

Phase Space Ray Tracing for Illumination Optics

Carmela Filosa

Cover art:
Photography:

A catalogue record is available from the Eindhoven University of Technology Library

ISBN: 978-90-386-3972-7

Copyright © 2017 by C. Filosa.

All rights are reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without prior permission of the author.

Phase Space Ray Tracing for Illumination Optics

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de
Technische Universiteit Eindhoven, op gezag van de
rector magnificus prof.dr.ir. F.P.T. Baaijens, voor een
commissie aangewezen door het College voor
Promoties, in het openbaar te verdedigen

door

Carmela Filosa

geboren te Torre del greco, Italië

Dit proefschrift is goedgekeurd door de promotoren en de samenstelling van de promotiecommissie is als volgt:

voorzitter: prof.dr. J.J. Lukkien
1^e promotor: prof.dr. W.L. IJzerman
copromotor: dr. J.H.M. ten Thije Boonkkamp
leden: prof.dr.ir. G.M.W. Kroesen
prof. Dr. A. Herkommer (Universität Stuttgart)
dr. Y. Meuret (KU Leuven)
prof.dr.ir. B. Koren
dr. S.M.B. Baumer (TNO Den Haag)
prof.dr. W.H.A. Schilders

Het onderzoek of ontwerp dat in dit proefschrift wordt beschreven is uitgevoerd in overeenstemming met de TU/e Gedragscode Wetenschapsbeoefening.

Contents

1	Introduction	7
1.1	Motivation	7
1.2	Phase space methods and results	10
1.3	Outline of this thesis	12
2	Illumination optics	15
2.1	Radiometric and photometric variables	15
2.2	Reflection and refraction law	20
2.3	Fresnel's coefficients for reflection and refraction	22
3	Ray tracing	31
3.1	Ray tracing for two-dimensional optical systems	31
3.2	Monte Carlo (MC) ray tracing	33
3.3	Quasi-Monte Carlo (QMC) ray tracing	37
4	Ray tracing on phase space	43
4.1	Phase space	43
4.2	The edge-ray principle	45
4.3	Phase space ray tracing	47
4.4	Conclusions	49
5	The α-shapes approach	55
5.1	α -shapes theory	55
5.2	Determination of α using étendue conservation	59
5.3	Results for a TIR-collimator	60
5.4	Conclusion	66
6	Boundaries reconstruction based on the triangulation refinement	71
6.1	Reconstruction of the boundaries	71
6.2	The two-faceted cup	73
6.3	A TIR-collimator	75
6.4	A Parabolic reflector	79
6.5	Discussion and conclusions	83

7 Concatenated backward ray mapping	89
7.1 Phase spaces of the two-faceted cup	89
7.2 Numerical results for the two-faceted cup	102
7.3 Extension of the method for the multi-faceted cup	106
7.4 Numerical results for the 20-faceted cup	109
7.5 Discussions	112
8 Direct backward ray mapping	113
8.1 Bisection method and backward ray tracing	113
8.2 Results for the TIR-collimator	119
8.3 Results for the parabolic reflector	122
8.4 Conclusions	125
9 Direct backward ray mapping for systems with Fresnel reflection	129
9.1 Introduction	129
9.2 Backward ray mapping for an ideal lens	131
9.3 Numerical results	136
9.4 Conclusion and outlook	140
10 Conclusions and Recommendations	141
10.1 Summary	141
10.2 Recommendations	144
A Monte Carlo integration versus Monte Carlo ray tracing	147
A.1 Monte Carlo integration	147
A.2 Monte Carlo ray tracing	149
B Two examples of low discrepancy sequences	151
B.1 Van der Corput sequences	151
B.2 Sobol sequences	152
C Exact intensity for the two-faceted cup	155
C.1 Analytic approach	155
Bibliography	161
Curriculum Vitae	169
Acknowledgments	172

Acronyms

TIR	Total internal reflection
MC	Monte Carlo
QMC	Quasi-Monte Carlo
PS	Phase space
CPC	Compound parabolic reflector
OPL	Optical path length

