Monthly Notices
of the
ROYAL ASTRONOMICAL SOCIETY

Mon. Not. R. Astron. Soc. **415,** 881–892 (2011) doi:10.1111/j.1365-2966.2011.18754.x

An analytic ray-tracing algorithm for weak lensing

Baojiu Li,[1,2]★ Lindsay J. King,[2,3]★ Gong-Bo Zhao[4]★ and Hongsheng Zhao[5]★

[1]*DAMTP, Centre for Mathematical Sciences, University of Cambridge, Wilberforce Road, Cambridge CB3 0WA*
[2]*Kavli Institute for Cosmology Cambridge, Madingley Road, Cambridge CB3 0HA*
[3]*Institute of Astronomy, University of Cambridge, Madingley Road, Cambridge CB3 0HA*
[4]*Institute of Cosmology & Gravitation, University of Portsmouth, Dennis Sciama Building, Portsmouth PO1 3FX*
[5]*SUPA, University of St Andrews, North Haugh, St Andrews, Fife KY16 9SS*

Accepted 2011 March 20. Received 2011 February 28; in original form 2010 December 20

ABSTRACT

We propose a new ray-tracing algorithm to measure the weak-lensing shear and convergence fields directly from *N*-body simulations. We calculate the deflection of the light rays lensed by the 3D mass density field or gravitational potential along the line of sight on a grid-by-grid basis, rather than using the projected 2D lens planes. Our algorithm uses simple analytic formulae instead of numerical integrations in the computation of the projected density field along the line of sight, and so is computationally efficient and straightforward to implement. This will prove valuable in the interpretation of data from the next generation of surveys that will image many thousands of square degrees of sky.

Key words: gravitational lensing: weak – methods: analytical – methods: numerical – cosmology: theory – large-scale structure of Universe.

1 INTRODUCTION

Weak gravitational lensing (WL) (Bartelmann & Schneider 2001) is a promising tool to map the matter distribution in the Universe and constrain cosmological models, using the statistical quantities primarily constructed out of the observed correlations in the distorted images of distant source galaxies. In 2000, four teams announced the first observational detections of cosmic shear (Bacon, Refregier & Ellis 2000; Kaiser, Wilson & Luppino 2000; van Waerbeke et al. 2000; Wittman et al. 2000; Maoli et al. 2001). Since then improved observational results have been published (Hoekstra et al. 2006; Fu et al. 2008; Schrabback et al. 2010), and it has been extensively used to investigate key cosmological parameters, such as the matter density parameter $\Omega_{\rm m}$ and the normalization of the matter power spectrum, σ_8, as well as to constrain the neutrino mass (Tereno et al. 2009). Much theoretical progress has also been made in assessing the utility of cosmic shear in, for example, estimating the equation of state of dark energy w (Bridle & King 2007; Crittenden, Pogosian & Zhao 2009; Li et al. 2009), as well as its role in testing theories of modified gravity (Schmidt 2008; Zhao et al. 2009, 2010a,b; Song et al. 2010) and constraining the quintessence dark energy (Chongchitnan & King 2010).

On linear scales, one can use the linear perturbation theory to calculate the WL observables for a given cosmology, such as the shear power spectrum or the aperture mass statistic, and compare these predictions to observational data to constrain the model parameters. However, the observables on non-linear scales, which cannot be predicted theoretically without the help of *N*-body simulations, can also provide valuable information to prove or falsify cosmological models. Making such predictions using *N*-body simulations becomes increasingly important as we move into a new era in weak lensing using large observational surveys. The next generation of cosmic shear surveys, for example, the Dark Energy Survey (DES; www.darkenergysurvey.org), will be more than an order of magnitude larger in area than any survey to date, covering thousands of square degrees and using several filters that allow photometric redshift estimates for the source galaxies to be derived. These surveys have the potential to map dark matter in 3D at unprecedented precision, testing our structure formation paradigm and cosmological model.

To obtain the statistics for WL from the outputs of *N*-body simulations, one needs to construct numerous virtual light rays propagating from the source to the observer. By tracing these light rays along the lines of sight (l.o.s.), one could, in principle, calculate how much the original source image is distorted, and magnified.

Conventional ray-tracing algorithms generally project the matter distribution along the paths of light rays on to a series of lens planes and use the discrete lensing approximation to compute the total deflection of the light rays on their way to the observer (Jain, Seljak & White 2000; Hilbert et al. 2009). The lens planes could be set up either by handling the simulation outputs after the *N*-body simulation is completed or by recording corresponding light-cones on-the-fly (Heinamaki et al. 2005) and projecting later. Although this algorithm is the most frequently used in the literature, it requires a large amount of data, such as particle positions, to be stored and this would be difficult for simulations with very high mass resolution

★E-mail: b.li@damtp.cam.ac.uk (BL); ljk@ast.cam.ac.uk (LJK); gongbo.zhao@port.ac.uk (G-BZ); hz4@st-andrews.ac.uk (HZ)

or very big box-sizes, which are increasingly more common today. Furthermore, projecting particles on to a number (∼20–30) of lens planes will inevitably erase the detailed matter distribution along the l.o.s. (Martel, Premadi & Matzner 2000; Vale & White 2003) and oversimplify the time-evolution of the large-scale structure.

One can also perform the lensing computation during the N-body simulation process to obtain the projected (surface) density and/or convergence field directly (White & Hu 2000). This method avoids the expensive storage of dump data at numerous redshifts and allows the detailed matter distribution to be probed. However, it does involve numerical integrations in the calculation of the projected density field and therefore certain overheads, because in order to make the integrals accurate one has to sample the density field along the l.o.s. rather densely.

Motivated by the promise of cosmic shear surveys, and the need to make predictions of observables on non-linear scales using cosmological simulations, in this work, we introduce a new algorithm to preform ray tracing on-the-fly, which is based on that of White & Hu (2000). We calculate the deflection of a light ray as it goes through the N-body simulation grids using the 3D density field inside the grids, instead of using the density field projected on to discrete 2D lens planes. Furthermore, the numerical integration is replaced by some exact analytic formulae, which could further simplify the computation. We will show our results in comparison with the fitting formula, and discuss how our algorithm can be applied to particle or potential outputs recorded in large simulations and how we can go beyond the Born approximation and include the lens–lens coupling effect.

This paper is organized as follows. We will introduce our algorithm in the next section, describe our simulation and present the results in Section 3, and close with a section of discussion and conclusions. Although we do not include lens–lens coupling and corrections to the Born approximation in our simulations, we will outline in Appendix A how these can be done. For simplicity, we shall consider a spatially flat universe throughout this work, but the generalization to non-flat geometries is straightforward. We shall use 'grid' and 'grid cell' interchangeably to stand for the smallest unit of the mesh in the particle-mesh (PM) N-body simulations.

2 METHODOLOGY

In this section, we will first briefly review the traditional 'plane-by-plane' ray-tracing algorithm and then detail our improved 'grid-by-grid' prescription.

2.1 Conventional ray-tracing algorithm

We work in the weak-lensing regime, meaning that the light rays can be well approximated as straight lines (Mellier 1999; Bartelmann & Schneider 2001). The metric element is given by

$$\mathrm{d}s^2 = a^2[(1+2\Phi)\mathrm{d}\tau^2 - (1-2\Phi)\,\mathrm{d}\boldsymbol{x}\cdot\mathrm{d}\boldsymbol{x}] \tag{1}$$

where a is the scalefactor normalized so that $a = 1$ today, τ is the conformal time, Φ is the gravitational potential and $\boldsymbol{x}$ is the comoving coordinate. We use units such that $c = 1$.

Then, the change in the photon's angular direction as it propagates back in time is (Lewis & Challinor 2009)

$$\boldsymbol{\xi}(\chi_\mathrm{s}) - \boldsymbol{\xi}_0 = -2\int_0^{\chi_\mathrm{s}} \frac{\chi_\mathrm{s}-\chi}{\chi\chi_\mathrm{s}}\nabla_{\boldsymbol{\xi}}\Phi\,\mathrm{d}\chi, \tag{2}$$

where χ is the comoving angular diameter distance, $\boldsymbol{\xi}$ is the angular position perpendicular to the l.o.s., $\boldsymbol{\xi}_0 \equiv \boldsymbol{\xi}(\chi = 0)$, $\nabla_{\boldsymbol{\xi}}$ denotes the covariant derivative on the sphere with respect to $\boldsymbol{\xi}$ and $\Phi = \Phi(\chi, \boldsymbol{\xi})$ is the gravitational potential along the l.o.s. The 2×2 distortion matrix is given by $A_{ij} \equiv \nabla_i\xi_j \equiv \nabla_{\xi_{0i}}\xi_j(\chi)$, where ξ_{0i} is the ith component of $\boldsymbol{\xi}_0$, and is equal to

$$\begin{aligned} A_{ij} &\equiv -2\int_0^{\chi_\mathrm{s}} \frac{g(\chi,\chi_\mathrm{s})}{\chi^2}\nabla_{\xi_{0i}}\nabla_{\xi_j}\Phi(\chi,\boldsymbol{\xi})\,\mathrm{d}\chi + \delta_{ij} \\ &\approx -2\int_0^{\chi_\mathrm{s}} \frac{g(\chi,\chi_\mathrm{s})}{\chi^2}\nabla_{\xi_i}\nabla_{\xi_j}\Phi(\chi,\boldsymbol{\xi})\,\mathrm{d}\chi + \delta_{ij}, \end{aligned} \tag{3}$$

with $i, j = 1, 2$ running over the two components of $\boldsymbol{\xi}$, and

$$g(\chi,\chi_\mathrm{s}) \equiv \frac{(\chi_\mathrm{s}-\chi)\chi}{\chi_\mathrm{s}}. \tag{4}$$

Note that to obtain equation (3) we have made the approximation $\nabla_{\xi_{0i}} \approx \nabla_{\xi_i}$, which means that lens–lens coupling is ignored. We shall discuss how to go beyond this approximation in Appendix A.

