A new ray tracing method in phase space using alpha-shapes

Citation for published version (APA):

Document status and date:
Published: 01/12/2015

Document Version:
Publisher’s PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:
• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher’s website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the “Taverne” license above, please follow below link for the End User Agreement:
www.tue.nl/taverne

Take down policy
If you believe that this document breaches copyright please contact us at:
openaccess@tue.nl
providing details and we will investigate your claim.
A new ray tracing method in phase space using α-shapes

by

C. Filosa, J.H.M. ten Thije Boonkkamp, W.L. IJzerman
A new ray tracing method in phase space using α-shapes

C. Filosa¹, J.H.M. Ten Thije Boonkkamp¹, and W.L. IJzerman¹,²

¹CASA, Eindhoven University of Technology, PO Box 513 5600 MB Eindhoven, The Netherlands (c.filosa@tue.nl, j.h.m.tenthijeboonkkamp@tue.nl)
²Philips Lighting, High Tech Campus 44, Eindhoven 5656 AE, The Netherlands (wilbert.ijzerman@philips.com).

Compiled December 18, 2015

Ray tracing is a forward method to calculate the photometric variables at the target of an optical system. In this paper a new ray tracing technique is presented to improve the accuracy and to reduce the computational time of the classical ray tracing approach. The method is based on the phase space representation of light, and it is applied to a two-dimensional TIR-collimator. The strength of the method lies in tracing fewer rays through the system. To do this, a procedure that disregards rays in smooth regions in phase space, where the luminance is continuous, is implemented and, only the rays close to discontinuities are traced. The phase space of the source and the target of the optical system is divided into regions formed by rays that follow the same path. To compute the photometric variables at the target the boundaries of these regions are found using the α-shapes method and a new technique based on a triangulation refinement of the source phase space. The efficiency of the method is demonstrated by numerical simulations that compare the new method with Monte Carlo ray tracing. The results show that the phase space approach is faster and more accurate than the already existing ray tracing method and converges as one over the number of rays. © 2015 Optical Society of America

OCIS codes: Ray tracing, Monte Carlo ray tracing, phase space, two-dimensional optical systems, TIR-collimator.

1. INTRODUCTION

Ray tracing is one of the most important tools for optical designers. Currently, an increasing interest in Light Emitting Diode (LED) lighting is arising. LED lighting has two main advantages: first, its high energy efficacy (from 80 lm/W to 150 lm/W, [1, 2]); second, its very long lifetime upto 50,000 hours. For these reasons, the use of LED lamps is rapidly growing. For a light lamp based on LED an optical system is required that consists of different optical components, such as lenses and reflectors. The goal in illumination optics is to obtain the desired light output at the target after the propagation of the light beam through the system.

To compute the photometric variables at the target of the optical system, the ray tracing procedure can be used. Ray tracing is a forward method which provides the light output given a light source and an optical system, [3]. There are many ways to implement the ray tracing process. The most commonly used is Monte Carlo (MC) ray tracing which is based on a probabilistic interpretation of the ray distribution at the target of the optical system [4-6]. In this method many rays are traced randomly from the source, and their distribution at the target is estimated to compute the photometric variables of the output light. Although the MC procedure constitutes a robust method, it remains a slow and costly numerical procedure especially when high precision is required. Moreover, as the complexity of optical designs increases, more calculation time is needed to obtain accurate results. Indeed, the error is proportional to the reciprocal value of the square root of the number of rays traced.

In this paper a new approach to ray tracing, which employs the phase space (PS) representation of the optical system, is proposed. In two dimensions, the phase space of an optical surface is defined as a two-dimensional space and every ray is described by its position coordinate \( x \) on that surface and its angular coordinate \( p = n_s \sin(t) \), where \( t \) is the angle that the ray forms with respect to the optical axis and \( n_s \) is the refractive index of the medium in which the source is located. Our approach takes into account only the phase space of the source and the target of the optical system. Ray tracing is a general technique applicable to any optical system; we develop a numerical algorithm to simulate ray tracing through the Total Internal Reflection (TIR) collimator in two dimensions. In addition, the algorithm computes the sequence of the optical surfaces that each ray hits when it propagates inside the system. We use this information to describe the ray distribution at source and target phase space, which can be partitioned in different regions \( R_{a}\) and \( R_{b}\) according to the path II followed by the rays. Ries and Rabl (1994) showed that the area of these regions is conserved:
all rays that are neighbors at the source phase space remain close to each other at the target [7]. Assuming constant brightness, we only need to compute the boundaries of the ray sets to obtain the photometric variables at the target.

For the sake of simplicity, we calculate the boundaries at the source phase space to determine the target boundaries. To this purpose, a nonuniform triangulation of the source phase space is constructed in such a way that new triangles are added to the triangulation only where boundaries occur. The triangulation is based on the fact that a boundary is expected between two rays that follow different paths. Two different criteria to stop the refinement procedure are considered. For the first criterion the rays traced must follow the same path; for the second, they have to be located close each other in phase space. The details of this procedure are explained in Section 2C. To compute the boundaries of the point clouds formed by the rays traced, the α-shapes procedure [8] is used. Next, the output intensity is computed through an integration of the luminance along the direction $\eta = n_t \sin(\theta)$, where $\theta$ is the angle that the ray forms with respect to the optical axis and $n_t$ is the refractive index of the medium where the target is located. Then, we compare our technique with MC ray tracing, and we show that we obtain a more accurate intensity distribution even when far fewer rays are traced. The error between the approximate intensity and an intensity taken as reference is computed both for Monte Carlo and phase space ray tracing. An error plot shows that the MC error is inversely proportional to the square root of the number of rays, whereas the PS error has a behavior proportional to the inverse of the number of rays traced. Hence, ray tracing in PS is more accurate and less computationally expensive than the already existing MC ray tracing.

