

Lens Modeling in SpaceWarps

First Author,¹ Second Author,² and Third Author³

ABSTRACT

In Spacewarps, volunteers are invited to search sky surveys for lens candidates. Here we report on a system that allows experienced volunteers to model lens candidates. A sample of 29 simulated lenses were modelled, each multiple times. The quality of the models was then examined in different ways. In particular, (i) the image parities and time orderings were identified correctly in most cases, but not always; (ii) the mean convergence (equivalent to the enclosed mass) was quite well constrained in the image region; (iii) the performance of experienced volunteers was comparable to professionals.

Subject headings:

1. Introduction

Gravitational lenses (GLs), predicted by General Relativity (GR), can be used as a tool by astronomers to examine many properties of the cosmos. They allow for example to estimate the masses and their distribution of galaxies, automatically including the hard to grasp dark matter. Further more, they allow an estimate on cosmological parameters like the Hubble constant (Saha et al. 2006) and the mass density \bullet .

How to find gravitational lenses? Huge amount of image data from surveys that needs to be processed. Even more coming in the future with new surveys \bullet Robotic procession has [cit](#) been suggested \bullet , and tested \bullet , but has failed so far to be convincing \bullet . Another suggestion [cit](#) involves humans, volunteers ¹. SpaceWarps uses this approach with great success so far. \bullet [cit](#)

Next step involves modeling. that needs advanced knolegde and takes a lot of time. Too much time to be done by astronomers them selves as the results from new surveys and [cit](#)

¹First place

²Second place

³Third place

¹A volunteer is considered everybody that has no background in astro physics.

identifiers like SpaceWarps come in. So there is a demand of a means to modell a great amount of identified lens data, that scales with increasing data amount.

The purpose of this study was to provide a means to model a large amount of gravitational lenses by showing that gravitational lens modelling can be learned / done by volunteers. We suggest, that volunteers will be as successful as professionals with modelling if provided with an easy to use tool with visual feedback (What you see is what you get, WYSIWYG) and a minimal set of instructions. Volunteers will then crowd work (using buzz word here ;)) / work collaborative on modelling lenses from several sources / groups at a central place. Since this is a iterative learning process, the more involved volunteers will quickly gain knowledge that can be passed down to new volunteers. That creates a social structure that scales well with the number of volunteers, as other projects have already shown. Finding people working as volunteers has been shown to be successful last but not least by SpaceWarps and the whole galaxy zoo project. To test the peoples abilities, we investigated the performance of a first set of volunteers modelling a set of simulated lenses. We tested the ability to correctly identify lensed images and reproduce similar mass maps of the lens.

2. A lens modelling program

It is well known that making a scientific contribution is a central motivation cited by citizen-science volunteers. So for a lens modeller in a citizen-science project, nobody wants that it should sacrifice scientific usefulness in order to be fun to use. That said, being aesthetically pleasing and having a short initial learning curve are also essential qualities. Also desirable is that the user is encouraged/challenged to go deeper and wider into the subject. Yet another aspect is enabling small incremental contributions by different people. SpaghettiLens tries to address all of these.

2.1. Theory: Fermat’s Principle

There are several ways to understand the formation of arcs and multiple images in gravitational lensing. We will follow some ideas originally introduced by Blandford & Narayan (1986), based on Fermat’s principle. The key to this approach is an abstract construct called the *arrival-time surface*. This surface cannot itself be observed, but several observable quantities can be derived from it.

Consider a gravitational lens. As in most astrophysical lensing, this lens is ‘thin’ along