This matrix is related to the convergence κ and shear components $\gamma_{1,2}$ by

$$\mathbf{A} = \begin{pmatrix} 1-\kappa-\gamma_1 & -\gamma_2-\omega \\ -\gamma_2+\omega & 1-\kappa+\gamma_1 \end{pmatrix}, \tag{5}$$

where ω stands for the rotation and $\gamma = (\gamma_1^2+\gamma_2^2)^{1/2}$ is the shear magnitude. In the weak-lensing approximation, once the convergence is obtained, the shear is determined as well; therefore, in practice, we only need to compute κ,

$$\kappa = 1 - \frac{\left(A^1{}_1 + A^2{}_2\right)}{2} = \int_0^{\chi_\mathrm{s}} g(\chi,\chi_\mathrm{s})\nabla_\xi^2\Phi\,\mathrm{d}\chi. \tag{6}$$

Under the Limber approximation (White & Hu 2000), the 2D Laplacian in equation (6) can be replaced with the 3D Laplacian, because the component of the latter parallel to the l.o.s. is negligible on small angular scales (Jain et al. 2000). Then, using the Poisson equation

$$\nabla^2\Phi = \frac{3}{2}\Omega_\mathrm{m}H_0^2\frac{\delta}{a}, \tag{7}$$

where δ is the matter overdensity and a the scalefactor, we can rewrite equation (6) as

$$\kappa(\boldsymbol{\xi}) = \frac{3}{2}\Omega_\mathrm{m}H_0^2\int_0^{\chi_\mathrm{s}} g(\chi,\chi_\mathrm{s})\frac{\delta(\boldsymbol{x})}{a}\,\mathrm{d}\chi, \tag{8}$$

in which we have written explicitly the $x_\perp$ dependence of κ (the χ dependence is integrated out in the projecting process). Equation (8) is the starting point of most ray-tracing simulations.

The most commonly used ray-tracing method is the discrete lens-plane approximation. In this approach, the density field is projected on to a number of lensing planes (usually ∼20–30) and the light rays are treated as if they were deflected only by these lens planes. Correspondingly, the term $g(\chi,\chi_\mathrm{s})$ in equation (8) is evaluated only at the positions of these planes.

The method of White & Hu (2000) incorporates the integration in equation (8) directly into their N-body simulation code and performs the integral at every time-step. To realize this, N_los straight l.o.s. are generated to be traced. The rays have specified origin (the observer at redshift 0), opening (e.g. $3\times 3\,\mathrm{deg}^2$) and orientation. As the N-body simulation process evolves to the source redshift, z_s, the convergence is computed along each l.o.s. using equation (6) or equation (8). The l.o.s. integration is then carried out numerically for each time-step, during which the photon travels from χ_i to χ_f, where the subscripts 'i' and 'f' literally stand for *initial* and *final*, respectively, and hence they are used as the integration boundaries. The integrand $g\nabla_\xi^2\Phi/\chi^2$ in equation (6) or $g\delta(\boldsymbol{x})$ in equation (8) is considered to be constant during each time-step and the integral

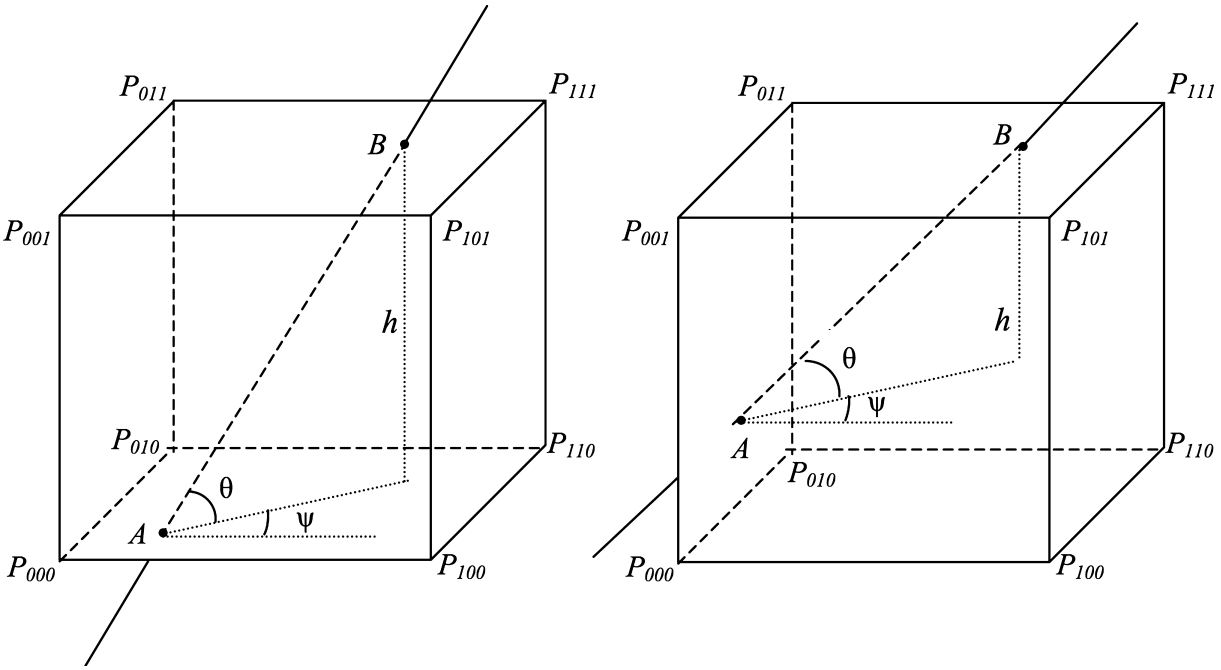

Figure 1. Two examples of the l.o.s. (lines) crossing a cubic cell of the simulation box at points A and B. The segment AB lies inside the cube. P_{xyz} (x, y, z = 0, 1) are the eight vertices of the cube. Projecting AB into the plane perpendicular to the z-direction and passing point A, ψ is the angle between the projection and x-direction, and θ is the angle between AB and the plane. For the given l.o.s. and cube, A, B, θ and ψ are known or can be computed easily, and we also know the density values at the eight vertices; we then want to integrate the density field along AB (or part of it).

is approximated by summing over all the time-steps. The time-sampling has to be sufficiently fine so as to guarantee the required numerical accuracy.

One advantage of this algorithm is that κ is computed step by step on-the-fly, so one can avoid the expensive disc storage required for storing particle dumps and the time-consuming post-processing analysis. Moreover, in this approach, there is no difficulty to make ultrafine time-sampling (the number of time-slices can be as many as the number of time-steps for the simulation after z_s) and probe the detailed 3D matter distribution (because the l.o.s. integration is performed on a grid-by-grid basis, and remember that there are at least hundreds of grid cells in each simulation box). Note that a relatively small number of lens planes in the discrete lens-plane approximation do have a non-negligible impact on the computed lensing observables; for example, Vale & White (2003) show that increasing the number from ∼20 to ∼90 could change the convergence power spectrum by ∼10 per cent.

However, one does have to carry out the numerical integration in equation (6) or equation (8), and to make the result accurate, one has to sample the value of the integrand very densely [e.g. ∼100 sampling points are dynamically chosen for each time-step in White & Hu (2000)], which might cause certain overheads when a large number of light rays are traced and ultrafine time-stepping is used.

2.2 Improved ray-tracing algorithm

In this work, we propose an improved ray-tracing algorithm by computing the convergence, shear and projected density fields on the very grid cells on which the N-body simulation is performed. In our grid-by-grid approach, the l.o.s integration can be carried out *analytically*, making the computation more efficient and accurate than the numerical integration. Also, the light rays are deflected by the detailed matter distribution exactly as seen in the N-body simulations, making the ray-tracing and N-body simulations more consistent with each other. A detailed derivation of the relevant formulae is given in Sections 2.2.1 and 2.2.2, and the basic idea is as follows. Take equation (8), for example, where the integrand is $g\delta(\boldsymbol{x})$. Since our PM code automatically computes $\delta(\boldsymbol{x})$ on the regular mesh, the value of $\delta(\boldsymbol{x})$ at any point can be obtained by interpolation, and in particular we can compute the value along the l.o.s. as a function of the comoving distance χ, and the values of $\delta(\boldsymbol{x})$ at the vertices of the grid containing the said point.[1] Note that the vertices themselves are the regular grid points and the values of $\delta(\boldsymbol{x})$ on the vertices are known. Using certain interpolation schemes, trilinear, for example, $\delta(\boldsymbol{x})$ can be expressed as a polynomial of χ; thus, the integral can be carried out analytically. Therefore, no numerical integration is needed to compute κ. Similarly, our algorithm can also be used to compute the integral of equation (6) analytically, as detailed in Section 2.2.2.

Note that when the algorithm is applied to the density field $\delta(\boldsymbol{x})$, there is some subtlety, and this will be clarified in Section 2.2.1.

2.2.1 *Method A*

To compute the projected density field, we need to integrate along the l.o.s., and in practice this integration can be carried out progressively along the segments of l.o.s. within individual cubic grid cells. The reason for such a prescription will become clear soon. Throughout this subsection, we will use z to denote the coordinate rather than redshift.

