of the analysis. First, our packages `lenskappa` and `heinlein` make designing and running these analyses much quicker than before, in addition to making it much simpler to add additional survey datasets. This will accelerate the pace of future analyses, and enable population-level analyses of lens environments. Additionally, we have made use of the entire distributions of weighted number counts, which accounts for covariance between weights and is generally more robust than just exploring a small region around the medians. We have applied these techniques to the J0924 field, and found that this field is a fairly typical line of sight, with a slightly negative median value of $\kappa_{\rm ext}$.

2.7.1 Future Development of `lenskappa`

Our primary goal for this project has been to build a software tool that can quickly and reliably analyze weak perturbers along lines of sight to strong gravitational lenses. We have accomplished this goal, but we plan to extend the capabilities of `lenskappa` to include tools for analyzing strong perturbers and coherent structures (such as galaxy groups). These objects must be handled individually, and require very different analysis tools. However we see a significant advantages to being able to do full environment analyses within a single software package.

2.7.2 Future Analyses

Leveraging the capabilities of `lenskappa`, we hope to better understand lines of sight to gravitational lenses on a population level. Fassnacht et al. (2010) and Wong et al. (2018) have shown that lenses seem to fall in preferentially overdense lines of sight, but that this overdensity seems to be confined to the immediate surroundings of the lens itself. `Lenskappa` gives us the tools to perform these population-level analyses quickly, with the freedom to adjust and re-run the analysis as needed. As a first step, we hope to complete an analysis with 4-5 new lens systems, as well as 2-3 systems that have previously been analyzed to check our results.

As an additional check, we would like to analyze a large number of non-lens lines of sight. This would ensure there are no biases introduced by comparing distributions based on real galaxy catalogs to those obtained from the synthetic catalogs in the Millennium Simulation.

Recently, Park et al. (2023) demonstrated the use of Bayesian Graph Neural Network to estimate the value of κ_{ext} in a simulated dataset. Their method out-performs a simplified version of the analysis performed here that uses only a single weight. Further work may demonstrate the ability of the technique to match or even outperform the weighted number counts technique, but a more complete comparision will need to be performed to assess this.

Chapter 3

Population Analysis of Lines of Sight of 25 Strong Galaxy-Galaxy Lenses with Extreme Value Statistics

This chapter has been accepted for publication in Astronomy & Astrophysics as a TDCOSMO Collaboration paper.

3.1 Abstract

Time-delay cosmography is a technique for measuring H_0 with strong gravitational lensing. It requires a correction for line of sight perturbations, and it is necessary to build tools to assess populations of these lines of sight efficiently. We aim to demonstrate the techniques necessary to analyze line of sight effects at a population level, and investigate whether strong lenses fall in preferably overdense environments. We analyze a set of 25 galaxy-galaxy lens lines of sight in the Strong Lensing Legacy Survey sample using standard techniques, then perform a hierarchical analysis to constrain the population-level parameters. We introduce a new statistical model for these posteriors that may provide insight into the underlying physics of the system. We find the median value of $\kappa_{\rm ext}$ in the population model to be 0.033 ± 0.010. The median value of $\kappa_{\rm ext}$ for the individual lens posteriors is 0.008 ± 0.015. Both approaches demonstrate that our systems are drawn from an overdense sample. The different results from these two approaches show the importance of population models that do not multiply the effect of our priors.

3.2 Introduction

Time-delay cosmography is a technique for inferring the value of the Hubble constant and other cosmological parameters based on multiply imaged time-variable sources. Its independence from standard early and late Universe probes make it an essential tool in resolving the ongoing tension between those techniques. Time-delay cosmography relies on four primary ingredients. The first is time delays, which are obtained by monitoring the system over many months or years. The second is a mass model based on high-quality imaging data. The third is stellar kinematics of the lensing galaxy, which is used to break the well-known mass sheet degeneracy. The fourth, and the subject of this paper, is the external convergence (denoted $\kappa_{\rm ext}$), which can be thought of as the cumulative effect of all additional perturbers along the line of sight.

As with many domains of astronomy and cosmology, time-delay cosmography is increasingly big data focused. The number of known time-delay lenses has increased dramatically with surveys such as the Dark Energy Survey (Dark Energy Survey Collaboration, 2005, hereafter DES) and the Subaru Hyper-Suprime Cam Strategic Survey program (Aihara et al., 2017, hereafter HSC), and is expected to increase by orders of magnitude with the Vera Rubin Observatory Legacy Survey of Space and Time (Ivezić et al., 2019, hereafter LSST). Not only does this present unprecedented opportunities to do interesting astronomy, but it also introduces new and unique technical challenges. In particular, population modeling is an increasingly important tool for deriving constraints on interesting quantities by leveraging the statistical power of many systems. However utilizing this approach effectively is not possible without building high-quality tools that can ingest, process, and track the large amount of data required to perform the inference.

Each of the ingredients discussed above involves its own set of challenges. The $\kappa_{\rm ext}$ measurement is different in that the challenge is largely a problem of data management. The relative density of a given field around some lens of interest is determined by comparing the field to a large number of fields randomly drawn from some large reference survey. The data for this procedure is readily available from the various survey teams. The challenge then is building tools that are capable of doing this comparison for dozens or even hundreds of lens fields at once, while being flexible enough to allow us to evolve our techniques forward without starting from scratch. However, if this challenge is solved, this analysis serves as an excellent testing ground for building systems that do astronomy at scale.

The long term goal of the cosmography community is to provide constraints on H_0 with a precision comparable to that of more mature probes such as the distance ladder (e.g., Riess et al., 2022) or the cosmic microwave background (e.g., Aghanim et al., 2020a). Being independent of these probes, cosmography is well positioned to provide an insight into the Hubble Tension (Di Valentino et al., 2021) once this higher level of precision is realized. The combined statistical power of the population of lenses that will become available in the next decade should be sufficient to provide such a constraint, but only if we have the analysis tools to match it. In this work, we analyze a population of 25 strong galaxy-galaxy lenses from the Strong Lensing Legacy Survey (SL2S) sample. We use the number counts techniques described in Wells et al. (2023) to estimate $\kappa_{\rm ext}$ along the line of sight to each individual lens, and then use the hierarchical techniques discussed in Park et al. (2023) to infer population parameters. We introduce a new statistical model for the resultant distributions, which provides a potential insight into the primary source of signal in κ_{ext}. This process additionally serves as a test of the ability of our techniques and software to work at scale.

In section 3.3, we present the essentials of time-delay cosmography and the techniques used to estimate $\kappa_{\rm ext}$ along a given line of sight. In section 3.4, we discuss the challenges of operationalizing this analysis to run at scale, and the tools used in the analysis presented in this work. In section 3.5, we introduce the lens sample used in this work and discuss the choices made to estimate κ_{ext} for the individual lines of sight. In section 3.6, we introduce a novel statistical model that provides insight into the source of signal in $\kappa_{\rm ext}$, while in section 3.7 we leverage these statistics to produce a population model of $\kappa_{\rm ext}$ for our systems. Finally, in section 3.8 we present and discuss the results of our hierarchical analysis, and look forward to future work on this topic.

3.3 Time-delay cosmography and line-of-sight analysis

In this section, we present the essentials of time-delay cosmography and the associated line-of-sight analysis. For a more complete overview of time-delay cosmography and its current status, we refer the reader to Treu & Marshall (2016), Birrer et al. (2022), and references therein, while for a more complete discussion of the line-of-sight analysis we refer the reader to Wells et al. (2023) and Rusu et al. (2017).

3.3.1 $\kappa_{\rm ext}$ and its application to time-delay cosmography

Time-delay cosmography relies on the strong gravitational lensing of a time-variable source to place constraints on the distance scales of the combined observer-lens-source system and, ultimately, use these constraints to derive a constraint on H_0. In a strongly lensed system in which multiple images are visible, the light from the various images will take different paths from the source to the observer. This difference will be directly visible when the luminosity of the source varies, as the brightness of the various images will change at different times. These techniques have been applied to strongly lensed quasars for a number of years (e.g., Kundić et al., 1997; Fassnacht et al., 2002; Vuissoz et al., 2008; Bonvin et al., 2016) More recently, time-delay techniques have been applied to the supernova Refsdal (Kelly et al., 2023), as well as cluster-scale lenses (Liu et al., 2023).

This time delay between two images at angular positions θ_A and θ_B of a source at (unobservable) angular position σ can be directly related to the gravitational potential of the lens by

$$\Delta t_{AB} = \frac{D_{\Delta T}}{c}[\tau(\theta_A, \sigma) - \tau(\theta_B, \sigma)], \tag{3.1}$$

where τ is the Fermat potential of the lens given by

$$\tau(\theta, \sigma) = \frac{(\overrightarrow{\theta} - \overrightarrow{\sigma})}{2} - \psi(\overrightarrow{\theta}), \tag{3.2}$$

and $\psi(\overrightarrow{\theta})$ is the scaled lensing potential.

The cause of the time delay is the difference in the path length taken by the light of the various images plus the difference in the Shapiro delay. The quantity $D_{\Delta t}$ from Eq. 3.1 is known as the time-delay distance and is given by

$$D_{\Delta t} = (1 + z_d)\frac{D_{\rm d} D_{\rm s}}{D_{\rm ds}} \propto \frac{1}{H_0}, \tag{3.3}$$

where $D_{\rm d}$, $D_{\rm s}$, and $D_{\rm ds}$ are the angular diameter distances to the deflector, source, and from the deflector to the source, respectively. Given an accurate lens model and a measurement of a given time delay, it is therefore possible to measure H_0.

However this analysis is complicated by the fact that lenses are embedded in the Universe, and therefore surrounded by mass structures that also have an impact on the lensing observables. This effect is parameterized with the external convergence ($\kappa_{\rm ext}$). The result of these perturbations

is difficult to pin down, both because the mass distribution along the line of sight cannot be directly observed, and because the effect is much less significant than the effect of the primary lens. However, correcting for the effect is essential in cosmography because it propagates directly to the inferred value of $D_{\Delta t}$ by

$$D_{\Delta t} = \frac{D'_{\Delta t}}{1 - \kappa_{\rm ext}}, \tag{3.4}$$

where $D'_{\Delta t}$ denotes the uncorrected value of the time-delay distance. The relationship to the inferred value of the Hubble constant is similarly straightforward:

$$H_0 = (1 - \kappa_{\rm ext}) H'_0, \tag{3.5}$$

where H'_0 represents the uncorrected value. In general, $\kappa_{\rm ext}$ is of order 10^{-2}, and so failing to correct for this effect introduces bias of a few percent in the average case, up to around 10% in the most extreme cases. It is worth noting that the total convergence is the combination of $\kappa_{\rm ext}$ and the convergence from the primary lens. In that sense, $\kappa_{\rm ext}$ can be thought of as the residual convergence that would be present if the primary lens were removed.

Because the underlying mass distribution in a given line of sight cannot be directly observed, we must infer it based on available data about the luminous matter in the field. The standard tools for doing this inference involve weighted number counts of galaxies within some distance of the lens. This technique has been used extensively by the TDCOSMO collaboration and its predecessors (see for example Fassnacht et al., 2010; Rusu et al., 2017; Wells et al., 2023) to provide an estimate of $\kappa_{\rm ext}$ along a given line of sight. $\kappa_{\rm ext}$ can be thought of as the density of mass sheet which, if placed coplanar to the lensing galaxy, would produce the same cumulative effect as all the perturbers along the line of sight. We note that this is distinct from the internal mass sheet transformation (see Chen et al., 2021; Gomer & Williams, 2020). The combination of these two effects leads to the well-known mass sheet degeneracy, which is usually broken with measurements of stellar kinematics (Shajib et al., 2023; Schneider & Sluse, 2013). Generically, $\kappa_{\rm ext}$ results in the magnification or de-magnification of the images of the background source, but this effect is not directly measurable.

3.3.2 Essentials of the technique

In the context of lensing, κ is a dimensionless measurement of the underlying matter distribution in units of the lensing critical density, Σ_{cr}.

$$\Sigma_{cr} = \frac{c^2 D_s}{4\pi G D_s D_{ds}} \tag{3.6}$$

In strong lensing, $\kappa > 1$ and an appropriately-placed source will be lensed into multiple images and/or an Einstein ring. For lower mass concentrations ($\kappa \ll 1$) lensing is instead evident by distortions in the shape of background sources, such as galaxies. Typical weak lensing techniques involve statistics based on distortions to the apparent shape of large numbers of galaxies. As a result, weak lensing analyses are typically done on much larger angular scales than is useful for the $\kappa_{\rm ext}$ measurement. While weak lensing analyses of $\kappa_{\rm ext}$ have been done (e.g., Tihhonova et al., 2018, 2020), these rely on high resolution imaging (typically space-based) of the field of interest. It is unrealistic to expect such imaging for the vast majority of lens fields in future surveys, and the techniques used to do this analysis must reflect this.

On larger scales, κ can be thought of a measurement of the relative density of a region of space as compared to the entire Universe. An "average" field will be assigned a value of κ near zero, while a slightly overdense field should receive a slightly positive value. For a given line of sight, we first make an empirical estimate of the density of the field. While we cannot directly observe the majority of the mass in a given field, we can use visible matter as a tracer of the underlying dark matter. In particular, a field with more galaxies is likely to have more dark matter than an identically shaped field with fewer galaxies. The relationship between luminous matter and dark matter is noisy, but we can readily estimate the amount of luminous matter in a given field with galaxy surveys. We measured the absolute density of a given field with summary statistics computed based on the galaxies in the region. Natural examples of summary statistics include an inverse distance summary statistic, where galaxies closer to the center of the line of sight are weighted more heavily, and a weighting based on redshift, where sources are weighted more heavily where the lensing efficiency is higher. The value of the summary statistic for a given line of sight is just the sum of the weights of the individual galaxies:

$$W_i = \sum_{j \in gal} w_j, \tag{3.7}$$

where w_j denotes the weight for a single galaxy along the line of sight. In this context, the full posterior in $\kappa_{\rm ext}$ can be written as

$$p(\kappa_{\rm ext}|\mathbf{d}) \propto p(\kappa_{\rm ext}|W)p(W|\mathbf{d}) \propto p(W|\mathbf{d})p(W|\kappa_{\rm ext})p(\kappa_{\rm ext}), \tag{3.8}$$

where the relationship between the second and third form follows from Bayes' theorem. In general, we use constraints for several summary statistics when estimating $\kappa_{\rm ext}$. The likelihood $p(W|\kappa_{\rm ext})$ cannot be written down in closed form, and we must turn to Approximate Bayesian Computation to estimate it. We seek to compare our line of sight to similar lines of sight in a simulated dataset, where values of $\kappa_{\rm ext}$ have already been calculated.