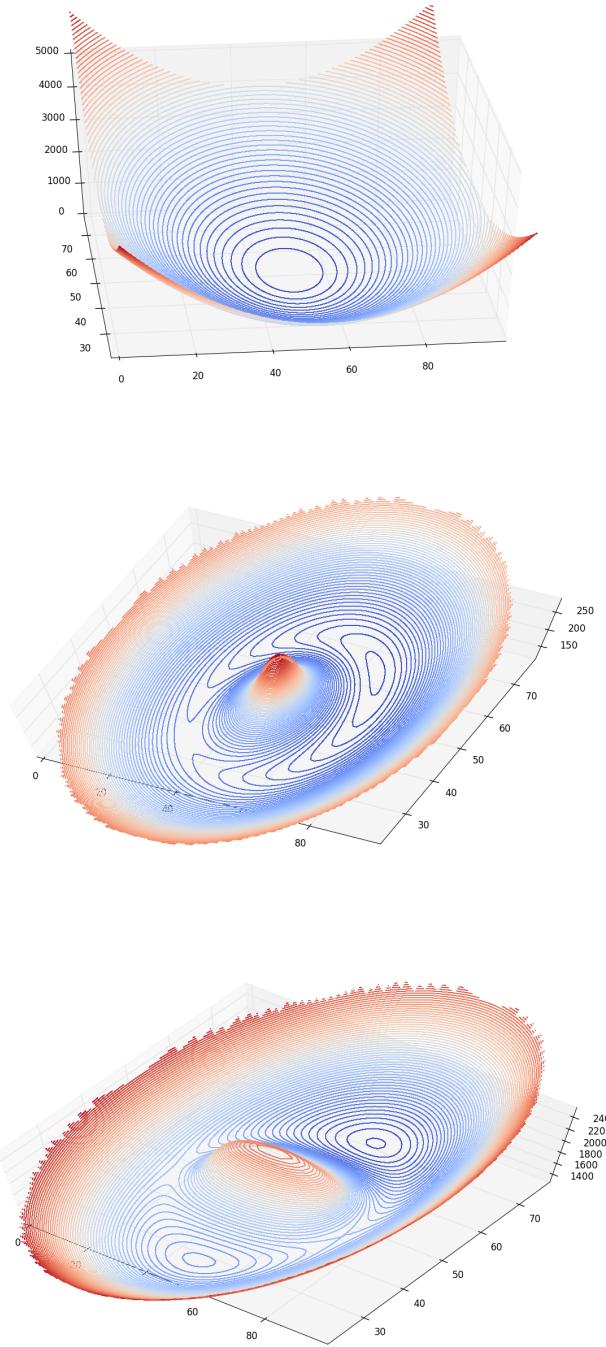


Fig. 1.— Arrival-time surfaces, with contour levels of equal arrival time. The upper surface is with no lens. The middle surface is when a circular lensing mass (offset from the source) is added; a maximum, a minimum and a saddle point can be seen. The last surface is the result of an elongated lensing mass; a maximum, two minima and two saddle points can be seen.

the line of sight, and effectively lies on a plane. Let x, y be coordinates on this plane. These coordinates measure ordinary physical lengths (metres, etc) on the lens plane. Let there be a point source of light at $x = 0, y = 0$ but far behind the lens plane. Now imagine a light ray coming from the source to the lens plane (x, y) and then changing its path to reach the observer. The observer would see the source apparently behind (x, y) on the lens. If this happened, the light ray would have taken a longer path than having come directly to the observer. The geometrical path difference would be $\propto x^2 + y^2$ for small angles. Let us define

$$A_{\text{geom}}(x, y) = \frac{1}{2}(x^2 + y^2). \quad (1)$$

As written, this looks is an area. But we can also think of it as a time delay introduced by the light ray bending — with an unwritten constant factor. That factor is basically the lens distance times the speed of light; the precise expression has to take the expansion of the Universe into account, and is given in the Appendix. The top part of Figure 1 shows what A_{geom} looks like: it is just a paraboloid. The minimum of the surface is at the source, and that is where, according to Fermat’s principle, the image will be. The rest of the arrival-time surface is just an imaginary surface.

The geometrical time delay (1) is not the only one. The warping of space by a gravitational field introduces a further time delay A_{grav} . The arrival time of the light ray, compared to what it would have been with no lensing, is

$$\text{arrival time} = A_{\text{geom}} + A_{\text{grav}}. \quad (2)$$

We can also express A_{grav} as an area, with the same implicit constant factor that makes it a time. The expression, however, is more complicated. We first introduce the notation $\nabla^2 f(x, y)$ to denote

$$\frac{f(x + \Delta x, y) + f(x - \Delta x, y) + f(x, y + \Delta y) + f(x, y - \Delta y) - 4f(x, y)}{\Delta x \Delta y} \quad (3)$$

in the limit of $\Delta x, \Delta y$ small. Then we write the projected mass density of the lens, which would be in kg m^{-2} , in special units (given in the Appendix) such that the surface density becomes a dimensional field $\kappa(x, y)$. Then

$$\nabla^2 A_{\text{grav}}(x, y) = -2\kappa(x, y). \quad (4)$$

Note that $A_{\text{grav}}(x, y)$ is not given explicitly in terms of $\kappa(x, y)$, but as a differential equation that must be solved. Equations of the type (4) are, however, well known (as Poisson equations in two dimensions), and there are techniques for solving them. Incidentally, κ has a second

meaning, as well as being a dimensionless form of the projected density: it also measures how a bundle of light rays is brought together by gravitational lensing, and is called the *convergence*.

What is the effect of mass on the arrival-time surface? If the mass distribution $\kappa(x, y)$ is circular, $A_{\text{grav}}(x, y)$ will have a peak at the centre of that circle. But if the mass is not precisely in front of the source, the arrival-time surface will not be circular any more. The result is illustrated in the middle part of Figure 1. In addition to the maximum, there is now a minimum and a saddle point. According to Fermat’s principle, images appear at maxima and saddle points as well as minima. If the mass distribution is non-circular, more images can appear. The lower example in Figure 1 shows the arrival-time surface due to an elongated mass. There is still a maximum, part of an elongated hill, and around it there are two minima and two saddle points.

So far we have considered a single point source. What if there is an extended source? To understand this, let us consider what happens if we move the original point source slightly, or equivalently, keep the original point source behind $x = 0, y = 0$ and move the lens slightly. The contours of constant arrival time will, naturally, move slightly, and so will the images. The movement of the contours will be most noticeable where the contours are far apart, that is, where the arrival-time surface is nearly flat. As is evident from Figure 1, this is the region where the minima and saddle points lie, or near the images. In this region, points on the source that are close together produce images that are comparatively far apart. In other words, the image is highly magnified. In summary, lower curvature in the arrival-time surface for a point source implied larger magnification of an extended source. Conversely, where the arrival-time surface is strongly curved, the image will be demagnified. We see from Figure 1 that the arrival-time surface tends to be highly curved near the maximum. Hence maxima tend to be demagnified. In practice, maxima of the arrival time are nearly always too faint to see. The minima and saddle points dominate.