Fig. 1 shows two examples of such configurations, in which the part AB of a l.o.s. lies in a grid cell (see the figure caption for

[1] Because the l.o.s. is by approximation a straight line, once the comoving distance χ to a point is known, the corresponding x, y, z-coordinates of that point can be expressed in terms of χ and orientation angles, which are fixed when the l.o.s. are assumed to be straight, and computable otherwise.

more information). The density value at a given point on AB could be computed using trilinear interpolation, as long as we know the corresponding values at the vertices, which are denoted by ρ_{xyz} (x, y, $z = 0, 1$). To be more explicit, let us define

$$\begin{aligned}
c_0 &\equiv \rho_{000},\\
c_1 &\equiv \rho_{100} - \rho_{000},\\
c_2 &\equiv \rho_{010} - \rho_{000},\\
c_3 &\equiv \rho_{001} - \rho_{000},\\
c_4 &\equiv \rho_{110} - \rho_{010} - \rho_{100} + \rho_{000},\\
c_5 &\equiv \rho_{011} - \rho_{001} - \rho_{010} + \rho_{000},\\
c_6 &\equiv \rho_{101} - \rho_{001} - \rho_{100} + \rho_{000},\\
c_7 &\equiv \rho_{111} - \rho_{011} - \rho_{101} - \rho_{110}\\
&\quad + \rho_{100} + \rho_{001} + \rho_{010} - \rho_{000}. \qquad (9)
\end{aligned}$$

Suppose the point on AB we are considering has the coordinate (x, y, z), then the density value is given by

$$\begin{aligned}
\rho(x, y, z) &= c_0 + c_1\Delta x + c_2\Delta y + c_3\Delta z + c_4\Delta x\Delta y\\
&\quad + c_5\Delta y\Delta z + c_6\Delta x\Delta z + c_7\Delta x\Delta y\Delta z, \qquad (10)
\end{aligned}$$

where

$$\begin{aligned}
\Delta x &\equiv \frac{1}{L}(x - x_0) = \frac{1}{L}\left[a + (\chi - \chi_A)\cos\theta\cos\psi\right],\\
\Delta y &\equiv \frac{1}{L}(y - y_0) = \frac{1}{L}\left[b + (\chi - \chi_A)\cos\theta\sin\psi\right],\\
\Delta z &\equiv \frac{1}{L}(z - z_0) = \frac{1}{L}\left[c + (\chi - \chi_A)\sin\theta\right], \qquad (11)
\end{aligned}$$

where L denotes the size of the cubic grid cell, (x_0, y_0, z_0) is the coordinate of the vertex P_{000}, χ_A is the χ value at point A, and a, b, c are the coordinates of point A relative to P_{000}.

Because we express $\rho(x, y, z)$ in terms of χ only, the line integral along AB could be rewritten as an integral over χ, and we have

$$\begin{aligned}
&\frac{1}{\chi_s}\int_{\chi_l}^{\chi_u}\chi(\chi_s - \chi)\rho(x, y, z)\,\mathrm{d}\chi\\
&= \frac{1}{\chi_s}\sum_{N=1}^{4} d_N \int_{\chi_l}^{\chi_u}\chi(\chi_s - \chi)(\chi - \chi_A)^{N-1}\,\mathrm{d}\chi\\
&= \frac{1}{\chi_s}\sum_{N=1}^{4} d_N \int_{\tilde{\chi}_l}^{\tilde{\chi}_u}(\tilde{\chi} + \chi_A)(\chi_s - \chi_A - \tilde{\chi})\,\tilde{\chi}^{N-1}\,\mathrm{d}\tilde{\chi}\\
&= \sum_{N=1}^{4}\frac{d_N}{N}\chi_A\left(1 - \frac{\chi_A}{\chi_s}\right)\left(\tilde{\chi}_u^N - \tilde{\chi}_l^N\right)\\
&\quad + \sum_{N=1}^{4}\frac{d_N}{N+1}\left(1 - 2\frac{\chi_A}{\chi_s}\right)\left(\tilde{\chi}_u^{N+1} - \tilde{\chi}_l^{N+1}\right)\\
&\quad - \sum_{N=1}^{4}\frac{d_N}{N+2}\frac{1}{\chi_s}\left(\tilde{\chi}_u^{N+2} - \tilde{\chi}_l^{N+2}\right), \qquad (12)
\end{aligned}$$

in which $\chi_l \geq \chi_A$ and $\chi_u \leq \chi_B$[2] are the lower and upper limits of the integral, respectively, $\tilde{\chi} \equiv \chi - \chi_A$, $\tilde{\chi}_u \equiv \chi_u - \chi_A$, $\tilde{\chi}_l \equiv \chi_l - \chi_A$ and we have also defined

$$\begin{aligned}
d_1 &\equiv c_0 + \frac{1}{L}(ac_1 + bc_2 + cc_3)\\
&\quad + \frac{1}{L^2}(abc_4 + bcc_5 + acc_6) + \frac{1}{L^3}abcc_7,\\
d_2 &\equiv \frac{1}{L}\cos\theta\cos\psi c_1 + \frac{1}{L}\cos\theta\sin\psi c_2 + \frac{1}{L}\sin\theta c_3\\
&\quad + \frac{1}{L^2}\cos\theta\sin\psi(ac_4 + cc_5)\\
&\quad + \frac{1}{L^2}\cos\theta\cos\psi(bc_4 + cc_6)\\
&\quad + \frac{1}{L^2}\sin\theta(bc_5 + ac_6) + \frac{1}{L^3}\sin\theta abc_7\\
&\quad + \frac{1}{L^3}\cos\theta\sin\psi acc_7 + \frac{1}{L^3}\cos\theta\cos\psi bcc_7,\\
d_3 &\equiv \frac{1}{L^2}\cos^2\theta\sin\psi\cos\psi c_4 + \frac{1}{L^2}\sin\theta\cos\theta\sin\psi c_5\\
&\quad + \frac{1}{L^2}\sin\theta\cos\theta\cos\psi c_6 + \frac{1}{L^3}\sin\theta\cos\theta\sin\psi ac_7\\
&\quad + \frac{1}{L^3}\sin\theta\cos\theta\cos\psi bc_7\\
&\quad + \frac{1}{L^3}\cos^2\theta\sin\psi\cos\psi cc_7,\\
d_4 &\equiv \frac{1}{L^3}\sin\theta\cos^2\theta\sin\psi\cos\psi c_7. \qquad (13)
\end{aligned}$$

Note that by writing the result in the above form, we have separated the treatments for four types of variables:

(i) a, b, c, χ_A, $\tilde{\chi}_{u,l}$: a, b, c and χ_A are determined by the direction of the light ray and the specific grid cell under consideration, and $\tilde{\chi}_{u,l}$ depend only on the considered time-step and χ_A. Note that a, b and c must be determined carefully, and for each grid cell at least one of them vanishes, but exactly which of them vanishes varies from ray to ray and from grid cell to grid cell;

(ii) θ, ψ: these specify the direction of the light ray, and terms involving them only need to be computed once, that is, at the beginning of the simulation, for a given l.o.s.;

(iii) c_{0-7}: these are determined by the values of ρ at the vertices of a grid cell and must be evaluated for each grid cell that the light ray passes through;

(iv) L, χ_s: these are constants for a given simulation.

Therefore, once ρ_{xyz} is known, the integral can be performed analytically without much computational effort. This is not unexpected, because once the density is known at the vertices of the grid, we should know the density at any point inside the grid using interpolation, and no more information is needed to carry out the integral. If we consider a different grid, a different set of ρ_{xyz} needs to be used, and this is why our algorithm is based on the individual grids.

There are two technical points which need to be noted. First, in equation (12), ρ should be replaced by ρ/a in practice. It is true that a could be expressed as a function of χ as well, once the background cosmology is specified, but this will lead to more complicated expressions. Therefore, in our simulations, we simply take a to be constant during each time-step. This is certainly only an approximation, but we should note that a is considered as constant during each time-step in the N-body simulations anyway. Indeed,

[2] Note that A and B are the intersections between the l.o.s. and the grid cell, and not necessarily the two ends of the l.o.s. in one time-step. However, the integration is carried out for each time-step and so we do not always have $\chi_l = \chi_A$ and $\chi_u = \chi_B$.

as we see in Section 2.2.2, the factor $1/a$ does not appear if we use $\nabla_\xi^2\Phi$ instead of ρ in the integral.[3]

Secondly, as has been mentioned by various papers (e.g. Jain et al. 2000; White & Hu 2000), the use of the 3D Laplacian [equation (8)] instead of the 2D one [equation (6)] is at best an approximation. We have to test the validity of this approximation. In fact, as we show below, the error caused by this approximation is actually not negligible. To see this, recall that

$$\begin{aligned}\kappa &= \int_0^{\chi_s} g\left(\nabla^2 - \nabla_\chi^2\right)\Phi\,\mathrm{d}\chi \\ &= \frac{3}{2}\Omega_\mathrm{m}H_0^2\int_0^{\chi_s} g\frac{\delta}{a}\mathrm{d}\chi - \left[g\nabla_\chi\Phi\right]_0^{\chi_s} \\ &\quad + \int_0^{\chi_s} g\dot{\nabla}_\chi\Phi\mathrm{d}\chi + \int_0^{\chi_s} g'\nabla_\chi\Phi\,\mathrm{d}\chi \end{aligned} \tag{14}$$

in which a prime (overdot) denotes the χ (time) derivative and the last three terms come from the treatment of $\nabla_\chi^2\Phi$, including the integration by parts. The common argument is that the second term actually vanishes as $g = 0$ and $\nabla_\chi\Phi < \infty$ at $\chi = \chi_\mathrm{s}$ and $\chi = 0$, and the last two terms are negligible. This is true in the ideal case, but while our algorithm [and that of White & Hu (2000)] is applied, the second term is no longer zero because of the following reasons:

(i) It is unrealistic to make the simulation boxes big enough to contain the whole light-cone and, in practice, people tile different simulations to form a complete light-cone. Unless a periodic tiling of the same box is adopted, we expect the matter distribution and thus the potential Φ to be discontinuous at the tiling boundaries. As a result, the second term in equation (14) should read

$$\left[g\nabla_\chi\Phi\right]_0^{\chi_s} = \left[g\nabla_\chi\Phi\right]_{\chi_{l1}}^{\chi_s} + \left[g\nabla_\chi\Phi\right]_{\chi_{l2}}^{\chi_{u2}} + \cdots + \left[g\nabla_\chi\Phi\right]_0^{\chi_{uN}}$$

in which $\chi_{\mathrm{u,l}}$ correspond to the values of χ when the light ray goes through a given box, which is labelled as $1, 2, \ldots, N$. If the matter distribution is smooth at the boundaries of the boxes, then $\nabla_\chi\Phi(\chi = \chi_{l1}) = \nabla_\chi\Phi(\chi = \chi_{u2})$ and so on, so all terms cancel. However, if the matter distribution is not smooth, as is the case for many tiling treatments, then such cancelling will not happen and $[g\nabla_\chi\Phi]_0^{\chi_s}$ will turn out to be non-zero in the numerical calculation, although it should be zero in theory.

