Adaptive hierarchical tree algorithm for microlensing wave effect

Xikai Shan [ID],[1, 2, 3] Guoliang Li*,[1] Xuechun Chen,[2, 3] and Bin Hu[2, 3]

[1]*Purple Mountain Observatory, Chinese Academy of Sciences, Nanjing, Jiangsu, 210023, China*
[2]*Institute for Frontier in Astronomy and Astrophysics, Beijing Normal University, Beijing, 102206, China*
[3]*Department of Astronomy, Beijing Normal University, Beijing, 100875, China*

ABSTRACT

The gravitational lensing wave effect generated by the microlensing field embedded in the lens galaxy is an inevitable phenomenon in the realm of strong lensing gravitational waves. This effect presents both challenges and opportunities for the detection and application of gravitational waves. Investigating this specific wave effect necessitates a complete diffraction integral over each microlens in the field. However, the number of microlenses is very large, ranging from thousands to millions. Therefore, simply adding all of the microlenses is impractical. Additionally, the numerical resolution remains an issue for the microlensing field wave effect scenario. In this paper, we propose a trapezoid approximation based adaptive hierarchical tree algorithm to address the challenges of calculation speed and resolution. Our findings demonstrate that this algorithm significantly accelerates the calculation by four orders of magnitude compared to the simple adding method and achieves one order of magnitude acceleration compared to the fixed hierarchical tree algorithm proposed for electromagnetic microlensing. Additionally, our algorithm ensures controllable numerical errors, instilling more confidence in the results. This paper, in conjunction with our previous work (Shan et al. 2023c), comprehensively resolves all numerical issues, including integral boundary, precision, and time consumption. Therefore, this algorithm will greatly facilitate future investigations on the microlensing field wave effect.

Keywords: Gravitational lensing — Diffractional integral — Algorithm — Gravitational wave

1. INTRODUCTION

Strong lensing gravitational waves (SLGW) as a novel probe have demonstrated numerous powerful cosmological applications, such as precision Hubble constant measurement (Liao et al. 2017), testing modified gravity models (Collett & Bacon 2017; Finke et al. 2021; Narola et al. 2023), and studying the nature of dark matter (Cao et al. 2022; Tambalo et al. 2023a; Seo et al. 2023; Fairbairn et al. 2023), among others. Further applications can be found in Oguri (2019); Bian et al. (2021); Liao et al. (2022).

In contrast to the past decade when GW data were too sparse to detect SLGW events, we are now on the verge of the first detection of SLGW, especially in the context of binary neutron star lensing Smith et al. (2023), which will facilitate multi-messenger lensing applications. Therefore, investigating more details on gravitational lens modeling and sub-structure effects is valuable to gain a better understanding of the nature of GW lensing and systematics.

The microlensing field, including stars, remnants, and primordial black holes, constitutes one of the most important components of the lens galaxy. These objects will introduce a unique lensing wave effect on the SLGW generated by stellar mass binary objects (Diego et al. 2019; Meena & Bagla 2020; Mishra et al. 2021), owing to the comparable scale of their Schwarzschild radius and the wavelength of the GW (Takahashi & Nakamura 2003). Although this wave effect may bias the estimation of SLGW parameters and cause the loss of SLGW identification (Mishra et al. 2023; Shan et al. 2023b), it can also contribute to the SLGW identification (Shan et al. 2023a). Therefore, conducting more and further studies on the microlensing wave effect will facilitate our understanding of its nature and help find robust

*Contact e-mail: guoliang@pmo.ac.cn (G. Li);

methods to convert its disadvantages into advantages. However, the first important issue that we need to overcome before application is the diffraction integral calculation problem.

In the past thirty years, pioneers have proposed many algorithms to overcome this oscillation integral, including contour integration (Ulmer & Goodman 1995), asymptotic expansion methods (Press et al. 1992), Levin's method (Levin 1982), Filon-type methods (Filon 1930; Xiang 2007; Iserles & Nørsett 2006), and two new hybrid algorithms (Guo & Lu 2020; Tambalo et al. 2023b). However, directly applying these algorithms in the microlensing field scenario is not convenient because there are thousands, even millions, of microlenses in this field, and numerical calculations will encounter convergence, precision, and time-consuming problems inevitably. In our previous work (Shan et al. 2023c), we have solved the convergence problem and advanced the numerical precision using a component decomposition algorithm; however, the resolution of the lens plane and time-consuming problems remain unresolved.

In this paper, we propose an adaptive hierarchical tree algorithm, inspired by the hierarchical tree algorithm designed for the generation of microlensing magnification maps (Wambsganss 1990; Zheng et al. 2022). In Section 2, we will introduce the basic theory of the diffraction integral and our algorithm. In Section 3, we will present the results and compare the acceleration achieved by our algorithm. Finally, in Section 4, we will draw conclusions and engage in discussions.

2. METHOD

2.1. *Diffraction integral*

In the context of the scalar wave approximation, the gravitational wave (GW) lensing effect can be quantified using the diffraction integral (Schneider et al. 1992; Nakamura & Deguchi 1999; Takahashi & Nakamura 2003) represented by the equation:

$$F(\omega, \boldsymbol{y}) = \frac{2GML(1+z_L)\omega}{\pi c^3 i} \int_{-\infty}^{\infty} d^2x \exp\left[i\omega t(\boldsymbol{x}, \boldsymbol{y})\right], \tag{1}$$

where $F(\omega, \boldsymbol{y})$ denotes the magnification factor, and ω and $\boldsymbol{y}$ represent the GW's circular frequency and its position (normalized by the Einstein radius) in the source plane, respectively. M_L is the lens mass, z_L is the lens redshift, and $\boldsymbol{x}$ is the coordinate (normalized by the Einstein radius) in the lens plane. $t(\boldsymbol{x}, \boldsymbol{y})$ is the time delay function which is given by (Wambsganss 1990; Schneider et al. 1992; Chen et al. 2021):

$$t(\boldsymbol{x}, \boldsymbol{x}^i, \boldsymbol{y} = 0) = \underbrace{\frac{k}{2}\left((1-\kappa+\gamma)x_1^2 + (1-\kappa-\gamma)x_2^2\right)}_{t_{\text{smooth}}(\kappa, \gamma, \boldsymbol{x})} - \underbrace{\left[\frac{k}{2}\sum_i^{N_*} \ln\left(\boldsymbol{x}^i - \boldsymbol{x}\right)^2 + k\phi_-(\boldsymbol{x})\right]}_{t_{\text{micro}}(\boldsymbol{x}, \boldsymbol{x}^i)} \tag{2}$$

,

for the microlensing field embedded in the lens galaxy/galaxy cluster scenario. Here, $k = 4GM_{\text{microlens}}(1+z_L)/c^3$, $\boldsymbol{x}^i$ is the coordinate of the ith microlens, and N_* is the number of microlenses. κ and γ are the macro lensing convergence and shear. $t_{\text{smooth}}(\kappa, \gamma, \boldsymbol{x})$ and $t_{\text{micro}}(\boldsymbol{x}, \boldsymbol{x}^i)$ are the macro lensing and micro lensing time delay, respectively. Here, we set the macro image point as the coordinate origin ($y = 0$). The contribution from a negative mass sheet, denoted by $\phi_-(\boldsymbol{x})$, is included to ensure the total convergence κ remains unchanged when adding microlenses (Wambsganss 1990; Chen et al. 2021; Zheng et al. 2022).