In the lensed examples in Figure 1, the maximum in each case is evident, but how can we distinguish the minima from the saddle points? The key is to examine the contours of equal arrival time. Contour curves loop around a minimum or a maximum, but near a saddle point, the contours do not loop around, instead they approach and then pull away. The contour curve precisely level with a saddle point forms an X at the saddle point. In Figure 1, one such X is evident. In other cases the X contour passing through the saddle point is not itself plotted, but we can tell the other contours where the X would be. These self-crossing contours will play a special role in the following.

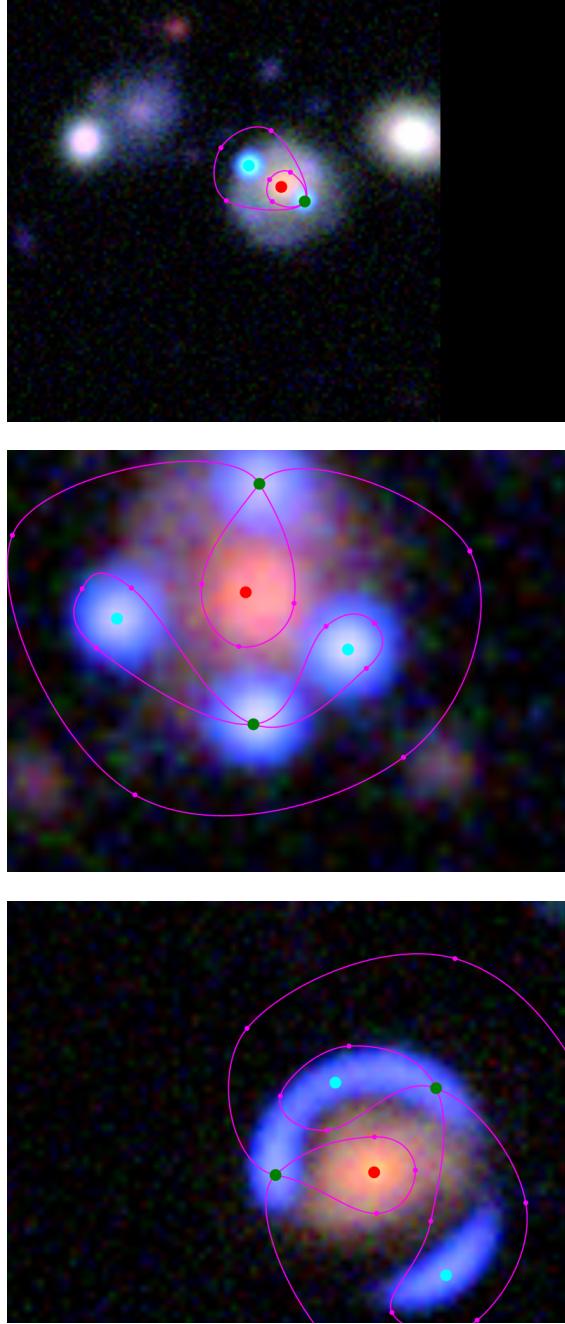


Fig. 2.— Examples of Spaghetti input. The models from these appear later in Figures ??, ?? and ??.

2.2. SpaghettiLens

SpaghettiLens implements the ideas of the arrival-time surface into a modeling program.

The first modeling step with SpaghettiLens is for the user to call up the SpaceWarp of interest, and make an educated guess for the topography of the arrival-time surface — specifically, the locations of the maxima, minima and saddle points that correspond to images of the brightest part of the source. Input is given by tracing a proposal for the saddle-point contours. Figure 2 shows some example inputs — and incidentally also indicates the origin of the name SpaghettiLens.

SpaghettiLens, as implemented so far, assumes that the lens is dominated by a single galaxy, and that the centre of that galaxy is the sole maximum. Once the maximum has been identified, we wish to characterize the rest of the arrival-time surface in relation to it. From geometry, any surface that has a central maximum and goes high at the edges must have a limaçon-like contour, meaning a two-looped curve with one loop inside the other. The maximum lies within the inner loop, a minimum lies between the two loops, and a saddle point lies at the self-intersection of the contour. The top panel of Figure 2 shows an example: red marks a suggested maximum, green a saddle point, and blue a minimum. (The small pink dots are just help sketch the proposed contour.) The middle and lower panels of Figure 2 show a more complex situation. The minimum has split into three images: two new minima with a saddle point in between. In this situation the contour through the new saddle point forms a figure of eight, its two loops enclosing the two new minima. The configuration of a limaçon, possibly with a figure of eight inside, suffices to give a preliminary description of the arrival-time surface for nearly all cases where the lens is dominated by one galaxy, and the source is a single small object.