List of symbols

Φ_r	radiant flux, [W]
Q	total energy emitted from a light source or received by a target, [J]
τ	time, [sec]
Φ	luminous flux, [lm]
y	luminous efficacy function, [lm/W]
λ	wavelength
Ψ_r	spectral radiant flux, [W/m]
Ψ	spectral luminous flux, [lm/m]
$\bar{y}(\lambda)$	luminosity efficiency function
E	illuminance, [lm/m ²]
P	irradiance (radiant flux density), [W/m ²]
Ω	solid angle
ν	surface normal
I	intensity, [cd = lm/sr]
L	luminance, [cd/m ²]
U	étendue, [m ² sr]
n	index of refraction
θ	polar angle between ν and the direction of the central line of $d\Omega$
ϕ	azimuthal angle between ν and the direction of the central line of $d\Omega$
t_i	direction of the incident ray
t_r	direction of the reflected ray
t_t	direction of the transmitted ray
s	arc length
n_i	index of refraction of the medium in which the incident ray travels
n_r	index of refraction of the medium in which the reflected ray is located
n_t	index of refraction of the medium in which the transmitted ray travels
θ_i	angle between the incident ray and the normal ν
θ_r	angle between the reflected ray and the normal ν
θ_t	angle between the transmitted ray and the normal ν
P_i	incident radiant flux density
P_r	reflected radiant flux density
P_t	transmitted radiant flux density
\mathcal{E}	electric field
\mathcal{B}	magnetic field
\mathcal{E}_0	amplitude of the electric field

\mathcal{B}_0	amplitude of the magnetic field
μ	permeability
ε	permittivity
ω	angular frequency
c	speed of light in vacuum
v	speed of light in medium
r	amplitude coefficient for the reflected light
t	amplitude coefficient for the transmitted light
r_s	amplitude coefficient for reflected s-polarized light
t_s	amplitude coefficient for transmitted s-polarized light
r_p	amplitude coefficient for reflected p-polarized light
t_p	amplitude coefficient for transmitted p-polarized light
\mathcal{R}	reflectance
\mathcal{T}	transmittance
\mathcal{R}_s	reflectance for s-polarized light
\mathcal{T}_s	transmittance for s-polarized light
\mathcal{R}_p	reflectance for p-polarized light
\mathcal{T}_p	transmittance for p-polarized light
θ_c	critical angle
θ_p	Brewster's angle for $n_t < n_i$
θ'_p	Brewster's angle for $n_t > n_i$
S	light source
T	light target
(x_j, z_j)	coordinates of the intersection point of the rays with line j in real space
t_j	angle that the ray incident line j makes with respect to the optical axis
$\mathbf{r}(s)$	ray parametrization
N_r	number of rays
N_b	number of bins
n_j	index of refraction of the medium in which line j is located
N_l	number of lines that constitute the system = Index of the target
\mathbf{Y}	random variable
\mathbf{y}	values that a random variable assumes
$\rho(\mathbf{y})$	probability density function
$\mathbb{E}[f]$	expected value of f
$\text{Var}[f] = \sigma^2$	variance of f
$\sigma^2[f]$	standard deviation of f
(x_j, z_j)	Position coordinates of the on line j of the j -th ray traced
t_j	angular coordinate of the j -th ray traced
\hat{I}_{MC}	averaged and normalized Monte Carlo intensity
\hat{I}_{QMC}	averaged and normalized Quasi-Monte Carlo intensity
$\text{card}(\mathbf{A})$	cardinality of the set A
$D_N(\mathbf{Y})$	discrepancy of the finite set Y of N points.
$D_N^*(\mathbf{Y})$	discrepancy of the finite set Y of N points.
$V(f)$	Hardy-Krause variation function.
\mathbf{Q}	set of the position coordinates of the rays
\mathbf{P}	set of the direction coordinates of the rays
$\mathbf{S}=\mathbf{Q} \times \mathbf{P}$	phase space of the rays with position in \mathbf{Q} and direction in \mathbf{P}