Usage of simulations in this context may introduce bias into the inference based on the underlying cosmology of the simulation. To control this bias, we wish to estimate the relative density of a given line of sight compared to all lines of sight in the Universe, and then find lines of sight in the simulated dataset with the same relative density to estimate $\kappa_{\rm ext}$.

To estimate the relative density of a given line of sight, we compute the same set of summary statistics in a large number of randomly selected fields in an appropriately large sky survey. For each random field, we compute the ratio of the summary statistic in the field of interest to the summary statistic of the given random field. The resultant distribution of ratios gives an empirical estimate of the relative density of the field compared to the Universe as a whole, so long as the reference survey is large enough to avoid sampling bias. This caveat is increasingly less of a concern. Modern surveys such as HSC and DES have hundreds to thousands of square degrees of contiguous, high quality sky coverage. In the near future LSST will image nearly all of the southern sky with many thousands of strongly lensed objects expected to be discovered. Of these thousands of systems, several hundred are expected to be suitable for time-delay cosmography (Verma et al., 2019).

We then compute the same set of summary statistics in a simulated dataset with values of $\kappa_{\rm ext}$ computed. We normalize the resultant distribution by its median. This allows us to more directly compare these lines of sight to lines of sight of interest. More concretely, if the median of a distribution for a summary statistic in the real data is 1.2, this implies that the value of the summary

statistic for our line of sight is 20% higher than the median line of sight in our comparison survey. A line of sight from the simulated dataset with a normalized summary statistic of 1.2 is also 20% greater than the median value (for a further discussion of the summary statistics we use, and our techniques for matching to the simulated dataset, see Wells et al. (2023)).

3.4 Line-of-sight analysis at scale

A crucial aspect of the work presented here is developing and validating the tools needed to perform line-of-sight studies at scale. In Wells et al. (2023), we presented `heinlein`, a data management tool for survey datasets, and `lenskappa`, which utilized the capabilities of `heinlein` to perform a line-of-sight analysis. `lenskappa` was quite limited in that it was only capable of performing a single lens analysis at a time, and had minimal flexibility to evolve our techniques forward. In particular, `lenskappa` required weighted number count analysis for each lens to be performed individually, even if all the analyses were using the same region of the sky for comparison. This problem was magnified when performing weighted number counts in the Millennium simulation, as many more samples are required to produce a reasonable posterior.

The core philosophy of the κ_{ext} analysis is that simpler statistics can produce meaningful results when evaluated over very large datasets. In the context of this work, the correlation between the summary statistics and the quantity of interest (κ_{ext}) is fairly weak. However, the advantage of these statistics is ease of computation. The true computational challenges of this style of analysis derive from the need to efficiently manage and query large survey datasets. This challenge is not unique to our analysis, and building tools to efficiently solve it will be useful for a wide variety of analyses over the coming decade. We call this style of analysis "cosmological data sampling," because it requires repeatedly drawing samples from a large survey dataset. We seek to build a tool that is capable of doing this style of analysis efficiently, and allows the user to iterate and build new analyses quickly.

To this end, we introduce `cosmap`[1], a Python package for defining and running "cosmological data sampling" analyses like those discussed in this work. In practice, `cosmap` can be used to apply any computation across a large survey dataset quickly and reliably. `cosmap` is an evolution of the `lenskappa` package first presented in Wells et al. (2023), and has been written from the

[1] availabile from `pip` or `https://github.com/PatrickRWells/cosmap`

ground up to provide an easy-to-use tool for doing analysis with big data astronomy. `cosmap` makes use of `pydantic`[2] for parameter validation and `Dask`[3] to distribute work across available computing resources. Data management and result outputs can be handled by the library without user involvement, but a plugin architecture is included to modify default behavior if the user finds it insufficient for their analysis.

Crucially, all user-defined behavior in `cosmap` is written outside of the core library. Analyses are defined as a series of transformations on the data organized as a Directed Acyclic Graph (DAG), a structure in common use in pipeline orchestration and task scheduling. Analysis parameters must be declared, and their runtime values parsed by `pydantic` to ensure correctness. These decisions ensure that failures occur early in the runtime of the program, to avoid situations where processor (and astronomer) time is wasted.

While `lenskappa` was only capable of analyzing a single lens at a time, `cosmap` allows us to write an analysis that handles all the lenses in our sample in a single run. This saves a large amount of computation time over the previous model. The fundamentally modular nature of individual analysis definitions makes it simple to iterate on an existing analysis or define a new analysis entirely. We estimate `cosmap` saves over 95% of the computational time that would be required if this analysis was done with `lenskappa`.

3.5 Data and procedures for individual κ measurements

In this section, we discuss the dataset we use in this analysis and our procedures for performing κ_{ext} measurements on our individual lens lines of sight. The basic procedure used to analyze the individual lenses is identical to the procedure discussed in section 3.3. One key difference is our ability to analyze many lenses at once, as we discussed in section 3.4. However, we note this is a computational optimization, and does not impact the results for individual lenses.

3.5.1 The CFHT Legacy Survey and the Strong Lensing Legacy Survey

The Canada-France-Hawaii Telescope Legacy survey (hereafter CFHTLS) is a 155 deg^2 multiband imaging survey completed in 2012 (Gwyn, 2012). After the completion of the survey, the data were re-processed with the goal of discovering strong lenses, resulting in the so-called Strong Lensing

[2] `https://pydantic.dev`
[3] `https://www.dask.org`

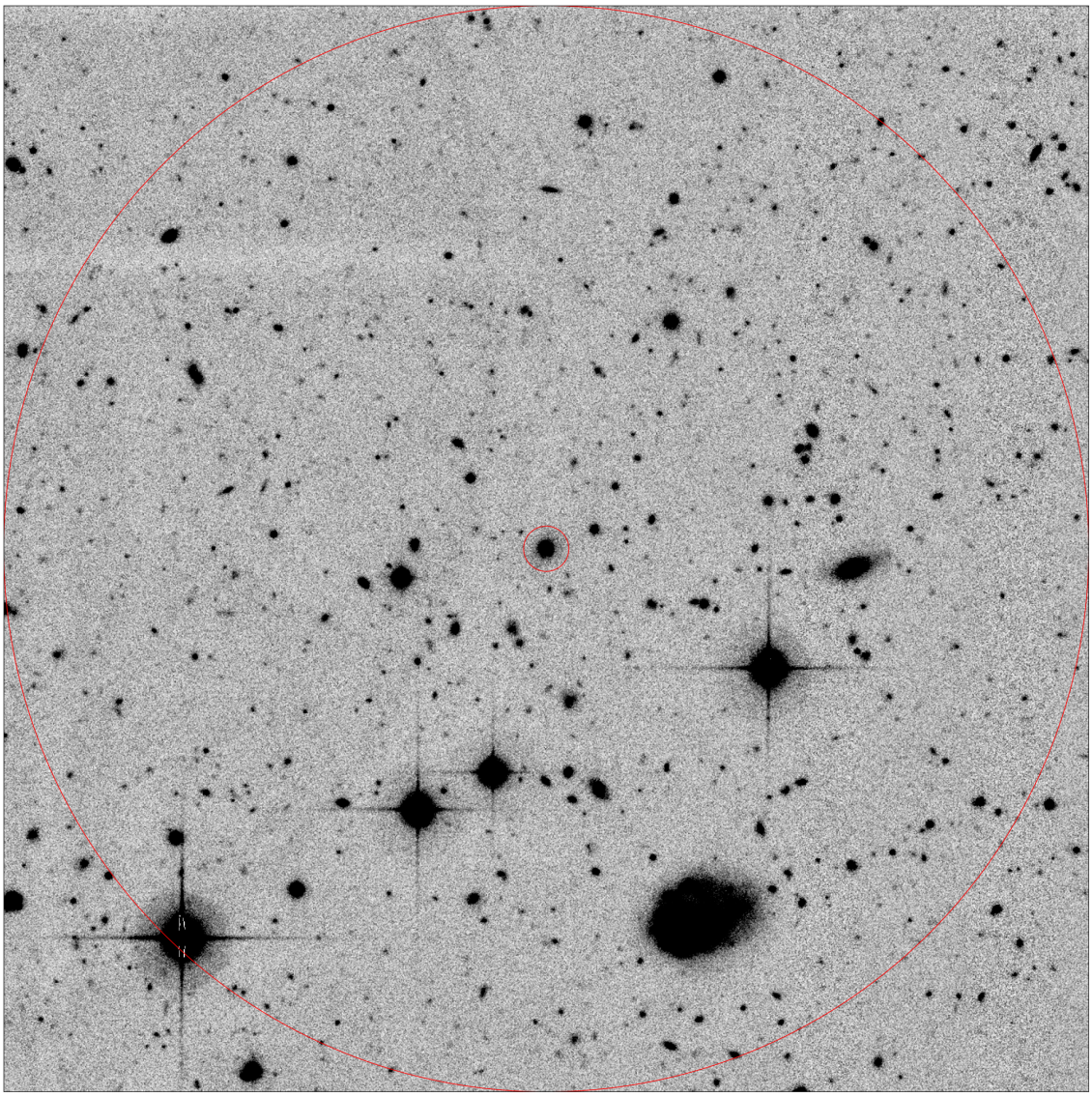

Figure 3.1 Image showing the field around SL2SJ1405+5243. The inner and outer red circles mark the inner and outer cutoff radius we use when computing weighted number counts.

Legacy Survey (Cabanac et al., 2006, hereafter SL2S). CFHTLS has been used previously to to analyze other lens lines of sight (see, for example, Rusu et al., 2017). It is useful due to its depth ($i \sim 24.5$) and relatively large size (at least historically) of its wide fields.

Our sample includes 28 lenses from this survey. The sample was selected to be analogues of the kinds of systems analyzed by TDCOSMO, with the original goal of obtaining population-level constraints on the mass distribution of the lensing galaxies in the sample. The choice of lenses is discussed in more detail in TDCOSMO Collaboration (2024, in prep). The lensing galaxies have redshifts between 0.238 and 0.884, while the source galaxies have redshifts between 1.19 and 3.39. Because our analysis relies on a high-quality galaxy catalog of the line of sight, we remove three lenses where a nearby bright star has corrupted the resultant catalog such that more than half of the field is missing galaxy photometry. Table 3.1 summarizes the essential information about each lens system included in this work, while Figure 3.1 illustrates the typical quality of imaging data used to derive catalog products. However, we do not re-derive any catalog products, instead using the fiducial measurements performed by the survey team.

3.5.2 Individual $\kappa_{\rm ext}$ measurements

We used the techniques discussed in Section 3.3 and in Wells et al. (2023) to infer the posterior on $\kappa_{\rm ext}$ for each individual line of sight. At this stage, there is no information about the population-level statistics. Each line of sight is analyzed on its own, with a prior set by the Millennium simulation.

3.5.2.1 Comparison field and cuts

To compute weighted number counts for the individual lenses, we used $50\,\mathrm{deg}^2$ from the CFHTLS W1 field, bound by $31^\circ < RA < 38.5^\circ$ and $-11^\circ < Dec < 4^\circ$ as a control field. We computed weighted number counts in a $120''$ aperture, and limited our counts to objects brighter than 24th magnitude in i-band. These choices were consistent with choices made in previous work (see for example Wells et al., 2023; Rusu et al., 2017). In particular, the magnitude limit is sufficiently bright as to be meaningfully above the survey's detection limit, while being sufficiently faint to catch all structures that are likely to contribute meaningfully to $\kappa_{\rm ext}$ (Collett et al., 2013). For each lens, we also ignored objects beyond the redshift of the source quasar, and performed the same cut when comparing to the reference survey. Additionally, we removed all objects from the underlying catalog within 5" of the center of the field. For time-delay lenses, objects near the center of the

field are typically included in the mass model explicitly, and removed during the $\kappa_{\rm ext}$ measurement accordingly.

3.5.2.2 Selecting Summary Statistics

Selecting appropriate summary statistics is an important step in the analysis described here. This has been explored extensively in previous TDCOSMO and H0licow papers (e.g., Rusu et al., 2017; Wells et al., 2023). We based our result in this work on the following summary statistics:

1. Pure number counts ($w_j = 1$)
2. Inverse Distance Weighting ($w_j = 1/r_j$)
3. Redshift-Distance Combination ($w_j = (z_s * z_j - z_j^2)/r_j)$)

A primary challenge of this techniques is the fairly limited data that are available on individual objects in wide-field galaxy surveys. These summary statistics provide information on how much mass is clustered near the center of the line of sight, and how much mass is clustered where the lensing efficiency is high. Importantly, these summary statistics depend on quantities which are reasonably robust in modern galaxy surveys. However in general these summary statistics are poor tracers of the underlying mass distribution, as evidenced by the width of the posteriors on indiviual lines of sight. This challenge is one of the primary motivations behind combining information behind many systems into a population-level inference. More sophisticated and/or higher order summary statistics (such as two-point galaxy clustering) may provide additional useful information but are left for a future analysis.

3.5.2.3 Uncertainty and comparison to the Millennium simulation

After computing these weighted number counts, we treated the median value of the distribution as the estimate of the overdensity or underdensity. To estimate an uncertainty in this quantity, we utilized the photometric redshift uncertainties present in the underlying catalogs. We produced 1000 copies of the original line-of-sight catalog, with redshifts for each object randomly sampled from that object's photo-z PDF. We computed weighted number counts for each of the resultant catalog and treated the fractional width of the resultant distribution as the fractional uncertainty in our measurement of the median.