However, due to the oscillatory nature of the integrand, conventional Gaussian numerical integration methods are inadequate. To address this limitation, many pioneer works have proposed many algorithms. Press et al. (1992); Levin (1982); Filon (1930); Xiang (2007); Iserles & Nørsett (2006); Guo & Lu (2020); Tambalo et al. (2023b), etc, proposed effective and fast algorithm for isolated microlens, and Ulmer & Goodman (1995) proposed an usual algorithm for both isolated and microlensing field. Due to in this paper, we focused on the microlensing field scenario, so, we use Ulmer & Goodman (1995)'s strategy to solve the diffractional integral. In detail, this algorithm Fourier transforms the amplification factor into the time domain:

$$\tilde{F}(t, \boldsymbol{y}) \equiv \frac{1}{2\pi} \int_{-\infty}^{\infty} d\omega \exp\left(-i\omega t\right) \frac{F(\omega, \boldsymbol{y})}{C_\omega}. \tag{3}$$

where $\tilde{F}(t, \boldsymbol{y})$ is the time domain amplification factor and C_ω is the coefficient before the integrand of Eq. (1):

$$C_\omega = \frac{2GM_L(1+z_L)\omega}{\pi c^3 i}. \tag{4}$$

Upon substituting Eq.(1) into Eq.(3), it is evident that the time domain amplification factor is proportional to the time delay probability density function (PDF):

$$\tilde{F}(t, \boldsymbol{y}) = \int_{-\infty}^{\infty} \mathrm{d}^2 x \, \delta\left[t(\boldsymbol{x}, \boldsymbol{y}) - t\right] = \frac{|\mathrm{d}S|}{\mathrm{d}t}, \tag{5}$$

which can be determined by calculating the surface area $\mathrm{d}S$ in the time delay interval $\mathrm{d}t$. The subsequent step is to inversely Fourier transform $\tilde{F}(t, \boldsymbol{y})$ to recover $F(\omega, \boldsymbol{y})$. One can find that this algorithm eliminates the integral oscillation problem.

One direct and easy realization method for the calculation of the time domain amplification factor is to pixelate the lens plane and then accumulate the area of the pixel within the same time delay interval $[t - \Delta t/2, t + \Delta t/2]$ (Diego et al. 2019). The advantage of this method is that it does not need to find out the image point at first, compared with the contour integral method (Ulmer & Goodman 1995; Mishra et al. 2021). However, this simple adding (refer to as SA in the later of this paper) method needs an extremely high-resolution pixelized lens plane to avoid numerical error induced by the time delay variation in the inner region of the pixel. In other words, if the maximum time delay difference in the pixel is greater than the time delay resolution, rudely arranging the whole pixel area into one time delay interval will introduce numerical error. Therefore, in the case of a highly dense field configuration, the total calculation time, $t_{\mathrm{SA}} = N_{\mathrm{pixel}} \cdot N_*$, will be very long. Here, N_{pixel} is the total number of pixels in the lens plane and N_* is the number of microlenses.

To reduce the numerical errors induced by the pixels that cross the time delay bins, and meanwhile to speed up the calculation by avoiding an extremely high resolution pixelized lens plane, we propose a trapezoid approximation based adaptive hierarchical tree algorithm. Hereafter, we short this method as TAAH (**T**rapezoid **A**pproximation based **A**daptive **H**ierarchical) tree algorithm.

2.2. *Trapezoid approximation in pixel*

The time-domain amplification factor within a pixel can be evaluated through the contour integral:

$$\frac{|\mathrm{d}S|}{\mathrm{d}t} = \oint_C \frac{\mathrm{d}s}{|\nabla_{\boldsymbol{x}} t(\boldsymbol{x}, \boldsymbol{y})|} \tag{6}$$

Here, $\mathrm{d}s$ represents the line element, and C is the length of the iso-time delay curve and the gradient of the time delay:

$$\nabla_{\boldsymbol{x}} t(\boldsymbol{x}, \boldsymbol{y}) = k \begin{pmatrix} 1 - \kappa_{\mathrm{s}} + \gamma & 0 \\ 0 & 1 - \kappa_{\mathrm{s}} - \gamma \end{pmatrix} \boldsymbol{x} - k \sum_i^{N_*} \frac{\boldsymbol{x} - \boldsymbol{x}^{(i)}}{\left|\boldsymbol{x} - \boldsymbol{x}^{(i)}\right|^2}, \tag{7}$$

where κ_{s} is the smoothing part of the convergence. This formula is only correct for a circle microlensing field, here, we just use this formula to illustrate our algorithm and the realistic calculation used a square field. One can refer Zheng et al. (2022) to find more details about the square field and the difference between square and circle field. One can find that the Eq. 7 is differentable except for the microlenses positions. Therefore, the gradient of the time delay function can be approximated as constant if the pixel is small enough and the pixel does not include any microlens. Based on this property, we show one sketch (Fig. 1) that describe the time delay gradient and time domain amplification factor within this kind of pixel.

The upper panel shows the gradient of the time delay (black dashed curve with an arrow, indicating the direction of the gradient) and the iso-time delay curves (black solid curves orthogonal to the dashed curve, labeled with different red tags representing different time delays $(t_{1\sim4})$). t_0 is the time delay in the middle of the square pixel, and α is the angle between the direction of the gradient and the diagonal of the pixel. One can find that the contour length between t_1 and t_2 is constant, and the contour length between t_1 and t_3 and between t_2 and t_4 monotonically decreases.
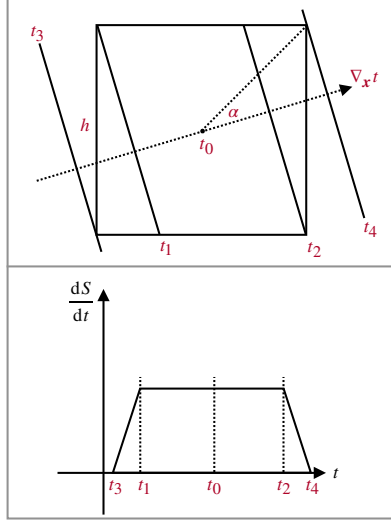
Figure 1. Deflection angle perturbation in the lens plane. $\kappa = \kappa_= 0.6$, $\gamma = 0.1$, $N_= 238$, with an equal mass of $1 M_\odot$.