(ii) Using the same argument as above, we could find that this discontinuity problem appears not only on the boundaries of the tiled simulation boxes, but also at *each time-step* in the simulations and *each time* the light ray passes through a grid of the simulation box. For the former case, suppose that during one time-step the l.o.s. ends at point C, then C is also the point where this l.o.s. starts during the next time-step. However, the values of $\nabla_\chi\Phi$ at point C are generally different in the two time-steps because particles have been advanced and so a discontinuity appears. For the latter case, our piecewise l.o.s. integral and the interpolation scheme dictate that the values of $\nabla_\chi\Phi$ at a point D on the interface of two neighbouring grids could depend on which grid is supposed to contain point D (remember that the interpolation scheme uses the values of $\nabla_\chi\Phi$ at the vertices of the *containing* grid cell), and naturally a discontinuity in $\nabla_\chi\Phi$ appears at the interface of the two grids. Note that these discontinuities are inevitable due to the nature of the numerical simulation (the discreteness in time), and decreasing the grid size or the length of time-steps does not help because then such discontinuities will only appear more frequently.[4]

The way to tackle these problems is as follows: we know that $[g\nabla_\chi\Phi]_0^{\chi_s}$ vanishes rigorously, in principle, but is non-zero because of the nature of the simulation; meanwhile, the same discontinuity problem also appears when calculating the first quantity on the right-hand side of equation (14). The errors in the numerical values for these two quantities are caused by the same discontinuity and could cancel each other. The exact value of this error can be obtained by computing $[g\nabla_\chi\Phi]$, because this quantity is zero in theory and its non-zero value is completely the error. In our simulations, we compute $[g\nabla_\chi\Phi]$ explicitly whenever the light ray passes a grid and subtract it according to equation (14); this way we can eliminate the error in the integration of $g\delta/a$ due to the discontinuities.

As for the third and fourth terms in equation (14), the third term is non-zero but small in reality, but in our simulations, it vanishes because Φ is assumed to be constant during any given time-step. This will cause certain unavoidable errors that we anyway expect to be small. The fourth term has as small a contribution, but, fortunately, we can perform the integral exactly and analytically as we have done for the first term in equation (14).

We have run several tests to check the accuracy of the approximations and found the following:

(i) If we simply replace the 2D Laplacian in equation (6) with a 3D one, as in equation (8), then the difference is of the order of 10 per cent and even much larger for the rays for which $|\kappa|$ is small. This agrees with the discovery of White & Hu (2000), which shows that the difference between the values of κ computed using these two expressions is of, on average, 0.003, while the theoretical value of κ is mostly of the order of 0.01–0.1. Note that equation (6) can be evaluated exactly as will be described in Section 2.2.2.

(ii) If we explicitly calculate the term $[g\nabla_\chi\Phi]_0^{\chi_s}$ for each cell crossed by a ray and subtract it according to equation (14), the difference between equations (6) and (8) is brought down to the level of 1–2 per cent.

(iii) If we further include the contribution from the fourth term of equation (14), the difference will fall well within the per cent level.

2.2.2 *Method B*

The method described in Section 2.2.1 is only applicable to equation (8), while there are also motivations for us to consider equation (6). For example, the use of the 3D Laplacian instead of the 2D Laplacian in equation (8) is at best an approximation and only works well on small angular scales. This is even worse in the discrete lensing approximation, because the photons at equal distance from the observer are certainly not in a plane but on a spherical shell, and this has motivated more accurate treatments such as the prescription proposed by Vale & White (2003). As another example, within the current framework, the shear is not computed directly but from its relation with κ. There is certainly no problem with this, but it will be even better if we can compute $\gamma_{1,2}$ directly and compare with the results obtained from κ.

[3] This just reflects the fact that during each time-step of the N-body simulation, the $1/a$ factor in the Poisson equation is treated as constant. The nature of the numerical simulation (discreteness in time) dictates that we cannot do better, say, by decreasing the length of time-steps, which we cannot always keep doing in reality.

[4] Interestingly, the discrete lens-plane approximation does not have this problem (as long as simulation boxes are tiled periodically so that the matter distribution is smooth on the tiling boundaries), because it does not treat the l.o.s. integral on a grid-by-grid basis.

Our generalized treatment here is quite simple, taking advantage of the fact that the PM codes also give us the values of $\Phi(\boldsymbol{x})$ and (if necessary) $\nabla_i\nabla_j\Phi$ at the regular grid points. For simplicity, let us assume that (1) the central l.o.s. is parallel to the x-axis; and (2) the opening of the l.o.s. bundle is a square with its sides parallel to the y-axis and z-axis. In the 2D plane perpendicular to the l.o.s., the $i = 1$ and 2 directions are set to be the longitude and latitude, respectively. We also define

$$\begin{aligned}
\mu &\equiv \nabla_x\nabla_x\Phi,\\
\nu &\equiv \nabla_y\nabla_y\Phi,\\
\eta &\equiv \nabla_z\nabla_z\Phi,\\
\zeta &\equiv \nabla_x\nabla_y\Phi = \nabla_y\nabla_x\Phi,\\
\omega &\equiv \nabla_y\nabla_z\Phi = \nabla_z\nabla_y\Phi,\\
\varpi &\equiv \nabla_x\nabla_z\Phi = \nabla_z\nabla_x\Phi,
\end{aligned} \tag{15}$$

to lighten the notation. Then, given the values of μ, ν, ... at the vertices of a grid, their values at any point inside that grid can be obtained using trilinear interpolation just as we have done for ρ in Section 2.2.1.

Now, for the configuration depicted in Fig. 1, we have, after some exercise of geometry,

$$\begin{aligned}
\nabla_1\nabla_1\Phi &= \chi^2\cos^2\theta(\mu\sin^2\psi + \nu\cos^2\psi - \zeta\sin 2\psi),\\
\nabla_2\nabla_2\Phi &= \chi^2(\mu\cos^2\psi\sin^2\theta + \nu\sin^2\psi\sin^2\theta + \eta\cos^2\theta)\\
&\quad + \chi^2\zeta\sin 2\psi\sin^2\theta - \chi^2\varpi\cos\psi\sin 2\theta\\
&\quad - \chi^2\omega\sin\psi\sin 2\theta,\\
\nabla_1\nabla_2\Phi &= \frac{1}{2}\chi^2\sin 2\theta\left[\frac{1}{2}(\nu-\mu)\sin 2\psi + \zeta\cos 2\psi\right]\\
&\quad + \chi^2\cos^2\theta\,(\varpi\sin\psi - \omega\cos\psi),\\
\nabla_\chi\nabla_\chi\Phi &= \mu\cos^2\psi\cos^2\theta + \nu\sin^2\psi\cos^2\theta + \eta\sin^2\theta\\
&\quad + \zeta\sin 2\psi\cos^2\theta + \varpi\cos\psi\sin 2\theta\\
&\quad + \omega\sin\psi\sin 2\theta.
\end{aligned} \tag{16}$$

Note that the above expressions are all linear in $\mu, \nu, \ldots$, making the situation quite simple. As an example, for $\nabla^2_\xi\Phi = \nabla^1\nabla_1\Phi + \nabla^2\nabla_2\Phi$, we have

$$\begin{aligned}
\nabla^2_\xi\Phi &= (\sin^2\psi + \cos^2\psi\sin^2\theta)\mu\\
&\quad + (\cos^2\psi + \sin^2\psi\sin^2\theta)\nu + \eta\cos^2\theta\\
&\quad - \zeta\sin 2\psi\cos^2\theta - \varpi\cos\psi\sin 2\theta - \omega\sin\psi\sin 2\theta,
\end{aligned}$$

as is consistent with Castro, Heavens & Kitching (2005), and because ψ and θ are constants for a given ray,

$$\begin{aligned}
&\int_0^{\chi_s}\frac{\chi(\chi_s-\chi)}{\chi_s}\nabla^2_\xi\Phi\,\mathrm{d}\chi\\
&= (\sin^2\psi + \cos^2\psi\sin^2\theta)\langle\mu\rangle - \langle\zeta\rangle\sin 2\psi\cos^2\theta\\
&\quad + (\cos^2\psi + \sin^2\psi\sin^2\theta)\langle\nu\rangle + \langle\eta\rangle\cos^2\theta\\
&\quad - \langle\varpi\rangle\cos\psi\sin 2\theta - \langle\omega\rangle\sin\psi\sin 2\theta,
\end{aligned} \tag{17}$$

where

$$\langle\mu\rangle \equiv \int_0^{\chi_s}\frac{\chi(\chi_s-\chi)}{\chi_s}\mu\mathrm{d}\chi \tag{18}$$

(and similarly $\langle\nu\rangle$, ...) are computed exactly as in equation (12). Note that we only need to compute $\langle\mu\rangle$, ... during the N-body simulations and multiply appropriate coefficients as in equation (17) to obtain κ finally. The components of the shear field (γ_1, γ_2) could be computed using the same formula as equation (17), but with $\nabla^2_\xi\Phi$ replaced with $\nabla^1_1\Phi - \nabla^2_2\Phi$ and $\nabla^1_2\Phi$, respectively, using the expressions given in equation (16).

3 *N*-BODY AND RAY-TRACING SIMULATIONS

To test our algorithm, we have performed a series of N-body simulations for a concordance cosmology using the publicly available code MLAPM (Knebe, Green & Binney 2001). As it is not our intention to carry out very high resolution simulations here, we only use the PM part of MLAPM so that our simulation grid is not self-adaptively refined. We have also developed a C code, RATANA (which stands for ANAlytic RAy-Tracing), to compute the convergence and shear fields on-the-fly as described in the above section. This section is devoted to a summary of our results.

3.1 Specifications for *N*-body simulations

We consider a concordance cosmology with the cosmological parameters $\Omega_\mathrm{m} = 0.257$, $\Omega_\Lambda = 0.743$, $h \equiv H_0/(100\,\mathrm{km\,s^{-1}\,Mpc^{-1}}) = 0.719$, $n_\mathrm{s} = 0.963$ and $\sigma_8 = 0.769$. The simulations start at an initial redshift $z_\mathrm{i} = 49.0$ and initial conditions (i.e. initial displacements and velocities of particles) are generated using GRAFIC (Bertschinger 1995). In this work, we only consider a source redshift $z_\mathrm{s} = 1.0$, though other values of z_s or even multiple source redshifts can easily be implemented. The field of view is $5 \times 5\,\mathrm{deg}^2$ and we trace 1024^2 light rays.