Finally, we note that the behavior of the error for the PS method also depends on the shape of the regions $R_{1,1}$ and, thus, on the optical system. We considered two TIR-collimators with different lenses, one flatter than the other, and with different distances between the target and the top of the collimator. The PS and MC methods are applied to both collimators. The boundaries of the regions in target phase space are found using two different methods: the already mentioned α-shapes procedure and, a method that exploits the triangulation refinement of source phase space, the details are presented in Section 3. An error plot for three procedures (MC ray tracing and PS ray tracing with two different methods for the boundaries calculation) shows that PS ray tracing is still faster than MC ray tracing. Comparing the two examples of the TIR-collimator, we note that the shape of the regions formed by the rays that hit the target does not have the spikes anymore and a more accurate approximation of the boundaries is obtained for a small number of rays.

This paper is organized as follows. In Section 2, the phase space representation and the geometry of the optical system are introduced. In the same section, the techniques used to compute the boundaries of the ray sets: the triangulation refinement of the source phase space and the α-shapes method are described, and the PS ray tracing procedure for the TIR-collimator is developed. Numerical results are shown in Section 3; a comparison between PS and MC ray tracing for two different shapes of the TIR-collimator is presented in the same section. Conclusions and remarks are given in Section 4.

2. THE PHASE SPACE METHOD FOR A TIR-COLLIMATOR

Ray tracing uses single rays to describe the propagation of the light through an optical system. The influence of diffraction for the transport of a ray is neglected and a geometrical modeling of an optical system is considered. In this work only the Snell’s law is taken into account and Fresnel reflection is not considered. Generally, the method can be implemented for two or more dimensions and for any optical system. We focus on two examples of a TIR-collimator in two-dimensions. First, we introduce the classical ray tracing method and specify the geometry of the optical system used in our model, then, we describe the phase space procedure.

A. Classical ray tracing and geometry of the optical system

The ray tracing process consists of tracing each ray, which is considered to be a broken line, through an optical system. In the following, the procedure in a two-dimensional optical system is considered; hence, given a Cartesian coordinate system $(x, z)$, a two-dimensional optical system is defined. The propagation of the light from the source $S$, which emits monochromatic light, to the target $T$, which receives light from $S$, is computed to obtain the photometric variables at the target. In this work, we compute the output intensity for the (TIR)-collimator; the profile is depicted in Figure 1.

![Fig. 1. Shape of the TIR-collimator. Each surface of the system is labeled with a number. The source $S = [−2, 2]$ (surface number 1) is located at a height $z_s = 0.3$ from the $x$-axis. The target $T = [−9.7, 9.7]$ (surface 12) is parallel to the source and is located at a height $z_t = 8.2$. The shape of the collimator is shown as a blue line. Three detectors depicted with green lines (surfaces 13, 14, and 15) are located at the left, the right, and the bottom of the optical system. $n_1 = 1$ is the refractive index of the medium (air) where the source and the target are located, and $n_2 = 1.5$ the refractive index of the medium (glass) inside the optical system. The sagitta of the lens is approximately 1.17.](image)
of refraction \( n_2 = 1.5 \) (e.g., glass). The collimator is surrounded by two vertical (surfaces 13 and 15) and two horizontal lines (12 and 14) that receive the light exiting from the optical system; among these the horizontal one at the top is assumed to be the light target, and it is located at a small distance from the top (surface 7).

From now on, the coordinates \((x_i, z_i)_{i=1,\ldots,15}\) denote the intersection of the rays with the surface \(i\) and \(s_i = (-\sin t_i, \cos t_i)\) is the direction vector of the rays from the surface \(i\), where \(t_i \in (-\pi/2, \pi/2)\) is the angle that the ray forms with respect to the optical axis and is measured counterclockwise. Therefore, a ray segment between \((x_i, z_i)\) and \((x_{i+1}, z_{i+1})\) is parameterized by:

\[
r(s) = \begin{pmatrix} x_i - s \sin(t_i) \\ z_i + s \cos(t_i) \end{pmatrix}, \quad s \geq 0, \tag{1}
\]

where \(s\) denotes the arc-length.

A Lambertian optical source is considered; hence, the intensity over an interval \(J = [-\alpha, \alpha]\) emitted in the direction \(t\) is given by:

\[
I(t) = I_0 \cos(t), \tag{2}
\]

where \(I_0 = 2aL, L\) is the luminance, and \(t\) is the angle that the ray forms with respect to the optical axis, and it is measured counterclockwise. As a result, in the case where \(L = 1, I_0\) coincides with the source length; from Equation (2), the intensity at the source is deduced.

To compute the target intensity, we need to know how the optical system changes the direction of the rays in such a way that the source and the target intensities are related to each other. To do this, we employ the ray-tracing technique which can be summarized as follows: first, a ray from the source with initial position given by the coordinates \((x_1, z_1)\) and initial angle \(t_1\) with respect to the \(z\)-axis is traced, and the ray parametrization is implemented according to Equation (1). Second, the intersection point \((x_i, z_i)\) between the ray and the surface, \(i\), that it hits is computed. Third, the normal to the surface hit at the point \((x_i, z_i)\) is calculated to compute the change of direction of the ray. For the last step, the laws of reflection and refraction are implemented. The direction of the refractive ray is given by:

\[
t = n_{1,2} i + \left[\frac{1 - n_{1,2}^2 + n_{1,2}^2 (n, i)^2 - n_{1,2} (n, i)}{n_{1,2}^2 (n, i)^2} \right] n, \tag{3}
\]

where \(n_{1,2} = n_1/n_2\) with \(n_1\) and \(n_2\) the refraction indexes of air and of glass, respectively. The unit vectors \(i\) and \(t\) describe the directions of the incident and refracted ray, respectively; \(n\) is the surface normal; it is also a unit vector and it is directed towards the interior of the optical system. Note that the positive sign before the square root is due to the convention of taking the inward direction of the normal \(n\). In the case where \(n_1 = -n_2\), Equation (3) can be rewritten as the law of reflection:

\[
t = i - 2(i, n)n. \tag{4}
\]