Therefore, the time-domain amplification factor within the pixel can be described by:

$$\frac{|\mathrm{d}s|}{\mathrm{d}t'} = \begin{cases} \text{const} \times \frac{t'-t_3}{t_1-t_3} & (t_3 \leq t' \leq t_1) \\ \text{const} & (t_1 < t' < t_2) \\ \text{const} \times \frac{t-t_2}{t_4-t_2} & (t_2 \leq t' << t_4) \\ 0 & (\text{else}) \end{cases}, \tag{8}$$

where const $= \frac{h}{\cos(\pi/4-\alpha) \times |\nabla_x t| \times \text{coeffi}}$. The lower panel in Fig. 1 illustrates this theoretical distribution of the time-domain amplification factor within this pixel.

For the pixel that include microlens, one can find that the span of the time delay in such kind of pixel will be ∞, therefore the time domain amplification factor within this kind of pixel will be zero. Therefore, one can eliminete the trapezoid appximation error from this kind of pixel by reduce the size of the pixel.

After calculate the time domain amplification factor in each pixel, one can get the final time-domain amplification factor at t_0 in the total lens plane by using:

$$\frac{|\mathrm{d}S|}{\mathrm{d}t}|_{t_0} = \frac{\sum_{i=0}^{N_{\text{pixel}}} \int_{t_0-\Delta t/2}^{t_0+\Delta t/2} \frac{|\mathrm{d}s|}{\mathrm{d}t'}|_i \mathrm{d}t'}{\Delta t} \tag{9}$$

where i represents the ith pixel, N_{pixel} is the total number of pixels in the image plane.

2.3. Adaptive hierarchical tree algorithm for diffraction integral

From the previous section, we know that the calculation time t_{SA} for the microlensing field diffraction integral is proportional to $N_{\text{pixel}} \times N_*$. Therefore, it is impossible to rudely sum over all of the microlenses' contribution in every pixel when the microlensing field contains a large number of microlenses. In this section, we will introduce how our TAAH tree algorithm solve the time consuming problem.

The TAAH tree algorithm include two fixed level grids (Level 1 and Level 2) and one adaptive grid (Level n, $n \geq 2$). The first two fixed level grid is based on the algorithm proposed in Zheng et al. (2022), which is focused on the generation of the microlensing magnification map. In detail, the Level 1 grid stores the indices of the microlenses, facilitating the construction of a lookup table for quick microlens identification. Here, we set the resolution of Level 1 as $L_1 = \min\{L_0/10, \sqrt{L_0^2/N_*}\}$, where L_0 is the length of the total lens plane. This ensures that each Level 1 grid has at least one microlens when the number of microlenses is not very small.

The Level 2 grid stores the deflection angle of the far-field microlenses. As outlined in Wambsganss (1990); Chen et al. (2021); Zheng et al. (2022), the evolution of the lens potential induced by far-field microlenses within the Level 2 grid is smoothing, and the distance between these microlenses and the center of the Level 2 grid is significantly larger than the resolution of the Level 2 grid. Consequently, the time delay and deflection angle of far-field microlenses within the Level 2 grid (in other words, Level n grid) can be effectively interpolated using the stored deflection angle (see Wambsganss (1990) for more details). Hence, for a more accurate grid Level n within Level 2, there is no need to resumover the far-field microlenses, and only need to use the interpolation polynomial to recover the potential from far-field microlenses. And then, one only need to sum over the near-field microlenses and add them to the interpolation result to obtain the final result. This leads to a substantial reduction in the total computing time

$$t_{\mathrm{FH}} \simeq N_{\mathrm{L}_2} \cdot N_* + N_{\mathrm{pixel}} \cdot N_{*,\mathrm{near}}, \tag{10}$$

where, in a typical microlensing system, $N_{\mathrm{L}_2} \ll N_{\mathrm{pixel}}$ and $N_{*,\mathrm{near}} \ll N_*$. Here, we use the same resolution of the Level 2 grid as in Zheng et al. (2022), which is $L_2 = L_1/20$ and far-field microlenses are the lenses outside the nearest eight L_1 grids surrounding the L_1 grid in which the L_2 grid resides (so, the number of near field microlenses $N_{*,\mathrm{near}} \simeq 9$). This is the computing time for the fixed hierarchical (FH) tree algorithm introduced in Zheng et al. (2022) for the generation of the microlensing magnification map.

Different from Zheng et al. (2022), our TAAH algorithm using an adaptive grid, which is named Level n grid to replace the fixed refined grid Level 3, within Level 2. As established in section 2.2, the time domain amplification factor within the pixel can be approximated by a trapezoid-shaped distribution. The accuracy of this approximation relies on the relative changing rate of the gradient of the time delay. Therefore, if this relative changing rate $|\Delta\nabla_{\boldsymbol{x}}t|/|\bar{\nabla}_{\boldsymbol{x}}t|$ is less than a tiny value ϵ, the approximation is accurate. Based on this property, we set the length of the level n grid by requiring $|\Delta\nabla_{\boldsymbol{x}}t|/|\bar{\nabla}_{\boldsymbol{x}}t| < \epsilon$. In detail, first, we calculate $e_2 = |\Delta\nabla_{\boldsymbol{x}}t|/|\bar{\nabla}_{\boldsymbol{x}}t|$ in the Level 2 grid. If $e_2 < \epsilon$, the trapezoid approximation in the Level 2 grid is accurate enough; otherwise, we reduce the grid length by half, and calculate $e_n = |\Delta\nabla_{\boldsymbol{x}}t|/|\bar{\nabla}_{\boldsymbol{x}}t|$ in each subpixel (where $n > 2$ is the resolution level number, and the number of e_n is 4^{n-2}). The refinement stops when all e_n are less than ϵ.

There are three kind of special points that need special deal, the first one is our previously mentioned microlenses points, the second one is the origion of the lens plane and the third one is the microlensing image points. In these three kind of points, the relative difference within the pixel will be ∞ and will not converge by reducing the length of the pixel. Therefore, in order to avoid the infinite refinement, we set a maximum resolution level $n_{\max}$. In the result section, one can find that in these three kinds of pixels, the resolution will be very high and the numerical error caused by the trapezoid approximation in these three kinds of pixels can be reduced by reducing the length of the pixel.