A source at redshift $z_\mathrm{s} = 1.0$ is about $2374\,h^{-1}$ Mpc away from us ($z = 0$) in terms of the comoving angular diameter distance and it is unrealistic for us to have a simulation box which is large enough to cover the whole light-cone. In this work, we adopt the tiling scheme introduced by White & Hu (2000). They use multiple simulation boxes to cover the light-cone between $z = 0$ and z_s, and the sizes of the simulation boxes are adjusted so that smaller boxes are used as the light rays get closer to the observer. It has been argued that the use of multiple tiling boxes can compensate the lack of the statistical independence of fluctuations caused by using the same simulation box repeatedly. Also the variable box-sizes mean that one can get better angular resolutions by using smaller boxes near the observer.

Similar to White & Hu (2000), we choose six different box-sizes and 20 tiles between $z = 0$ and z_s, and the details are summarized in Table 1. For the N-body simulations (regardless of the box-sizes), we use a regular mesh with $512 \times 512 \times 512$ cubic cells. We use the triangular-shaped cloud (TSC) scheme to assign the matter densities in the grid cell and to interpolate the forces (Hockney & Eastwood 1981; Knebe et al. 2001). Given the matter densities in the cells, the gravitational potential Φ is computed using fast Fourier transform (FFT) and the gravitational forces (first derivatives of Φ) as well as the second derivatives of Φ are then obtained by performing finite differences. These derivatives of Φ are subsequently utilized by RATANA to compute the convergence and shear fields as described in the above section.

Note that unlike in many other works, we use the same grid for both N-body and ray-tracing simulations. The TSC scheme we are using then results in some small-scale details of the matter distribution being smoothed out, as compared to the conventional

Table 1. The tiling solution of our N-body simulations. Here $a_{\rm out}$ is the scalefactor at the time when the light rays which are traced leave a given tile and B is the size of the simulation box in units of h^{-1} Mpc. Each simulation uses exactly 400 time-steps from $z = 49$ to 0. $N_{\rm real}$ is the number of realizations for each value of box-size. To obtain a tiling solution, we *randomly* pick out two *different* simulation boxes with $B = 240$, two with $B = 200$, two with $B = 160$, two with $B = 120$, two with $B = 100$ and 10 with $B = 80$ – a total of 20 simulation boxes of different sizes.

$a_{\rm out}$	$B\,(h^{-1}\,{\rm Mpc})$	$N_{\rm real}$	$a_{\rm out}$	$B\,(h^{-1}\,{\rm Mpc})$	$N_{\rm real}$
0.527	240	10	0.799	80	20
0.561	240	–	0.819	80	–
0.593	200	10	0.838	80	–
0.628	200	–	0.860	80	–
0.657	160	10	0.880	80	–
0.686	160	–	0.902	80	–
0.711	120	10	0.927	80	–
0.735	120	–	0.951	80	–
0.757	100	10	0.976	80	–
0.780	100	–	1.000	80	–

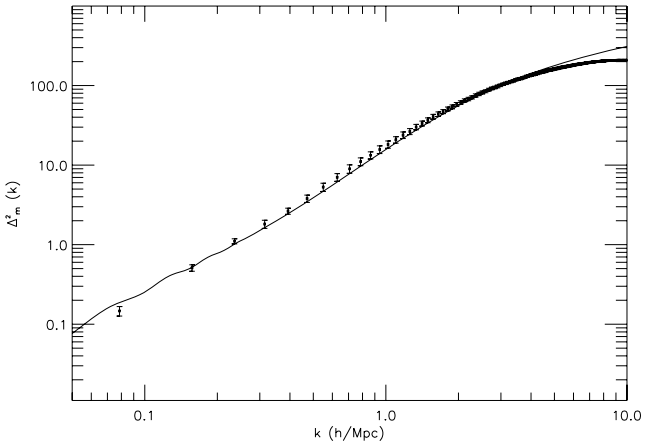

Figure 2. Plotted is the $\Delta^2_{\rm m}(k) \equiv k^3 P(k)/(2\pi^2)$, in which $P(k)$ is the matter power spectrum, as a function of the wavenumber k in units of $h\,{\rm Mpc}^{-1}$. The symbols with error bars represent averaged results at $z = 0$ from 10 realizations for the $B = 80\,h^{-1}$ Mpc simulations. The solid curve is the corresponding result using the Smith et al. (2003) fit and the same set of cosmological parameters.

nearest grid point (NGP) or cloud-in-cell (CIC) density-assignment schemes.[5] We will comment on this point later.

3.2 Numerical results

In this section, we summarize the numerical results from our N-body and ray-tracing simulations.

In Fig. 2, we compare the matter power spectra [or equivalently $\Delta^2_{\rm m}(k)$ defined in the figure caption), computed from our N-body simulations (box-size $80\,h^{-1}$ Mpc), to the prediction of the analytic fitting formula of Smith et al. (2003). We can see a good agreement, except in the range of $0.5 \leq k \leq 2.0\,h\,{\rm Mpc}^{-1}$ where the N-body simulations predict a slightly higher power. However, the agreement becomes poor for $k > 5.0\,h\,{\rm Mpc}^{-1}$ because the resolution of our simulations is not high enough, but this could be overcome in future higher resolution simulations.

To show that our ray-tracing simulations produce reasonable results, we first consider the convergence and shear maps from a chosen realization of the tiling solution; these are shown in Fig. 3. We have computed the convergence field $\kappa(\boldsymbol{\xi})$, using the two methods outlined in Sections 2.2.1 [Method A, equation (14), left-hand panel of Fig. 3] and 2.2.2 [Method B, equation (17), right-hand panel of Fig. 3]. The two methods give almost identical results and, as we have checked, the difference is in general well within the per cent level. The shear field (γ_1, γ_2) is also calculated using two methods: method A using an equivalence of equation (17) as described in the figure caption (left-hand panel) and method B which is often used in the literature, namely by Fourier transforms of the convergence field which only works in the weak-lensing regime (right-hand panel). The shear fields are shown in rods along with the convergence map shown as images. Again, the agreement for the shear field is very good, indicating that our ray-tracing algorithm works well.

We also show the lensing convergence power spectrum measured from our ray-tracing simulations in Fig. 4. Due to our limited field of view of $5 \times 5\,{\rm deg}^2$, we cannot measure the spectrum at multiple moments $\ell < 100$. Also, there is a roll-off of power at $\ell > 2000$, which is because (1) the resolution for our N-body simulations is not high enough; and (2) the TSC density-assignment scheme smoothes out the small-scale structure more than the CIC and NGP schemes do. Both factors tend to suppress the convergence power spectrum at high ℓ and we hope to solve this problem by using higher resolution simulations and more suitable interpolation schemes, which is left for our future study. Otherwise, we find that the ray-tracing result agrees reasonably well with the analytic prediction using the fitting formula for the matter power spectrum by Smith et al. (2003) in some ℓ range, that is, $100 < \ell < 2000$. On some scales, we see that the numerical result is slightly higher than the theoretical prediction. This is, however, as expected because we have seen from Fig. 2 that the N-body simulations give a higher matter power spectrum than the Smith et al. (2003) fit on some scales. The fact that a difference in the matter power spectra from simulations and analytic fitting could cause differences in the computed convergence power spectra has been reported and discussed by many authors, for example, Vale & White (2003), Hilbert et al. (2009) and Pielorz et al. (2010). Note that on small scales ($\ell > 3000, k > 1\,{\rm Mpc}^{-1}$) baryons, with their more complex physics, become important in predicting the matter power spectrum (e.g. White 2004; Zhan& Knox 2004; Rudd, Zentner & Kravtsov 2008; Guillet, Teyssier & Colombi 2010); thus, one has to include the hydrodynamical process for baryons in the simulation to obtain an accurate prediction for the weak-lensing statistics on those scales.

In Fig. 4, we overplot the expected observational uncertainty from the DES using the survey parameters $f_{\rm sky} \sim 0.12$, $\bar{n}_{\rm g} = 10\,{\rm arcmin}^{-2}$ and $\gamma^2_{\rm int} \sim 0.16$, where $f_{\rm sky}$, $\bar{n}_{\rm g}$ and $\gamma^2_{\rm int}$ denote the sky coverage, number of galaxies per arcminute squared and the mean-square intrinsic ellipticity, respectively. We can see that the cosmic variance and the shot noise dominate the noise budget on large and small scales, respectively, and our simulation result agrees with the Halofit prediction well within the DES sensitivity.

Here we have only used the DES as an example for comparison, for which the error bars are quite large compared with those of the simulations, which seems to imply no point of using high-precision numerical simulations instead of the analytical fittings. However, other surveys, such as the Large Synoptic Survey, Kunlun Dark Universe Survey Telescope (Zhao et al. 2010c) and Extremely Large Telescopes, will have smaller error bars on the lensing power

[5] In the TSC scheme, the density on a grid cell depends on the distribution of particles on all the 26 neighbouring grid cells; in the CIC (NGP) scheme, it depends on the matter distribution on the six direct neighbouring grid cells (the particles in that cell only).

