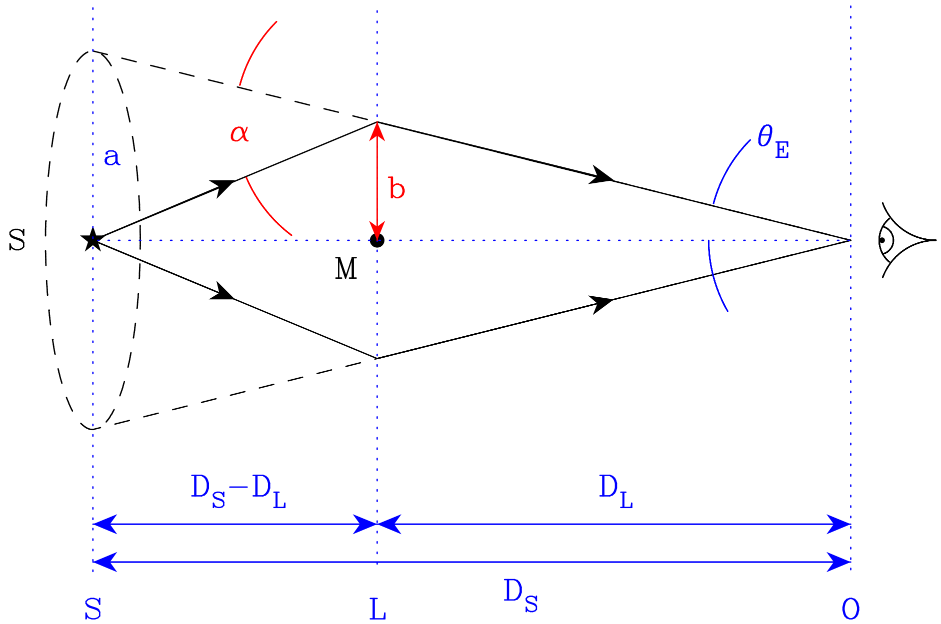
A Semi-Analytical Model for Gravitational Microlensing Events

Denis J Sullivan
(and Paul Chote, Michael Miller)
Victoria University of Wellington, NZ

January 31, 2011

The Einstein ring radius for microlensing

$$\theta_E = \sqrt{\frac{2R_S}{D}} \quad \text{where} \quad R_S = \frac{2GM}{c^2} \quad \text{and} \quad D = \frac{D_L}{D_S - D_L} D_S$$



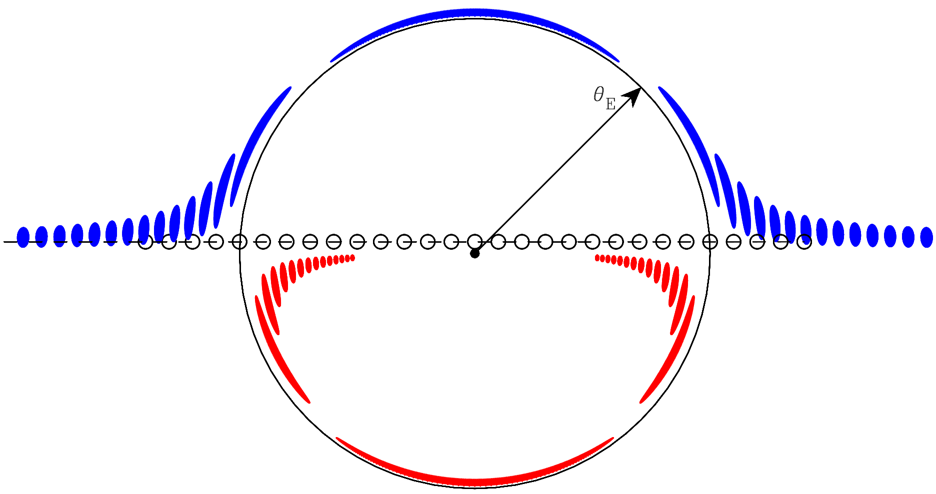
Point source point mass microlensing – single lens