Up to now, the completed TAAH tree algorithm has now been constructed. In Fig. 2, we present the flow chart of this algorithm, encompassing the following nine main steps:

- Input lens parameters, including macro lensing κ, γ, and micro lensing κ_*.

- Use the method introduced in Shan et al. (2023c) to calculate the microlensing field boundary L_0.

- Randomly place $N_* = \kappa_* \times L_0^2/\pi$ microlenses within the microlensing field boundary.

- Set Level 1 grid with a length $L_1 = \min\{L_0/10, \sqrt{L_0^2/N_*}\}$, and assign each microlens to a Level 1 grid.

- Set Level 2 grid with a length $L_2 = L_1/20$, and store the deflection angle contributed by the far-field microlenses.

- Calculate $|\Delta\nabla_{\boldsymbol{x}}t|/|\bar{\nabla}_{\boldsymbol{x}}t|$ within the Level 2 or Level n grid. If $|\Delta\nabla_{\boldsymbol{x}}t|/|\bar{\nabla}_{\boldsymbol{x}}t| < \epsilon$ or $n > n_{\max}$, proceed to the next step; otherwise, reduce the grid length by half and $n = n + 1$.

- Calculate the trapezoid approximation result within the Level n pixel using Eq. 8.

- Sum over the trapezoid approximation results at all Level n grids using Eq. 9.

- Use the Component Decomposition algorithm proposed in Shan et al. (2023c) to recover the frequency domain amplification factor $F(\omega)$.
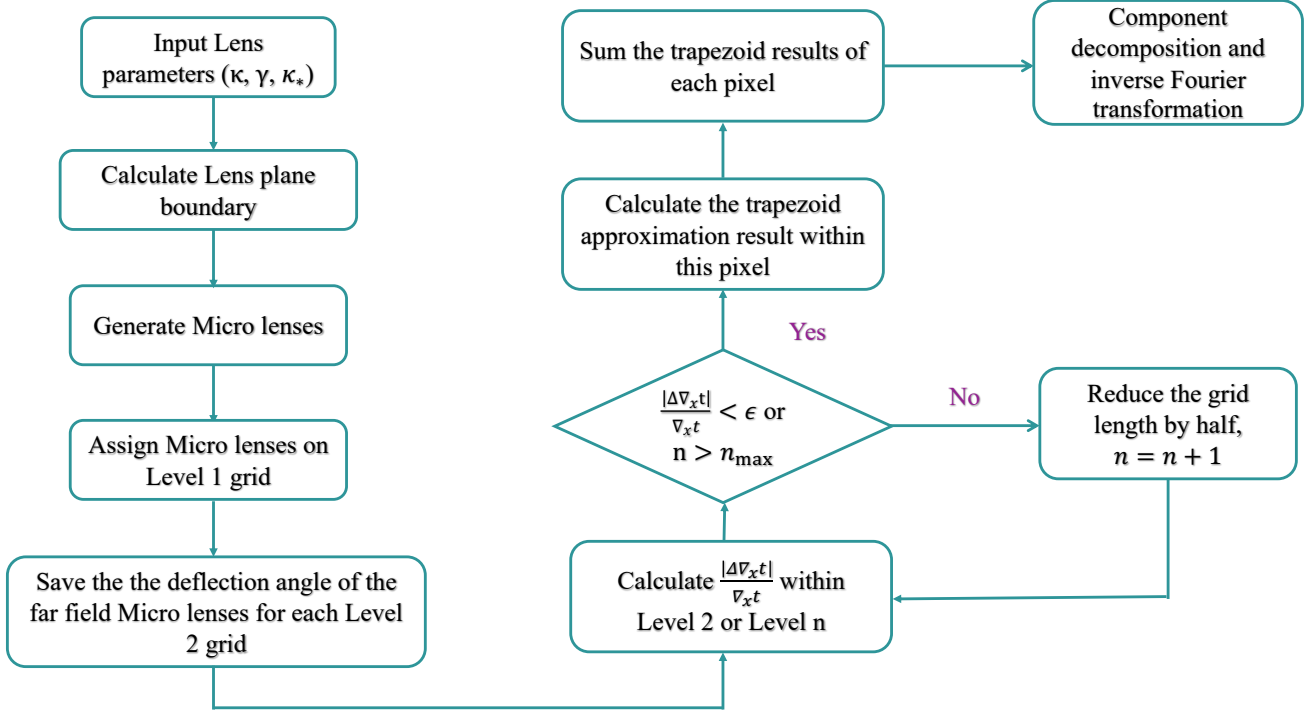
Figure 2. Flow chart of the TAAH tree algorithm.

In the final of this section, we want to highlight the adaptive manner of the Level n grid. Compared with the FH tree algorithm (Wambsganss 1990; Chen et al. 2021; Zheng et al. 2022) introduced for the generation of the microlensing magnification map, our algorithm does not need a uniformly refined gird over Level 2. Therefore, our algorithm can furture accelerate the calculation by reduce the second term of Eq. 10:

$$t_{\mathrm{TAAH}} \simeq N_{\mathrm{L_2}} \times N_* + N_{\mathrm{L_n}} \times N_{*,\mathrm{near}}, \tag{11}$$

where $N_{\mathrm{L_n}}$ is the subpixel number of the Level n adaptive grid. One can find that, if $N_{\mathrm{L_n}} \ll N_{\mathrm{pixel}}$, the acceleration will be significant and this acceleration can help us avoid using GPU processor.

3. RESULT

3.1. *Algorithm test using analytical result*

In this section, we evaluate the performance of our TAAH tree algorithm by comparing its results with analytical solutions. We consider three distinct macro-lensing scenarios, and the corresponding macro and micro-lensing parameters are provided in Table 1. Specifically, for Type I and Type III macro-lensing images, we set the shear (γ) value to 0. This choice is made because when a micro lens is embedded in external convergence without external shear, the diffraction has an analytical resolution described by:

$$
\begin{aligned}
F(w, \boldsymbol{x}) = &-\frac{1}{\lambda^2 w} 2^{-2-\frac{iw}{2}} |w|^{-1+\frac{iw}{2}} \Gamma\left(1 - \frac{iw}{2}\right) |\lambda|^{-1+\frac{iw}{2}} e^{\frac{iw}{2}\left(\phi_m + \lambda(x_1^2 + x_2^2)\right)} \quad \left((w + 2i) |\lambda|^3 |w|^3 \left(x_1^2 + x_2^2\right)\right. \\
&\left(\sinh\left(\frac{\pi w}{4}\right) \operatorname{sgn}(\lambda w) + \cosh\left(\frac{\pi w}{4}\right)\right) {}_2F_3\left(1 - \frac{iw}{4}, \frac{3}{2} - \frac{iw}{4}; 1, \frac{3}{2}, \frac{3}{2}; -\frac{1}{16}\lambda^2 w^2 \left(x_1^2 + x_2^2\right)^2\right) \\
&\left. -4\lambda^2 w^2 \left(\cosh\left(\frac{\pi w}{4}\right) \operatorname{sgn}(\lambda w) + \sinh\left(\frac{\pi w}{4}\right)\right) {}_2F_3\left(\frac{1}{2} - \frac{iw}{4}, 1 - \frac{iw}{4}; \frac{1}{2}, \frac{1}{2}, 1; -\frac{1}{16}\lambda^2 w^2 \left(x_1^2 + x_2^2\right)^2\right)\right)
\end{aligned} \tag{12}
$$

.