The exact placement of the pink spaghetti loops shown in Figure 2 has no significance. Their utility is just to help mark up the image as maximum, minima and saddle points. As the figure suggests, the marking up can be done very easily with a mouse. Human intuition is required to (a) identify lensed images and separate them from other background light sources, and (b) classify and order images according to arrival time. But the computer helps by allowing only valid lensing configurations to be entered, and ensuring that the odd image theorem is taken care of.

In the second modelling step, once the user has sketched a proposed spaghetti configurations, SpaghettiLens sends the input to its server-side modelling engine, called GLASS. The task of GLASS is to find a mass distribution $\kappa(x, y)$ that exactly reproduces the locations of the maximum, minima and saddle points. Now, this criterion by itself is extremely under-determined — there are infinitely many mass distributions that will reproduce a given set

of maxima, minima and saddle points, but typically they (a) produce lots of extra images, and (b) look very unlike galaxies. Additional assumptions (a prior) are necessary. GLASS uses a prior based on suggestions by Saha & Williams (1997).

1. The mass distribution is built out of non-negative tiles of mass. (Sometimes these tiles are called mass pixels, but we should emphasize that they are unrelated to image pixels, and are much larger.)
2. There is a notional lens centre, say (x_0, y_0) which is identified with the maximum of the arrival time. The source can have an arbitrary offset with respect to the lens centre.
3. The mass distribution must be centrally concentrated, in two respects. First, the circularly averaged density must fall away like

$$[(x - x_0)^2 + (y - y_0)^2]^{-1/2}$$

or more steeply. Second, the direction of increasing density at any (x, y) can point at most 45° away from from (x_0, y_0) .

4. The lens must symmetric with respect to 180° rotations about (x_0, y_0) . This symmetry assumption can be relaxed if the user wishes.

There are still infinitely many models that satisfy both data and prior constraints, but now they are more credible as galaxy lenses. It is then possible to generate an ensemble of models. The sampling technique used by GLASS is described in (Lubini & Coles 2012), and improves upon earlier techniques (Williams & Saha 2000; Saha & Williams 2004). Typically, ensembles of 200 models are used. That is to say, what we call a SpaghettiLens model is really the mean of an ensemble of 200 models, and its estimated uncertainty is the range covered by the whole ensemble.

In the third modelling step, having generated a model ensemble, SpaghettiLens post-processes it to present results and diagnostics to the user for inspection. This takes the form of three figures.

1. A grayscale plus contour map of the mass distribution.
2. A contour map of the arrival-time surface.
3. A synthetic image of the lensed features.

After examining this feedback, the user can archive the results for discussion, or modify their input and try again, or discard the attempt altogether.

The fourth modelling step is discussion among modellers and iteration on the model. Any archived model can be revised by another user: they can modify the spaghetti configuration slightly or drastically, or change options like the size of the mass tiles. Particularly interesting lens candidates lead to trees of models in this way. Forum discussion then prunes the tree, focussing attention on one or a few models.

3. A lens modeling challenge

Interested volunteers from the SpaceWarp forum were initially introduced to SpaghettiLens through a video tutorial and by videocon. After this introductory stage, a modelling challenge was presented. This consisted of 29 simulated lenses (sims) covering a range of lensing configurations.

3.1. The simulated lenses

The sims were generated by AM, in consultation with PM and AV. In the interest of blind testing, the information in this section was revealed neither to RK, while choosing the challenge set, nor to the people making models (EB, CC, CM, JO, PS and JW) until they were done.

The sims produced using `gravlens` (Keeton 2001a,b) and were of three kinds, as follows.

- *Quasar*: A singular elliptical isothermal plus constant external shear, with a circular Gaussian source.
- *Galaxy*: Lens as with quasar, but elliptical de Vaucouleurs source.
- *Galaxy*: Lens is circular NFW plus one dominant elliptical SIE and perturbing elliptical SIEs, source as with galaxy.

Singular isothermal is in equations (33-35) of Keeton (2001a) with core radius set to zero. NFW is equations (48,50) of the same paper, and shear is the γ term in equation (76) of that work.

3.2. Some example models

Figure ?? shows a simple example.

Figure ?? shows an example where substructure introduces a complication, but model ok.

Figure ?? shows a case where substructure leads to a poorer model.

Figure ?? shows incorrect identification, but the enclosed mass still quite well recovered.

Figure ?? shows a nice symmetric quad.

Figure ?? shows a long-axis quad.

Figure ?? shows a short-axis quad.

Figure ?? shows an inclined quad.

3.3. The test setup

To estimate the performance of the volunteers and the quality of the generated models, two test T1 and T2 were done.

T1 tested the volunteers ability to reconstruct the arrival time surface given a survey image containing a sim. This task consists of two parts. First, the correct identification and location of lensed images (T1a). Second to find the correct ordering in respect of the arrival time for the identified lensed images (T1b).

While we expected T1a to be trivial, given the nature of the survey images and the success of SpaceWarps, we expect T1b to be more difficult. T1b tests the volunteers understandings of the theory of arrival time surfaces and the odd number theorem. While we can provide the volunteers with some general rules of thumb, T1b involves imagination and guessing and therefore training could improve their skills in a later stage.

T1 was also designed to give some feedback on the difficulties volunteers encounter, to further improve the tutorial materials.

