A simple simulation technique for gravitational lenses

CRISTIAN EDUARD RUSU and BETRIA SILVANA ROSSA

ABSTRACT. The present article discusses the simulation of Gravitational Lensing with an algorithm developed in C++ and using the EasyBMP library. The algorithm numerically solves the general gravitational lens equation in the astrophysically significant weak field case, for any single-plane lens configuration. Examples of execution are considered, and a discussion is carried out on the advantages and disadvantages of the direct simulation technique employed.

1. INTRODUCTION

The Gravitational Lensing phenomenon ([6], [11], [12]) is a topic of intense research in modern Astrophysics. There is a large variety of techniques and algorithms ([1], [5], [7] and others) available for the simulation of Gravitational Lensing. One of the features these techniques have in common is that they try to solve the so-called "forward problem": given a distant background source (for example a quasar) and a foreground mass distribution (a galaxy or a galaxy cluster), how will the lensed image of the source look like?

In fact, since so many techniques are available, the following question arises: why does a simple and straightforward ray-tracing method that uses the rectangular pixel grid in a BMP (a bitmap, which is a very simple picture format) for representing data, along with the gravitational lens equation, not suffice? Are there any advantages to this technique at all?

This article tries to answer the question by presenting how this technique was implemented as a C++ application, as well as showing some results and necessary considerations.

2. FORMALISM OF GRAVITATIONAL LENSING

In order to simulate Gravitational Lensing, a basic introduction to the physical phenomenon is required.

Here, we are only concerned with Gravitational Lensing in weak fields, normally encountered in the proximity of galaxies and galaxy clusters. For the case of strong fields such as in the close vicinity of black holes, the reader is referred to the theory presented, for example, in [10]. The fundamental theory of Gravitational Lensing uses the following assumption, known as the "thin lens approximation": most of the light deflection occurs within $\Delta z \sim \pm b$ (Figure 1) of the point of closest approach between the light ray and a point mass acting as the lens. This Δz can typically be ignored in comparison to the much larger distances between observer and lens, and between lens and source. In effect, the

lens can be considered as thin compared to the total extent of the light path. A quantitative justification of this approximation for relevant astrophysical cases is provided in [12].

FIGURE 1. The deflection angle of light by a point mass M, with impact parameter b.
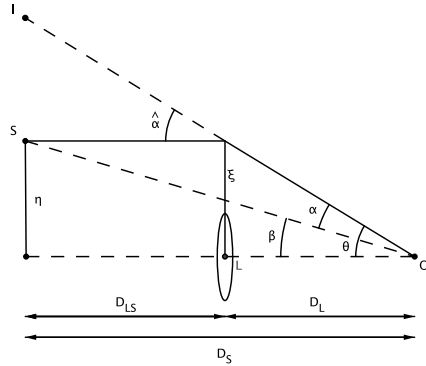
FIGURE 2. Lensing geometry. The distances D_L, D_S and D_{LS} are from observer to the lens plane, to the source plane, and between the lens and source plane, respectively. See the text for details.

The "thin lens approximation" allows to project the mass distribution of the lens along the line of sight, replacing it with a mass sheet perpendicular to the line-of-sight. The plane of this mass sheet is called the lens plane. The mass sheet corresponding to the lens is characterized by its surface mass density

$$\Sigma(\vec{\xi}) = \int \rho(\vec{\xi}, z) dz, \tag{2.1}$$

where $\vec{\xi}$ is a two-dimensional vector in the lens plane and ρ is the volume density.

The deflection angle of light due to a mass point M is directed towards the point mass (Figure 1) and given by [9]:

$$\hat{\alpha} = \frac{4GM}{c^2 b}, \tag{2.2}$$

where G is the gravitational constant, c is the speed of light and b is the impact parameter. In the presence of the mass sheet, the deflection angle at every position $\vec{\xi}$ is the sum of the deflections due to all the mass elements in the lens plane, and is given by

$$\vec{\alpha}(\vec{\xi}) = \frac{4G}{c^2} \int \frac{(\vec{\xi} - \vec{\xi'})}{\left|\vec{\xi} - \vec{\xi'}\right|^2} \Sigma\left(\vec{\xi'}\right) d^2\xi'. \tag{2.3}$$

This is simply the integral of (2.2), where the variable is a vector. A more thorough proof of this relation is given in [8].