Table 1. This table listed the macro and micro parameters for algorithm testing. κ and γ are macro lensing convergence and shear, Micro x_1 and Micro x_2 are micro lens coordinate. Here, we set the mass of microlens as 100 M$_\odot$ for Type I and Type III scenario.

	κ	γ	Micro x_1	Micro x_2
Type I	0.7	0	0.1	0
Type II	0.875	0.325	none	none
Type III	1.3	0	10	0
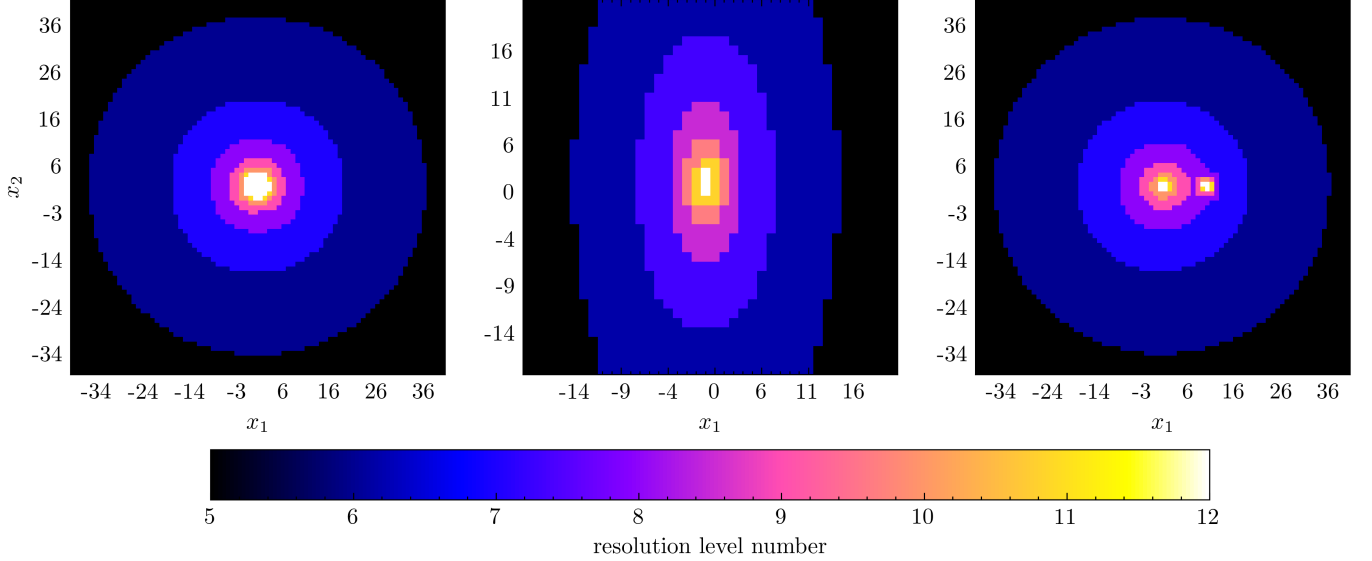
Figure 3. Resolution level number n in the lens plane. Columns from left to right is Type I, Type II and Type III scenarios listed in Table. 1.

For Type II macro-lensing images, we exclusively test our algorithm under a pure macro-lensing scenario. This choice is deliberate since logarithmic divergence is Type II's most important characteristic, and there are not analytical solutions when any micro lens is embedded.

Fig. 3 illustrates the resolution level number n (Level n) in the lens plane for three different scenarios, with Type I on the left, Type II in the middle, and Type III on the right. Here, we set $L_2 = 1.024$, $\epsilon = 0.01$, and $n_{\max} = 12$, resulting in a minimum pixel length of $L_{12} = 0.001$. In this figure, the inner region exhibits a higher resolution number, primarily due to macro lensing effects, as $|\Delta\nabla_{\boldsymbol{x}}t|/|\bar{\nabla}_{\boldsymbol{x}}t| \propto 1/|\boldsymbol{x}|$, where $|\boldsymbol{x}|$ is the distance from the center. Additionally, the third panel shows that at the micro lens points, the resolution level is very high, as this region is dominated by micro lensing effects and $|\Delta\nabla_{\boldsymbol{x}}t|/|\bar{\nabla}_{\boldsymbol{x}}t| \propto 1/|\boldsymbol{x} - \boldsymbol{x}_{\mathbf{micro}}|$, where $\boldsymbol{x}_{\mathbf{micro}}$ is the coordinate of the micro lenses. This refinement result is aligned with our anticipation in Section 2.3.

Fig. 4 illustrates the results of the diffraction integral for the three micro + macro/pure macro lensing configurations. Each column corresponds to a different macro lensing image, with Type I on the left, Type II in the middle, and Type III on the right. The top (bottom) two rows display the absolute value (phase) of the amplification factor and its relative error between numerical and analytical results. The blue curves represent our TAAH tree algorithm, while the red curves represent the SA method. For the SA method, the resolution at the lens plane was set to 0.001, equivalent to the grid length of the maximum resolution level of the TAAH tree algorithm. It is evident that the precision of our method is comparable to or more accurate than the SA method but requires less time due to the use of fewer pixels at the lens plane. The acceleration rate is approximately two orders of magnitude. However, we do not emphasize this number here, as this is a simple case primarily intended to test our method's accuracy.