The second test T2 was to compare the desired results of a modeling process, the mass distribution of the lens $\kappa(x, y)$. To get a means of comparing the simulated data (sims) to the generated modeled data (models), the total convergence, also called enclosed mass $\kappa_{\text{encl}}(r)$ for both was calculated. The Einstein radius Θ_E is defined by $\kappa_{\text{encl}}(\Theta_E) = 1$ and gives a number that allows the comparison between the sims and the models. We also let

an expert model three selected systems to compare the results from volunteers to those of a professional.

- Should I write down prior expectation for T2 too?

!

3.4. First test T1

The evaluation of the volunteers performance for T1 was done manually, comparing their input from SpaghettiLens and the resulting reconstructed arrival time surface contour line plot (arrival plot) to the arrival plot generated using simulation parameters. Figure 3 shows the setup used to evaluate the models.

To evaluate T1a and T1b, each model was evaluated with two binary tests. The first is passed, if all the images have been identified and are approx. within $\pm 0.05 \cdot \text{imgage width}$. The second test is passed, if those identified points have the right parity, ordering with respect to arrival time.

Additionally, ten types of errors (E01 – E10, listed in Table 1) that could occur were analyzed, where each model could contain multiple of those. The complete table with the results for each model can be found in the appendix, Table ??.

In Table 1 a summary of this evaluation is presented. We conclude that the volunteers are performing very well identifying and positioning images (T1a), with a performance of 92% (R1, $p=0.92$). Most of the problems where due to unclear arc-like structures (E01, $p=0.18$; E04, $p=0.03$; E08, $p=0.04$). Critical errors like the failure to identify all five images in an five image system (E03, $p=0.04$) or to include too many images (E10, $p=0.01$) did almost never happened. From this we conclude that the introduction materials was adequate and the volunteers understand the basics of gravitational lensing.

- Add more details like total amount of images detected / tot am images (rightPlace add more detail? fraction)??

The assignment of the parity of the images (T1b) was a more difficult task. In 59% (R2, $p=0.59$, $N=70$) of the cases the volunteers succeeded to identify the right configuration. Most of the failures are due to E06, with $N=38$, $p=32\%$, followed by E05 (PI) with $N=7$, $p=6\%$. For an example of E06, see Figure ???. It basically describes a situation, where the minima and saddle points of a five image configuration were exchanged (rotated by $\pm 90^\circ$). E05 describes the situation, where the ordering of the saddle points was wrong (rotation by 180°). While these errors occurred, we suspect they can be avoided with better training material and some examples for the obvious cases. For more challenging cases, like very

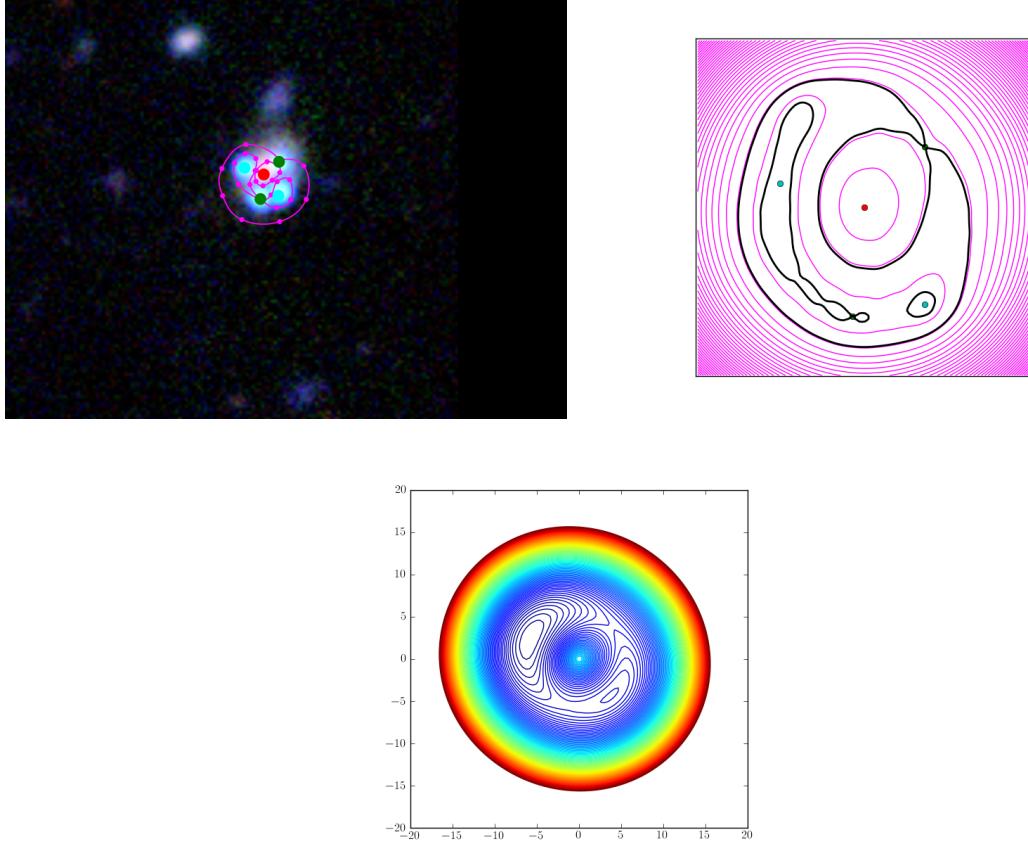


Fig. 3.— Examples of Spaghetti input (left top) and modeled arrival time surface (right top) vs reconstructed arrival time surface from simulation parameters (bottom).