In order to obtain the lens equation, consider the lens geometry in Figure 2. A light ray from source S is deflected by the angle $\vec{\alpha}$ at the position of the lens and reaches the

observer at O. The angle between the (arbitrarily chosen) optic axis and the true source position is $\vec{\beta}$, and the angle between the optic axis and the image is $\vec{\theta}$. The (angular-diameter) distances between observer and lens, lens and source, and observer and source are D_L, D_{LS} and D_S, respectively. We are only interested in angular distances, so we consider the source and the image to be in the same plane perpendicular to the optic axis. We also employ the approximation that the deflection angle is small, so that for example $\tan \theta \cong \sin \theta \cong \theta$, where the angle is in radians.

From Figure 2, we introduce the reduced deflection angle

$$\vec{\alpha} = \frac{D_{LS}}{D_S} \vec{\hat{\alpha}}, \tag{2.4}$$

and by noting the simple relation $\theta D_S = \beta D_S + \hat{\alpha} D_{LS}$, we get the following general gravitational lens equation (for a single lens plane):

$$\vec{\beta} = \vec{\theta} - \vec{\alpha}(\vec{\theta}). \tag{2.5}$$

If we consider cosmological scales, the distances D_L, D_{LS} and D_S must be taken as angular-diameter distances, as explained in [3].

3. IMPLEMENTATION

The algorithm developed for the purpose of this paper is, to the author's knowledge, the only one that simulates gravitational lensing by using BMP files for input/output (via the EasyBMP script; [4]). The advantage is that sources and lenses of any desired shape or mass (the latter being implemented via the pixel color palette) may be used as input data, and they can easily be edited in widely available applications such as Microsoft Paint. The disadvantage is that, especially at low resolutions, various errors due to the pixel grid system occur.

Apart from the physical parameters, which are specified from inside the algorithm, the input data consists of two bitmaps, depicting the distribution of the sources and the lenses, respectively. The output data is the bitmap showing the lensed images of the sources.

The principles of the technique are as follows. First, it is a natural choice to codify the mass distribution of the lens into pixel colors. This may be inspired, for instance, by the mass-to-light ratios for galaxies (from [2]). There are 256^3 colors and 256 values of transparency available for each pixel in an 8 or more bpp (bits per pixel) BMP file, and thus as many distinct values can be attributed to the surface density of the portion of the lens that we represent by an individual pixel.

Secondly, it is difficult both analytically and numerically to solve the lens equation from left to right (trying to find the images that correspond to each source point). The difficulties are presented in [5]. So the idea is to solve the equation from right to left (to map each image into its corresponding source). Roughly put, the algorithm finds the defection angle of each possible image point (each pixel of the BMP representing the image plane) by employing (2.3) for the given lens distribution, and then checks if the position of its source, given by (2.5), actually corresponds to a point on the source (a pixel on the bitmap representing the source plane). The number of calculations for such an algorithm is strongly dependent on the resolution of the input bitmaps and the size of the lens (but not on the size of the source), and this is reflected in the running time (shown in the captions to the simulation bitmaps in the next chapter).

4. APPLICATIONS AND CONSIDERATIONS

We present some simulation examples and the corresponding conclusions about the simulation method. The execution times correspond to running the C++ algorithm on a commercial machine with 2 GHz Intel Core 2 Duo processor and 1Gb RAM. These include the time required to write the 24bit BMP files (this size has been considered as offering an appropriate balance between the richness of the color palette and the rapidity of the writing process).

The first example (Figure 3) represents a point source mapped into an Einstein Ring by a foreground point lens lying on the line of sight. The diagonal symmetry due to the rectangular grid (coordinate system) and the incompleteness of the ring are clearly apparent. As the resolution is increased, the result of this simulation is more accurate, at the expense of computation time.

As can be seen, the nonlinear lens equation may map a single source into a collection of image points. The magnification of the source is also contained in the lens equation, and in this algorithm it is understood as the ratio of the number of pixels of the images to the number of pixels of the source. This convention is justified by the surface brightness conservation theorem in Gravitational Lensing (explained in [11]). In this particular example, the theoretical magnification from the lens equation is infinity, which would require infinite resolution in order to be reproduced by the algorithm.

FIGURE 3. Incomplete Einstein Ring in the case of small resolution (8 sec execution time, 100×100 pixels, 9 pixels Einstein ring radius).

The second example shows the appearance of the Andromeda Galaxy if it were lensed through a black hole lying on the line of sight (Figure 4). Analytically, this case corresponds to deflection by a point mass lens and the Gravitational Lensing equation in the week field limit still applies, if we are not concerned with light passing very close to the black hole (i.e. very close to its Schwarzschild radius). The parameters are shown in the figure caption. In this case, when just a visual representation of the lensing phenomenon is requested, the simulation may be considered as accurate.