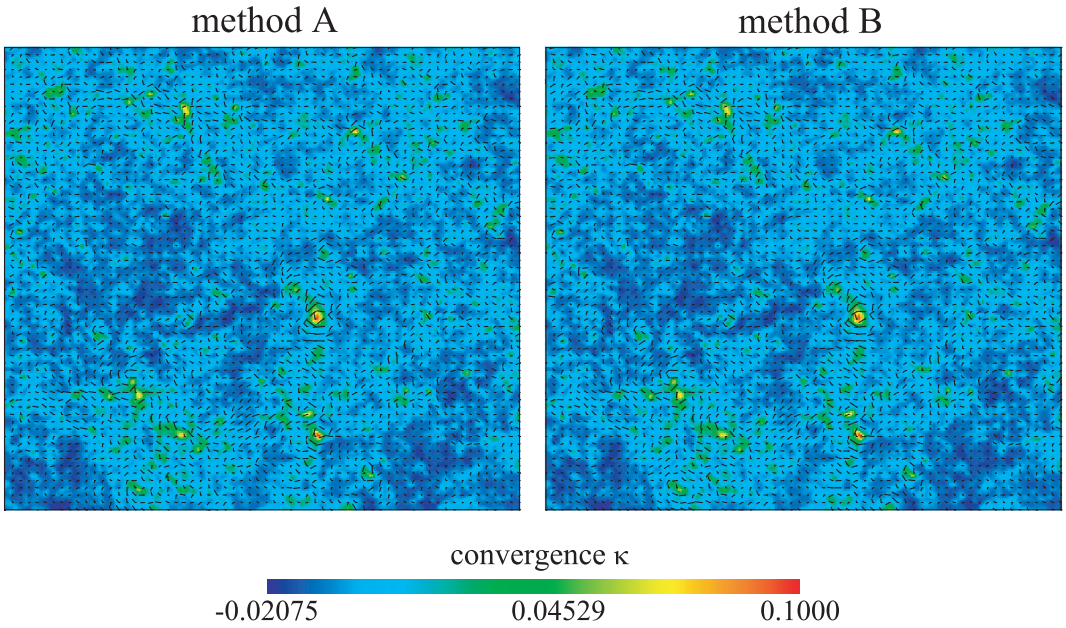

Figure 3. Convergence and shear maps ($5 \times 5\,\mathrm{deg}^2$) for one realization from the tiling solutions. The convergence field κ is shown as a colour-scale plot and the values are indicated by the colour bar below; the shear field (γ_1, γ_2) is shown as a flow plot and superposed on the κ field for comparison. As expected, the shear field is tangential around high-κ regions. Note that: (1) the κ field in the left-hand panel is computed according to equation (14) with the last three correction terms incorporated as described in Section 2.2.1; (2) the κ field in the right-hand panel is computed using equation (17); (3) the (γ_1, γ_2) field in the left-hand panel is computed using equation (17) but with $\nabla_\perp^2\Phi$ replaced by $\nabla_1^2\Phi - \nabla_2^2\Phi$ (for γ_1) and $\nabla_1\nabla_2\Phi$ (for γ_2); and (4) the (γ_1, γ_2) field in the right-hand panel is computed indirectly by Fourier transforming $\kappa(\boldsymbol{\theta})$ to $\tilde{\kappa}(\boldsymbol{l})$, computing $(\tilde{\gamma}_1, \tilde{\gamma}_2) = (\frac{l_1^2 - l_2^2}{l_1^2 + l_2^2}\tilde{\kappa}, \frac{2l_1 l_2}{l_1^2 + l_2^2}\tilde{\kappa})$ and finally inverse Fourier transforming $(\tilde{\gamma}_1, \tilde{\gamma}_2)$ to (γ_1, γ_2).

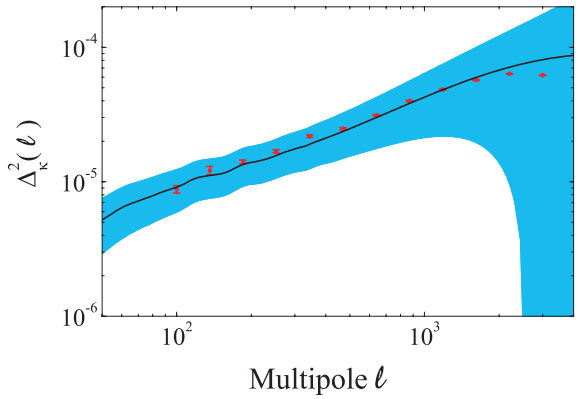

Figure 4. The convergence power spectrum $\Delta_\kappa^2 \equiv l^2 C_l/(2\pi)$ as measured from our ray-tracing simulations (symbols with error bars). The result is obtained by averaging 120 realizations of the tiling solution. The solid curve is again obtained using the Smith et al. (2003) fit of the matter power spectrum (Kaiser 1992; Jain & Seljak 1997) and the filled band illustrates the expected observational uncertainty from the DES.

spectrum. The accuracy with which simulations can be performed matters more and more for these and further-future facilities. Also, since our algorithm will be used to study cosmologies with more exotic physics, for which we do not generally have fitting formulae such as that of Smith et al. (2003), it is important to test its reliability using the Λ cold dark matter paradigm, which is what we have done here. Furthermore, other statistics, such as cross-correlation functions, are straightforward to construct from our ray-tracing simulations, but these will be left for future work.

Note that in our numerical simulations we have not included the lens–lens coupling and second-order corrections to the Born approximation. In Appendix A, we will outline how these can be incorporated in future higher resolution simulations.

4 DISCUSSION AND CONCLUSIONS

The correlations in the distorted images of distant galaxies, induced by cosmic shear, hold information about the distribution of matter on a wide range of scales in the universe. In order to take the full advantage of current and future weak-lensing data sets to constrain cosmology, using information from both the linear and non-linear regimes, one needs a sophisticated algorithm to measure the shear and convergence fields from N-body simulations, and to construct statistical quantities. This is traditionally done using the 'plane-by-plane' discrete lens-plane algorithm – trace the virtual light rays and calculate the deflection caused by the density field projected on to a number of 2D lens planes.

In this work, we propose an improved ray-tracing algorithm. We calculate the deflection of the light rays caused by the detailed 3D density fields living on the natural simulation mesh, rather than the simplified density distribution projected on to some 2D planes. We evaluate the shear and convergence fields by analytically integrating the deflection as the light rays go through the individual simulation grid cells. This approach is easy to implement. It avoids numerical integration and expensive data storage since it is performed on-the-fly. We apply the algorithm to our simulations and find good agreement with the Smith et al. (2003) fit and consistency with the published results in Sato et al. (2009).

The on-the-fly l.o.s. integration is computationally economic. In the RATANA code, most computation time is spent on the N-body part. Suppose N_d^3 is the number of grid cells in our mesh, then the FFT requires $3N_d^3 \log_2 N_d$ operations each time-step, not including other

operations such as differencing the potential to obtain the force on the mesh, assigning particles, computing densities on all the grid cells, and updating particle positions and velocities. In contrast, if we let $N_{\rm los} = N_d$ (which is enough for accuracy), then there are only N_d^2 rays to trace, and for each ray we have $\leq 10^2$ operations. We have checked the simulation log file and found that there is no observable difference in the times used by each step before and after the ray-tracing part of RATANA has been triggered.

Analytic formulae are often more useful than purely numerical results in tracing the physical contents of a theory. For example, in equations (12) and (13), it is easy to check which terms contribute the most to the final result: obviously, in the small-angle limit, that is, θ, $\psi \ll 1$, terms involving d_3 and d_4, and a large part of d_2 could be neglected because $\sin\theta$, $\sin\psi \ll 1$; also at least one of a, b and c vanishes and $abc = 0$ for all grid cells, further simplifying d_1 and d_2; furthermore, terms in equation (12) with the coefficient $1/\chi_{\rm s}$ contribute little because $\chi_{\rm s} \gg \tilde{\chi}_{\rm u,l}$. Such observations can be helpful in determining which terms have most important effects in certain regimes.

Note that the dependence on $\chi_{\rm s}$ [cf. equation (12)] could be taken out of the analytical integration, meaning that the algorithm can be straightforwardly generalized to include multiple source redshifts with very little extra computational effort (mainly in determining where to start the integration for a given source redshift). The algorithm can also be easily generalized to compute the flexion, which depends on higher order derivatives of the lensing potential, and is expected to give more accurate results than the multiple-lens-plane approximation.

The algorithm has many other flexibilities too. As an example, the analytic integration of the projected density and potential fields along the l.o.s. can be performed on an adaptive rather than a regular grid with careful programming, which means that higher resolution can be achieved in high density regions, as in the adaptive PM simulations. Also, the analytic integration can be easily generalized to other algorithms to compute the 3D shear field (Couchman, Barber & Thomas 1999).

We also give prescriptions to include second-order corrections to the results, such as the lens–lens coupling and corrections to the Born approximation, in Appendix A. It is interesting to note that, by running the N-body simulations backwards in time, we can still compute the convergence and shear fields on-the-fly even if the light rays are not straight.

To conclude, the algorithm described here is efficient and therefore suitable for the future ray-tracing simulations using very large N-body simulations. It will be interesting to apply it to study the higher order statistics of the shear field and the lensing excursion angles, and these will be left for future work.

ACKNOWLEDGMENTS

The work described here has been performed under the HPC-EUROPA project, with the support of the European Community Research Infrastructure Action under the FP8 'Structuring the European Research Area' programme. The N-body simulations are performed on the SARA supercomputer in the Netherlands and the post-precessing of data is performed on COSMOS, the UK National Cosmology Supercomputer. The Smith et al. (2003) fit results for the matter and convergence power spectra are computed using the CAMB code. The non-linear matter power spectrum is measured using POWMES (Colombi et al. 2009). We would like to thank Henk Hoekstra for being the local host for the HPC-EUROPA project, and David Bacon, Henk Hoekstra, Kazuya Koyama and Peter Schneider for helpful comments. BL is supported by Queens' College at University of Cambridge and STFC rolling grant in DAMTP, LJK is supported by the Royal Society and G-BZ is supported by STFC grant ST/H002774/1.