S	source phase space
T	target phase space
(q_1, q_1)	coordinates of the ray on source PS S
(q, q)	coordinates of the ray on target PS T
Π	ray path
M	ray mapping form source to target PS
$R_s(\Pi)$	$R_1(\Pi)$ regions in source PS related to path Π
$R_t(\Pi)$	$R_t(\Pi)$ positive luminance regions in target PS related to path Π
(q_1^k, p_1^k)	source PS coordinates of the k -th ray traced
Π^k	path followed by the k -th ray traced
Np	Number of paths
$\hat{I}_{PS}(p)$	averaged and normalized phase space intensity at the target
$\hat{I}_{exact}(p)$	exact averaged and normalized target intensity
$\hat{I}_{ref}(p)$	reference averaged and normalized target intensity
Π_j	j -th possible path ($j = 1, \dots, N_p$)
ε_q^{\min}	minimum triangulation parameter for the q -axis
ε_p^{\min}	minimum triangulation parameter for the p -axis
ε_q^{\max}	maximum triangulation parameter for the q -axis
ε_p^{\max}	Maximum triangulation parameter for the p -axis
$q^i(\Pi, p)$	q -coordinate on $\partial R(\Pi)$ along direction p
$q^{\min}(\Pi, p)$	minimum q -coordinate on $\partial R(\Pi)$ along direction p
$q^{\max}(\Pi, p)$	maximum q -coordinate on $\partial R(\Pi)$ along direction p
T	triangulation in \mathbb{R}^2
T'	Delaunay triangulation in \mathbb{R}^2
V_i	Voronoi cell
V	Voronoi diagram
S_i	source phase space of line i
T_i	target phase space of line i
$T_{i,j}$	part of T_i that is illuminated by line j with $i \neq j$
$q_{s,i}$	position coordinates of the rays on S_i
$q_{t,j}$	Position coordinates of the rays on T_j
ν_i	normal to line i
$p_{s,i}$	direction coordinates on S_i , $p_{s,i} = n_i \sin(\tau_i)$
$p_{t,i}$	direction coordinates on T_i , $p_{s,i} = n_i \sin(\tau_i)$
$M_{1,Nl}$	map from S_1 into T_{Nl}
$M_{1,Nl}(\Pi)$	map from S_1 to $R(\Pi)$
$P_{i,j}$	propagation map from S_i to T_j
R_j	reflection map from T_j to S_j
$u_{j,i}^{\min}(p)$	minimum position coordinate on $\partial T_{j,i}$ along direction $p = p_{t,j}$
$u_{j,i}^{\max}(p)$	maximum position coordinate on $\partial T_{j,i}$ along direction $p = p_{t,j}$
$v_{j,i}^{\min}$	$\max(q_{j,i}^{\min}, u_{j,i}^{\min})$
$v_{j,i}^{\max}$	$\min(q_{j,i}^{\max}, u_{j,i}^{\max})$
(q^a, p)	coordinates of the left end point of T
(q^b, p)	coordinates of the right end point of T

Chapter 1

Introduction

1.1 Motivation

A significant amount of the total electricity is consumed by residential and commercial illumination. According to the U.S. Energy Information Administrator (EIA) about 7% of the total electricity consumed in the U.S. in 2017 was for lighting [1]. These figures are impressive if one considers that the global electricity consumption was more than 20,000 TWh in 2016 [2].

The energy consumed by an illumination device is clearly related to the efficiency of the device itself. In particular, part of the energy consumption consists of excessive, misdirected and inefficient use of light, which could be used in a smarter way. A tangible aspect of such lack of efficiency is light pollution, a persistent and increasing problem in most cities. Light pollution is not only wasted energy but can also have adverse impacts on for example humans' sleep patterns, bird and fish migration and plants' growth [3]. Figures 1.1 and 1.2 show the light pollution of the Gulf of Naples.

Using efficient light sources could help to mitigate both the amount of energy consumption and light pollution. Solid-state lighting systems using light-emitting diodes (LEDs) can reach a luminous efficacy up to 120 lm/W versus only around 60 lm/W achieved by a fluorescent lamp [3, 5]. Because of this, many optical industries are interested in fabricating efficient light sources. Currently, LEDs are replacing traditional sources [6]. This is mainly due to their high energy efficacy and very long lifetime. LED systems consist of several components. Electronic devices convert the AC mains voltage into a voltage suitable for the LED source. An optical system is required, it is formed by optical components such as lenses and reflectors [7] (see Figures 1.3 and 1.4).

Light emitted from the LED propagates within the system which reshapes light according to physical laws (e.g. reflection, refraction and scattering phenomena). LED systems are increasingly used in many application such as street lights, automotive lighting and home illumination. The efficiency of the source does not reflect how much light is transferred from the source to the target, and for each application a certain target distribution is desired. For example, street lights should provide a uniform illumination of the streets; car headlights should be built such that they have



Figure 1.1: Gulf of Naples at night.



Figure 1.2: Satellite view of Naples [4].

a powerful and uniform light while avoiding uncomfortable glare for the oncoming traffic. The material and the shape of the optical components are very important for the efficiency of the entire lighting system. Thus, the aim of optical design is to create optical systems which direct light only where illumination is desired, avoiding to spread light everywhere [8, 9].