When computing weighted number counts in the Millennium simulation, we used the same limits described above. We used the semi-analytic galaxy catalogs of De Lucia & Blaizot (2007), which were shown in Rusu et al. (2017) to provide the best results for this analysis.

To match summary statistics, we selected lines of sight from the Millennium simulation that were similar in density to the lens lines of sight based on the value of the summary statistics. The values of $\kappa_{\rm ext}$ for each line of sight were drawn from the maps produced in Hilbert et al. (2009), which cover the simulation in a grid with spacing between points $\approx 3.5''$. The contribution from a given line of sight was weighted by a multidimensional Gaussian centered on the distribution medians with widths set by the uncertainties discussed above. We took into account correlations between the weights when constructing this Gaussian. This is, in essence, an Approximate Bayesian Computation computation, with one key limitation. We were limited by the lines of sight available to us in the Millennium simulation, and by the computational time required to search for lines of sight matching a given lens. For the majority of lenses this is not an issue, as there were more than enough similar lines of sight in the simulation to produce posteriors that are well fit by a smooth Generalized Extreme Value (GEV) distribution. For very overdense lenses where the posterior is noisy due to a small number of matching sightlines, we widened the search Gaussian. The majority of lenses in our sample do not require this intervention, or require only a modest widening to achieve acceptable results. We note that this procedure may bias more extreme lenses towards more moderate values of $\kappa_{\rm ext}$, as widening the search region will naturally include more lines of sight from the central peak of the distribution.

For each lens posterior, we determined a best-fit Generalized Extreme Value distribution with a least-squares optimizer. Using these distributions allowed us to quickly and easily draw from our priors and posteriors when sampling the population posterior. We discuss the GEV distribution and its interpretations in the following section.

3.6 Extreme-value statistics and applications in astronomy

Extreme-value statistics describe the expected distribution of extreme values (maxima or minima) of samples drawn from a single underlying distribution. Despite relatively minimal use in astronomy, they have numerous applications in many applied disciplines. For example, extreme value statistics can be used to model the maximum daily rainfall expected over some number of consecutive days at

some location. This model is crucial for engineers working to design flood-resistant infrastructure (e.g., Papalexiou & Koutsoyiannis, 2013).

3.6.1 The generalized extreme-value distribution and subtypes

The generalized extreme value distribution is a continuous, unimodal distribution with location parameter μ, scale parameter σ, and shape parameter ξ. Its probability distribution is given by

$$\frac{1}{\sigma} t(x)^{\xi+1} e^{-t(x)} \tag{3.9}$$

where

$$t(x) = \left\{ \begin{array}{ll} \left[1 + \xi\left(\frac{x-\mu}{\sigma}\right)\right]^{\frac{1}{\xi}}, & \xi \neq 0 \\ \exp\left(-\frac{x-\mu}{\sigma}\right), & \xi = 0 \end{array} \right\} \tag{3.10}$$

This generalized distribution is broken down into three subtypes based on the value of ξ. Type I (or "Gumbel") when $\xi = 0$, type II (or "Fréchet") when $\xi > 0$, and type III (or "Weibull") when $\xi < 0$.[4]

The Gumbel distribution typically arises when the underlying sample is normally or exponentially distributed. However for cases where the underlying distribution is bounded, the Fréchet and Weibull distributions are more appropriate. For an example of this distribution and an application to our work, see Figure 3.2. We do not restrict our GEV fits to any of these sub-distributions at any point within this work. The corresponds to allowing the shape parameter to take on both positive and negative values, or zero as dictated by the data.

3.6.2 Extreme-value statistics in astronomy

While extreme-value statistics have found limited use in astronomy, previous work has been done showing applications in cosmic structure problems. Davis et al. (2011) showed analytically that the most massive halo in a given region of the Universe should follow Gumbel statistics (that is, the resultant distribution should be a GEV distribution with $\xi = 0$). Antal et al. (2009) demonstrated that the number of galaxies within some physical distance of a given location on the sky also followed Gumbel statistics at a wide range of scales. This second result is of particular interest, because it is nearly identical to some of the techniques discussed in section 3.3 of this work.

[4]Throughout this work, we report values with the standard sign convention used here. The `scipy` implementation of the GEV distribution which we use for our computational work uses the opposite convention.

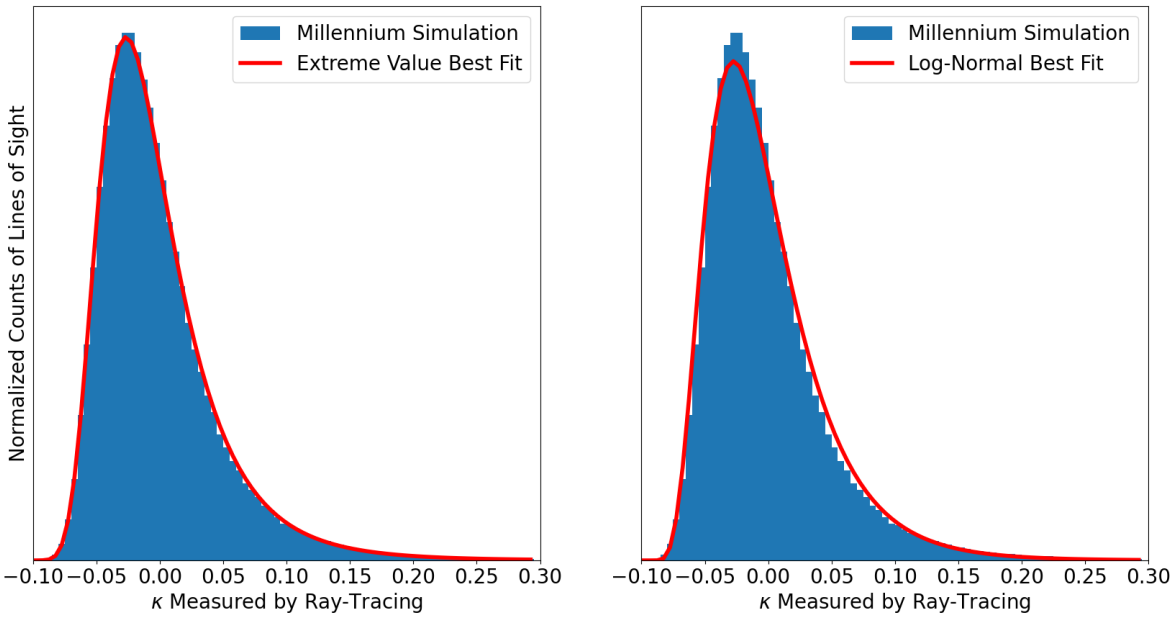

Figure 3.2 Comparison of best-fit GEV distribution (left) and best-fit log-normal distribution(right) to the κ_{ext} distribution of lines of sight in the Millennium simulation at redshift z = 2.34 from Hilbert et al. (2009). The best-fit GEV parameters are $\xi = 0.145$, $\mu = -0.0235$, and $\log(\sigma) = -3.46$, while the best-fit log-normal parameters are $\mu = -0.098$ and $\log(\sigma) = -0.78$.

The Gumbel distribution has additionally found use in areas such as modeling the weak lensing (Capranico et al., 2013) and modeling anomalies in the Cosmic Microwave Background (Mikelsons et al., 2009). Crucially, all these analyses have implications about cosmic structure which would have a direct impact on the measured mass density in a given region of space.

3.6.3 Application to Millennium simulation and individual lens lines of sight

The values of κ measured in the Millennium simulation by Hilbert et al. (2009) clearly follow extreme value statistics, as can be seen in Figure 3.2. Our posteriors on individual lenses are effectively this distribution convolved with the gaussian we use to select matching lines of sight (see Section 3.5.2), so it is sensible they too would follow these statistics. We emphasize this choice is empirical, but the excellent fit does suggest interesting interpretations.

The distribution of κ in the Millennium simulation and in our posteriors suggests that its value along a given line of sight may be dominated by contribution from a small number of mass structures which themselves follow extreme value statistics. The result presented in Davis et al. (2011) is of particular interest, because it demonstrates that the largest halo measured in a given region

of the Universe should also follow extreme value statistics. This relationship suggests several interpretations that may be worth investigating in the future.

3.6.3.1 $\kappa_{\rm ext}$ may be dominated by a single mass structure

The appearance of extreme-value statistics in our model suggests κ may be dominated by a single massive structure along the line of sight. This structure is not necessarily the most massive halo in the field, but may be a more moderately sized halo situated near the center of the field. In this context, the posterior on $\kappa_{\rm ext}$ could be interpreted as the range of mass structures (or more accurately their relative density) that are possible given some set of observables (i.e. luminous galaxies). Placing better constraints on this particular mass structure may allow us to improve the precision of the $\kappa_{\rm ext}$ measurement.

3.6.3.2 Number counts may be able to distinguish between different halo mass models

The appearance of extreme-value statistics both in galaxy number counts and and the underlying halo mass function suggests an interesting relationship. Number count statistics are nothing new in cosmology. Many analyses have used cluster number counts within surveys to place constraints on cosmological parameters (e.g., Costanzi et al., 2021). A primary limiting factor of these techniques is the measurement of the cluster masses themselves. The power of the number counts techniques is its ease of applicability. Given some set of statistics, we can use `cosmap` to evaluate its behavior over a large region of the sky quickly and reliably. Given a set of dark matter models which make quantitatively different statements about "clustering" on small angular scales, it may be possible to quickly assess which of these models is consistent with the data available in some large galaxy survey. In this case, the constraining power of a well-measured mass structure is traded for the constraining power that derives from the scale of the dataset.

3.6.4 Comparison to log-normal distribution

A general rule of thumb in statistics is to use models with the fewest number of parameters that fit the data well. As the GEV distribution is a three parameter distribution, it is reasonable to question whether it is necessary in this context when two parameter distributions with similar shape exist. In particular, log-normal distributions have found frequent use in cosmic structure problems (e.g., Coles & Jones, 1991; Xavier et al., 2016) and generally provide good fits to large scale convergence and shear data (e.g., Taruya et al., 2002; Clerkin et al., 2016).

Our work here differs in several key ways. In particular, we are working on very small angular scales (2′) and only measuring out to $z \approx 2$. Nonlinear structure becomes a significant concern in this regime and it is reasonable to suggest that this may introduce complications to the standard log-normal picture.

The best-fit log-normal distribution is included in figure 3.2 in addition to the GEV fit. The log-normal fit meaningfully underestimates the peak, overestimates the decay, and slightly underestimates the tail of the emperical distribution. Both best-fit models were determined using the `stats.fit` function of `scipy` on unbinned values of κ. The best-fit log-normal distribution results in a Bayesian Information Criteria (BIC) of 219.5, while the GEV best-fit distribution yields a BIC of 174.8. This clearly favors the GEV over log-normal.

3.7 Population-level environment studies in time-delay cosmography

Astronomy is increasingly a big data field, and time-delay cosmography is no exception. The LSST is expected to uncover many thousands of lenses in its full footprint, with many hundreds of time-delay lenses that will be suitable for cosmographic analysis (Verma et al., 2019). Time-delay cosmography is still in a regime where its errors are dominated by the random variance that is to be expected from a small sample of systems. Increasingly, the community is turning to population analyses to estimate cosmological parameters as well as to more informed priors on other ingredients of the analysis such as lensing galaxy mass profiles (see for example Birrer et al., 2020).

The measurement of $\kappa_{\rm ext}$ for a single lens is very much prior dominated. While easy to compute, the summary statistics we use are fairly weak tracers of the underlying matter distribution. This results in wide posteriors that often deviate only modestly from the prior except in particularly extreme cases. By combining statistical information from many lines of sight at the likelihood level, deviations of the population from the prior become more apparent.

There is a general expectation from previous studies that lenses lie in preferably overdense lines of sight (e.g., Wong et al., 2018; Fassnacht et al., 2010). This is consistent with well-known work from Dressler (1980) which showed that massive elliptical galaxies are more likely to be found in overdense regions. This suggests that the primary contributor to $\kappa_{\rm ext}$ in many lines of sight may be

this group or cluster, as is suggested in Fassnacht et al. (2010). As will be demonstrated shortly, the analysis presented here confirms this previous work while allowing us to quantify this overdensity in a way that can be directly useful in time-delay cosmography.

3.7.1 Procedures for population analysis

To estimate the population-level parameters of the lines of sight in our sample, we used the framework developed in Wagner-Carena et al. (2021) and applied in Park et al. (2023) to mock lines of sight. We used results from individual lines of sight to fit a distribution for the entire population. For a given value of location parameter μ, scale parameter σ, and shape parameter ξ the posterior takes the form:

$$p(\mu, \sigma, \xi|\mathbf{d}) \propto p(\mu, \sigma, \xi) \prod_{i=1}^{N} \frac{1}{M} \sum_{\kappa \in p(\kappa_i|\mathbf{d})} \frac{p(\kappa|\mu, \sigma, \xi)}{p(\kappa|\Omega_{\rm sim})} \tag{3.11}$$

where $p(\mu, \sigma, \xi)$ is our hyperprior on the population-level parameters, and $p(\kappa|\Omega_{sim})$ is the probability of a given value of κ in the prior imposed by the simulation. The sum was done over 20,000 samples taken from the individual posteriors for each line of sight. We use σ as the scale parameter in the equation above to emphasize that our target distribution is not necessarily Gaussian. We discuss our choice of target distribution in the following section.

Each value of the product term can be thought of as a likelihood for a given lens. By dividing out the prior, we avoid multiplying its effects across the population. As a result, it is reasonable to expect the population constraints to favor a more significant overdensity than a naive average of the individual posteriors. This is a statistical effect. Bayes theorem is designed as a tool for updating posteriors as new information comes in. In our case the "new information" in our analysis is that our lines of sight, looked at as a population, show significant signs of being biased when compared to the population of all lines of sight in the Universe.