REFERENCES

Bacon D. J., Refregier A. R., Ellis R. S., 2000, MNRAS, 318, 625
Bartelmann M., Schneider P., 2001, Phys. Rep., 340, 291
Bertschinger E., 1995, preprint (astro-ph/9506070)
Bridle S., King L. J., 2007, New J. Phys., 9, 444
Castro P. G., Heavens A. F., Kitching T. D., 2005, Phys. Rev. D, 72, 023516
Chongchitnan S., King L. J., 2010, MNRAS, 407, 1989
Colombi S., Jaffe A., Novikov D., Pichon C., 2009, MNRAS, 393, 511
Couchman H. P. M., Barber A. J., Thomas P. A., 1999, MNRAS, 310, 453
Crittenden R. G., Pogosian L., Zhao G.-B., 2009, J. Cosmol. Astropart. Phys., 0912, 025
Fu L. et al., 2008, A&A, 479, 9
Guillet T., Teyssier R., Colombi S., 2010, MNRAS, 405, 525
Heinamaki P., Suhhonenko I., Saar E., Einasto M., Einasto J., Virtanen H., 2005, preprint (astro-ph/0507197)
Hilbert S., Hartlap J., White S. D. M., Schneider P., 2009, A&A, 499, 31
Hockney R. W., Eastwood J. W., 1981, Computer Simulation Using Particles. McGraw-Hill, New York
Hoekstra H., Mellier Y., van Waerbeke L., Semboloni E., Fu L., Hudson M. J., Parker L. C., Tereno I., Benabed K., 2006, ApJ, 647, 116
Jain B., Seljak U., 1997, ApJ, 484, 560
Jain B., Seljak U., White S. D. M., 2000, ApJ, 530, 547
Kaiser N., 1992, ApJ, 388, 272
Kaiser N., Wilson G., Luppino G., 2000, preprint (astro-ph/0003338)
Knebe A., Green A., Binney J., 2001, MNRAS, 325, 845
Lewis A., Challinor A., 2009, Phys. Rep., 429, 1
Li H., Liu J., Xia J.-Q., Sun L., Fan Z.-H., Tao C., Tilquin A., Zhang X., 2009, Phys. Lett. B, 675, 164
Maoli R., van Waerbeke L., Mellier Y., Schneider P., Jain B., Bernardeau F., Erben T., Fort B., 2001, A&A, 368, 766
Martel H., Premadi P., Matzner R. A., 2000, ApJ, 537, 28
Mellier Y., 1999, A&A, 37, 127
Pielorz J., Rodiger J., Tereno I., Schneider P., 2010, A&A, 514, A79
Rudd D. H., Zentner A. R., Kravtsov A. V., 2008, ApJ, 672, 19
Sato M., Hamana T., Takahashi R., Takada M., Yoshida N., Matsubara T., Sugiyama N., 2009, ApJ, 701, 945
Schmidt F., 2008, Phys. Rev. D, 78, 043002
Schrabback T. et al., 2010, A&A, 516, 63
Smith R. E., Peacock J. A., Jenkins A. et al., 2003, MNRAS, 341, 1311
Song Y.-S., Zhao G.-B., Bacon D., Koyama K., Nichol R. C., Pogosian L., 2010, preprint (arXiv:1011.2106)
Tereno I., Schimd C., Uzan J.-P., Kilbinger M., Vincent F. H., Fu L., 2009, A&A, 500, 657
Vale C., White M., 2003, ApJ, 592, 699
van Waerbeke L. et al., 2000, A&A, 358, 30
White M., 2004, Astropart. Phys., 22, 211
White M., Hu W., 2000, ApJ, 537, 1
Wittman D. M., Tyson J. A., Kirkman D., Dell'Antonio I., Bernstein G., 2000, Nat, 405, 143
Zhan H., Knox L., 2004, ApJ, 616, L75
Zhao G.-B., Pogosian L., Silvestri A., Zylberberg J., 2009, Phys. Rev. D, 79, 083513
Zhao G. -B., Pogosian L., Silvestri A., Zylberberg J., 2010a, Phys. Rev. Lett., 103, 241301
Zhao G.-B., Giannantonio T., Pogosian L., Silvestri A., Bacon D. J., Koyama K., Nichol R. C., Song Y. S., 2010b, Phys. Rev. D, 81, 103510
Zhao G.-B., Zhan H., Wang L., Fan Z., Zhang X., 2010c, preprint (arXiv:1005.3810)

APPENDIX A: BEYOND THE FIRST-ORDER APPROXIMATIONS

In the attempt to trace light rays on-the-fly, we set up a bundle of l.o.s. before the N-body simulation starts. However, because we do not know the exact paths of those light rays which finally end up at the observer, we have to assume that they are straight lines even though they are not in reality. This so-called Born approximation is generally quite good in the weak-lensing regime, but can lead to non-negligible errors on small scales (Hilbert et al. 2009). Furthermore, in the above treatment, we have also neglected the lens–lens coupling, which accounts for the fact that the lenses themselves (the large-scale structure) are distorted by the lower redshift matter distribution.

Hilbert et al. (2009) take account of the lens–lens coupling and corrections to the Born approximation using the multiple-lens-plane approximation. In such an approach, the light rays get deflected and their paths are recomputed when and only when they pass by a discrete lens plane.

Since our algorithm goes beyond the discrete lens-plane approximation and is able to trace the detailed matter distribution, we want to generalize it to include those corrections as well. In this appendix, we shall derive an analytical formula for the distortion matrix with the lens–lens coupling taken into account and describe how the corrections to the Born approximation can be incorporated as well.

Obviously, to go beyond the Born approximation, the light rays are no longer straight and thus the l.o.s cannot be set up before the N-body simulation has finished. Instead, we have to start from the observer today and go backwards in time to compute the distortion matrix equation (3). We shall discuss below how this could be realized in practice, but at this moment let us simply assume that we can go backwards in time and know the value of the lensing potential Φ and its derivatives along the l.o.s.

A1 Corrections to the Born approximation

The corrections to the Born approximation are easy to implement. According to equation (2), the total deflection of a light ray is the sum of the deflections by the matter in each grid that the ray passes on its way towards the lensing source. Suppose $\boldsymbol{\xi}^{(n)}$ denotes the value of $\boldsymbol{\xi}$ after the light ray crosses the nth grid on its way (n increases with the distance from the observer, $n = 1$ corresponds to the grid which the observer is in, and $\boldsymbol{\xi}^{(0)} = \boldsymbol{\xi}_0$), then

$$\boldsymbol{\xi}^{(n)} = \boldsymbol{\xi}^{(n-1)} - 2\int_{\chi_{\rm l}^{(n)}}^{\chi_{\rm u}^{(n)}} \frac{\chi_{\rm s} - \chi}{\chi\,\chi_{\rm s}} \nabla_{\boldsymbol{\xi}} \Phi \,{\rm d}\chi, \tag{A1}$$

where $\chi_{\rm u}^{(n)} \equiv \min\{\chi_{\rm u}^{\rm ts}, \chi_B^{(n)}\}$ and $\chi_{\rm l}^{(n)} \equiv \max\{\chi_{\rm l}^{\rm ts}, \chi_A^{(n)}\}$, in which $\chi_{\rm u}^{\rm ts} > \chi_{\rm l}^{\rm ts}$ are the χ values at the two ends of the current time-step and $\chi_B^{(n)} > \chi_A^{(n)}$ are the χ-values of the two intersections between the light ray and the nth grid. Using the expressions given in Section 2.2.2, it is easy to write $\nabla_{\xi_1}\Phi$ and $\nabla_{\xi_2}\Phi$ in terms of polynomials of χ. Then, the above integral can be performed analytically as before. In this way, each time the light ray crosses a grid, we update its orientation according to the above equation, and thus the corrections to the Born approximation can be incorporated.

Note that in this approach the light rays are deflected many more times than in the multiple-lens-plane approximation and the detailed matter distribution has been fully taken account of.

A2 Lens–lens coupling

As mentioned earlier, the lens–lens coupling has been neglected in the above treatment because in equation (3) we have used the approximation $\nabla_{\chi 0i} \approx \nabla_{\chi_i}$. Let us now have a look at what happens when this approximation is dropped.

Note that in the expression

$$A_{ij} \equiv -2\int_0^{\chi_{\rm s}} \frac{g(\chi, \chi_{\rm s})}{\chi^2} \nabla_{\xi_{0i}} \nabla_{\xi_j} \Phi(\chi, \boldsymbol{\xi})\,{\rm d}\chi + \delta_{ij}, \tag{A2}$$

the argument of Φ is $\boldsymbol{\xi}$ while one of the derivatives is with respect to $\boldsymbol{\xi}_0$. We can utilize the chain rule to write $\nabla_{\xi_{0i}} = (\nabla_{\xi_{0i}}\xi_j)\nabla_{\xi_j} = A_{ij}\nabla_{\xi_j}$ where we have used the definition of A_{ij} given in Section 2.1. Then, the above equation becomes

$$A^i_j(\chi_{\rm s}, \boldsymbol{\xi}) \equiv \delta^i_j - 2\int_0^{\chi_{\rm s}} g(\chi, \chi_{\rm s}) \nabla^i \nabla_k \Phi(\chi, \boldsymbol{\xi})\, A^k_j(\chi, \boldsymbol{\xi})\,{\rm d}\chi, \tag{A3}$$

where for simplicity we have used $\nabla_i = \nabla_{\xi_i}$. With the A^k_j term in the integrand, equation (A3) now includes the lens–lens coupling and will be our starting point here.

Again, let us consider the integral in equation (A3) after the light ray crosses the nth grid on its way towards the lensing source. The discrete version of equation (A3) is

$${}^{(n)}A^i_j = {}^{(n-1)}A^i_j - 2\int_{\chi_{\rm l}^{(n)}}^{\chi_{\rm u}^{(n)}} \frac{\chi(\chi_{\rm s} - \chi)}{\chi_{\rm s}}\, {}^{(n)}A^k_j \nabla^i \nabla_k \Phi \,{\rm d}\chi, \tag{A4}$$

where ${}^{(n)}A^i_j$ is the value of A^i_j after the light ray has crossed the nth grid and ${}^{(0)}A^i_j = \delta^i_j$ as is easy to see. This formula has three advantages as compared to the multiple-lens-plane approximation:

(i) As before, the light rays between $z = 0$ and $z_{\rm s}$ are divided into many more segments, and the fine structure of the matter distribution is included naturally, without squeezing the matter and using impulse approximations.

(ii) As will be shown below, the integration can be evaluated analytically rather than numerically.

(iii) Note that we can use ${}^{(n)}A^i_j$ rather than ${}^{(n-1)}A^i_j$ in the integrand, which will give more accurate results, because using ${}^{(n-1)}A^i_j$ would mean that the contribution to the lens–lens coupling from the matter in the nth grid is ignored. In the multiple-lens-plane approximation, which typically uses $20 \sim 30$ lens planes, the nth plane could contain a significant amount of matter and neglecting its contribution could make the results less accurate.