The basic thin lens equation in units of the Einstein ring radius θ_E is:

$$u = \theta - \frac{1}{\theta} \quad \text{where} \quad u \text{ is source distance; } \theta \text{ is image distance}$$

$$A(t) = \sum \left| \frac{\theta d\theta}{u du} \right| = \frac{u^2(t) + 2}{u(t)\sqrt{u^2(t) + 4}} \quad \text{and}$$

$$u^2(t) = u_0^2 + \left(\frac{t - t_0}{t_E} \right)^2 \quad \Longrightarrow \quad \text{the Paczyński light curve}$$



Point source point mass lensing – N lenses

- In units of θ_E (or R_E) for the total lensing mass the basic thin lens equation relating the vector source position $\mathbf{s}$ to the vector positions $\mathbf{r}$ of the multiple images is:

$$\mathbf{s} = \mathbf{r} - \sum_{j=1}^{\mathrm{N}} \epsilon_j \frac{\mathbf{r} - \mathbf{r}_j}{|\mathbf{r} - \mathbf{r}_j|^2}$$

  where $\mathbf{r}_j$ and ϵ_j are the lens positions and mass fractions respectively.

- The required solution to this equation, ie determine the (multiple) $\mathbf{r}$ values from the (single) $\mathbf{s}$ value is not straight forward, so $\Longrightarrow$

- Two basic approaches to solving problem:

  1. Inverse ray tracing
  2. Semi-analytical methods

Solution methods

- Inverse ray-tracing:
  A brute force technique that exploits simplicity of equation in the reverse direction and locates image plane pixelated positions consistent with given source positions.

  Finite source effects are directly included, and creation of magnification maps makes this method relatively efficient for static lens systems.

  At VUW: Korpela 2007 (PhD thesis)

- Semi-analytical methods
  Start by directly determining image positions and proceed from there.

  At VUW: MSc thesis project (2010)

The lensing equation in complex coordinates

- Using complex coordinates yields great simplifications:
  1. source position $\mathbf{s} \longrightarrow \omega$
  2. image positions $\mathbf{r} \longrightarrow \mathbf{z}$
  3. lens positions $\mathbf{r}_j \longrightarrow \mathbf{r}_j$ (now complex)
- Lens equation becomes:

$$\omega = \mathbf{z} - \sum_{j=1}^{N} \epsilon_j \frac{\mathbf{z} - \mathbf{r}_j}{|\mathbf{z} - \mathbf{r}_j|^2} \quad \Longrightarrow \quad \omega = \mathbf{z} - \sum_{j=1}^{N} \frac{\epsilon_j}{\bar{\mathbf{z}} - \bar{\mathbf{r}}_j}$$

  Which is still complicated in separate real and imaginary form,
- But combining this with the complex conjugate version:

$$\bar{\omega} = \bar{\mathbf{z}} - \sum_{j=1}^{N} \frac{\epsilon_j}{\mathbf{z} - \mathbf{r}_j}$$

  allows the elimination of $\bar{\mathbf{z}}$, and rearranging gives an equation in the unknown complex variable $\mathbf{z}$ alone.

Obtaining the lensing polynomial

- The lensing equation for N lensing masses in the unknown (complex) image position $\mathbf{z}$ is:

$$\mathbf{z} - \omega - \sum_{k=1}^{N} \frac{\epsilon_k}{\mathbf{D}_k} = 0 \quad \text{where} \quad \mathbf{D}_k = \sum_{j=1}^{N} \frac{\epsilon_j}{\mathbf{z} - \mathbf{r}_j} + (\bar{\omega} - \bar{\mathbf{r}}_k)$$

- For two lensing masses this looks like:

$$\mathbf{z} - \omega - \frac{\epsilon_1}{\frac{\epsilon_1}{\mathbf{z} - \mathbf{r}_1} + \frac{\epsilon_2}{\mathbf{z} - \mathbf{r}_2} + (\bar{\omega} - \bar{\mathbf{r}}_1)} - \frac{\epsilon_2}{\frac{\epsilon_1}{\mathbf{z} - \mathbf{r}_1} + \frac{\epsilon_2}{\mathbf{z} - \mathbf{r}_2} + (\bar{\omega} - \bar{\mathbf{r}}_2)} = 0$$

- A polynomial is produced by appropriate multiplications in order to eliminated the fractions.
- Even for the two mass system the direct expressions (produced by Maple for example) are relatively complicated, and this complexity rapidly increases with N.
- So, ...