3.7.2 Hierarchical analysis with SL2S lenses

Once we have posteriors for each line of sight, we move on to a hierarchical analysis of the population. We use the `emcee` Python package (Foreman-Mackey et al., 2013) to sample from the posterior given in Section 3.7.1.

We emphasize that the value of the prior in the Millennium simulation is not unique to a particular value of the hyperparameters, as it also depends on the redshift of the lens. It is reasonable to

suggest that the redshift of sources in our sample should be included as a population parameter, but given the tight constraints on individual redshifts our sample is too small to determine this distribution. When drawing from individual posteriors, we use the best-fit GEV distribution as provided by `scipy`.

We use a flat hyperprior with $-0.5 < \xi < 0.5$, $-0.1 < \mu < 0.3$ and $-5.0 < \log(\sigma) < -2.0$. These ranges include the values of the best-fit parameters for the full population of lines of sight in the Millennium simulation.

3.7.3 Interpretation of population posteriors

When interpreting our results, it is crucial to appreciate the difference between the posterior on $\kappa_{\rm ext}$ produced for a single lens and the posterior on the population parameters. For a single lens, the value of $\kappa_{\rm ext}$ is (presumably) nearly constant across the surface of the primary lens. The posterior therefore is largely a statement about our uncertainty based on the incomplete information that goes into our analysis. With better information or a more sophisticated model, it may in principle be possible to shrink the width of the posterior. However this is a posterior on only a single parameter: $\kappa_{\rm ext}$.

However when doing a population analyses, the location, width, and shape of the population distribution are themselves parameters with associated uncertainties. These parameters are making a statement about the distribution of lines of sight in which we find strong lenses, while the issue of "incomplete information" appears in the uncertainty on the individual parameters. For the sake of intuition, it is helpful to compare the distribution produced by the parameter point estimates to the prior from the simulated dataset. However unlike the distribution for individual lines of sight, the width of this distribution has an astrophysical interpretation and may be fundamental to the population. Additionally, individual lens posteriors are prior dominated, as the data available is a fairly weak tracer of the underlying mass distribution in the field. By combining the constraining power of many lenses, it may be possible to constrain this population better than we could constrain any single lens system.

3.8 Results

Our results provides strong evidence that the lines of sight in our sample are drawn from a population that is more dense than the population of all lines of sight in the Universe.

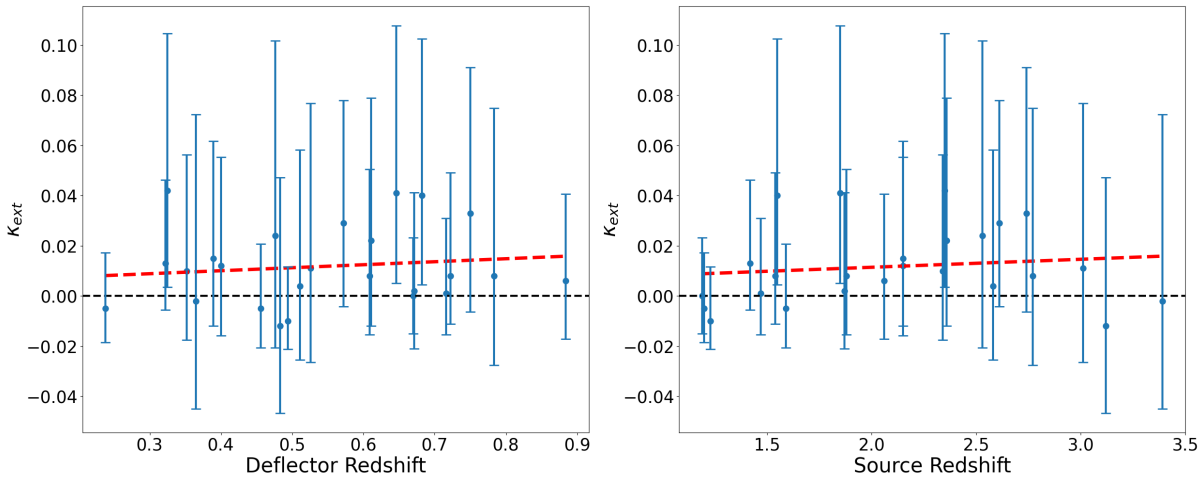

Figure 3.3 Measured median values of κ_{ext} for the lenses in our sample as a function of deflector and source redshift, respectively. Error bars denote the 68% confidence interval. The red dotted line represents the trend of the median value of κ_{ext}

.

3.8.1 Individual lines of sight

A summary of the best-fit parameters for individual lines of sight can be found in Table 3.1, and a plot of our measured value of κ_{ext} with respect to redshift in Figure 3.3. From the individual results alone, it is reasonable to suggest that our lines of sight are drawn from a population that is more dense on average than the population of all lines of sight in the Universe. Our results also demonstrate a modest trend towards greater density for longer lines of sight, but this is inconclusive.

Figure 3.3 also demonstrates the necessity of population studies. The posteriors on individual lines of sight are quite wide, and most lines of sight individually are consistent with $\kappa_{\text{ext}} = 0$. This is a result of the fact that the summary statistics we use are fairly weak tracers of the underlying mass distribution. This amount of scatter for individual κ_{ext} is consistent with previous analyses on other systems.

3.8.2 Population constraints

Although individual sightlines provide limited information, our population model demonstrates clearly that our lines of sight are drawn from a biased sample. A corner plot of our $MCMC$ samples and a comparison of best-fit distributions can be found in Figure 3.4. The median value of the population distribution on κ_{ext} inferred by our model $\kappa_{med} = 0.033 \pm 0.010$. This result demonstrates that our lines of sight are drawn from a sample that is more dense than the general

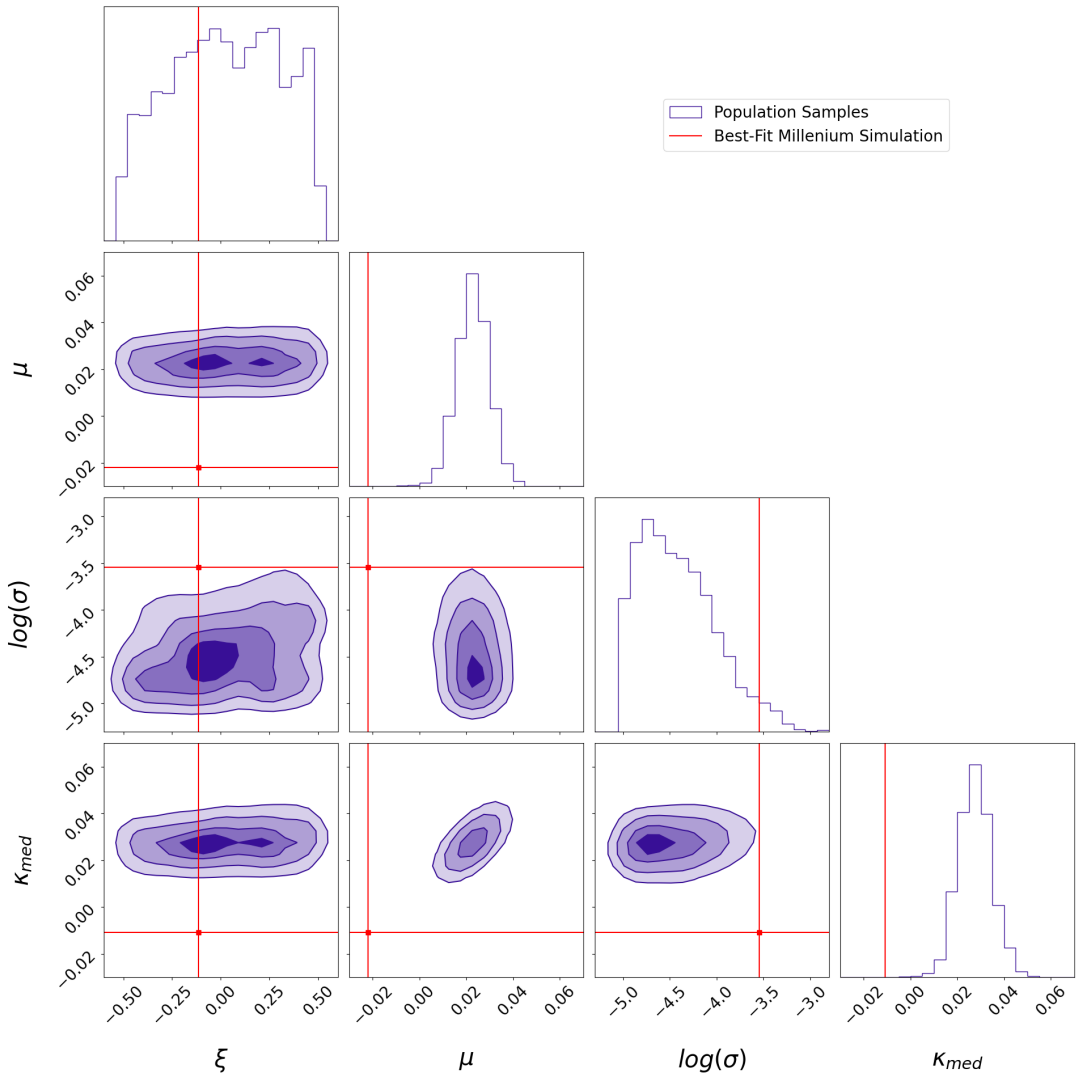

Figure 3.4 Corner plot showing results of our MCMC, with the median value of each sample distribution included as a derived parameter. The red mark indicates the best-fit values for the entire population of lines of sight in the Millennium simulation.

population at this redshift under the assumption the Millennium simulation provides a reasonable prior for the distribution of κ on small scales in the Universe. However for the purposes of time-delay cosmography, the important factor is whether the median value of $\kappa_{\rm ext}$ is above or below zero. If our population median was exactly zero, we would anticipate that no real population-level correction from $\kappa_{\rm ext}$ would be necessary. Our model does not show this conclusively, but does provide evidence for this conclusion at $\approx 3\sigma$ confidence. Our sample is not large enough to provide strong constraints on the shape and scale parameters of the distribution, but this may change in the

future with larger samples. In our tests, our posterior on $\log(\sigma)$ remains nearly constant in the range $-8 < \log(\sigma) < -5$, indicating the inability of our data to constrain these quantities cleanly. Such a narrow posterior would indicate an extremely specific selection function, in clear contention with our knowledge of these systems. We therefore choose to cut off our hyperprior at $\log(\sigma) = -5$.

It is interesting to compare the population median to the median of the individual results (the "median of medians"). Simply averaging the individual lens posteriors results in a "population" median of 0.008 ± 0.015. We emphasize that this approach implicitly includes the effect of the prior once for each lens posterior, whereas the population model divides out this prior before averaging. The overdensity from our population model is not excessive, but is more significant than would be expected by a naive averaging of the results for the individual sightlines. This demonstrates that correcting for line of sight effects on a population level is necessary when performing time-delay cosmography on large samples of lenses. We remind the reader that κ_{ext} measures the residual overdensity that remains after removing the lens and any immediate neighbors from the line of sight. The actual value of the total convergence κ at the location of the lens itself will be quite different.

3.9 Conclusions and future work

In this paper, we have demonstrated a technique for estimating $\kappa_{\rm ext}$ along strong lens lines of sight at a population level and applied it to a sample of 25 lenses in the Strong Lensing Legacy Survey. This work has been built on previous work that allows us to perform $\kappa_{\rm ext}$ inferences on individual lenses much faster than was previously possible. We have demonstrated the infrastructure and statistical frameworks necessary to apply this technique at massive scale and provide constraints on populations of lines of sight. We have shown that the populations of lines of sight that are used in this analysis are likely drawn from a biased sample that is overdense when compared to the population of a lines of sight in the Universe, with median $\kappa_{\rm ext} = 0.033 \pm 0.010$.

3.9.1 Future improvements to $\kappa_{\rm ext}$ measurement

The primary goal of this work is to develop and demonstrate the tools and frameworks necessary to provide constraints on large populations of lens lines of sight. While we have made much progress in this direction, there are still a number of improvements that should be made:

3.9.1.1 Upgrade our simulation

While of significant historical significance, the Millennium simulation has been surpassed in recent years by larger and more sophisticated simulations, which take the last two decades of improved understanding into account. We continue to use Millennium because of the high-resolution weak lensing maps that are available. However the `MillenniumTNG` team has produced high-resolution weak lensing maps that include the effects of baryons (Ferlito et al., 2023) which would be suitable for our analysis once the data products are released publicly. In particular `MillenniumTNG` has a mass resolution around one order of magnitude better than the original `Millennium` simulation, which may allow us to more cleanly map small-scale real-Universe sightlines onto equivalent simulated sightlines.

3.9.1.2 More efficient summary statistic mapping with machine learning

Additionally, finding matching lines of sight in the simulated dataset is quite slow, as we must iterate through the entire dataset. Training a neural network to reproduce the relationship between summary statistics and $\kappa_{\rm ext}$ at a single redshift should be straightforward. However expanding this to encompass the entire volume of the dataset would be a much more significant challenge. Taking on this challenge may be unavoidable given the number of lenses that will discovered in LSST.

3.9.1.3 Summary statistics that better target the primary mass structure

Our work here has suggested that the primary contribution to $\kappa_{\rm ext}$ may be a single mass structure. Placing further constraints on this mass structure may be a way to improve the precision of the measurement. As always, a primary challenge is finding techniques that can easily be applied to a large number of systems.

3.9.2 Further work on population modeling

This work has demonstrated the techniques necessary for placing constraints on the distribution of $\kappa_{\rm ext}$ populations of strong lens lines of sight. While our work demonstrates with a high degree of confidence that this population of lenses fall in overdense environments, we cannot place clear constraints on the width or scale of that distribution. Additionally, we cannot say with confidence that the result derived from the SL2S sample is applicable to strong gravitational lenses as a whole. A much larger set of lenses (on the order of a few hundred) is needed to better constrain this population and boost confidence that our conclusions can be generalized. The strong lenses

expected to be discovered in LSST will be an ideal sample for this type of work, and we look forward to working with these data when they become available.