3.2. *TAAH tree algorithm results for microlensing field*

In this section, we present the results of the diffraction integral for microlensing field scenarios, considering three different macro lensing image types: Type I, Type II, and Type III. The parameters used in the calculations are listed
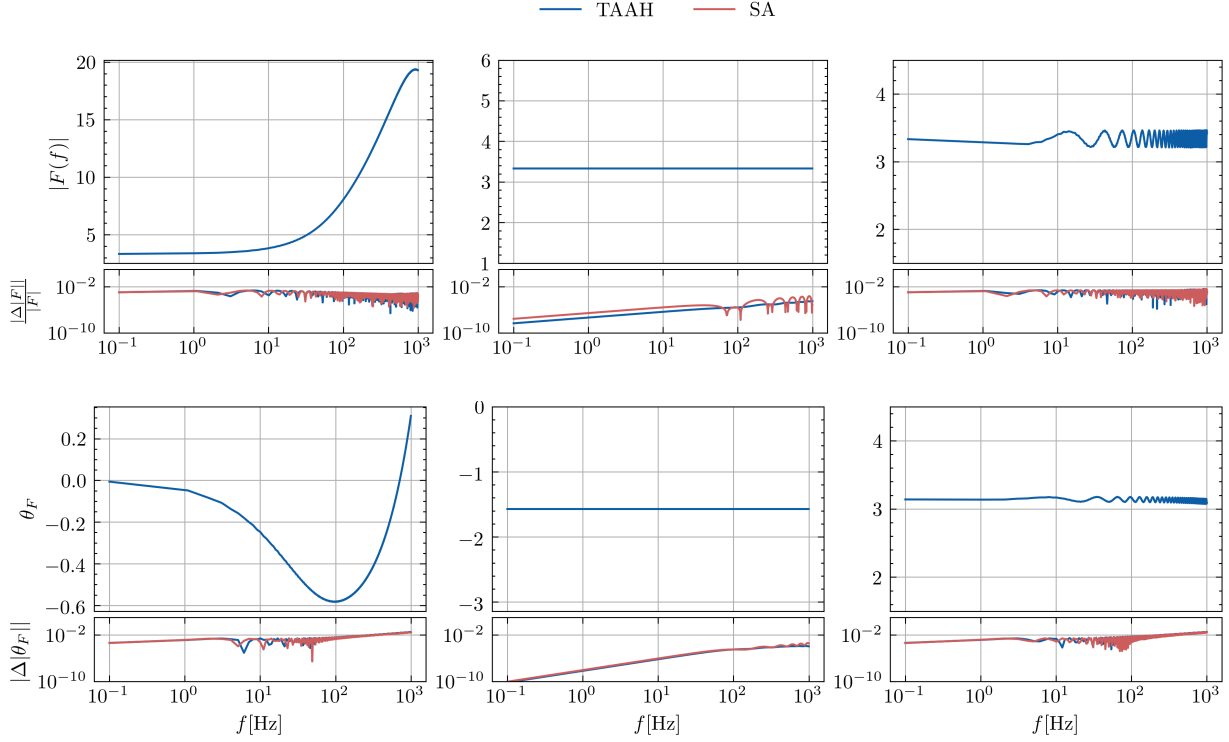
Figure 4. Diffraction integral result for three micro + macro or pure macro lensing scenarios. Each column corresponds to a different macro lensing image, with Type I on the left, Type II in the middle, and Type III on the right. The top (bottom) two rows display the absolute value (phase) of the amplification factor and its relative error between numerical and analytical results. The blue curves represent our TAAH tree method, while the red curves represent the SA method.
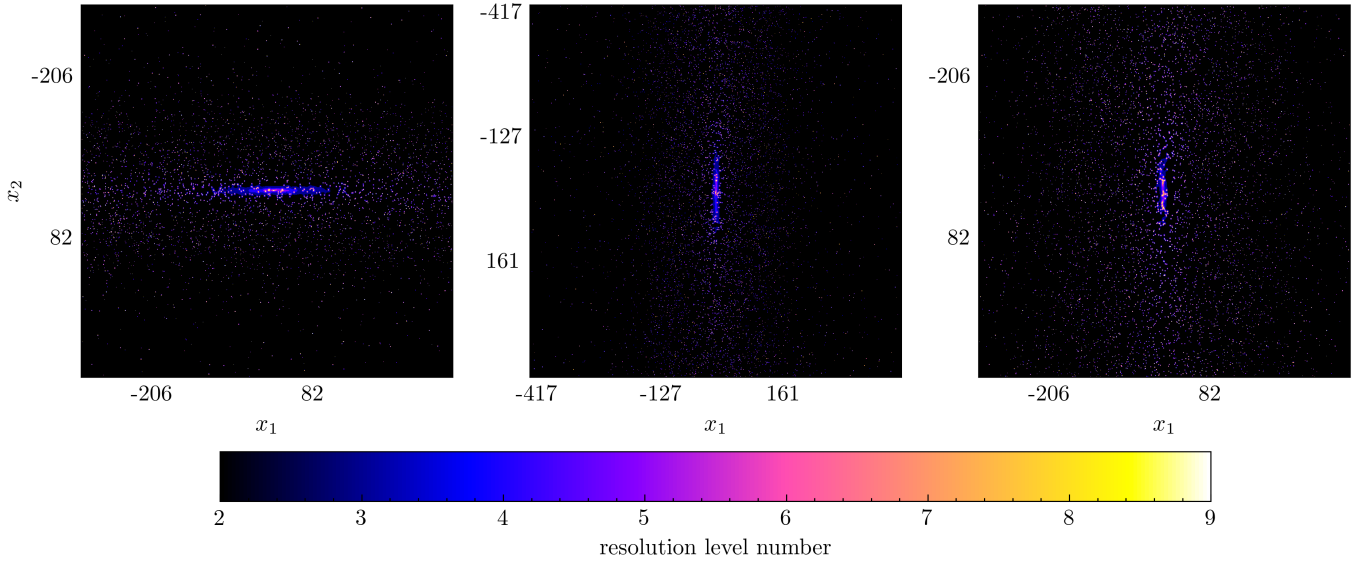
Figure 5. Resolution level numbers for microlensing field. Columns from left to right represent Type I, Type II, and Type III scenarios.

in Table.2. The lens plane boundary L_0 is determined using the method introduced in Shan et al. (2023c). We choose a precision parameter $\epsilon = 0.1$ and set the maximum grid level $n_{\max} = 9$ (resulting in a resolution of $L_9 = L_2/128$).

Table 2. This table lists the lens parameter values of Type I, Type II, and Type III macro images used in simulations. κ and γ are the macro convergence and shear. κ_* is the microlens convergence. L_0 is the lens plane boundary. ϵ is the precision parameter. Here, we set the mass of the microlens as 1 M$\odot$.

Parameter	κ	γ	κ_*	L_0	ϵ
Type I	0.7	-0.25	0.06	333.3	0.1
Type II	0.8	0.25	0.06	433	0.1
Type III	1.2	-0.15	0.06	333.3	0.1

Table 3. This table lists the lens parameter values of Type I, Type II, and Type III macro images used in simulations. κ and γ are the macro convergence and shear. κ_* is the microlens convergence. L_0 is the lens plane boundary. ϵ is the precision parameter. Here, we set the mass of the microlens as 1 M$\odot$.