The lensing equation polynomials

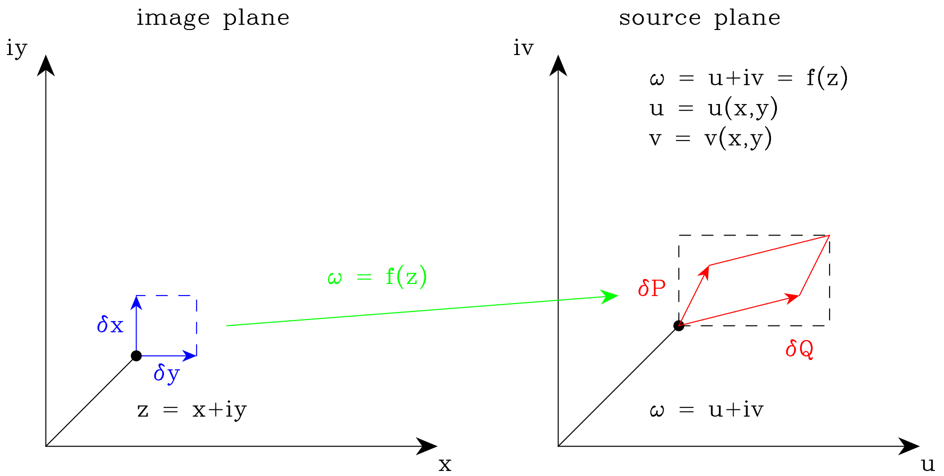
- In order to obtain efficient expressions for the polynomial coefficients one needs to identify by eye common factors and use a program like Maple to check for mistakes.
- We have used two approaches – the VUW method and the Rhie method (Rhie 2002). And we have coded the results as C functions.
- **Some properities of our C functions**:

number of lenses	polynomial order	lines of code	internal variables	relative speed
2	5	30	30	1
3	10	86	87	6.6
4	17	189	172	54
5	26	?	?	?

Finding the polynomial roots:
This can only be done numerically. We use the method of Jenkins & Traub (1972) combined with root polishing using the Laguerre method

Point source magnification

The magnification of each image is obtained from the infinitesimal area change produced by the transformation $z = g(\omega)$ from source plane $\omega(u, v)$ to image plane $z(x, y)$



But, it is the inverse transformation $\omega = f(z)$ that can be readily determined $\Longrightarrow$ the determinant of the Jacobian.

Jacobian & infinitesimal area change

We have:

$$\delta P = \frac{\partial u}{\partial x}\delta x + \frac{\partial u}{\partial y}\delta y \quad \text{and} \quad \delta Q = \frac{\partial v}{\partial x}\delta x + \frac{\partial v}{\partial y}\delta y$$

And image and source infinitesimal areas are:

$$\begin{aligned}
\delta A_{xy} &= \delta x \delta y \\
\delta A_{uv} &= \delta P_x \delta Q_y - \delta Q_x \delta P_y \\
&= \left(\frac{\partial u}{\partial x}\frac{\partial v}{\partial y} - \frac{\partial v}{\partial y}\frac{\partial u}{\partial x} \right) \delta x \delta y
\end{aligned}$$

Hence source/image(k) ratio is:

$$\frac{\delta A_{uv}}{\delta A_{xy}} = \frac{\partial u}{\partial x}\frac{\partial v}{\partial y} - \frac{\partial v}{\partial y}\frac{\partial v}{\partial y} \equiv J_k$$

Therefore total point source magnification is:

$$\text{magnification} = \sum_k^N \frac{1}{|J_k|}$$

10 / 19

The simplified form for the Jacobian

- The Jacobian for the area transformation therefore takes the form

$$J = \frac{\partial u}{\partial x}\frac{\partial v}{\partial y} - \frac{\partial u}{\partial y}\frac{\partial v}{\partial x}$$

- But, we can change variables as follows using the chain rule and exploit the properties of complex variables for this situation

$$\begin{aligned}
u &= \frac{\omega + \bar{\omega}}{2} \quad \text{and} \quad v = \frac{\omega - \bar{\omega}}{2i} \\
\mathbf{z} &= x + iy \quad \text{and} \quad \bar{\mathbf{z}} = x - iy
\end{aligned}$$