		n	p
N:	Total number of models	119	1.00
R1:	images approx. on right location	110	0.92
R2:	images with correct parity	70	0.59
E01:	inaccurate in arc	21	0.18
E02:	wrong parity in 3 lens conf.	2	0.02
E03:	identified 3 of 5 imgs.	5	0.04
E04:	modeled arc with single img.	4	0.03
E05:	π rotated parity	7	0.06
E06:	$\pi/2$ rotated parity	38	0.32
E07:	missed faint img.	1	0.01
E08:	too many imgs in arc.	5	0.04
E09:	missed double img	3	0.03
E10:	too many imgs.	1	0.01

Table 1: T1 evaluation statistics

symmetrical distribution of the lensed images (for example model 7022, Figure ??), those errors should still produce plausible results, as will be explained in the next section.

3.5. Second test T2

To compare the enclosed mass profile and the Einstein radius of the simulation and the models, $\kappa_{\text{encl}}(r)$ was calculated using the mass map $\kappa(x, y)$ directly generated in the modeling process. From the ensemble of models generated by one modeling process, the mean is taken as the resulting $\kappa(r)$ to calculate θ_E . To estimate the errors, also the extremal models are used to estimate a lower and upper limit for θ_E . These results can be seen in Figure 4. This figure shows that this technique of estimating the error is underestimating the error significantly and should be improved for further analysis.

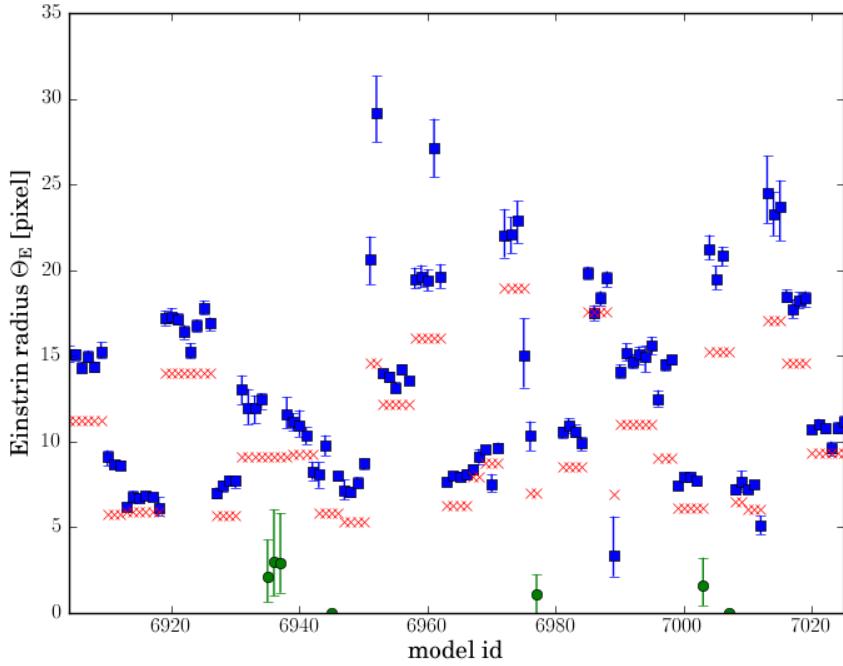


Fig. 4.— Einstein radius Θ_E for all models with estimated errors in blue squares, Θ_E of simulation in red crosses

In Figure 5 can be seen, that the calculated Einstein radius Θ_E of the models tend to be too high. The overshoot varies from around 0.2 to 0.4 for for good models. One of the reasons for this is, that it's hard to get the center of the lens on spot. An offset leads to a a flatter mass profile for the model compared to the simulation.

