Hands-on Workshop:
How to calculate Quasar Microlensing

Joachim Wambsganss
Zentrum für Astronomie der Universität Heidelberg (ZAH/ARI)

XIth School of Cosmology
September 17 - 22, 2012 — IESC, Cargèse
Gravitational lensing: numerical simulations with a hierarchical tree code

Joachim Wambsganss

Astrophysikalisches Institut Potsdam, An der Sternwarte 16, 14482 Potsdam, Germany

Received 12 May 1998; received in revised form 6 January 1999

Abstract

The mathematical formulation of gravitational lensing — the lens equation — is a very simple mapping $\mathbb{R}^2 \rightarrow \mathbb{R}^2$, between the lens (or sky) plane and the source plane. This approximation assumes that all the deflecting matter is in one plane. In this case the deflection angle $\alpha$ is just the sum over all mass elements in the lens plane. For certain problems — like the determination of the magnification of sources over a large number of source positions (up to $10^{10}$) for very many lenses (up to $10^6$) — straightforward techniques for the determination of the deflection angle are far too slow. We implemented an algorithm that includes a two-dimensional tree-code plus a multipole expansion in order to make such microlensing simulations “inexpensive”. Subsequently we modified this algorithm such that it could be applied to a three-dimensional mass distribution that fills the universe (approximated by many lens planes), in order to determine the imaging properties of cosmological lens simulations. Here we describe the techniques and the numerical methods, and we mention a few astrophysical results obtained with these methods.
Efficient Inverse Ray Shooting: A Tree-Code Approach  
(Wambsganss 1990, 1999)

Deflection angle for \( n \) lenses:

\[
\tilde{\alpha}_i = \sum_{j=1}^{n} \tilde{\alpha}_{ji} = \frac{4G}{c^2} \sum_{j=1}^{n} M_j \frac{r_{ij}}{r_{ij}^2}
\]

Number of computational operations:

\[
N_{\text{total}} = N_{\text{op}} \times N_{\text{pix}} \times N_{\text{av}} \times N_* \simeq 10 \times 2500^2 \times 500 \times 10^6 \simeq 3 \times 10^{16}
\]

Calculation of deflection angle for \( N^* \) lenses split into two parts:

\[
\tilde{\alpha} = \sum_{i=1}^{N_*} \tilde{\alpha}_i \simeq \sum_{j=1}^{N_L} \tilde{\alpha}_j + \sum_{k=1}^{N_C} \tilde{\alpha}_k =: \tilde{\alpha}_L + \tilde{\alpha}_C.
\]

The \( N \)'s denote the following:

- \( N_* \) is the number of all lenses,
- \( N_L \) the number of lenses to be included directly,
- \( N_C \) the number of cells (= pseudo-lenses) to be included.
Efficient Inverse Ray Shooting: A Tree-Code Approach
(Wambsganss 1990, 1999)

Lens Equation:

\[
y = \begin{pmatrix} 1 - \gamma & 0 \\ 0 & 1 + \gamma \end{pmatrix} x - \sigma_c x - \sum_{i=1}^{N_*} \frac{m_i(x - x_i)}{(x - x_i)^2}
\]

Tree code approach:

\[
\tilde{\alpha} = \sum_{i=1}^{N_*} \tilde{\alpha}_i \approx \sum_{j=1}^{N_L} \tilde{\alpha}_j + \sum_{k=1}^{N_C} \tilde{\alpha}_k =: \tilde{\alpha}_L + \tilde{\alpha}_C.
\]
Efficient Inverse Ray Shooting: A Tree-Code Approach
(Wambsganss 1990, 1999)
Fig. 3. The accuracy increase of the detection angle obtained with the inclusion of higher-order multipole moments (see Section 2.3.4) is shown here. In each panel the difference between the directly determined detection angle ($\tilde{\alpha}_{\text{direct}}$) and the one obtained with the tree code ($\tilde{\alpha}_{\text{tree}}$) is plotted as a function of the former (in arbitrary units). For the top left panel all matter is assumed to be concentrated in the centers of mass of the cells (i.e. only the monopole term is included). In the top right panel, the quadrupole moment is included. In the bottom panels moments up to order 4 (left) and 6 (right) are included. It is obvious how much the accuracy increases with the inclusion of higher-order moments.

2.3.6. Final speed-up: expansion and interpolation

The code can be accelerated even more: the detection angle $\tilde{\alpha}_C$ (cf. Eq. (8)) due to the cells (containing the far away lenses) changes only very slightly across the n$_\times$ rays with fixed cell/lens configuration. This again speeds up the code by almost a factor $n^2/x$. 

---

Efficient Inverse Ray Shooting: A Tree-Code Approach

(Wambsganss 1990, 1999)

---

September 18, 2012; XI-th School of Cosmology, IESC Cargese; Joachim Wambsganss: “Workshop: How to calculate Quasar Microlensing”
Efficient Inverse Ray Shooting: A Tree-Code Approach
(Wambsganss 1990, 1999)
Efficient Inverse Ray Shooting: A Tree-Code Approach

(Wambsganss 1990, 1999)
1) copy file Wambsganss-MicrolensingCode-Cargese-2012.tar to your disk
2) untar this file ... should produce directory:
   Wambsganss-MicrolensingCode-Cargese-2012
Quasar Microlensing: How to do simulations!