Equation (A4) is exact, but we only want the result to second order in $\nabla\nabla\Phi$. Therefore, we can iterate once and write an approximate solution as

$$
\begin{aligned}
{}^{(n)}A^i_j \approx {}^{(n-1)}A^i_j - 2^{(n-1)}A^k_j \int_{\chi_{\rm l}^{(n)}}^{\chi_{\rm u}^{(n)}} {\rm d}\chi \, \frac{\chi\,(\chi_{\rm s}-\chi)}{\chi_{\rm s}} \nabla^i\nabla_k\Phi \\
+ 4\int_{\chi_{\rm l}^{(n)}}^{\chi_{\rm u}^{(n)}} {\rm d}\chi \, \frac{\chi\,(\chi_{\rm s}-\chi)}{\chi_{\rm s}} \nabla^i\nabla_k\Phi(\chi,\boldsymbol{\xi}) \int_{\chi_{\rm l}^{(n)}}^{\chi} {\rm d}\chi' \frac{\chi'(\chi-\chi')}{\chi} \nabla^k\nabla_j\Phi(\chi',\boldsymbol{\xi}).
\end{aligned}
\tag{A5}
$$

Following the approach taken in Section 2.2.1, we can write

$$
\nabla^i\nabla_j\Phi(\chi,\boldsymbol{\xi}) = \sum_{N=1}^{4} (\mathbf{K}_N)^i_j \left[\chi - \chi_A^{(n)}\right]^{N-1},
\tag{A6}
$$

where $\chi_A^{(n)}$ is defined in equation (A1) and $\mathbf{K}_N$ ($N \in \{1,2,3,4\}$) is a 2×2 matrix whose ij-component depends on the orientation of the l.o.s. segment inside the nth grid (where it is taken to be straight) and the values of $\nabla^i\nabla_j\Phi$ at the vertices of the nth grid. Note, however, that $\mathbf{K}_N$ is independent of χ. The expressions are similar to equation (13) defined in Section 2.2.1 and we shall not write them explicitly here.

Substituting equation (A6) into equation (A5), we find

$$
{}^{(n)}A^i_j = {}^{(n-1)}A^i_j - 2\sum_{N=1}^{4} I_1(N)\,(\mathbf{K}_N)^i_k \; {}^{(n-1)}A^k_j + 4\sum_{N=1}^{4}\sum_{M=1}^{4} I_2(N,M)\,(\mathbf{K}_N)^i_k\,(\mathbf{K}_M)^k_j
\tag{A7}
$$

in which we have written [again, by defining $\tilde{\chi} \equiv \chi - \chi_A^{(n)}$, $\tilde{\chi}' \equiv \chi' - \chi_A^{(n)}$ and $\tilde{\chi}^{(n)}_{\rm u,l} \equiv \chi^{(n)}_{\rm u,l} - \chi_A^{(n)}$]

$$
\begin{aligned}
I_1(N) &\equiv \int_{\chi_{\rm l}^{(n)}}^{\chi_{\rm u}^{(n)}} {\rm d}\chi \, \frac{\chi\,(\chi_{\rm s}-\chi)}{\chi_{\rm s}} \left[\chi - \chi_A^{(n)}\right]^{N-1} \\
&= \frac{1}{N}\chi_A^{(n)}\left[1-\frac{\chi_A^{(n)}}{\chi_{\rm s}}\right]\left[\left(\tilde{\chi}_{\rm u}^{(n)}\right)^N - \left(\tilde{\chi}_{\rm l}^{(n)}\right)^N\right] + \frac{1}{N+1}\left(1-\frac{2\chi_A^{(n)}}{\chi_{\rm s}}\right)\left[\left(\tilde{\chi}_{\rm u}^{(n)}\right)^{N+1} - \left(\tilde{\chi}_{\rm l}^{(n)}\right)^{N+1}\right] \\
&\quad - \frac{1}{N+2}\frac{1}{\chi_{\rm s}}\left[\left(\tilde{\chi}_{\rm u}^{(n)}\right)^{N+2} - \left(\tilde{\chi}_{\rm l}^{(n)}\right)^{N+2}\right]
\end{aligned}
\tag{A8}
$$

and

$$
\begin{aligned}
I_2(N,M) &\equiv \int_{\chi_{\rm l}^{(n)}}^{\chi_{\rm u}^{(n)}} {\rm d}\chi \, \frac{\chi_{\rm s}-\chi}{\chi_{\rm s}} \left[\chi - \chi_A^{(n)}\right]^{N-1} \int_{\chi_{\rm l}^{(n)}}^{\chi} {\rm d}\chi' \chi'(\chi-\chi')\left[\chi' - \chi_A^{(n)}\right]^{M-1} \\
&= -\frac{\left(\tilde{\chi}_{\rm u}^{(n)}\right)^{M+N+3} - \left(\tilde{\chi}_{\rm l}^{(n)}\right)^{M+N+3}}{(M+1)(M+2)(M+N+3)}\frac{1}{\chi_{\rm s}} + \frac{\left(\tilde{\chi}_{\rm u}^{(n)}\right)^{M+N+2} - \left(\tilde{\chi}_{\rm l}^{(n)}\right)^{M+N+2}}{(M+1)(M+2)(M+N+2)}\left[1-\frac{\chi_A^{(n)}}{\chi_{\rm s}}\right] \\
&\quad - \frac{\left(\tilde{\chi}_{\rm u}^{(n)}\right)^{M+N+2} - \left(\tilde{\chi}_{\rm l}^{(n)}\right)^{M+N+2}}{M(M+1)(M+N+2)}\frac{\chi_A^{(n)}}{\chi_{\rm s}} + \frac{\left(\tilde{\chi}_{\rm u}^{(n)}\right)^{M+N+1} - \left(\tilde{\chi}_{\rm l}^{(n)}\right)^{M+N+1}}{M(M+1)(M+N+1)}\chi_A^{(n)}\left[1-\frac{\chi_A^{(n)}}{\chi_{\rm s}}\right] \\
&\quad + \left[\frac{1}{M}\frac{\chi_A^{(n)}}{\chi_{\rm s}} + \frac{1}{M+1}\frac{\chi_{\rm l}^{(n)}}{\chi_{\rm s}}\right]\left(\chi_{\rm l}^{(n)}\right)^M \frac{1}{N+2}\left[\left(\tilde{\chi}_{\rm u}^{(n)}\right)^{N+2} - \left(\tilde{\chi}_{\rm l}^{(n)}\right)^{N+2}\right] \\
&\quad - \left(1-\frac{\chi_A^{(n)}}{\chi_{\rm s}}\right)\left[\frac{1}{M}\chi_A^{(n)} + \frac{1}{M+1}\chi_{\rm l}^{(n)}\right]\left(\chi_{\rm l}^{(n)}\right)^M \frac{1}{N+1}\left[\left(\tilde{\chi}_{\rm u}^{(n)}\right)^{N+1} - \left(\tilde{\chi}_{\rm l}^{(n)}\right)^{N+1}\right] \\
&\quad - \left[\frac{1}{M+1}\frac{\chi_A^{(n)}}{\chi_{\rm s}} + \frac{1}{M+2}\frac{\chi_{\rm l}^{(n)}}{\chi_{\rm s}}\right]\left(\chi_{\rm l}^{(n)}\right)^{M+1} \frac{1}{N+1}\left[\left(\tilde{\chi}_{\rm u}^{(n)}\right)^{N+1} - \left(\tilde{\chi}_{\rm l}^{(n)}\right)^{N+1}\right] \\
&\quad + \left[1-\frac{\chi_A^{(n)}}{\chi_{\rm s}}\right]\left[\frac{1}{M+1}\chi_A^{(n)} + \frac{1}{M+2}\chi_{\rm l}^{(n)}\right]\left(\chi_{\rm l}^{(n)}\right)^{M+1}\frac{1}{N}\left[\left(\tilde{\chi}_{\rm u}^{(n)}\right)^N - \left(\tilde{\chi}_{\rm l}^{(n)}\right)^N\right].
\end{aligned}
\tag{A9}
$$

The above expressions look rather heavy; however, they are analytic and as a result are very easy to implement in the ray-tracing simulation codes, by writing functions that take M, N, $\chi_A^{(n)}$, $\chi_{\rm u}^{(n)}$ and $\chi_{\rm l}^{(n)}$ as parameters and return I_1 and I_2 as outputs. Furthermore, since the grid size ($<0.2\,h^{-1}$ Mpc) in the N-body simulations is small enough compared with the typical interplane distances in the multiple-lens-plane approximations ($10 \sim 100\,h^{-1}$ Mpc), we can drop the $I_2(N,M)$ terms to a very good approximation, which will greatly simplify the results.

Note that the distortion matrix A^i_j computed in this way is not symmetric because of the matrix multiplications. However, using equation (5), it is straightforward to compute $\gamma_2 = -(A^1{}_2 + A^2_1)/2$. In addition, we could also calculate the rotation ω as $\omega = (A^2_1 - A^1_2)/2$.

A3 Going back in time

As mentioned above, to include the actual deflections of the light rays which end up at the observer, we have to start from the observer and go backwards in time until encountering the source. This obviously can only be done after the N-body simulation has finished.

One way to go backwards is to record the information about the gravitational potential Φ and its derivatives in a light-cone during the simulation, and then post-process the light-cone data. This means that a large amount of dump data have to be stored.

Alternatively, one can think of running the N-body simulation 'backwards'. To be more explicit, the simulation is first run in the forward direction from a high redshift until today and we obtain the particle positions and velocities at present; then, we reverse the directions of the gravitational force and the particle velocities, and evolve the system back until z_s using the same time-stepping scheme as in the forward simulation. In this way, the actual light rays and distortion matrix could be built up on-the-fly, and there is no need to store a lot of dump data.

However, we must add a warning note here. There are currently no quantitative studies about reversing N-body simulations which the authors are aware of and there is a possible danger that numerical noise would prevent one from doing this accurately. However, for our simulations, we do not really need to go back to very high redshift (at most $z_s \sim$ a few). Also, if we have stored a sufficient number of intermediate snapshots of the N-body history, then we can inverse the systems for short intervals of time with an algorithm similar to linear or quadratic interpolation. We will test these in a future work.

This paper has been typeset from a TEX/LATEX file prepared by the author.