The last example is a circular source (Figure 5) lensed through two constant surface density "galaxies" (represented by uniform black color) of elliptical and circular shape, in the foreground. Again, if just a visual representation is aimed at, then the result (Figure
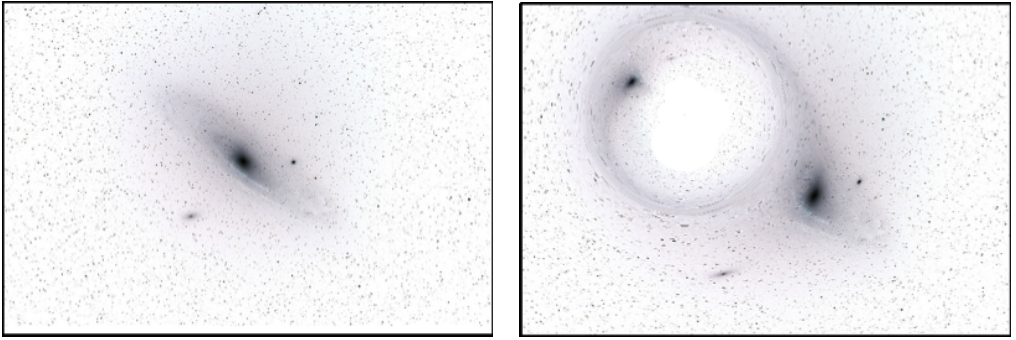
FIGURE 4. Inverted color image of the Andromeda Galaxy (public domain, left) and the simulation result (right) if a point mass (black hole) lens were on the line of sight. 229 sec execution time, 500×340 pixels (the version shown here is shrunk), parameters: pixel length 29 arcsec, $M = 1 \times 10^{13}$ M$_\odot$ (solar mass), $D_{LS} = 7.7 \times 10^5$ pc (parsec), $D_S = 7.8 \times 10^5$ pc, $D_L = 1.2 \times 10^4$ pc. Black hole Schwarzschild radius 16 arcsec. Einstein Ring radius 44 arcmin.
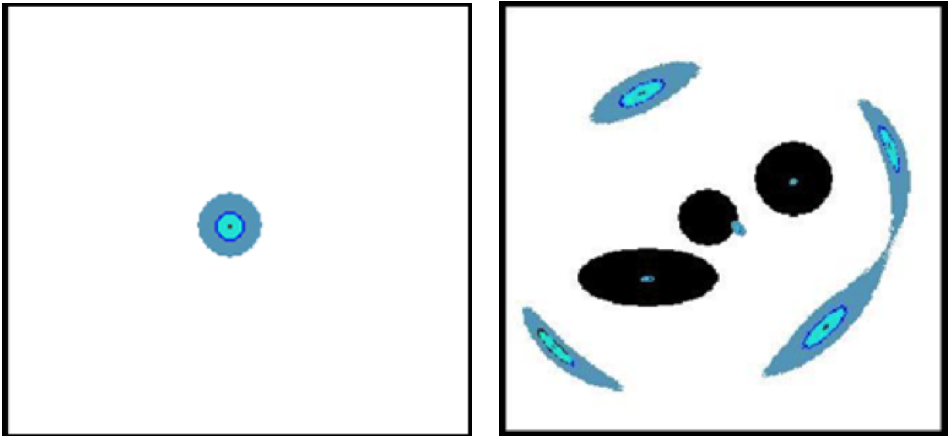

FIGURE 5. Circular source (left) and the result of the lensing (right) through the two galaxies in black. The position of the circular source (left figure) is indicated by the central black disk. 7 images are visible. 500×500 pixels (the version reproduced here is shrunk), 572 sec execution time. Parameters: M= 10^8M$_\odot$ (the mass represented by an individual pixel), $D_{ds} = 3 \times 10^9$ pc, $D_s = 3 \times 10^9$ pc, $D_d = 3 \times 10^9$ pc, pixel length: 0.01 arcsec.

7) is accurate, in the sense that it gives a reasonable visual approximation of the relative positions, magnifications and number of images of the source.