1) copy file  Wambsganss-MicrolensingCode-Cargese-2012.tar  to your disk
2) untar this file ... should produce directory:

   Wambsganss-MicrolensingCode-Cargese-2012

3) cd  cfitsio
4) ./configure
5) make  (still in directory  cfitsio)
6) ..  (now in directory  Wambsganss-MicrolensingCode-Cargese-2012)
7) make  (should produce executable "microlens")
8) run the program by typing:   ./microlens
Quasar Microlensing: How to do simulations!

1) copy file   Wambsganss-MicrolensingCode-Cargese-2012.tar   to your disk
2) untar this file ... should produce directory:
   Wambsganss-MicrolensingCode-Cargese-2012
3) cd  cfitsio
4)  ./configure
5)   make                 (still in directory  cfitsio)
6)   ..                        (now in directory  Wambsganss-MicrolensingCode-Cargese-2012)
7)   make                 (should produce executable “microlens” )
8) run the program by typing:              ./microlens
9) newly produced files:
    dat.401                           log-file
    IRIS401                          magnification pattern (unformatted)
    IRIS401.fits                     magnification pattern (FITS format)
10) display magnification pattern with IDL:      ./rnew dis_1000
Quasar Microlensing: How to do simulations!

11) in order to extract a lightcurve: compile lightcurve.f
   (I use: gfortran lightcurve -o lightcurve)
12) run lightcurve routine:
   ./lightcurve
13) output produced:
   out_line (lightcurve data, pixels convolved with source profile)
   IRIS401-track (magnification pattern WITH track marked)
14) display magnification pattern with track AND lightcurve:
   dis_light
14) display magnification pattern with track AND lightcurve:

**dis_light**

**Source size: sigma = 3**
14) display magnification pattern with track AND lightcurve:

Source size: \( \sigma = 1 \)

Source size: \( \sigma = 3 \)
14) display magnification pattern with track AND lightcurve:

source size: \( \sigma = 3 \)

source size: \( \sigma = 10 \)
11) in order to extract a lightcurve: compile lightcurve.f
   (I use: gfortran lightcurve -o lightcurve)

12) run lightcurve routine:
    ./lightcurve

13) output produced:
    out_line         (lightcurve data, pixels convolved with source profile)
                   IRIS401-track  (magnification pattern WITH track marked)

14) display magnification pattern with track AND lightcurve:
    dis_light

15) modify input file for microlens:
Quasar Microlensing: How to do simulations!

15) modify input file for microlens: replace nray = “20” by “100” ... and run again!

<table>
<thead>
<tr>
<th>Line</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.243</td>
<td>urand (random seed number; 0.345)</td>
</tr>
<tr>
<td>0</td>
<td>debug (parameter: 0 - standard run)</td>
</tr>
<tr>
<td>0.200</td>
<td>kappas (matter in stellar objects; 0.000: 1 lens; 0.001: 2 lenses)</td>
</tr>
<tr>
<td>0.000</td>
<td>kappac (continously distributed matter)</td>
</tr>
<tr>
<td>0.000</td>
<td>gamma (external shear)</td>
</tr>
<tr>
<td>0.600</td>
<td>eps (accuracy parameter)</td>
</tr>
<tr>
<td>20</td>
<td>nray (# of rays per row in level 1)</td>
</tr>
<tr>
<td>1.000000</td>
<td>minmass (lower cut off in mass spectrum)</td>
</tr>
<tr>
<td>1.000000</td>
<td>maxmass (upper cut off in mass spectrum)</td>
</tr>
<tr>
<td>-2.350</td>
<td>power (index in mass spectrum)</td>
</tr>
<tr>
<td>10.000</td>
<td>pixmax0 (radius of field (ex.: 10.000)</td>
</tr>
<tr>
<td>-10.000</td>
<td>pixminx (ex.: -2.000)</td>
</tr>
<tr>
<td>-10.000</td>
<td>pixminy (ex.: -2.000)</td>
</tr>
<tr>
<td>20.000</td>
<td>pixdif (ex.: 4.000)</td>
</tr>
<tr>
<td>0.150</td>
<td>fracpixd (ex.: 0.020, 0.050, 0.250 ..., 2.000)</td>
</tr>
<tr>
<td>0</td>
<td>iwrite (iwrite &gt;0: write these values on fort.99)</td>
</tr>
</tbody>
</table>

surface mass density kappas (or sigma)
Quasar Microlensing: **Now YOU do simulations!**

**Deal:**

1) You can use the code “microlens” freely
2) On first scientific paper using “microlens”: J.W offered co-authorship
3) This (and subsequent) papers cite:

*September 18, 2012; XI-th School of Cosmology, IESC Cargese; Joachim Wambsganss: “Workshop: How to calculate Quasar Microlensing”*