Acknowledgements

P.R.W and C.D.F acknowledge support for this work from the National Science Foundation under Grant No. AST-1907396

This work is partially based on observations with the NASA/ESA Hubble Space Telescope obtained at the Space Telescope Science Institute, which is operated by the Association of Universities for Research in Astronomy, Incorporated, under NASA contract NAS5-26555. Support for Program number HST-GO-17130 was provided through a grant from the STScI under NASA contract NAS5-26555.

P.W. thanks Kenneth Wong for reviewing this work prior to submission.

P.W. thanks the TDCOSMO environment working group for useful discussion throughout the preparation of this work.

Name	RA (deg)	Dec (deg)	z_d	z_s	μ_κ	ξ_κ	$log(\sigma_\kappa)$	κ_{med}	16%	84%
SL2SJ0212-0555	33.199	-5.931	0.750	2.740	0.018	0.031	-3.202	0.033	-0.007	0.087
SL2SJ0214-0405	33.547	-4.084	0.609	1.880	-0.001	0.179	-3.686	0.008	-0.017	0.036
SL2SJ0217-0513	34.405	-5.225	0.646	1.850	0.027	0.206	-3.260	0.041	0.003	0.084
SL2SJ0218-0802	34.505	-8.047	0.884	2.060	-0.003	0.057	-3.738	0.006	-0.017	0.037
SL2SJ0219-0829	34.759	-8.493	0.389	2.150	0.005	0.128	-3.542	0.015	-0.013	0.050
SL2SJ0220-0949	35.192	-9.824	0.572	2.610	0.017	0.002	-3.356	0.029	-0.005	0.077
SL2SJ0225-0454	36.296	-4.909	0.238	1.200	-0.010	0.083	-4.235	-0.005	-0.019	0.013
SL2SJ0226-0420	36.544	-4.337	0.494	1.230	-0.014	0.184	-4.383	-0.010	-0.022	0.005
SL2SJ0232-0408	38.215	-4.140	0.352	2.340	-0.000	0.091	-3.515	0.010	-0.019	0.048
SL2SJ0233-0438	38.280	-4.644	0.671	1.870	-0.007	0.153	-3.726	0.002	-0.022	0.030
SL2SJ0848-0351	132.197	-3.851	0.682	1.550	0.026	0.176	-3.286	0.040	0.003	0.083
SL2SJ0849-0412	132.290	-4.207	0.722	1.540	0.001	0.258	-3.825	0.008	-0.014	0.031
SL2SJ0855-0147	133.917	-1.792	0.365	3.390	-0.018	0.124	-3.073	-0.002	-0.047	0.055
SL2SJ0901-0259	135.275	-2.985	0.670	1.190	-0.006	0.102	-4.178	-0.000	-0.015	0.019
SL2SJ0904-0059	136.033	-0.998	0.611	2.360	0.009	0.120	-3.326	0.022	-0.014	0.065
SL2SJ1359+5535	209.957	55.597	0.783	2.770	-0.006	0.218	-3.270	0.008	-0.030	0.050
SL2SJ1404+5200	211.227	52.007	0.456	1.590	-0.011	0.117	-4.114	-0.005	-0.021	0.015
SL2SJ1405+5243	211.443	52.720	0.526	3.010	-0.003	0.139	-3.210	0.011	-0.029	0.059
SL2SJ1406+5226	211.710	52.439	0.716	1.470	-0.005	0.155	-4.021	0.001	-0.016	0.022
SL2SJ1411+5651	212.904	56.855	0.322	1.420	0.006	0.163	-3.917	0.013	-0.007	0.036
SL2SJ1420+5630	215.249	56.502	0.483	3.120	-0.025	0.121	-3.294	-0.012	-0.048	0.034
SL2SJ1427+5516	216.880	55.279	0.511	2.580	-0.007	0.172	-3.441	0.004	-0.028	0.041
SL2SJ2203+0205	330.871	2.089	0.400	2.150	0.001	0.092	-3.552	0.012	-0.017	0.047
SL2SJ2205+0147	331.279	1.784	0.476	2.530	0.007	0.146	-3.045	0.024	-0.023	0.081
SL2SJ2221+0115	335.453	1.262	0.325	2.350	0.027	0.117	-3.217	0.042	0.002	0.091

Table 3.1 Lens information with best fit parameters and 1σ confidence for $\kappa_{\rm ext}$ posterior.

Chapter 4

Software Engineering for Big Data Astronomy and Applications to κext Measurement

The focus of my scientific work has been discussed at length in the previous chapters of this dissertation. When I started my first line of sight project, the basic techniques discussed in Chapter 2 were generally well-specified and significant work had been done to quantify sources of systematic uncertainty (see for example Rusu et al., 2017). A focus of mine for the past few years has been building next-generation software tools that could perform this analysis on hundreds of lenses in a reliable and consistent way, and using that capability to model lines of sight at the population level. This has led to a significant amount of time spent learning design patterns, best practices, and technologies for producing high-quality software.

The core scientific philosophy behind much of the software work I have done is a belief in sustainability. In this context, "sustainability" means an explicit focus on the ability of the tools I build to reliably perform scientific analyses that have not yet been designed. In the beginning, this requires a significant investment of time and resources into building high-quality software. In the long term, this initial investment pays off in the form of sustainable long-term productivity.

This overarching *scientific* philosophy breaks down into three core *design* philosophies that make their way into the resultant software: reliability, ease-of-use, and flexibility. In this chapter, I examine the software design principles I have applied to build my analysis tools. I demonstrate how I have used these tools to perform the analyses presented in the previous chapter, and discuss

how these tools will enable me to continue to do original scientific work in the future.

The practical examples in this chapter will focus mainly on the data management and analysis orchestration tools I have developed and used to produce weighted number counts as discussed in the previous two chapters. The process of combining weighted number counts into a posterior and posteriors into a population model are effectively a numerical integration followed by Monte-Carlo sampling. While scientifically important, they are straightforward from a software perspective.

4.1 Principles for Enabling Good Science with Good Software

A primary motivation for my software work has been an intrinsic interest in computing and software design for their own sake. As my understanding of these disciplines and skills has developed, my understanding of their relationship to good science has developed in parallel.

4.1.1 Scientific Advantages of Good Software Design

Good software does not automatically produce good science. However it does make certain aspects of doing good science meaningfully easier. Examples include:

- Management of Systematic Errors: Well-engineered software allows scientists to perform an analysis in a reliable and consistent way. This makes it easier to manage systematic errors. For example in the context of the work presented in Chapter 3, I can guarantee each system in the sample was compared to the same set of lines of sight in the reference survey. If there was a systematic bias in the reference data, it would appear consistently across the sample and would be easy to mitigate.

- Accessibility and Speed of Onboarding: In the long term, it is inevitable that working groups will lose people and gain new ones. Easy-to-use software makes it much easier for new scientists to pick up where someone else left off and minimizes the potential disruption of losing an expert. This is especially important when working with junior scientists who may have minimal background in software and/or are still learning the relevant science details.

- Speed of iteration: As our scientific understanding of a problem matures, the techniques we use inevitably evolve. Good software is able to accommodate these changes in methodology without significant extra work. Reducing the cognitive load around defining an analysis in

software gives the scientist more time to focus on designing the analysis and interpreting the results.

- Reproducibility: Good software makes it easier to reproduce the results of another group or individual. This reproducibility is a core principle of the scientific method. However it also a contributing factor in the principles discussed above. For example, reproducing a previous result can be a good first-step for a junior scientist working on a problem for the first time.

These principles are largely in line with the core philosophy discussed previously. Designing a piece of software with these goals in mind will generally take longer than writing code that gets to a result as soon as possible, but dramatically increases long-term output.

4.1.2 Design Principles

Throughout my work, I have identified three core design principles that guide the process of building new scientific software. These principles should not be taken as prescriptive. There will always be cases where one or more should be relaxed or ignored entirely, but I believe they apply to most scientific software most of the time. These are:

- Reliability: The software should not error except when the user supplies invalid information or its execution environment cannot support its operation. Assuming no such problem arises, the software should perform consistently. In practice, this means efficient and reliable error handling should be a significant concern.

- Ease-of-use: The software should hide as many low-level details as possible from the user. Any configuration or behavior that is the responsibility of the user should be completely separated from the software's core routines to avoid confusion.

- Flexibility: The software should be flexible enough to allow the user to perform novel scientific work without modifying the software's core routines. However it should also be clear what the software's limits are, such that the user can look elsewhere when appropriate.

The implementation of these principles will always depend on the particular piece of software being built. However they should be front of mind from the moment the first line of code is written.

4.2 Software Considered in this Chapter

Over the past several years, I have written several packages that have been essential in the production of the science results presented in the other chapters in this dissertation. Some of these packages have eventually been replaced by newer, more sophisticated packages. In this chapter, I focus on three which represent the current "state of the art" of my implementation of solutions to three specific problems:

- `heinlein`[1] is a pure-python data management package for survey astronomy. It is designed to provide a high-level query interface to any underlying survey dataset. It works with catalog data, bright star masks, and imaging, and includes caching routines for accelerating repeated queries in a given region of the sky. It is the oldest of the software considered in this chapter.

- `cosmap`[2] is a pure-python analysis orchestration tool designed to provide a high-level declarative interface for performing analyses similar to the ones discussed in the other chapters of this dissertation. It handles data management (via `heinlein`), parameter validation, result collection, and distribution of work to all available computing resources. Its core user-facing construct is an *analysis template*, where a user defines a set of transformations to be performed on data samples drawn from from a given region of the sky.

- `godata`[3] is a data management tool designed to simplify the process of tracking data produced by scripts or long-running processes. It handles reading and writing data to and from disk in a consistent manner, regardless of the working directory of the Python script or notebook. It is capable of operating exclusively on Python objects, such that the implementation details of data storage are completely hidden from the user. Its user-facing interfaces and I/O routines are written in Python, while its file system and data tracking algorithms are implemented in Rust. It is the newest piece of software in this list.

All three of these pieces of software have been built with the intention of enabling analyses beyond those that are useful for my work, while still maintaining a sufficiently narrow scope to

[1] https://github.com/AstroPatty/heinlein
[2] https://github.com/AstroPatty/cosmap
[3] https://github.com/AstroPatty/godata

avoid becoming bloated. I will present examples involving all three of these pieces of software to demonstrate the principles I believe to be important for data engineering in an astronomy context.

4.2.1 Definitions

Throughout this chapter, I will discuss a number of concepts that are commonly used in a software engineering context. Below, I provide several definitions of these concepts as they will be used in this chapter.

- **Domain:** The specific set of problems a piece of software intends to solve.
- **Paradigm:** The set of principles guiding how a particular problem in the *domain* will be solved. Examples of programming paradigms include imperative, declarative, functional, and object-oriented. This can be heavily influenced by language selection.
- **Design Pattern:** A set of standard approaches for solving a particular type of problem that occurs frequently when designing and building software. Design patterns are often (though not always) agnostic to the *domain* of the software. However the *paradigm* an engineer is working in will often have significant impacts on which design patterns are available.
- **Interface:** A specification for how two pieces of a program interact, typically in the form of functions or methods. In the context of interactive computing, the interface is often directly used by the programmer.
- **Abstraction:** A part of a piece of software that hides a complex behavior behind a simpler *interface*. Although almost all abstractions *include* an interface, this is only the outward facing layer of the abstraction. Many times, abstractions will be composed to create even larger abstractions.
- **Coupling:** How closely related two pieces of a larger software system are, and how reliant one piece is on the implementation details of another. It is widely considered good practice to reduce coupling when possible, such that two modules with identical interfaces can be swapped without requiring changes to the remainder of the program.
- **Big Data:** A dataset that cannot be managed on a single machine.

- **Data-Driven Astronomy and Data Science:** In this chapter, I use to the term *data-driven astronomy* to refer to astronomical work that is built on top of large datasets, such as galaxy surveys. *Data science* refers to techniques that are often leveraged to analyze large datasets regardless of the problem domain. Although good data science relies on strong domain knowledge, many data-science techniques (such as machine learning and even specific machine learning architectures) can be shared across many different domains.

4.3 `heinlein` and Data Management

Data management is the foundational problem in any analysis system designed to work in a big-data context. This problem exists entirely outside the context of any actual analysis. The dataset in question must be stored across many machines, which must coordinate to respond to incoming queries. The data may also be heterogeneous, involving several different file formats even if the actual data is structurally identical[4]. The management system may cache information that is frequently accessed, and must track the state of many different user sessions at once.

This type of problem is ubiquitous in the private sector. The technologies needed to build such a system are mature and largely open-source. But this does not mean the process of building such a system is trivial. The architect must appreciate the way the system will be used and the constraints of resources it will be deployed on. The process of building a system that will help scientists do original work looks very different from a system that allows the public to browse images of interesting systems, even if the underlying dataset is the same.

This complexity should be hidden from the user whenever possible. Individual scientists should not be writing raw `SQL` or deciding what library should be used to read in a particular data type. Data-driven astronomers should be allowed to focus on what they do best (and likely prefer): designing new analyses and interpreting their results. Any system that promotes this kind of focused work must first solve this problem of data management.

4.3.1 Separating Software from Astronomy

This problem is largely a software and systems design problem. It involves tracking data on disk, database connections, caching, and low-level interfaces to various formats, all of which should be hidden behind a unified query interface with high-level operations that will be familiar to

[4] e.g. CSV files and SQL databases

astronomers. Although much of the details of such a system will look quite similar to analogous systems built for different kinds of data, there are several aspects unique to galaxy surveys that require astronomy expertise to handle correctly including:

- Relationships between data types (such as catalogs and bright star masks)
- Spherical geometry and coordinates
- Interpretation of database columns
- Underlying format of some of the data, such as FITS images

It is crucial to understand where these "bits of astronomy" are relevant and where they are not. For example, the routines that track the location of data on disk (or on the network) do not need to be aware of any of the above concerns. A client-side routine that handles data caching similarly only needs to be aware that it will be receiving and serving Python objects.