	N_{L_2}	N_{L_n}	N_{L_9}	N_*	$\frac{t_{\rm SA}}{t_{\rm TAAH}}$	$\frac{t_{\rm FH}}{t_{\rm TAAH}}$
Type I	8.4×10^5	10^8	1.3×10^{10}	8486	1.4×10^4	16
Type II	1.4×10^6	1.6×10^8	2.3×10^{10}	14322	1.5×10^4	10
Type III	8.4×10^5	1.5×10^8	1.3×10^{10}	8486	1.3×10^4	15

Figure 5 displays the resolution levels for these three macro + micro lensing configurations. One can find that most grids are at Level 2 resolution (black region), with only a small part of the region requiring higher resolution. Therefore, our TAAH tree algorithm can further expedite the calculation compared to the results obtained when applying the FH tree algorithm introduced in Zheng et al. (2022).

In Table 2, we present a comparison of calculation times using different algorithms. N_{L_2} represents the pixel number of Level 2 grid, N_{L_n} denotes the total pixel number ($n > 2$) for the TAAH tree algorithm, N_{L_9} is the pixel number for the FH tree algorithm (equivalent to Level 3 grid in this algorithm) when the pixel length is equal to the pixel length of the maximum resolution ($n = 9$) of the TAAH tree algorithm, and N_* is the number of microlenses.

The sixth column represents the calculation speedup of the TAAH tree algorithm relative to the SA method. Here, the calculation time of the SA method $t_{\rm SA} = N_{L_9} \cdot N_*$, and the calculation time of the TAAH tree algorithm $t_{\rm TAAH} = 9 \cdot N_{L_n} + N_{L_2} \cdot N_*$.

The seventh column represents the calculation speedup of the TAAH algorithm relative to the FH tree algorithm. Here, the calculation time of the FH tree algorithm $t_{\rm FH} = 9 \cdot N_{L_9} + N_{L_2} \cdot N_*$.

From this table, it can be observed that the TAAH tree algorithm exhibits a speedup of approximately four orders of magnitude compared to the SA method and approximately one order of magnitude compared to the FH tree algorithm. One can find that, the acceleration are mostly from the interpolation in Level 2 grid, and the adaptive grid in Level n can further accelerate the calculation to avoid the using of GPU processor as what does in the generation of the microlensing magnification map (Zheng et al. 2022).

Figure 6 compares the diffraction integral results of our TAAH tree algorithm with the FH tree method. Here, we have not presented the results obtained using the SA method due to its excessively long calculation time, making it impractical to complete within a reasonable timeframe. In this figure, different columns represent different macro lensing image types, with Type I on the left, Type II in the middle, and Type III on the right. The top (bottom) two rows depict the absolute (phase) value of the amplification factor using our method and the differences between our method and the FH tree method. The differences between these two algorithms are less than 1%. Therefore, we conclude that our method can accelerate the diffraction integral by four orders of magnitude compared to the SA method and one order of magnitude compared to the FH tree algorithm, while maintaining high precision (numerical error of less than 1% compared to the FH tree algorithm).

4. CONCLUSION AND DISCUSSION

The gravitational lensing wave effect generated by the microlensing field embedded in the lens galaxy is an inevitable phenomenon, bringing forth both systematic challenges and opportunities for SLGW detection and application. Despite pioneers proposed various algorithms to overcome the oscillation diffraction integral (Ulmer & Goodman 1995; Press et al. 1992; Levin 1982; Filon 1930; Xiang 2007; Iserles & Nørsett 2006; Guo & Lu 2020; Tambalo et al. 2023b), directly applying these algorithms in the microlensing field scenario remains challenging. This is attributed to the multitude
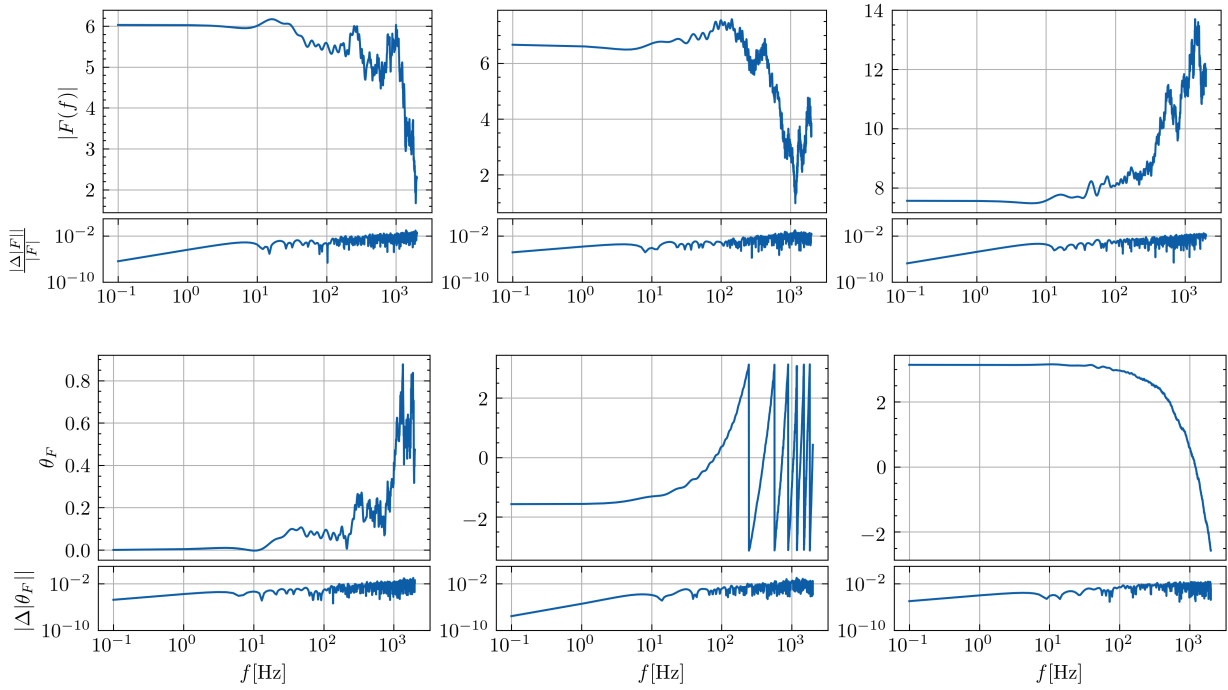
Figure 6. Diffraction integral results for the microlensing field. Columns from left to right represent Type I, Type II, and Type III macro lensing scenarios. The top (bottom) two rows are the absolute (phase) value of the amplification factor using our TAAH tree algorithm and the differences between our method and the Direct Adding method.

of microlenses in this field, numbering in the thousands, even millions, resulting in numerical calculations inevitably encountering convergence, precision, and time-consuming problems.