- Which yields the surprisingly simple and tractable form for J

$$J = \frac{\partial \omega}{\partial \mathbf{z}}\frac{\partial \bar{\omega}}{\partial \bar{\mathbf{z}}} - \frac{\partial \omega}{\partial \bar{\mathbf{z}}}\frac{\partial \bar{\omega}}{\partial \mathbf{z}} = 1 - \left|\frac{\partial \bar{\omega}}{\partial \mathbf{z}}\right|^2 = 1 - \left|\frac{\partial \omega}{\partial \bar{\mathbf{z}}}\right|^2$$

11 / 19

Critical and caustic curves

- The critical curves in the image plane are the locii of points where $J = 0$ and the corresponding caustics are the transformed points into the source plane.
- Thus using the above form for J we have

$$\left|\frac{\partial \bar{\omega}}{\partial \mathbf{z}}\right|^2 = 1 \quad \Longrightarrow \quad \frac{\partial \bar{\omega}}{\partial \mathbf{z}} = \sum_{j=1}^{N} \frac{\epsilon_j}{(\mathbf{z} - \mathbf{r}_j)^2} = e^{i\phi}$$

  where $e^{i\phi}$ defines the unit circle at the origin in the complex plane
- Transforming this equation to standard polynomial form yields a polynomial of order 2N in the complex variable $\mathbf{z}$. This can then be solved numerically for values $0 \leq \phi \leq 2\pi$ to yield the critical curve.

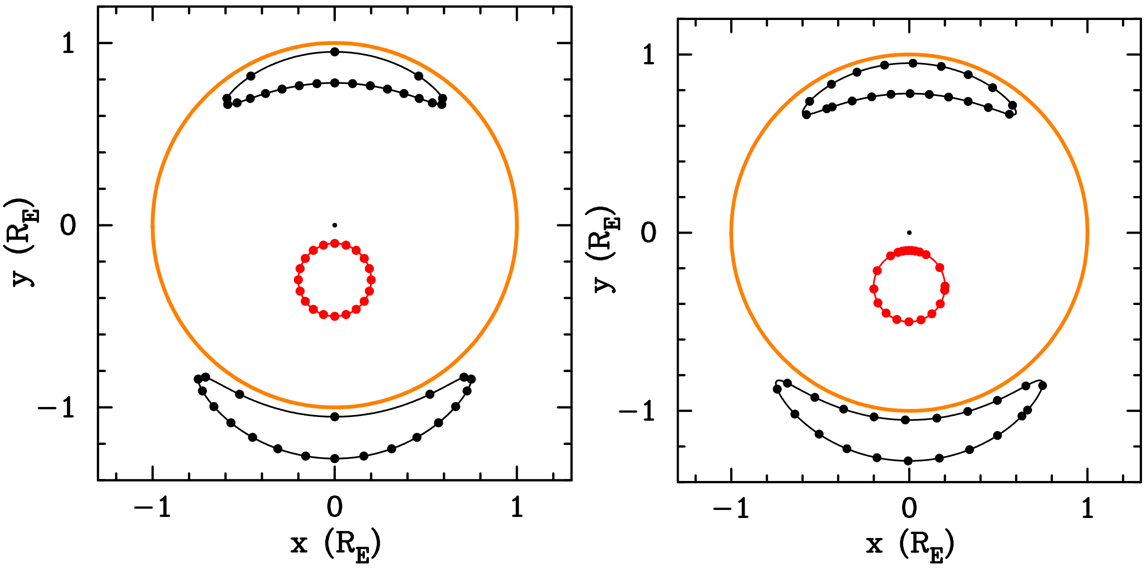
Including finite source effects

- **Hexadecapole approximation** [Gould (2008)]
  Evaluate a number (13) of point-source magnifications across the source disk and use these to approximate the analytical weighted integral of magnification field.
- **Polygon method** [Gould & Gaucherel (1997) – "Stokes' Theorem"]
  1. **Uniformly bright sources**: represent images using polygons with N vertices, then determine polygon area:

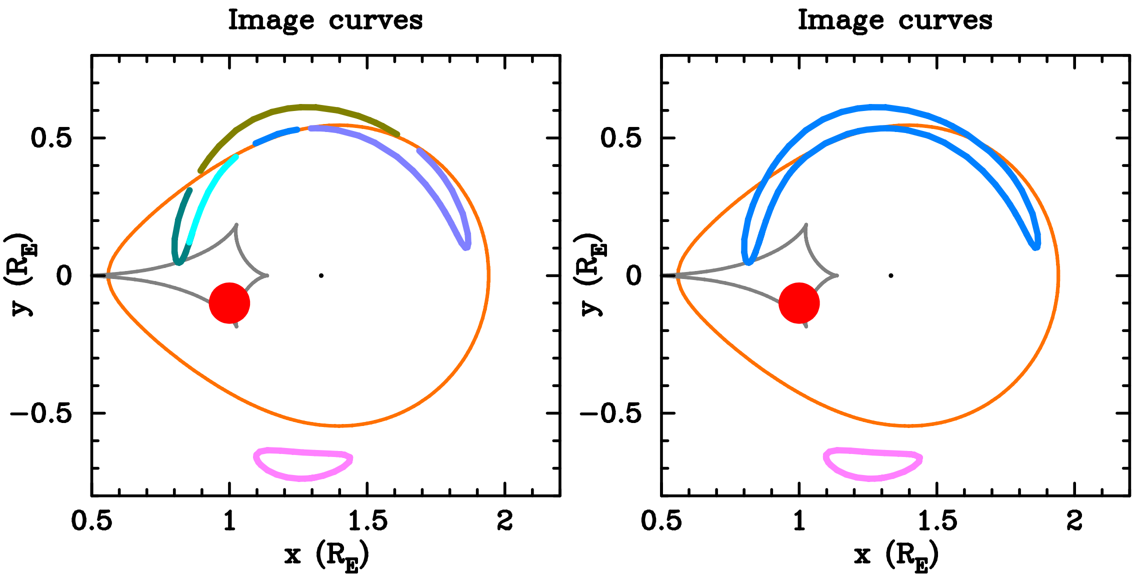
$$A = \sum_{i=1}^{N} (x_{i-1}y_i - x_i y_{i-1})$$

  2. **Including limb darkening**: Use a network of "concentric" polygons and appropriately weight annular areas
- Or, ...
- **Image centred inverse ray tracing** [Bennett (2010)]