4.3.2 Principles of the User-Facing Interface

`heinlein` seeks to provide an easy-to-use interface for querying astronomical surveys. Currently, it is only capable of working with data on a single, local machine.[5] It is written such that routines for network access would be easy to add, but these have not been necessary for the work presented in this dissertation. However despite its limited scope[6], I still approached the process of building `heinlein` as an opportunity to learn how to build a data system that understands astronomy and astronomers.

The interface of `heinlein` was built with ease of use as a core tenent. In particular, the goal was to build something that could be described in the following ways:

- *Accessible:* The interface should be readily available from any part of the program.
- *Declarative:* The interface should allow users to simply state the data they want.
- *Domain-Aware:* The interface should provide high-level operations that are familiar to its users, and return data in familiar formats.

[5] It should also work with data shared via a NFS

[6] And a user count which currently rounds up to one.

- *Consistent:* The interface should behave reliably. Identical input should produce identical output, even if the underlying tool handles the request in a different way. Backend details should not affect the way the interface is used.

Consider the following example usage of `heinlein`.

```
from heinlein import load_dataset
import astropy.units as u

des = load_dataset("des")
data = des.cone_search((62.0905, -53.8999), radius = 120*u.arcsec,
                       dtypes=["catalog", "mask"])
masked_catalog = data["mask"].mask(data["catalog"])
```

This code demonstrates the principles described above, though this is not necessarily obvious from a first glance. The dataset can be loaded quickly and easily from any part of the program with the `load_dataset` command. The user simply states the dataset they want to load, what spatial location they wish to draw from, and which types of data they wish to be included. The `cone_search` operation is a very natural way to interact with a catalog. However it is simply a convenience method that wraps a more general "search in region" method.

The consistency of the interface is best illustrated by a second example:

```
from heinlein import load_dataset
import astropy.units as u

hsc = load_dataset("hsc")
data = hsc.cone_search((141.23246, 2.32358), radius = 120*u.arcsec,
                       dtypes=["catalog", "mask"])
masked_catalog = data["mask"].mask(data["catalog"])
```

This code performs the exact same operation as the one above, but with a different dataset. The backend logic is meaningfully different between the two datasets. The columns in the two

catalogs have different names, including the names for the RA and Dec of the objects. The Hyper Suprime-Cam Subaru Strategic Program dataset ("hsc" in the above listing) uses bright star masks stored as DS9 Regions files, whereas the Dark Energy Survey ("des") uses FITS files. The software must understand these differences and behave accordingly. This complexity appears at several stages throughout the above example. When working with DES, `heinlein` must know in advance which FITS extension contains the mask data. While working with HSC, it must know how to parse regions files and filter them based on location. The business logic for removing objects from the catalog that fall behind the mask is entirely different between these two cases. The user on the other hand, need not know any of this. In fact they *benefit* from not having to worry about these details because they can focus on their science.

4.3.3 Principles of Backend Design

The simple interface discussed above hides a significant amount of backend complexity. When a cone search is performed, the system must ask a series of questions before it can respond to the user such as:

- Does the requested region of the sky fall within the survey footprint?
- If so, are the data from the surrounding region already in the cache?
- If the data must be loaded, does the system currently have access to all the requested data types?
- Is there any relevant configuration information that must be loaded in order to read the data from disk?

The backend must be designed to answer as many of these questions as possible without reference to any configuration that is specific to a given dataset. Any configuration that *is* required for individual datasets should be hidden behind an interface that is consistent across datasets. In particular, there are three pieces of configuration that are relevant for individual galaxy surveys as used from `heinlein`.

- The *regions* on the sky which the data is spread across. In many surveys, these regions correspond to individual pointings of the telescope.

- The data format particular data is in (particularly star masks)
- Any metadata that may be needed to use the given data (e.g. catalog column names)

In `heinlein`, the `dataset` is a high-level abstraction over these complexities that provides a common interface. The `dataset` object itself does not contain any logic that is dependent on the exact survey being used. It is simply an orchestrator which delegates work to the appropriate sub-systems. These subsysystems in turn delegate work to their own subsystems as needed, some of which must be custom built to interface with a single kind of data.

Understandably, the consistency in result does not derive from consistency in process. The standard definition of *declarative programming* is a style of programming where the programmer tells *what* should be accomplished but not *how* it should be done. If we write a python script that runs the same query twice, we would notice that the second run is considerably faster than the first. The `load_dataset` performs initial setup when it is first called, returning an object that is capable of executing queries. If `load_dataset` is called a second time, the initial setup is skipped and a cached version of the object is returned.

But the more significant speedup is in the `cone_search` method. While this method only returns the data in the specified region, it loads a significantly larger region from disk and stores it in a cache. On the second `cone_search` call the data can be returned from the cache, significantly speeding up the response. By loading a larger region into cache, we are indicating that we believe that a user making a request in a given region on the sky is likely to eventually make a similar request from a nearby region on the sky (or perhaps a larger region centered on the same location).

These details are transparent to the user, as they should be. The user will appreciate the speedup, but is not required to know anything about how it is implemented to make full use of it. It is worth noting that these caching routines were built with my particular line-of-sight analysis in mind. `heinlein` was built because I recognized that solving the problem of data management effectively would allow me to build more sophisticated analysis tools that did not have to be worried with these details. But it was still built with a particular analysis in mind, and these lead to some design decisions that I have regretted in the long term.

Most of these decisions were made when the analysis *I* wanted to perform was front of mind, rather than the general design principles I was aiming to implement. For example, data caching is

a very natural feature to add when repeatedly drawing samples from one region of the sky. Instead of making many small database queries, `heinlein` instead performs one large database query and then filters the resultant data as further requests in the same general area of sky are made by the user. This is very useful in certain contexts, but meaningfully *slows down* the response time and results in unnecessary memory usage in cases where a user is only making one or two queries.

4.4 `cosmap`: Analysis Orchestration

With data management handled, we now turn to the problem of analysis orchestration. In this context "orchestration" describes the set of tools responsible for actually *performing* an analysis based on a set of parameters provided by the user. Crucially it must also provide an interface for a user to *define* an analysis they wish to perform. As with the original data management code, the orchestration tools built into `lenskappa` were limited in scope. `lenskappa`, as the name suggests, was built to perform the number counts analysis discussed in previous chapters. It was in large part a piece of software that resulted from a direct translation of the *physical* ideas in the analysis into code.

As the analyses I wished to perform grew more sophisticated, `lenskappa` quickly proved ineffective. For example, it was only capable of performing an analysis on a single lens at a time and could only handle a single set of parameters (such as limiting magnitude and aperture radius) per run. In the past this would have been acceptable, as the rate at which results were needed was quite low. However in the future this will no longer be tenable. The work presented in Chapter 3 is already demonstrating the limits of this approach.

The solutions to these shortcomings became apparent by re-framing the analysis from an engineering perspective. From this angle, the core loop of the weighted number counts analysis can be described as follows:

1. Retrieve data within a set distance of a given point in a large two-dimensional space
2. Compute quantities of interest using the returned data
3. Save the results of the computation
4. Repeat

Many of the engineering problems associated with the above procedure are well understood. There are challenges particular to working with galaxy surveys, but many of those challenges are not relevant to the large-scale design of the software. Furthermore, much of the complexity of working with galaxy surveys is in data management, and this complexity has already been handled by `heinlein`.

4.4.1 Design Principles

`cosmap` is designed to enable large, long-running analyses of similar to those presented in the previous chapters of this dissertation. The design principles presented in Section 4.1.2 therefore translate into several specific design goals that were identified before `cosmap` was written:

1. The system should enable the user to define and perform any analysis that fits the general pattern described above, without modifying the system's core code directly.

2. The system should provide sensible defaults for "bookkeeping" behavior (such as saving results) but allow the user to override them if needed

3. The system should scale to a large number of computing resources without additional work by the user and

4. When provided with invalid input, the system should fail quickly and provide a understandable reason for the failure.

These principles, with the exception of the first, are not related to astronomy in any way. They are a set of core philosophies about how the system should behave. These principles are derived from the overarching idea discussed earlier in this chapter. Astronomers should be able to focus on astronomy. The software system they use should not burden them with computational details when it can be avoided.

As with `heinlein`, `cosmap` is a work in progress. These design principles are not perfectly executed by the software in its current state. However, it is mature enough to perform sophisticated analyses like the one described in the previous chapter.

4.4.2 Configuration Management with `Pydantic`

A core requirement of the above design principles is the ability to define and provide a set of configuration options that are needed to perform the analysis. For example, the weighted number counts analysis removes objects beyond the redshift of the source quasar. This redshift should be provided when the analysis is initialized. If this parameter is missing or invalid (e.g. by being negative) the system should quickly fail.

Configuration management in `cosmap` is provided by `pydantic`. Consider the following simplified example from the weighted number counts analysis:

```python
# file config.py
from cosmap.config.models.sky import SkyCoord
from pydantic import Field, BaseModel


class LensParameters(BaseModel):
    lens_coordinate: SkyCoord
    source_redshift: Field(1.0, ge=0.01)
```

The above class defines a set of parameters that are expected to be provided for any given lens. In particular, the lens must have an associated coordinate and a redshift, the latter of which must be greater than or equal to 0.01. These parameters can be provided to `cosmap` at runtime within a TOML file:

```toml
[lens_parameters.DES0029]
source_redshift = 2.815


[lens_parameters.DES0029.lens_coordinate]
coordinate = [7.419268,-38.240602]
units = "degree"
```

The custom SkyCoord object defined in `cosmap`[7] provides routines for parsing and validating sky coordinates passed to the library as shown. The result of this process is a standard

[7] I generally like to avoid long module paths. This will eventually be moved to `cosmap.models`

`astropy.coordinates.SkyCoord` that will be familiar to anyone who has used Python for astronomical analysis. This parsing and validation is among the first things performed by `cosmap` when an analysis is performed. If a problem is discovered with the provided parameters, the process immediately exits.

These parameter blocks can be nested to create structured configuration that is available to the system while it is running. For an example of a more complete configuration definition, see Appendix A.

4.4.3 Transformations and the Analysis Pipeline

During runtime, a given analysis involves passing a set of data through a series of *transformations*, with the final transformation providing the output. By modeling the pipeline as a series of successive transformations, we allow the user to break down the calculation into a series of logical steps.

```
# File transformations.py
class Main:
    @staticmethod
    def apply_mask(catalog, mask):
        # Remove objects that are close to bright stars
        masked_catalog = mask.mask(catalog)
        return masked_catalog


    @staticmethod
    def count(masked_catalog, lens_redshifts):
        outputs = {}
        for lens_name, lens_configuration in lens_parameters.items():
            mask = catalog["redshift"] < lens_configuration["source_redshift"]
            n_objs = len(catalog[mask])
            outputs[lens_name] = {"count": n_objs}
        return outputs
```

This definition is accompanied by a second file, which defines the relationships between the

transformations and their expected inputs and outputs.

```
{
    "Main": {
        "apply_mask" : {
            "needed-data": ["catalog", "mask"]
        },
        "count" : {
            "dependencies": {
                "apply_mask": "masked_catalogs"
            },
            "needed-parameters": ["lens_parameters"],
            "is-output": true
        }
    }
}
```

The "apply_mask" transformation takes in the sampled survey data provided by `heinlein`. It removes objects from the catalog that are covered by a bright star mask, and returns a new catalog. This catalog is defined as a dependency to the `count` transformation, which aliases it to `masked_catalog` when it is passed into the function. It also takes in the parameters for each of the lens, and uses it to count the number of objects in the line of sight up to the source redshift.

The flow of data through these transformations is handled entirely by `cosmap`, allowing the user to simply focus on defining the series of operations that should be performed on the data. At runtime, these transformations are analyzed as a Directed Acyclic Graph (DAG), which is a common structure in pipeline orchestration. If the transformations can be represented as a DAG, it is mathematically guaranteed that they can be scheduled such that each transformation is only performed once, and dependency transformations can always be evaluated before the dependent transformation. This evaluation must only be performed once, and occurs at the beginning of the runtime. As with invalid configuration, the program will almost immediately exit if the provided transformations cannot be evaluated properly.

The output of the "count" transformation is collected by cosmap and stored after a set number of iterations.

4.4.4 Default Behavior, Plugins, and Compute Scaling

One of the core design principles presented in this chapter is "configurability with sensible defaults." In other words, analysis systems should provide default behavior that is acceptable for the largest number of analyses possible, but provide easy-to-use configurability for edge cases.

A good example of this need arises when producing weighted number counts in the Millennium Simulation. `heinlein` treats the Millennium Simulation (or more accurately, the semi-analytic catalogs) as it would any other sky survey. But when κ was measured in Hilbert et al. (2009) was performed on a pre-defined grid. The samples used to produce weighted number counts must be drawn from this grid, rather than being sampled randomly from the sky. `cosmap` contains built-in routines for grid sampling, but does not know anything about the geometry of the Millennium Simulation.

`cosmap` includes a plugin system that allows the user to overwrite specific pieces of internal behavior at runtime. For example, the Millennium Simulation weighted number counts includes the following file:

```
# file plugins.py
from cosmap.plugins import register

@register
def generate_samples(sampler, *args, **kwargs):
    x_grid, y_grid = ms_sampler._generate_grid(sampler)
    coordinate_tuples = [ms_sampler.get_position_from_index(*index)
        for index in product(x_grid, y_grid)]
    coordinate_inputs = list(zip(*coordinate_tuples))
    ras = list(coordinate_inputs[0])
    decs = list(coordinate_inputs[1])
    coordinates = SkyCoord(ras, decs)
```

```
    return coordinates
```

The functionality in the `ms_sampler` class is not included in this listing, but it contains all the logic necessary to produce samples that fall on the grid points. The `generate_samples` function contains a default implementation within `cosmap` that randomly samples over the provided region. By using the `register` function, we overwrite this default behavior.

However there is an additional piece of behavior which must be overwritten. In a multi-core environment with n cores, `heinlein` splits the sampling region into n sub-regions of equal area. Each core is then responsible for computing the weighted number counts for its given subregion.