This paper, as a companion to our previous work (Shan et al. 2023c), where we addressed the convergence problem and improved numerical precision using a component decomposition algorithm, introduces an adaptive hierarchical tree algorithm based on the trapezoid approximation within the pixel. This algorithm effectively overcomes the remaining challenges of lens plane resolution and time-consuming.

In Fig.4, we compare our TAAH tree algorithm with the theoretical and traditional SA algorithm for a microlens embedded in Type I, Type III, and pure Type II scenarios. Our findings indicate that our algorithm achieves comparable or greater accuracy than the traditional SA algorithm while utilizing less time. Subsequently, we apply our TAAH algorithm to the microlensing field scenario. In Fig.5, we illustrate the resolution level of the TAAH tree algorithm, revealing that only a small portion of pixels requires refinement relative to the Level 2 grid. Therefore, the TAAH tree algorithm accelerates the calculation compared to the FH tree algorithm, approximately by one order of magnitude. When compared with the SA algorithm, our TAAH algorithm can have four orders of magnitude acceleration. In Fig. 6, a comparison is presented between the results obtained with our TAAH tree algorithm and those from the FH tree algorithm. Notably, our algorithm demonstrates a maximum error of only 1% under this acceleration.

In summary, we have introduced a novel numerical algorithm for the diffraction integral, specifically designed for the microlensing field embedded in the lens galaxy. This algorithm effectively resolves the numerical resolution problem and makes the numerical errors controllable. Furthermore, the incorporation of an adaptive hierarchical tree algorithm enhances computational efficiency. Consequently, this algorithm holds great potential for advancing studies on microlening wave effect, particularly in the context of density microlensing fields.

APPENDIX

REFERENCES

Bian, L., et al. 2021, Sci. China Phys. Mech. Astron., 64, 120401, doi: 10.1007/s11433-021-1781-x

Cao, S., Qi, J., Cao, Z., et al. 2022, Astron. Astrophys., 659, L5, doi: 10.1051/0004-6361/202142694

Chen, X., Shu, Y., Li, G., & Zheng, W. 2021, The Astrophysical Journal, 923, 117, doi: 10.3847/1538-4357/ac2c76

Collett, T. E., & Bacon, D. 2017, Phys. Rev. Lett., 118, 091101, doi: 10.1103/PhysRevLett.118.091101

Diego, J. M., Hannuksela, O. A., Kelly, P. L., et al. 2019, Astronomy & Astrophysics, 627, A130, doi: 10.1051/0004-6361/201935490

Fairbairn, M., Urrutia, J., & Vaskonen, V. 2023, JCAP, 07, 007, doi: 10.1088/1475-7516/2023/07/007

Filon, L. N. G. 1930, Proceedings of the Royal Society of Edinburgh, 49, 38

Finke, A., Foffa, S., Iacovelli, F., Maggiore, M., & Mancarella, M. 2021, Phys. Rev. D, 104, 084057, doi: 10.1103/PhysRevD.104.084057

Guo, X., & Lu, Y. 2020, Physical Review D, 102, 124076

Iserles, A., & Nørsett, S. P. 2006, BIT Numerical Mathematics, 46, 549

Levin, D. 1982, Mathematics of Computation, 38, 531

Liao, K., Biesiada, M., & Zhu, Z.-H. 2022, Chin. Phys. Lett., 39, 119801, doi: 10.1088/0256-307X/39/11/119801

Liao, K., Fan, X.-L., Ding, X., Biesiada, M., & Zhu, Z.-H. 2017, Nature Communications, 8, doi: 10.1038/s41467-017-01152-9

Meena, A. K., & Bagla, J. S. 2020, Mon. Not. Roy. Astron. Soc., 492, 1127, doi: 10.1093/mnras/stz3509

Mishra, A., Meena, A. K., More, A., & Bose, S. 2023. https://arxiv.org/abs/2306.11479

Mishra, A., Meena, A. K., More, A., Bose, S., & Bagla, J. S. 2021, Monthly Notices of the Royal Astronomical Society, 508, 4869–4886, doi: 10.1093/mnras/stab2875

Nakamura, T. T., & Deguchi, S. 1999, Progress of Theoretical Physics Supplement, 133, 137, doi: 10.1143/PTPS.133.137

Narola, H., Janquart, J., Haegel, L., et al. 2023. https://arxiv.org/abs/2308.01709

Oguri, M. 2019, Rept. Prog. Phys., 82, 126901, doi: 10.1088/1361-6633/ab4fc5

Press, W. H., Vetterling, W. T., Teukolsky, S. A., & Flannery, B. P. 1992, Numerical Recipes Example Book (FORTRAN) (Cambridge University Press Cambridge)

Schneider, P., Ehlers, J., & Falco, E. E. 1992, in Gravitational Lenses (Springer), 467–515

Schneider, P., Ehlers, J., & Falco, E. E. 1992, Gravitational Lenses, doi: 10.1007/978-3-662-03758-4

Seo, E., Li, T. G. F., & Hendry, M. A. 2023. https://arxiv.org/abs/2311.05543

Shan, X., Chen, X., Hu, B., & Cai, R.-G. 2023a. https://arxiv.org/abs/2301.06117

Shan, X., Chen, X., Hu, B., & Li, G. 2023b. https://arxiv.org/abs/2306.14796

Shan, X., Li, G., Chen, X., Zheng, W., & Zhao, W. 2023c, Sci. China Phys. Mech. Astron., 66, 239511, doi: 10.1007/s11433-022-1985-3

Smith, G. P., Robertson, A., Mahler, G., et al. 2023, Mon. Not. Roy. Astron. Soc., 520, 702, doi: 10.1093/mnras/stad140

Takahashi, R., & Nakamura, T. 2003, The Astrophysical Journal, 595, 1039–1051, doi: 10.1086/377430

Tambalo, G., Zumalacárregui, M., Dai, L., & Cheung, M. H.-Y. 2023a, Phys. Rev. D, 108, 103529, doi: 10.1103/PhysRevD.108.103529

—. 2023b, Phys. Rev. D, 108, 043527, doi: 10.1103/PhysRevD.108.043527

Ulmer, A., & Goodman, J. 1995, The Astrophysical Journal, 442, 67, doi: 10.1086/175422

Wambsganss, J. 1990, PhD thesis, -

Xiang, S. 2007, Numerische Mathematik, 105, 633

Zheng, W., Chen, X., Li, G., & Chen, H.-z. 2022, An Improved GPU-Based Ray-Shooting Code For Gravitational Microlensing, arXiv, doi: 10.48550/ARXIV.2204.10871