Choosing the best polygon vertices

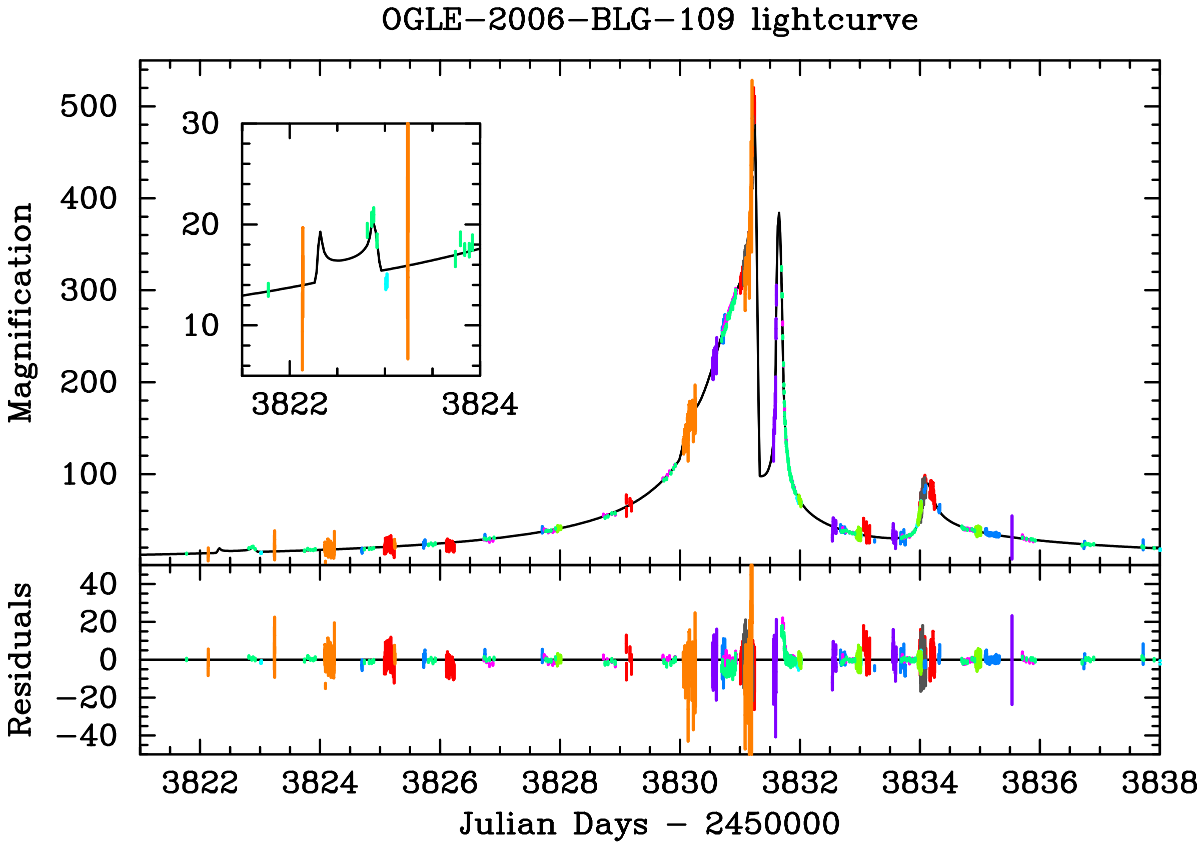


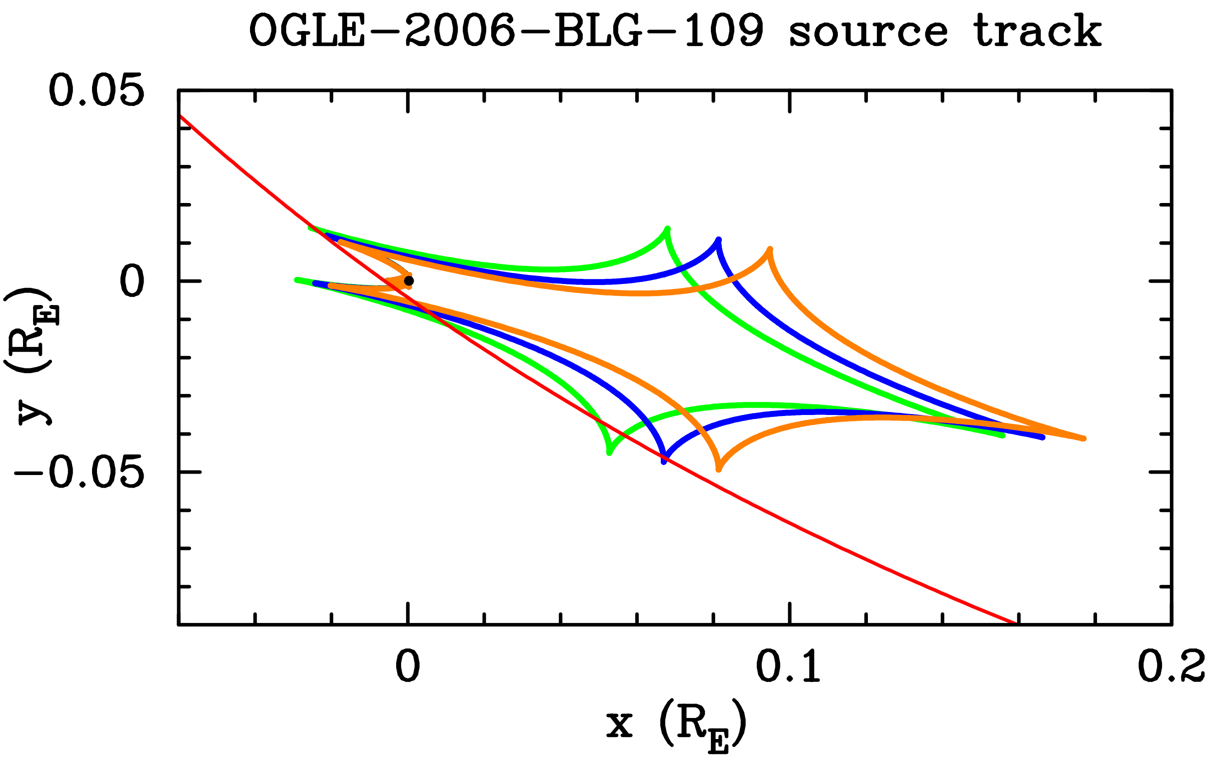
Finite source crossing a caustic

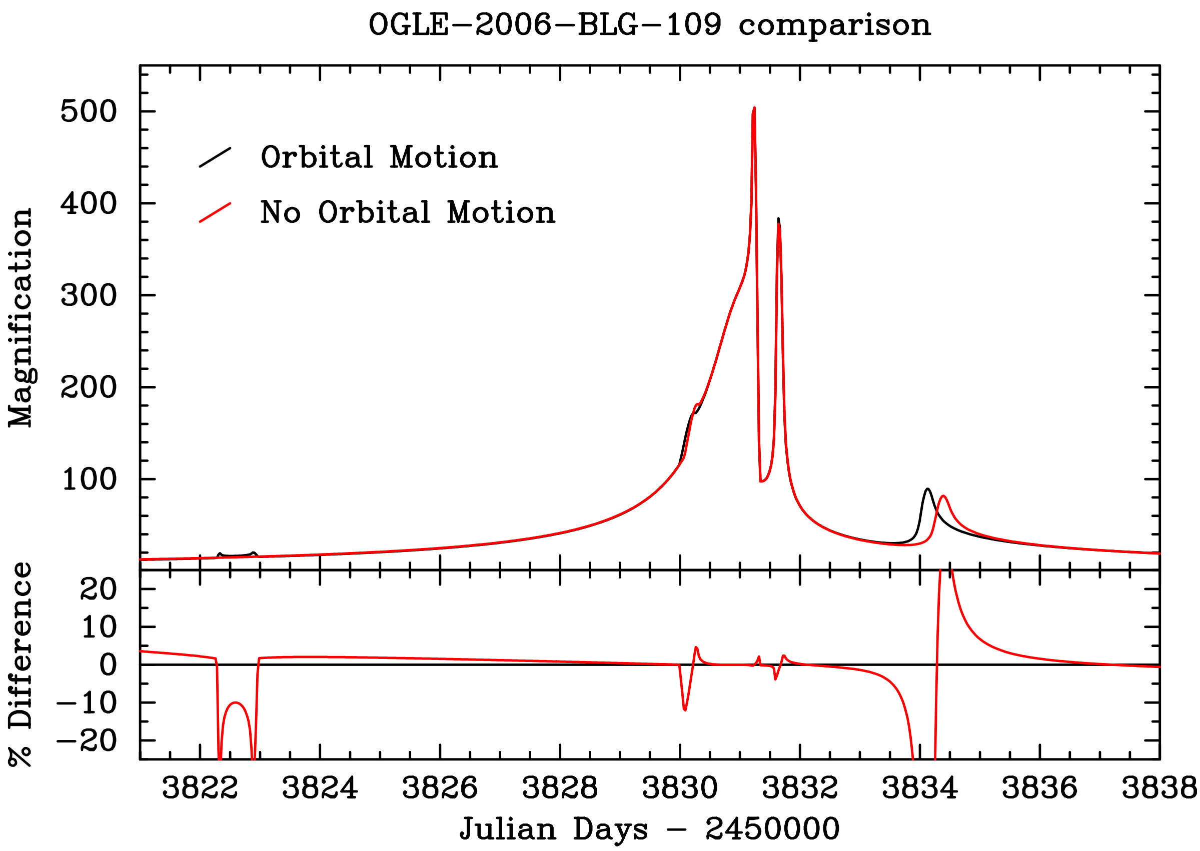


Light curve fitting procedures

We are using a combination of targetted grid search methods along with MCMC χ^2 surface explorations.




OGLE−2006−BLG−109 source track
0.05
0
−0.05
y (R_E)
0
0.1
0.2
x (R_E)


OGLE−2006−BLG−109 comparison
Orbital Motion
No Orbital Motion
Magnification
500
400
300
200
100
% Difference
20
10
0
−10
−20
3822
3824
3826
3828
3830
3832
3834
3836
3838
Julian Days − 2450000