However the Millennium Simulation is divided into 64 independent fields, each of the same size. It therefore makes significantly more sense to give one *field* to each core, and only subdivide fields if there are more cores available than the number of fields. As with the sample generator, `cosmap` contains a routine called `generate_tasks` which is responsible for producing task information that will be consumed by the various worker threads. This routine may be overwritten using with the same `register` function used previously.[8]

`cosmap` uses the `Dask`[9] framework to handle distribution of work. `Dask` workers consume tasks, which consist of a series of samples (represented by sky coordinates and apertures) to evaluate. Each worker receives a full copy of all configuration and transformation definitions, allowing it to operate completely independently of other workers. Each worker contains an independent copy of the appropriate `heinlein` dataset object, which requests and manages data for a given worker.[10] Individual samples are sent to workers as chunks. After completing a chunk, the worker sends its results to the main thread and requests a new chunk. The main thread is responsible for collecting and writing outputs.

This decentralized approach has advantages. If a worker fails, it does not necessarily bring down the entire analysis. A failed worker can simply be restarted and provided a new chunk from the queue. However it also has disadvantages. Independent workers mean that `heinlein's` caching routines will inevitable cache multiple copies of the same data across multiple workers. This can

[8] The documentation recommends you *don't* overwrite this unless you really know what you're doing. But there's a clear use case for it and marking a given function as "overwriteable" is very straightforward.

[9] https://www.dask.org

[10] I'd really like to change this, as it is inefficient from a memory perspective.

be mitigated by ordering samples in a way that is aware of data layout and the way data will be cached. `heinlein` provides built-in routines for this purpose.

4.5 Handling Outputs at Scale with `godata`

One of the core motivations behind `cosmap` was the desire to perform an analysis on several lenses, and with several sets of parameters, on a single run. This necessarily produces a large number of output files[11], which must be handled effectively.

The core challenge in the context of `cosmap` is that many processes will be producing output simultaneously. A single process may also produce output that must be spread across multiple files, depending on how the analysis is designed. Sending all this data to a single process responsible for handling output is the most obvious approach, but this can cause memory issues if all worker processes are sending output to a single process at roughly the same time.

`godata` solves this problem by using a central server to track the location of data on disk, while delegating the actual *writing* of data to the individual worker processes producing the output. In keeping with the design principles discussed in this chapter, the interface of `godata` is designed to be easy to use and to hide as many details as possible from the user:

```
from godata import create_project
import numpy as np

project = create_project("my-project", collection="my-collection")
my_cool_data = np.random.rand((100,100))
project.store(my_cool_data, "outputs/my-cool-data")
```

Once the data has been written to disk, it is instantly available in any python interpreter on the system:

```
from godata import load_project

project = load_project("my-project", collection="my-collection")
my_cool_data = project.get("outputs/my-cool-data")
```

[11] The largest test I've done involved around 1200 individual CSV files.

Having a single, centralized authority tracking the status of the project sidesteps many of the challenges of working with Python in a multi-core environment. It ensures that large numbers of processes producing output will not conflict, and hides the details of reading and writing data from the user. Although the examples here demonstrate storing and loading Python objects directly, `godata` is also capable of tracking and loading data that already exists on disk. The `project` interface makes no distinctions between data handled with these two methods. Everything appears to the user as part of a unified virtual file tree.

As with `heinlein`, `godata` is designed to handle a particular menial task very well. By building it to work at large scale, it is also guaranteed to work on small scales. A scientist can use `godata` to track the location of data on their local machine and retrieve it instantly from any Python process, no matter where it is executing. After processing results from a large analysis on an external machine, `godata` can export the project into a single file which can be imported on a second machine, greatly simplifying the process of doing follow-up analysis.

4.6 Putting it all Together

The guiding principle of the packages presented above is simple: *well designed software makes doing science easier and more enjoyable.* Consistent context switching between a scientific objective and the details of implementing it is a drain on mental resources that hurts the quality of both efforts. We can never eliminate this context switching entirely, but we can build software that minimizes it.

`cosmap` represents an attempt to build a system that implements this principle for a particular style of cosmological analysis. While it was used in this dissertation for a particular science goal, it could be applied in a variety of scientific contexts. For example, it could easily be adapted to calculate galaxy two-point statistics across a wide survey region. It could be used to systematically produce cutouts of galaxies that meet a specific criteria, or search for galaxy clusters using a pre-trained machine learning algorithm. It was not designed to perform any of these analyses individually. Instead, I sought to identify the set of problems that were common to many possible analyses that could be performed with a galaxy survey, and built a system to handle those problems systematically.

It is also worth pointing out that there are plenty of cosmological analyses to which `cosmap`

would not be well suited. Identifying these is crucial in building any orchestration tool. A tool that tries to do too many things risks doing none of them well. However even those analyses that are not well suited to `cosmap` will benefit from using `heinlein` if they rely on an underlying astronomical survey dataset. And even those analyses that do not rely on survey data will benefit from using `godata` to manage the outputs they produce. Originally, all three of these functionality were mixed in together in a single software package. By recognizing the set of common problems that must be solved across many different use cases and building software to solve them well, we can create an ecosystem of high-quality tools that eliminate much of the computational busy work of doing science and make the process more enjoyable.

As we look forward to science at LSST scale and beyond, these principles will need to be generalized to much bigger systems. At the scale of the work presented in this thesis, good software design is enough to realize most of the benefits discussed in the chapter. However in the future, there will be a need to build and maintain large computing systems that coordinate many pieces of software across many machines to provide high-quality tools to working astronomers. In large part this will be a practical necessity. Modern survey datasets are simply too large to readily move over the internet, and we should seek to allow as much of the community as possible to access these data and perform original scientific work. Building these kinds of systems will be a significant portion of my postdoctoral work.

None of the software presented in this chapter could be deployed as part of a large, centralized computing system without significant modification. But the principles behind their design generalize well. Reliability is even more important when building large systems that must coordinate with each other and serve the needs of many users simultaneously. Flexibility and ease-of-use enable more people to do more science, a clear win for our community and the world we serve. Good software and systems design is not a silver bullet, but provides a solid foundation for the transformative science 21st century datasets will enable.

Chapter 5

Summary and Conclusions

In this dissertation, I have discussed the current state of environment studies in time-delay cosmography and the contributions I have made over the last several years. In particular, I have presented the software I have developed that allows us to begin to understand these lines of sight at a population level. I have applied these techniques to a sample of lenses from the Strong Lensing Legacy Survey and demonstrated that these lines of sight are drawn from a biased population. Although this result is consistent with previous expectations, my work represents the first time this overedensity has been quantified in terms of $\kappa_{\rm ext}$ with real data.

A consistent throughline in my work is the development of next-generation software tools to compliment the next-generation data sets that will become available to the astronomy community over the coming decades. I have discussed my approach to building software to work with these data sets, and demonstrated the ways it allows me to produce original scientific results that would otherwise be much more challenging.

5.1 Future Work

There are in practice three separate aspects of the work presented in this thesis, each of which requires its own discussion about the future.

5.1.1 Improving Individual κext Measurements

My work has not introduced significant change in the underlying methods we use to model individual lines of sight. Although the process is significantly more streamlined than when I began my work, it is still fundamentally the same process presented in Rusu et al. (2017).

However one important change is the introduction of extreme-value statistics to model lines of sight. This statistical model suggests that the $\kappa_{\rm ext}$ is dominated by a single mass structure along the line of sight. Given that most galaxy clusters contain massive elliptical galaxies near their center, it is reasonable to suggest that this structure is such a group or cluster. If this is the case, than the posterior on $\kappa_{\rm ext}$ *could* be interpreted as a quantification of the range of possible cluster halos that could produce galaxy distributions consistent with the available photometric data.

Quantifying this relationship more precisely may allow us to produce tighter constraints on individual lines of sight. This likely requires more sophisticated summary statistics, including possibly two-point statistics. It should also be possible to test the "most significant structure" hypothesis in simulations, which may provide insight into the best methods for constraining them in real data.

However improved constraints on individual lines of sight may be less important in the context of population modeling. Ultimately, TDCOSMO is moving in the direction of making inferences based on populations of lenses, and the line-of-sight analysis should follow suit.

5.1.2 Continued Work on Population Modeling

A centerpiece of this dissertation was my work modeling populations of lines of sight. While the information available was sufficient to provide evidence that our lenses were drawn from an over dense sample, it was not sufficient to constrain the other population parameters meaningfully. At this stage, the best next step for population modeling of $\kappa_{\rm ext}$ is "same analysis, more data." Evaluating it with a much larger sample should provide an exciting opportunity to further constrain the population and evaluate the efficacy of the model.

One big advantage of using lenses from the Strong Lensing Legacy survey is the homogeneity of the data. Once models of individual lenses have been constructed, we do not make use of the underlying survey data when building a population modeling. Ultimately this should allow us to combine lenses from different surveys. However we should ensure that individual surveys produce consistent results. A good next step would be to analyze a sample of lenses against multiple surveys, and quantify any differences between the results. The software tools presented in this work should make this analysis straightforward.

5.1.3 Next Steps for Cosmological Software

The software presented in this work is capable of significantly more than it has produced. However in the long term, it suffers from a number of shortcomings that will prevent it from truly scaling into LSST-sized analyses.

5.1.3.1 `heinlein` and Data Management

Because of the way `heinlein` is built, it must be replicated inside of every individual core of a machine. This results in significant redundancies in memory usage. A shared-memory backend which serves data to client processes should significantly improve our ability to scale work across extremely large computing environments. However Python is not a good language for this type of problem. In practice, `heinlein` should remain as a front-end client which queries data served by a back-end server process written in a language like Rust or Go. Such a server would likely require only a few cores to serve an entire node worth of worker processes.

Because of the design decisions, these changes could be made and immediately implemented into `heinlein` without changing anything about its interface to `cosmap`. However `cosmap` will also need updates if it is to be used to sample across huge regions of sky.

5.1.3.2 `cosmap` and Analysis Orchestration

`cosmap` is a powerful tool for performing analyses across large survey datasets. However it was built to work within the constraints of a single node on a computing cluster. While it should in principle scale beyond this, it should be thought of as more of a testing ground for the concepts that will go into a much larger system designed to truly operate at the scale of an entire survey.

This system will be much too big to build as a single Python package. It will involve coordination between many machines and certainly cannot be built by a single person. My postdoctoral work at Argonne National Lab will give me the opportunity to start to work on this vision with some of the most knowledgeable and skilled computational scientists in the world.

5.2 Conclusion

In this dissertation, I have demonstrated the ability to estimate κ_{ext} for strong gravitational lenses at scale. I have used this ability to produce the first constraints on the distribution of κ_{ext} across a large population of strong lenses, and demonstrated that these lenses fall in preferentially overdense lines of sight. This is in line with previous expectations.

This ability will prove critical to the future of time-delay cosmography. The field is working hard to adapt to an environment with a much greater volume of data. My work has produced the tools necessary to quickly and reliably estimate the value of $\kappa_{\rm ext}$ for large numbers of individual systems, and a framework for combining those results into an estimate of the properties of the population the lenses are drawn from.

Unfortunately, it is not clear how much I will continue to be involved in TDCOSMO and the greater strong-lensing community in the near term. I plan to use the skills I have learned over the last several years to build tools to accelerate science for the whole astronomy community, which will likely come at the expense of my personal science output. I am content with this trade off, as I believe I can ultimately have a much greater impact by building high-quality tools than I would have focusing most of my time on science. But the ultimate goal is still to understand our universe, and I cannot wait to see what our community will produce in the coming decades.

Appendix A

Details of Weighted Number Counts Implementation in cosmap

This appendix includes a more complete example of the cosmap analysis that is used to perform the weighted number counts for the lens systems. We begin with the definition of the runtime configuration:

```
# file config.py
from cosmap.config.models.sky import SkyCoord, Quantity
from cosmap.config.analysis import CosmapAnalysisParameters
from pydantic import Field, BaseModel

class LensParameters(BaseModel):
    lens_coordinate: SkyCoord
    source_redshift: Field(1.0, ge=0.01)

class Filters(BaseModel):
    limiting_magnitude: float | list[float]
    filters: dict = {}

class GeometryParameters(BaseModel):
    radius: Quantity
```

```
        inner_radius: Quantity

    class Main(CosmapAnalysisParameters):
        n_samples: int = Field(
            1000, description="Number of samples to draw from the sky")
        lens_parameters: dict[str, LensParameters]
        geometry_parameters: GeometryParameters
        to_remove: dict[str, any]
        filters: Filters
```

This is an example of nested configuration. `cosmap` searches for the `Main` configuration block at runtime, but that configuration block may itself be composed of other configuration blocks.

Any needed runtime configuration must be stored inside one of these blocks. But the content of the blocks does not *necessarily* need to be directly provided in the runtime configuration provided by the user. `cosmap` allows for the creation of setup transformations, which allow the analysis to perform additional setup based on user input

```
# file transformations.py
class Setup:


    @staticmethod
    def build_filter_sets(lens_parameters, inner_radius, radius,
                          limiting_magnitude, dataset_name) -> dict:
        if dataset_name not in columns.keys():
            raise ValueError(
                f"Dataset {dataset_name} not found in columns.json")
        column_aliases = columns[dataset_name]
        #make sure radius is a list
        if radius.isscalar:
            radius = [radius]
```

```
    #make sure limiting_magnitude is a list
    if isinstance(limiting_magnitude, float):
        limiting_magnitude = [limiting_magnitude]
    filters = {}
    # Create keys for the filter sets by combining the
    # radius and limiting magnitude
    for lens_name, lens_data in lens_parameters.items():
        for r in radius:
            for m in limiting_magnitude:
                #get limiting magnitude without any a period
                mag_key = str(m).replace(".", "")
                #drop any trailing zeros
                mag_key = mag_key.rstrip("0")
                #make a key by combing the radius and limiting magnitude
                key = f"{lens_name}_{int(r.value)}_{mag_key}"
                filter = build_filter(lens_data.source_redshift,
                                      inner_radius, r, m, column_aliases)
                filters[key] = filter
    return filters
```

As with standard transformations, this setup transformation is accompanied by an associated entry in the `transformations.json` file:

```
"Setup": {
    "build_filter_sets": {
        "needed-parameters": ["geometry_parameters.inner_radius",
            "geometry_parameters.radius", "filters.limiting_magnitude",
            "lens_parameters", "Main.dataset_parameters.dataset_name"],
        "output": true,
        "output-name": "filters.filters"
```

```
        }
    }
```

This transformations produces a series of functions that may be called on the catalogs of the randomly-drawn fields. These functions filter the catalogs based on parameters such as the limiting magnitude and maximum radius. These filters are then stored in the `filters` parameter of the `filters` configuration block, which is accessible to the standard transformations as needed.