Equation (4) is used when the total internal refraction condition holds, that is given by the following inequality:

\[
1 - n_{1,2}^2 + (n, i)^2 < 0. \tag{5}
\]

For the TIR-collimator the previous condition occurs for the curved lines (surfaces 6 and 8 in Figure 1). Finally, the new parametrization of the ray is described by:

\[
r(s) = \begin{pmatrix} x_i + st_z \\ z_i + st_z \end{pmatrix}, \tag{6}
\]

where \(t_x\) and \(t_z\) are the \(x\) and \(z\)-components of the new ray direction and are calculated from Equation (3) or (4). The points \((x_i, z_i)\) and the new direction \(t\) are calculated until the ray hits the target and the previous procedure is repeated for each traced ray.

To obtain a reasonable approximation of the target intensity, a large number of rays has to be traced; the more rays are traced, the more accurate the target intensity is. Moreover, for the TIR-collimator shown in Figure 1 we do not have an explicit equation to describe the lens and the reflectors, but only a set of data points located on the surfaces is known. Therefore, it is required to use spline interpolation to obtain a good approximation of the curved surfaces. In addition, to calculate the intersection points between the rays and these surfaces, the Newton-Raphson procedure (with a tolerance equal to \(10^{-14}\)) is employed. Due to all these reasons, the ray-tracing method is a very slow procedure.

The most common ray tracing method used in optics is MC ray tracing [4] in which the rays are emitted from a random location and at a random angle and, they are traced through the system until they reach the target receiver. To calculate the output intensity, the target screen is divided into bins and the frequency of the rays that arrive at each bin is considered. The intensity restricted to a certain bin is obtained dividing the number of rays that fall into that bin by the total number of rays traced. Although MC ray tracing is highly robust and does not require difficult calculations, it has two main disadvantages. First, while the position and the angle of the rays that reach the target are known, some information is lost because the flux of a ray is averaged over a bin. Second, some parts of the target are reached by a very small fraction of rays and, consequently, the intensity is unreliable in those parts. As a consequence, a large number of rays needs to be traced to obtain an accurate intensity, making the MC method computational costly.

We provide a new method that employs the phase space representation of the optical system avoiding to trace rays where the luminance does not present any discontinuities. Phase space ray tracing is explained in the next section.

**B. Phase space representation of the source and the target**

In phase space each ray is described by its intersection point with the surface it hits and the sine of the angle it forms with respect to the optical axis multiplied by the refractive index (see [11–13] for details). In the following, the phase space is considered only for the source \(S\) and the target \(T\) and for no other surface of the optical system. The rays in a two-dimensional system correspond to points with coordinates \((x, t)\) and \((q, \eta)\) in \(S\) and \(T\) phase space, respectively. We have indicated the ray positions with \(x\) and \(q\), the angles formed with the normal with \(t\) and \(\theta\), the refractive indexes with \(n_s\) and \(n_t\) for \(S\) and \(T\), respectively and \(\tau = n_s \sin(t)\) and \(\eta = n_t \sin(\theta)\). The rays are represented by a unique point in phase space, both for \(S\) and \(T\). More formally, the optical phase space for the light source is defined as:

\[
\mathcal{P}_s = S \times [-n_s, n_s]. \tag{7}
\]

The target phase space is defined as

\[
\mathcal{P}_t = T \times [-n_t, n_t]. \tag{8}
\]

The map \(\mathcal{M} : \mathcal{P}_s \rightarrow \mathcal{P}_t\) that describes how the optical system changes the rays is defined as:

\[
\mathcal{M}(x, t) = (q, \eta). \tag{9}
\]
For most optical systems, there is no way to find an explicit expression for the map \( M \) defined above. The idea is to apply the edge-ray principle \([7]\) to a given set of rays at the source. The principle states that to map one region from the source to the target phase space it is sufficient to map the boundaries of those regions. Therefore, the boundaries of the source are mapped to the boundaries of the target and the regions where the luminance is different from zero are calculated. The intensity in target phase space is defined as a function of the output luminance:

\[
I_{PS}(\eta) = \int_{\eta} L_{t}(q, \eta) \, dq, \tag{10}
\]

where we indicate the luminance at the target with \( L_{t}(q, \eta) \). Since we use the target phase space to compute the output intensity, it is convenient to define it as a function of \( \sin(\theta) \) instead of \( \theta \). Note that the luminance is positive in the entire \( P_{s} \), but not all parts of \( P_{t} \) receive light emitted by the source. As a result, \( L_{t} \) has jump discontinuities where it changes from zero to positive values. To understand where these discontinuities occur, further information about the rays is required. Because of this, for PS-ray tracing not only the initial positions and the initial angles of the rays are stored, but also the optical surfaces they hit when they propagate through the system. A ray path \( \Pi \) is defined as a collection of the surfaces hit by the ray. Rays that are close to each other at the source and leave the source at close angles follow the same path and hit the target at close positions and under close angles. All the rays that follow the same path, \( (\Pi_{j})_{j=1, \ldots, p} \), with \( p \) the number of paths encountered by the rays, are grouped together into the same subset of phase space, which is indicated, for a given \( j \in \{1, \ldots, p\} \), with \( R_{k,1} \), and \( R_{t,1} \), for the source and the target, respectively. The map \( M \), defined in equation (9), relates the regions \( R_{s,1} \), to the regions \( R_{t,1} \), for every \( j \in \{1, \ldots, p\} \). Again the edge-ray principle guarantees that the boundaries \( \partial R_{s,1} \), and \( \partial R_{t,1} \), are connected by the same map \( M \). Given two different paths \( \Pi_{1} \) and \( \Pi_{2} \), the regions \( R_{s,1} \) and \( R_{t,1} \) do not overlap; they can have at most a common boundary. As a result, the discontinuities of the luminance occur exactly at the boundaries \( \partial R_{t,1} \). Finally, the luminance at the target satisfies the following relations:

\[
L_{t}(q, \eta) > 0 \quad \text{for} \quad (q, \eta) \in (R_{s,1})_{j=1, \ldots, p},
\]

\[
L_{t}(q, \eta) = 0 \quad \text{otherwise}. \tag{11}
\]

In addition, since the luminance is conserved along a ray \([10]\), it remains constant inside every region \( (R_{s,1})_{j=1, \ldots, p} \). The output intensity is obtained from Equation (10). Therefore, the problem to compute the target intensity can be interpreted as the calculation of the boundaries \( \partial R_{t,1} \). To this end, we define a triangulation on source phase space in such a way that more rays close to the boundaries are traced. To compute the shape of \( (R_{s,1})_{j=1, \ldots, p} \) the \( a \)-shapes method is applied. The details of these two procedures are explained in the next sections.

**C. Triangulation refinement of source phase space**

The regions \( (R_{s,1})_{j=1, \ldots, p} \) can be defined only when some rays are traced. An algorithm is provided in such a way that, given an initial ray set, it selects a new ray set that determines a more accurate location of the boundaries \( \partial R_{s,1} \). Rather than randomly choosing the rays that leave the source, as MC ray tracing does, a triangulation in \( P_{s} \) is defined and a ray from every vertex \((x_{k}, \eta_{k})\) of the triangle is traced. The procedure starts tracing four rays with coordinates \((x_{k}, \eta_{k})_{k=1, \ldots, 4}\) that are located exactly at the corners of \( P_{s} \) and, for each of them, the paths \((\Pi_{j})_{j=1, \ldots, 4}\), are stored. Next, for some \( j \in \{1, \ldots, 4\} \), the grid is divided into two equal triangles joining two opposite vertices. The iteration procedure continues as explained in the following. For each triangle the rays at its corner are traced and then, if the paths corresponding to those rays are different, one or more boundaries \( \partial R_{t,1} \) are expected to cross the triangle. In that case, the middle points \((x_{2}, \eta_{2}) \approx 5.6, 7\) of each side of the triangle are added and three more rays with coordinates \((x_{2}, \eta_{2}) \approx 5.6, 7\) are traced. Each refinement step leads to four new triangles (see Figure 2).

**Fig. 2.** Triangulation refinement: when the rays related to the vertices of the triangles follow a different path a new refinement step is required. Each refinement step leads to four new triangles. The parameters values are \( \epsilon_{x_{\text{max}}} = 2, \epsilon_{x_{\text{min}}} = 1, \epsilon_{\tau_{\text{min}}} = 4 \) and \( \epsilon_{\tau_{\text{max}}} = 2 \).

When all the rays have the same path, it is not necessary to refine the triangles anymore. Note that it can happen that a region formed by rays that follow a path \( \Pi_{j} \) is located completely inside a triangle whose vertices are related to the same path \( \Pi_{i} \) with \( j \neq i \). In that case the algorithm is not able to detect that region, see Figure 3. To avoid this, two parameters \( \epsilon_{x_{\text{max}}} \) and \( \epsilon_{\tau_{\text{max}}} \) are defined for the \( x \)-axis and the \( \tau \)-axis, respectively. When the length of the sides of the triangle are greater than these parameters, a new triangle is defined even if its vertices correspond to the same path. Furthermore, two other parameters \( \epsilon_{x_{\text{max}}} \) and \( \epsilon_{\tau_{\text{max}}} \) are introduced to defined a stopping criterium. The algorithm stops when the length of the sides of the triangles is smaller than \( \epsilon_{x_{\text{max}}} \) and \( \epsilon_{\tau_{\text{max}}} \).

**Fig. 3.** The red line encloses a region of rays that follow the path \( \Pi_{1} \) and is completely located inside a triangle. The algorithm is not able to detect that region and, a further refinement is required. The parameters values are \( \epsilon_{x_{\text{max}}} = 2, \epsilon_{x_{\text{min}}} = 1, \epsilon_{\tau_{\text{min}}} = 4 \) and \( \epsilon_{\tau_{\text{max}}} = 2 \).

The values of the parameters \( \epsilon_{x_{\text{max}}} \), \( \epsilon_{x_{\text{min}}} \), \( \epsilon_{\tau_{\text{min}}} \) and \( \epsilon_{\tau_{\text{max}}} \) determine the number of rays traced. Indeed, on the one hand, \( \epsilon_{x_{\text{max}}} \) and \( \epsilon_{\tau_{\text{max}}} \) can be decreased to obtain more rays close to the boundaries; on the other hand, a large number of rays in the
interior of the regions can be traced decreasing the values of \( \epsilon_{\text{max}} \) and \( \epsilon_{\text{min}} \).