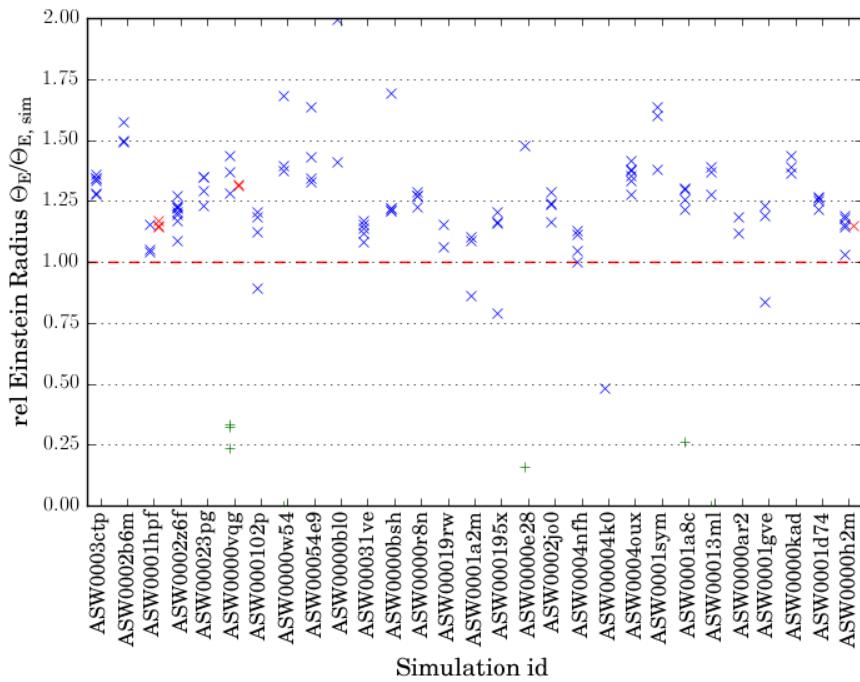


Fig. 5.— relative Einstein radius $\Theta_E / \Theta_{E, \text{sim}}$ for models by volunteers (blue cross), models made by an expert (red cross with offset), including rejected models (green squares); binned per sim.

E05 and E06 happen mostly in very symmetric images, that are hard to come up with a unique valid solution. These errors change the orientation of the mass distribution $\kappa(x, y)$, but $\kappa_{\text{encl}}(r)$ and thus the Einstein radius Θ_E is not influenced that much, and thus the final result is still a valid model. This can be seen by comparing the results for ASW0000h2m, shown in Figure 6: The correct model 7022 ($\Theta_E = 10.76px$), which was done by an expert, and 7020 ($\Theta_E = 10.72px$), 7024 ($\Theta_E = 10.80px$), 7025 ($\Theta_E = 11.16px$), 7021 ($\Theta_E = 11.04px$). The first three are of type E06, while the last is of type E05. This can be easily corrected, if further analysis is done for the modeled system and time delays are known.

Comparing the models of volunteers and experts can be done in Figure 6, where only the expert got the right configuration, but all the resulting models are comparable, besides rotation.

Two additional sims were modeled by an expert, ASW0001hp and ASW0000vqg. Looking at the results for Θ_E for those models in Figure 5, we conclude that the performance of volunteers (blue crosses) and experts (red crosses, offset) is comparable. Note that the models of ASW0000vqg with Θ_E, rel around 0.25 (6935 – 6937) are failed models that show the attempts of a single user, that came finally up with model 6938 as final result.● remove?