We will not include details of all the individual transformations performed during a standard run here. However, it is worth mentioning the way outputs are handled. Any set of transformations must define one output transformation. The return values of this transformation are considered to be the output of the block and are passed to `cosmap`'s output routines. In the case of the analysis discussed here, the output transformation looks like the following:

```
class Main
    # additional transformations not included...
    @staticmethod
    def count(catalogs, sample_region, dataset_name, lens_parameters, *args,
              **kwargs):
        outputs = {}
        column_aliases = columns[dataset_name]
        redshift_alias = column_aliases["redshift"]

        for lens_name, lens_catalogs_ in catalogs.items():
            z_source = lens_parameters[lens_name]["source_redshift"]
            lens_catalogs = lens_catalogs_["lens_catalogs"]
            field_catalogs = lens_catalogs_["field_catalogs"]
            for name, catalog in lens_catalogs.items():
                # compute_weightfns function not included
                lens_weights = compute_weightfns(catalog, redshift_alias,
                                                 z_source)
                field_weights = compute_weightfns(field_catalogs[name],
```

```
                                                    redshift_alias, z_source)

                final_weights = {
                    name: lens_weights[name] / field_weights[name]
                    for name in lens_weights.keys()
                }
                outputs[name] = {
                    "ra": sample_region.coordinate.ra.deg,
                    "dec": sample_region.coordinate.dec.deg,
                    **final_weights
                }

        return outputs
```

and its associated configuration looks like the following. Note that the configuration for the preceding transformations is left out of this listing.

```
"Main": {
    "count": {
        "dependencies": {
            "apply_filters": "catalogs"
        },
        "needed-parameters": ["Main.dataset_parameters.dataset_name",
            "lens_parameters"],
        "is-output": true
    }
}
```

The `apply_filters` transformation listed as a dependency performs the final filtering using the filters produced during setup. There may be multiple filters for each lens, involving a combination of limiting magnitudes and apertures. Each is identified with a unique key, and each unique key is associated with a single output file. In this way, `cosmap` allows the user to perform the analysis not

only on several *lenses* at once, but also on several *combinations of parameters* per lens. `cosmap`'s output systems are robust. In the largest test, `cosmap` successfully manged over 1000 output files for a single run with no signs of instability.[1]

The beauty of this approach is that all the code presented in this appendix may be re-written or swapped out at any time. Tweaking the analysis is as simple as updating a transformation or parameter block.

[1] It's generally bad to produce so many individual files, especially on shared resources. A better approach to output may be in order.

References

Aghanim, N. et al. 2020a, Astronomy & Astrophysics, 641, A6

Aghanim, P. C. N. et al. 2020b, Astronomy & Astrophysics, 641, A6

Aihara, H. et al. 2019, Publications of the Astronomical Society of Japan, 71

——. 2017, Publications of the Astronomical Society of Japan, 70, https://academic.oup.com/pasj/article-pdf/70/SP1/S4/23692189/psx066.pdf, s4

Alpher, R. A., & Herman, R. K. 1948, Nature, 162, 774

Anand, G. S., Tully, R. B., Rizzi, L., Riess, A. G., & Yuan, W. 2022, The Astrophysical Journal, 932, 15

Antal, T., Labini, F. S., Vasilyev, N. L., & Baryshev, Y. V. 2009, EPL (Europhysics Letters), 88, 59001

Birrer, S., Millon, M., Sluse, D., Shajib, A. J., Courbin, F., Koopmans, L. V. E., Suyu, S. H., & Treu, T. 2022

Birrer, S. et al. 2020, Astronomy & Astrophysics, 643, A165

Bom, C. R., Alfradique, V., Palmese, A., Teixeira, G., Santana-Silva, L., Santos, A., & Darc, P. 2024, A dark standard siren measurement of the Hubble constant following LIGO/Virgo/KAGRA O4a, 2404.16092

Bonvin, V. et al. 2016, Monthly Notices of the Royal Astronomical Society, 465, 4914–4930

Bosch, J. et al. 2017, Publications of the Astronomical Society of Japan, 70

Buckley-Geer, E. J. et al. 2020, Monthly Notices of the Royal Astronomical Society, 498, 3241–3274

Cabanac, R. A. et al. 2006, Astronomy & Astrophysics, 461, 813–821

Capranico, F., Kalovidouris, A. F., & Schaefer, B. M. 2013, 1305.1485

Chen, G. C.-F. et al. 2022, Monthly Notices of the Royal Astronomical Society, 513, 2349

Chen, G. C.-F., Fassnacht, C. D., Suyu, S. H., Yıldırım, A., Komatsu, E., & Luis Bernal, J. 2021, Astronomy & Astrophysics, 652, A7

Clerkin, L. et al. 2016, Monthly Notices of the Royal Astronomical Society, 466, 1444–1461

Coles, P., & Jones, B. 1991, Monthly Notices of the Royal Astronomical Society, 248, 1

Collett, T. E. et al. 2013, Monthly Notices of the Royal Astronomical Society, 432, 679

Costanzi, M. et al. 2021, Physical Review D, 103

Coupon, J., Czakon, N., Bosch, J., Komiyama, Y., Medezinski, E., Miyazaki, S., & Oguri, M. 2017, Publications of the Astronomical Society of Japan, 70

Cuceu, A., Farr, J., Lemos, P., & Font-Ribera, A. 2019, Journal of Cosmology and Astroparticle Physics, 2019, 044

Dark Energy Survey Collaboration. 2005, The Dark Energy Survey, astro-ph/0510346

Davis, O., Devriendt, J., Colombi, S., Silk, J., & Pichon, C. 2011, Monthly Notices of the Royal Astronomical Society, 413, 2087, https://academic.oup.com/mnras/article-pdf/413/3/2087/2898212/mnras0413-2087.pdf

De Lucia, G., & Blaizot, J. 2007, Monthly Notices of the Royal Astronomical Society, 375, 2, https://academic.oup.com/mnras/article-pdf/375/1/2/3087832/mnras0375-0002.pdf

Di Valentino, E. et al. 2021, Classical and Quantum Gravity, 38, 153001

Dressler, A. 1980, ApJ, 236, 351

Dyson, F. W., Eddington, A. S., & Davidson, C. 1920, Philosophical Transactions of the Royal Society of London Series A, 220, 291

Eigenbrod, A., Courbin, F., Dye, S., Meylan, G., Sluse, D., Vuissoz, C., & Magain, P. 2006, A&A, 451, 747

Einstein, A. 1915, Sitzungsberichte der Königlich Preussischen Akademie der Wissenschaften, 844

Fassnacht, C. D., Gal, R. R., Lubin, L. M., McKean, J. P., Squires, G. K., & Readhead, A. C. S. 2006, The Astrophysical Journal, 642, 30

Fassnacht, C. D., Koopmans, L. V. E., & Wong, K. C. 2010, Monthly Notices of the Royal Astronomical Society, 410, 2167

Fassnacht, C. D., Xanthopoulos, E., Koopmans, L. V. E., & Rusin, D. 2002, The Astrophysical Journal, 581, 823

Ferlito, F. et al. 2023, Monthly Notices of the Royal Astronomical Society, 524, 5591–5606

Foreman-Mackey, D., Hogg, D. W., Lang, D., & Goodman, J. 2013, Publications of the Astronomical Society of the Pacific, 125, 306

Freedman, W. L. et al. 2019, The Astrophysical Journal, 882, 34

Gomer, M., & Williams, L. 2020, Journal of Cosmology and Astroparticle Physics, 2020, 045–045

Greene, Z. S. et al. 2013, The Astrophysical Journal, 768, 39

Gwyn, S. D. J. 2012, The Astronomical Journal, 143, 38

Hernández-Aguayo, C. et al. 2022

Hilbert, S., Hartlap, J., White, S. D. M., & Schneider, P. 2009, Astronomy & Astrophysics, 499, 31

Holzman, B. et al. 2017, Computing and Software for Big Science, 1

Hsieh, B. C., & Yee, H. K. C. 2014, The Astrophysical Journal, 792, 102

Hubble, E. 1929, Proceedings of the National Academy of Science, 15, 168

Hubble, E. P. 1925, The Observatory, 48, 139

Inada, N. et al. 2003, The Astronomical Journal, 126, 666

Ivezić, v., et al. 2019, The Astrophysical Journal, 873, 111, 0805.2366

Kaiser, N., Squires, G., & Broadhurst, T. 1995, ApJ, 449, 460, astro-ph/9411005

Kelly, P. L. et al. 2023, Science, 380

Knox, L., & Millea, M. 2020, Physical Review D, 101

Kundić, T. et al. 1997, The Astrophysical Journal, 482, 75

Kusnierz, J., Padulano, V. E., Malawski, M., Burkiewicz, K., Saavedra, E. T., Alonso-Jorda, P., Pitt, M., & Avati, V. 2022, in 2022 22nd IEEE International Symposium on Cluster, Cloud and Internet Computing (CCGrid) (IEEE)

Leavitt, H. S., & Pickering, E. C. 1912, Harvard College Observatory Circular, 173, 1

Lemaître, G. 1927, Annales de la Société Scientifique de Bruxelles, 47, 49

Liu, Y., Oguri, M., & Cao, S. 2023, Phys. Rev. D, 108, 083532

McCully, C., Keeton, C. R., Wong, K. C., & Zabludoff, A. I. 2017, The Astrophysical Journal, 836, 141

Mikelsons, G., Silk, J., & Zuntz, J. 2009, Monthly Notices of the Royal Astronomical Society, 400, 898

Millon, M. et al. 2020, Astronomy & Astrophysics, 639, A101

Nishizawa, A. J., Hsieh, B.-C., Tanaka, M., & Takata, T. 2020

Papalexiou, S. M., & Koutsoyiannis, D. 2013, Water Resources Research, 49, 187, https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2012WR012557

Park, J. W., Birrer, S., Ueland, M., Cranmer, M., Agnello, A., Wagner-Carena, S., Marshall, P. J., & Roodman, A. 2023, The Astrophysical Journal, 953, 178, 2211.07807

Penzias, A. A., & Wilson, R. W. 1965, ApJ, 142, 419

Perlmutter, S. et al. 1999, The Astrophysical Journal, 517, 565–586

Planck Collaboration. 2020, Astronomy & Astrophysics, 641, A6

Refsdal, S. 1964, MNRAS, 128, 307

Riess, A. G., Casertano, S., Yuan, W., Bowers, J. B., Macri, L., Zinn, J. C., & Scolnic, D. 2021, The Astrophysical Journal Letters, 908, L6

Riess, A. G. et al. 1998, The Astronomical Journal, 116, 1009–1038

——. 2022, The Astrophysical Journal Letters, 934, L7

Rubin, V. C., & Ford, W. Kent, J. 1970, ApJ, 159, 379

Rusu, C. E. et al. 2017, Monthly Notices of the Royal Astronomical Society, 467, 4220, https://academic.oup.com/mnras/article-pdf/467/4/4220/11041001/stx285.pdf

——. 2019, Monthly Notices of the Royal Astronomical Society, 498, 1440

Schneider, P., & Sluse, D. 2013, Astronomy & Astrophysics, 559, A37

Shajib, A. J. et al. 2023, Astronomy & Astrophysics, 673, A9

Sluse, D. et al. 2019, Monthly Notices of the Royal Astronomical Society, 490, 613

Springel, V. et al. 2005, Nature, 435, 629

Suyu, S. H., Marshall, P. J., Auger, M. W., Hilbert, S., Blandford, R. D., Koopmans, L. V. E., Fassnacht, C. D., & Treu, T. 2010, The Astrophysical Journal, 711, 201

Tanaka, M. et al. 2017, Publications of the Astronomical Society of Japan, 70, https://academic.oup.com/pasj/article-pdf/70/SP1/S9/23692265/psx077.pdf, s9

Taruya, A., Takada, M., Hamana, T., Kayo, I., & Futamase, T. 2002, The Astrophysical Journal, 571, 638–653

Tihhonova, O. et al. 2020, Monthly Notices of the Royal Astronomical Society, 498, 1406

——. 2018, Monthly Notices of the Royal Astronomical Society, 477, 5657

Treu, T., & Marshall, P. J. 2016, The Astronomy and Astrophysics Review, 24

Verma, A., Collett, T., Smith, G. P., Collaboration, S. L. S., & the DESC Strong Lensing Science Working Group. 2019, Strong Lensing considerations for the LSST observing strategy, 1902.05141

Vuissoz, C. et al. 2008, Astronomy & Astrophysics, 488, 481–490

Wagner-Carena, S., Park, J. W., Birrer, S., Marshall, P. J., Roodman, A., & Wechsler, R. H. 2021, The Astrophysical Journal, 909, 187

Weinberg, D. H., Bullock, J. S., Governato, F., Kuzio de Naray, R., & Peter, A. H. G. 2015, Proceedings of the National Academy of Sciences, 112, 12249–12255

Wells, P., Fassnacht, C. D., & Rusu, C. E. 2023, Astronomy & Astrophysics, 676, A95

Wong, K. C. et al. 2018, The Astrophysical Journal, 867, 107

Xavier, H. S., Abdalla, F. B., & Joachimi, B. 2016, Monthly Notices of the Royal Astronomical Society, 459, 3693–3710

Zwicky, F. 1933, Helvetica Physica Acta, 6, 110