Using the above procedure, rays increasingly closer to the boundaries are traced. For our optical system, the width of the \( x \)-axis in source phase space is two times the wide of the \( \tau \)-axis. Then, our choice is to consider \( \epsilon_{\text{max}} = \frac{1}{2} \epsilon_{\text{min}} \) and \( \epsilon_{\text{min}} = \frac{1}{2} \epsilon_{\text{max}} \), respectively. Figure 4 shows an example of a triangulation refinement of the source phase space with \( \epsilon_{\text{max}} = 0.1 \) and \( \epsilon_{\text{min}} = 1 \). The triangulation refinement provides more triangles close to the boundaries and only few triangles occur inside the regions \( \Pi_{s,j} \).

Finally, a set of rays that leave the optical source \( S \) is generated. The path followed by each ray is computed and then the regions \( \Pi_{s,j} \) and \( \Pi_{t,i} \) are defined for a given path \( \Pi_j \) with \( j \in \{1, \cdots, p\} \). To obtain the boundaries \( \partial \Pi_{s,j} \) in \( \Pi_t \), we applied the \( \alpha \)-shape method to the point cloud given by the rays traced. In the next section we briefly explain how this method works (for more details see [8, 14–16]).

### D. Photometric variables at the target and \( \alpha \)-shapes method

Given a set of points \( Q \), \( \alpha \)-shapes are geometrical objects which give us a good approximation of the boundary of \( Q \). The method starts from a triangulation of the point cloud (a possible choice could be the Delaunay triangulation [17]). For each triangle, the radius of the circumsphere is calculated; if this radius is larger than \( \alpha \), the triangle is removed from the shape. When \( \alpha \to 0 \), the \( \alpha \)-shape is simply the point set \( Q \); when \( \alpha \to \infty \), the \( \alpha \)-shape coincides with the convex hull. The rule of the parameter \( \alpha \) is crucial in this procedure. There are several ways to determine the value of \( \alpha \) [18]; we provide a technique that exploits the conservation of the étendue in phase space [10]. The essence of our approach is as follows.

We consider the étendue for the whole \( \mathcal{P}_t \) which is given by:

\[
E_s = 2n_s \Delta \sin(t_{\text{max}}),
\]

where \( n_s \) is the length of the source and \( t_{\text{max}} \) is the maximum value of the angle that the rays make with the \( z \)-axis. The étendue of a set of rays is defined by the area they occupy in phase space. For the TIR-collimator we considered (Figure 1), the area of \( \mathcal{P}_s \) is equal to 7.92, since \( a = 4 \) and \( \sin(t_{\text{max}}) = 0.99 \). The rays traced are uniformly distributed over \( \mathcal{P}_s \), and they cover it almost entirely. More precisely, given two adjacent paths \( \Pi_i \) and \( \Pi_j \), the regions \( \Pi_{s,j} \) and \( \Pi_{s,i} \) have a common boundary and only small parts of the source do not illuminate the target of the TIR-collimator (see Figure 5). Indeed a small number of rays reach the left and the right reflectors of our optical system (surfaces 13 and 15 in Figure 1), which are not considered to determine the target intensity. As a result, the étendue at the source \( E_s \) can be calculated removing from the area of the entire \( \mathcal{P}_s \) those areas occupied by the regions formed by the rays that hit the left and the right detector (white regions in Figure 1). Therefore, \( E_s \) can be approximate by the following relation:

\[
E_s \approx 7.92 - 2A_T
\]

where \( A_T \) is the approximated area of one of the white regions in Figure 1 and, it is computed approximating those regions with the area of the triangles shown in Fig. 5 with black lines.

On the contrary, large parts of the target phase space \( \mathcal{P}_t \) are not illuminated by any ray and the regions \( \Pi_{t,i} \) are always disconnected. Hence, the total étendue at the target, \( E_t \), is given by the sum of the étendues related to each region \( \Pi_{t,i} \):

\[
E_t = \sum_{j=1}^{p} E(\Pi_{t,i}),
\]

where \( E(\Pi_{t,i}) \) is the contribution to the étendue at the target given by the rays inside the region \( \Pi_{t,i} \). \( E(\Pi_{t,i}) \) is defined by:

\[
E(\Pi_{t,i}) = \int_{\Pi_{t,i}} dq d\eta.
\]

To calculate the previous integral the \( \alpha \)-shapes method is applied to the regions \( \Pi_{t,i} \) for a range of values for \( \alpha \); for each value an approximation of the boundaries \( \partial \Pi_{t,i} \) is obtained and the intersection points \( q_{t,i,j}(\eta)_{i=1,\cdots,p} \) between \( \partial \Pi_{t,i} \) and the horizontal lines \( \eta = \text{const.} \), with \( \eta \in [-1,1] \), are computed for each \( j \in \{1, \cdots p\} \). Ordering the points \( q_{t,i,j}(\eta)_{i=1,\cdots,p} \) in ascending order, Equation (15) becomes:

\[
E(\Pi_{t,i}) = \sum_{j=1}^{m} \int_{-1}^{1} (q_{t,i,j}(\eta) - q_{t,i,j-1}(\eta)) dq d\eta,
\]
where \( m \) is the integer part of \( r/2 \) and \( r \) is the number of the intersection point between \( \partial R_{k,l} \) and the horizontal lines \( \eta = const \).

The integral in Equation (16) is calculated discretizing the interval \([-1, 1]\) into \( Nb = 100 \) sub-intervals of equal width and using the trapezoidal rule to approximate the integral. Note that by increasing the value of \( \alpha \), the area inside the \( \alpha \)-shape increases and, consequently, the étendue raises. Eventually, matching the value of the étendue at the source \( E_s \) with the value of the étendue at the target \( E_t \), a unique value of \( \alpha \), \( \alpha_c \), is determined. Figure 6 shows that a value of \( \alpha_c = 0.041 \) is obtained for a set of \( 1.9 \cdot 10^4 \) rays.