It is important to keep in mind the following observation when using such a simulation technique. Although a source/lens comprised of a collection of neighboring pixels looks like a continuous mass distribution when displayed on the input bitmap (such as the two

lenses in Figure 5), the algorithm treats it as a collection of point masses positioned at the center of each pixel. Care must thus be taken as to what physical model we attach to the lenses depicted in the input lens plane BMP. Comparing the results obtained with this algorithm to simulations using analytical methods (for continuous mass distribution in lenses), it has been noticed that a discrete mass distribution lens (where individual mass points are distributed in a grid), acts approximately like an equivalent continuous mass distribution lens with a different (higher/lower) surface mass density. Should the lens model be interpreted as having a continuous mass distribution, a detailed mathematical investigation is required to calibrate the value of the corresponding density, as a function of resolution.

5. COMPARISON WITH OTHER TECHNIQUES

A brief introduction to more advanced numerical simulation techniques for gravitational lensing is provided in [11]. The technique presented here is the most straightforward to implement of the so-called "grid search methods". A more careful approach would use the grid to map triangular tiles between the planes of the images and the source (as described in [11] and implemented in [5]), whereas in the present method the tiles degenerate into pixels. The tile method gives the possibility to increase the resolution of the mapping to specific regions of interest (close to caustics/critical curves − locations in the source/lens plane where magnification is infinite in the geometrical-optics limit), where two or more images may be found close together, yet may be unresolved by the present algorithm.

Other simulation techniques, such as [1], concentrate on implementing analytical solutions to the lens equation, for particularly simple lens density distributions.

6. CONCLUSIONS

This article has considered a technique for the simulation of Gravitational Lensing, technique which is straightforward to implement in a programming language such as C++. This is, to the authors knowledge, the only technique that simulates the Gravitational Lensing phenomenon by using bitmap files for direct data input and output.

It has been shown that, similar to other methods, the precision of the technique increases with the input bitmap resolution, at the expense of execution time.

The technique may be used to approximate Gravitational Lensing, in the week field case, for arbitrary distributions of sources and lenses, provided that the lenses lie in a single plane. As compared to more advanced techniques, the present method is easier to implement, but does not permit locally adjustable resolution. This means that it does not work very well for the study of critical lines and caustics, but it is appropriate to use when an approximate visual simulation, for a given global resolution, is needed.

REFERENCES

[1] Alfaro, F. F., *A Computer Program to visualize Gravitational Lenses*, arXiv:astro-ph/0204513v1 (30 Apr 2002)

[2] Binney, J. and Tremaine, S., *Galactic Dynamics* (Second Edition), Princeton Series in Astrophysics, 2008

[3] Carroll, S., *Spacetime and Geometry: An Introduction to General Relativity*, Publisher: Benjamin Cummings, 2003

[4] EasyBMP homepage: http://easybmp.sourceforge.net/ (1 Dec 2006)

[5] Keeton, C. R., *Computational Methods for Gravitational Lensing*, arXiv:astro-ph/0102340v1 (20 Feb 2001)

[6] Narayan. R. and Bartelmann, M., *Lectures on Gravitational Lensing*, arXiv:astro-ph/9606001v2 (2 Oct 1997)

[7] Oguri, M., *The Mass Distribution of SDSS J1004+4112 Revisited*, PASJ, Vol. 62, No. 4, pp. 1017-1024

[8] Padmanabhan, T., *Theoretical Astrophysics: Volume 3, Galaxies and Cosmology*, Cambridge University Press, New York, 2002

[9] Rusu, C. E., *Simple considerations on the deflection angle of light in General Relativity*, Creat. Math. Inform. **18** (2009), No. 2, 200–207

[10] Sass., I. H. A., *Late stages in stellar evolution*, Monographical Booklets in Applied and Computer Mathematics, Budapest University and of Technology and Economics, Budapest, 2002

[11] Schneider, P., Ehlers, J. and Falco, E. E., *Gravitational Lenses*, A&A Library, Springer, 1999

[12] Wambsganss, J., *Gravitational Lensing in Astronomy*, Living Reviews in Relativity. Retrieved from: http://www.livingreviews.org/lrr-1998-12

THE UNIVERSITY OF TOKYO
DEPARTMENT OF ASTRONOMY
GRADUATE SCHOOL OF SCIENCE,
7-3-1, HONGO, BUNKYO-KU, TOKYO 113-0033, JAPAN
E-mail address: eduard.rusu@nao.ac.jp

OSAKA UNIVERSITY
DEPARTMENT OF INFORMATION AND COMPUTER SCIENCE
SCHOOL OF SCIENCE ENGINEERING
1-1 MACHIKANEYAMA-CHO, TOYONAKA-SHI, OSAKA 560-0043, JAPAN
E-mail address: rossa@am.sanken.osaka-u.ac.jp