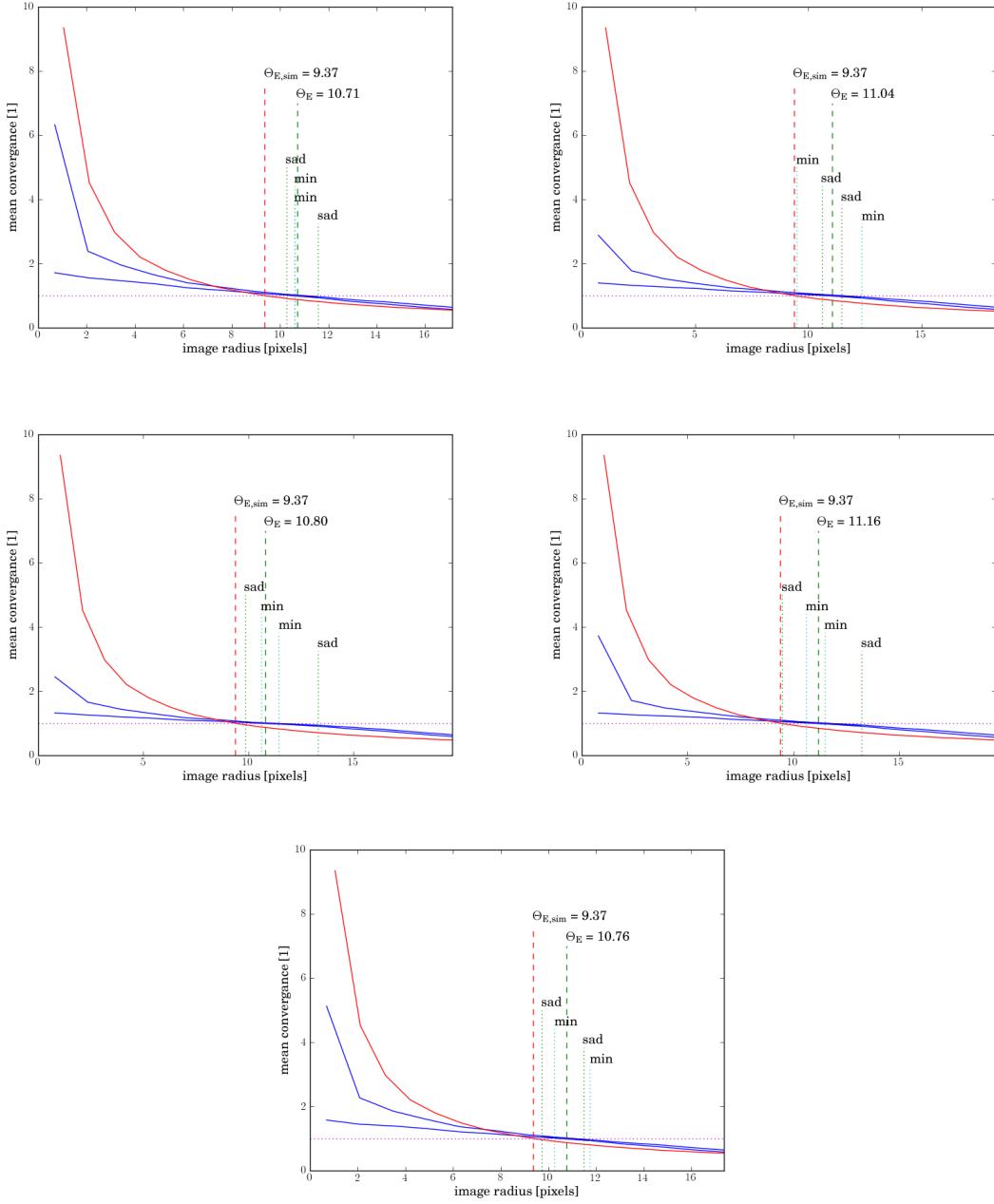


Fig. 6.— $\kappa_{\text{encl}}(r)$ for models of ASW0000h2m: 7020 (top left), 7021 (top right), 7024 (mid left), 7025 (mid right) by volunteers and a correct model 7022 (bottom) by an expert.

4. Extensions needed

As seen in Section 3.4, volunteers fail mostly in two situations: when to identify an arclike structure while placing the points and in 5 image configurations, that are quite symmetric, then to identify the correct ordering of the points.

The first can be improved by additional images from different filters (?) Will be implemented with a next major SpaghettiLens upgrade.

The second can be made less by better training (introduce the ruler), maybe more filters help too?

We hope that those problems are also taken care of when starting collaborative modelling. (Already in progress) Basics are available ('revise functionality') but better community tools would help (having overview, comparison between different models etc...)

Lenses with more than one maximum can be allowed for. See Figure 5c in (Rusin et al. 2001) and Figure 4b in Keeton & Winn (2003).

Supply several images from several bands.

A. More on Lensing Theory

In Section 2.1, for the sake of a more intuitive explanation, we suppressed some constant factors in equations (1), (1) and (4). Here we fill in the details.

First we define comoving distances.

$$\int \frac{dz}{\sqrt{\Omega_m(1+z)^3 + \Omega_\Lambda}} \quad (A1)$$

$$\begin{aligned} D_S & \quad \int_0^{a_S} \\ D_{LS} & = \frac{c}{H_0} \times \int_{z_L}^{z_S} \\ D_L & \quad \int_0^{z_L} \end{aligned} \quad (A2)$$

Can replace comoving distances D with angular-diameter distances d , and so on:

$$\begin{aligned} D_S & = (1+z_S) d_S \\ D_{LS} & = \frac{1+z_S}{1+z_L} d_{LS} \\ D_L & = (1+z_L) d_L \end{aligned} \quad (A3)$$

Angular-diameter distances are always smaller than comoving distances. Locations on the lens plane can be replaced angular coordinates on the sky, as

$$(x, y) = d_L(\theta_x, \theta_y). \quad (A4)$$

The A times and the κ density are related to physical arrival time t and density Σ as

$$\begin{aligned} A & = \frac{cD_L}{(1+z_L)^2} \frac{D_{LS}}{D_S} \times t \\ \kappa(x, y) & = \frac{4\pi G}{c^2} \frac{D_L}{1+z_L} \frac{D_{LS}}{D_S} \times \Sigma(x, y) \end{aligned} \quad (A5)$$

Letting the source be behind (s_x, s_y) rather than behind the origin, we have

$$t_{\text{geom}} = \frac{(1+z_L)^2}{2cD_L} \frac{D_S}{D_{LS}} ((x-s_x)^2 + (y-s_y)^2) \nabla^2 t_{\text{grav}} = -(1+z_L) \frac{8\pi G}{c^3} \Sigma(x, y) \quad (A6)$$

We can now compare with equations (2.1) to (2.6) from Blandford & Narayan (1986).

REFERENCES

- Blandford, R. & Narayan, R. 1986, *ApJ*, 310, 568
- Keeton, C. R. 2001a, ArXiv: astro-ph/0102341
- . 2001b, ArXiv: astro-ph/0102340
- Keeton, C. R. & Winn, J. N. 2003, *ApJ*, 590, 39
- Lubini, M. & Coles, J. 2012, *MNRAS*, 425, 3077
- Rusin, D., Kochanek, C. S., Norbury, M., Falco, E. E., Impey, C. D., Lehár, J., McLeod, B. A., Rix, H.-W., Keeton, C. R., Muñoz, J. A., & Peng, C. Y. 2001, *ApJ*, 557, 594
- Saha, P., Coles, J., Maccio', A. V., & Williams, L. L. R. 2006, *ApJ*, 650, 17
- Saha, P. & Williams, L. L. R. 1997, *MNRAS*, 292, 148
- . 2004, *AJ*, 127, 2604
- Williams, L. L. R. & Saha, P. 2000, *AJ*, 119, 439