Fig. 6. The source and the target étendue are depicted with the dotted red line and the blue line, respectively. \( E_s \) is computed for a range of values for \( \alpha \). \( E_t \approx 7.68 \) The green dot indicates the value of \( \alpha_c = 0.041 \) which gives the best approximation of the boundaries \( \partial R_{k,l} \) at the target. Around \( 1.9 \cdot 10^4 \) rays have been traced in PS.

The \( \alpha \)-shapes method with \( \alpha = \alpha_c \) is applied to the target regions \( R_{k,l} \), and the best approximations of the boundaries \( \partial R_{k,l} \) are obtained (see Figure 7).

To conclude, we compute the target intensity, which is determined in \( P_t \) by Equation (10). From Equation (11), we obtain:

\[
I_{PS}(\eta) = \sum_{ij} \int_{q_{ij,2i-1}(\eta)}^{q_{ij,2i}(\eta)} L_i(q, \eta) dq ,
\]

(17)

where the summation for the indexes \( i \) is over all \( i = 1, 2, \ldots, m \), and the summation for the \( j \) indexes is over all \( j \) for which the intersection between \( \eta = const \) and \( R_{k,l} \) is not empty. In the case of a Lambertian source with \( L_1(x, \tau) = 1 \), the following relation for the intensity at the target holds:

\[
I_{PS}(\eta) = \sum_{ij} (q_{ij,2i}(\eta) - q_{ij,2i-1}(\eta)) ,
\]

(18)

where the relation \( n_1 = 1 \) and the property of conservation of luminance along a ray [10] are exploited. We again notice that Equation (17) and (18) are valid when only two intersection points are found. If \( r > 2 \) intersection points occur the sum of the distances \( (q_{ij} - q_{ij-1}) = 1, \ldots, m \) needs to be computed. A uniform partitioning \( P : -1 \leq \eta_0 < \eta_1 < \cdots < \eta_{Nb} \leq 1 \) of the interval \( J = [-1, 1] \) is considered, where \( Nb = 100 \) and \( -1 \) and 1 are the minimum and maximum value for \( \eta \), respectively. Eventually, the intensity for each \( \eta_k \), with \( h = 0, 1, \ldots, Nb \), is computed using relation (18). We compare the new method with the already existing MC ray tracing to show its efficiency.

The intensity for MC ray tracing is computed as follows. The partitioning \( P \) of \( J = [-1, 1] \), used for the target phase space, is considered and the number of rays that fall into each bin \((\eta_{k}, \eta_{k+1})\) is calculated for every \( h \in [0, \ldots, Nb - 1] \). The intensity in the direction \( \eta_k \) is \( I_{PS}(\eta_k) \) approximated by:

\[
I_{MC}(\eta_k) = \frac{Nr([\eta_{h}, \eta_{h+1}])}{Nr([-1, 1])} ,
\]

(19)

for every \( \frac{q_{k} + q_{k+1}}{2} \), where we have indicated with \( Nr([\eta_{h}, \eta_{h+1}]) \) the number of rays that fall into the bin \( [\eta_{h}, \eta_{h+1}] \) and with \( Nr([-1, 1]) \) the total number of rays. Since \( I_{MC} \) is normalized, to compare the two intensities also a normalization of \( I_{PS} \) is required. This normalization is calculated dividing the intensity by the total étendue \( E_T \):

\[
I_{PS}(\eta_k) = \frac{1}{E_T} \int_{\eta_k}^{\eta_{k+1}} I_{PS}(\eta) \quad \text{for} \quad k = 1, 2, \ldots, Nb ,
\]

(20)

where \( E_T \) is obtained removing from the total étendue \( E_T \), computed in Equation 16 and showed in Figure 13, the étendue corresponding to the regions formed by rays that hit the left and the right detectors.

Note that the intensities are vectors of length \( Nb \) and \( I_{PS}(\eta_k) \) represents the intensity in the direction \( \eta_k \). To obtain a more precise intensity profile, the number of bins for the partitioning \( P \) should be increased. Choosing \( Nb = 100 \), a smooth profile of the intensity is found; hence, we decide to fix that value of \( Nb \). Moreover, the intensity calculation in phase space strongly depends on the computation of the boundaries. Therefore, the more precise the choice of \( \alpha \) for the \( \alpha \)-shapes method is, the more accurate the intensity profile is. The consequence is that, in the case where the regions \( R_{k,l} \) contain too few rays, there is no way to estimate their boundaries \( \partial R_{k,l} \). When the point cloud is formed by a small number of rays, the Delaunay triangulation provides large triangles that do not permit to achieve a good approximation of the boundaries of a concave hull. As a consequence, the calculated area is bigger than the area covered by the region \( R_{k,l} \).
and the intensity is too high for the angles where the area is inaccurate. By increasing the number of points in the point set \( Q \), the number of the triangles in the Delaunay triangulation increases, the size of each triangle decreases and therefore the value of \( \alpha_c \) decreases in such a way that the best approximation of the boundaries can be obtained. The photometric variables at the target are now determined for both MC and PS method. The numerical results are shown in the next section.

3. NUMERICAL RESULTS AND DISCUSSION

In this section a comparison between the MC and PS methods is presented. The MC and PS intensities are calculated several times increasing the number of rays to improve the accuracy. Both the approximate intensities are compared with an intensity taken as a reference. For some optical systems, there is an explicit solution for the target intensity but this is not the case of the TIR-collimator. Therefore, a MC simulation with \( 1, 7 \cdot 10^8 \) rays is run to obtain the normalized intensity \( \hat{I}_{\text{ref}} \) which is used as reference.

We show how the error, defined as:

\[
\text{error} = \frac{\sum_{h=1}^{N_b} | I_{\text{PS}}(\eta_h) - \hat{I}_{\text{ref}}(\eta_h) |}{N_b},
\]

decreases for increasing number of rays. Table 1 describes how the number of rays traced affects the error and shows the correlation between \( \alpha_c \) and the number of rays, which is determined by the values of \( \epsilon_{\text{max}} \), \( \epsilon_{\text{min}} \), \( \epsilon_{\text{rms}} \), and \( \epsilon_{\text{rms}} \), as explained in Section 2C. The values of \( \alpha_c \) are selected according to the method presented in Section 2D. Then, the intensity for the MC method, \( \hat{I}_{\text{MC}} \), is computed. Replacing \( \hat{I}_{\text{PS}} \) with \( \hat{I}_{\text{MC}} \) in Equation (21), the error between the reference intensity and the intensity computed with the MC method is calculated. Increasing the number of rays traced in MC ray tracing, the error gradually decreases. In Table 2 the numerical results found are reported.

### Table 1. Error values of the PS intensity

<table>
<thead>
<tr>
<th>Number of rays</th>
<th>( \epsilon_{\text{min}} )</th>
<th>( \epsilon_{\text{max}} )</th>
<th>( \epsilon_{\text{rms}} )</th>
<th>( \epsilon_{\text{rms}} )</th>
<th>( \alpha_c )</th>
<th>( \text{error} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>3363</td>
<td>0.9</td>
<td>0.1</td>
<td>0.025</td>
<td>0.119</td>
<td>1.20 \cdot 10^{-3}</td>
<td></td>
</tr>
<tr>
<td>6949</td>
<td>0.5</td>
<td>0.050</td>
<td>0.020</td>
<td>0.098</td>
<td>2.50 \cdot 10^{-4}</td>
<td></td>
</tr>
<tr>
<td>15870</td>
<td>0.4</td>
<td>0.025</td>
<td>0.001</td>
<td>0.050</td>
<td>5.49 \cdot 10^{-4}</td>
<td></td>
</tr>
<tr>
<td>37455</td>
<td>0.2</td>
<td>0.020</td>
<td>0.005</td>
<td>0.037</td>
<td>2.00 \cdot 10^{-5}</td>
<td></td>
</tr>
<tr>
<td>66855</td>
<td>0.1</td>
<td>0.009</td>
<td>0.004</td>
<td>0.020</td>
<td>1.00 \cdot 10^{-5}</td>
<td></td>
</tr>
</tbody>
</table>

### Table 2. Error values of the MC intensity

<table>
<thead>
<tr>
<th>Number of rays</th>
<th>( \text{error}_{\text{MC}} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>972</td>
<td>2.10 \cdot 10^{-3}</td>
</tr>
<tr>
<td>9714</td>
<td>6.69 \cdot 10^{-4}</td>
</tr>
<tr>
<td>97103</td>
<td>2.08 \cdot 10^{-4}</td>
</tr>
<tr>
<td>971627</td>
<td>7.00 \cdot 10^{-5}</td>
</tr>
<tr>
<td>9716519</td>
<td>2.00 \cdot 10^{-5}</td>
</tr>
</tbody>
</table>

In Figure 8, the results listed in Table 1 and Table 2 are shown where the red line depicts the behavior of the error for the PS intensity, and the blue line indicates the error for the MC simulation.

The intensity profile \( \hat{I}_{\text{PS}} \) obtained with the phase space method and tracing 66 855 rays is depicted in Fig. 9 with a red line. \( \hat{I}_{\text{PS}} \) is hardly distinguishable from \( \hat{I}_{\text{ref}} \) (dashed and blue line in Figure 9). Figure 8 shows that an error equal to 2.00 \cdot 10^{-5} can be obtained tracing at least 10^5 times fewer rays in phase space.

The intensity profile \( \hat{I}_{\text{PS}} \) obtained with the phase space method and tracing 66 855 rays is depicted in Fig. 9 with a red line. \( \hat{I}_{\text{PS}} \) is hardly distinguishable from \( \hat{I}_{\text{ref}} \) (dashed and blue line in Figure 9). Figure 8 shows that an error equal to 2.00 \cdot 10^{-5} can be obtained tracing at least 10^5 times fewer rays in phase space.
are implemented for the collimator in Figure 11 where the lens is approximated with an analytic expression and the target is located at a distance $d = 0.01$ from the top. Figure 12 shows that a flatter lens removes one of the two spikes of the region formed by the rays that hit the lens. Moreover a target located very close to the top makes the shape of that region less stretched along the $q$-axis. Therefore the $\alpha$-shapes method performs well, even for a small number of rays.

Next, another method to compute the boundaries $\partial R_{s,\Pi_j}$ is provided. This method is based on the triangulation refinement presented in Section 2C. As explained in that section, more rays close to the boundaries can be traced selecting increasingly smaller values for the parameters $\epsilon_{x_{\text{max}}}$ and $\epsilon_{x_{\text{max}}}$. Once the algorithm stops, only the triangles that are expected to be crossed by a boundary are taken into account. By construction, each of these triangles has two vertices that follow the same path and one vertex that follows another path. The triangles are ordered in such a way that two triangles are neighbors if they have a side in common. Given a path $\Pi_j$ with $j \in \{1, \cdots, p\}$ the boundary $\partial R_{s,\Pi_j}$ of the region corresponding to that path in source phase space is approximated by those vertices of the triangles that follow the path $\Pi_j$. With this procedure we can easily construct the boundaries $\partial R_{s,\Pi_j}$ in source phase space for every $j \in \{1, \cdots, p\}$. The boundaries $\partial R_{s,\Pi_j}$ at the target are given by $M(\partial R_{s,\Pi_j})$. To establish the minimum value of the parameter $\epsilon_{x_{\text{max}}}$ that gives a good approximation of the boundaries $\partial R_{s,\Pi_j}$, again the conservation of the étendue in phase space [10] is exploited. The essence of our approach is as follows.

We consider the étendue for the whole $\mathcal{P}_s$ which is given by Equation 12. The rays traced are uniformly distributed over $\mathcal{P}_s$, and they cover it entirely. The total étendue at the target is given by Equation 14. Furthermore, to calculate the integral in Equation 15 the triangulation refinement method is applied to the regions $R_{s,\Pi_j}$ for a range of value for $\epsilon_{x_{\text{max}}}$; for each value an approximation of the boundaries $\partial R_{s,\Pi_j}$ is obtained. Since the computation of the boundaries does not depend on the number of rays inside the regions, the étendue remains constant when
the value of $\epsilon_{\text{max}}$ changes. Then, we choose $\epsilon_{\text{max}}$ as large as possible to avoid tracing rays that do not significantly contribute to the computation of the photometric variables at the target. That value depends on the distribution of the rays in phase space. For our optical system the parameter $\epsilon_{\text{max}} = 1$. Figure 13 shows that decreasing the value of the parameter $\epsilon_{\text{max}}$ the values of the étendue at the target $E_t$ increases and achieves a value of 7.9 when $\epsilon_{\text{max}} = 3 \cdot 10^{-3}$. We decide to stop the phase space refinement procedure when a good approximation of the étendue (note that the total étendue is equal to 7.92).

Once the boundaries are computed the output intensity is calculated. Finally an error estimation between the exact intensity and the approximate intensity is obtained. The approximate intensity is computed with three different methods: MC ray tracing, PS ray tracing using the $a$-shapes method to approximate the boundaries and, PS ray tracing using the triangulation method to obtain the boundaries approximation. In Figure 14 the error plot for the TIR-collimator in Figure 11 is depicted. The blue line indicates the error for MC ray tracing, the red line indicates the behavior of the error for the PS intensity using the $a$-shapes method to calculate the boundaries, the cyan line depicts the error for the PS intensity using the triangulation refinement to calculate $\partial R_{t,11}$. Increasing the number of rays the error decreases as $\frac{1}{\sqrt{N_r}}$ for the MC method and as $\frac{1}{N_r}$ for the PS simulation.

4. CONCLUSION

A new ray tracing method based on the phase space representation of the source and the target of the optical system was presented. The method was presented for two-dimensional optical systems, numerical simulations were implemented for a TIR-collimator and the intensity at the target was computed for this system. For each ray, the patches that are hit were calculated, and all the rays with the same path were grouped together in the same region in phase space. To compute the photometric variables at the target, the boundaries of these regions were found using the $a$-shapes method in the target phase space. The intensity at the target was obtained using the location given by the rays on the boundaries of the regions; hence, the rays inside those regions do not significantly contribute to the photometric variables. To trace only rays close to the boundaries, a refinement of the source phase space was constructed. Eventually, applying the edge ray principle, we concluded that the traced rays are located close to the boundaries of the regions even in the target phase space. As a result, the output intensity is calculated using far less rays. The efficiency of the method was proved using a highly accurate intensity profile as a reference that was found tracing randomly around $1.7 \cdot 10^8$ rays through the optical system and using the Monte Carlo procedure. Then, the phase space and the Monte Carlo ray tracing were compared. The error between the approximate intensity and the exact intensity was calculated for both methods. An initial ray set was fixed, and the number of rays was gradually increased to obtain a more accurate intensity profile. Numerical simulations have demonstrated that Monte Carlo ray tracing the error decreases proportionally to the inverse of the square root of the number of rays, while for PS ray tracing it has a much higher speed of convergence.

To conclude, the ray tracing procedure is implemented for a different design of the TIR-collimator where the lens is flatter than the previous one and the target is very close to the top of the collimator. To compute the boundaries in phase space an alternative technique that exploits a triangulation refinement of source phase space is presented. An error between the approximate intensity and the exact intensity is computed for each procedure. Note that still the PS method converges faster than the MC method, indeed in PS the speed of convergence is proportional to the inverse of the number of rays traced. The PS and MC ray tracing were implemented considering only Snell’s law and a constant luminance. The method can be generalized using a non-constant luminance which can be computed through an interpolation algorithm. In the future, we plan to take into account Fresnel reflection each time that a ray hits a surface of the optical system.

REFERENCES

13. A. Torre, Linear ray and wave optics in phase space: Bridging ray and wave optics via the Wigner phase-space picture (Elsevier, 2005).
16. H. Edelsbrunner, Weighted alpha shapes (University of Illinois at Urbana-Champaign, Department of Computer Science, 1992).
<table>
<thead>
<tr>
<th>Number</th>
<th>Author(s)</th>
<th>Title</th>
<th>Month</th>
</tr>
</thead>
<tbody>
<tr>
<td>15-37</td>
<td>N. Kumar, J.H.M. ten Thije, B. Koren</td>
<td>Flux approximation scheme for the incompressible Navier-Stokes equations using local boundary value problems</td>
<td>Dec. '15</td>
</tr>
<tr>
<td>15-38</td>
<td>C. Filosa, J.H.M. ten Thije, W.J. IJzerman</td>
<td>A new ray tracing method in phase space using α-shapes</td>
<td>Dec. '15</td>
</tr>
</tbody>
</table>