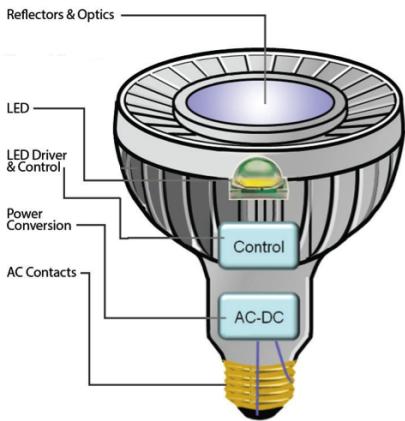


Figure 1.3: LED system [10].

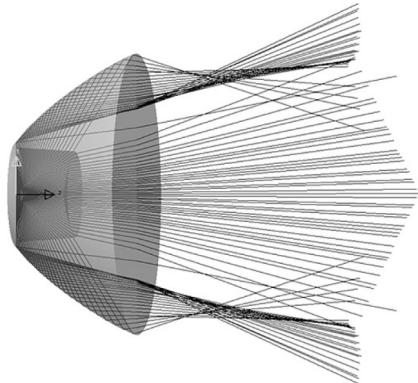


Figure 1.4: Example of an optical system. A total internal reflector in 3D [11].

Illumination optics is the branch of optics that deals with the design of optical systems. It concerns the transfer of light from the source to the target. The goal in illumination optics is to obtain the desired light distribution at the receiver after its propagation through the optical system. Depending on the intensity distribution required, an optical engineer can determine what type of optic is needed. To this purpose, it is important to understand how light propagates through a given optical system.

Light can be seen as the electromagnetic radiation perceived by the human eye [12]. Therefore, it propagates as an electromagnetic wave and, as such, it has a certain

wavelength. Visible light has wavelength in the range between 400 and 700 nm. Usually the wavelength of light is extremely small compared to the dimensions of the optics with which it interacts, so it is reasonable to consider the limit of infinitely short wavelengths. The field of optics that discards the microscopic wave behavior of light is called *geometric optics*. In the geometric optics approximation, light is described by rays that propagate perpendicular to the wavefront, transporting electromagnetic energy. Since the dimensions of the optical components used for illumination systems are very big compared to the wavelength of visible light, illumination optics is usually described in terms of geometric optics [13].

To compute the photometric variables at the target of an optical system, the ray tracing procedure is widely used in the field of geometric optics [14]. Ray tracing is a forward method which provides the light target distribution given a light source and an optical system [15]. Every ray is considered to be a straight line when propagating in free space. Once it hits an optical surface its direction changes according to reflection and refraction laws. Ray tracing computes the intersection points of every ray emitted from the source with *all* the optical surfaces that it encounters. Next, it calculates the surface at the closest distance from the point where the ray was emitted. The new ray direction at the intersection point ray is computed. The procedure continues until the ray reaches the target. These steps are repeated for all the rays traced. There are many ways to implement the ray tracing process.

Monte Carlo (MC) ray tracing is often used in non-imaging optics. This method is based on a probabilistic interpretation of the rays distribution at the source of the optical system [16, 17]: 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. For example, the output intensity distribution is provided dividing the target into equal cells and counting the number of rays that arrive at each cell of the target. Although the MC procedure constitutes a robust method, it remains a slow and numerically costly procedure, as it converges proportionally to the reciprocal value of the square root of the number of rays traced. Therefore, to obtain a good accuracy of the intensity distribution at the target, millions of rays need to be traced.

To speed up MC ray tracing, deterministic Quasi-Monte Carlo (QMC) ray tracing was introduced. The difference between MC and QMC is that in the latter the coordinates of the rays traced are distributed according to *low-discrepancy sequences*. Intuitively, the discrepancy indicates how much the rays distribution differs from a uniform distribution. Hence, low discrepancy sequences are close to uniformly distributed sequences [18]. Numerical simulations show that in most cases QMC ray tracing is an improvement of MC ray tracing [19, 20]. However, it is not possible to predict the convergence of QMC ray tracing as the error is always bounded by a term proportional to the discrepancy of the initial sample of rays [21].

The purpose of this thesis is to provide tools to improve illumination optics design by using faster and more accurate methods than the current state-of-the-art ray tracing methods. To do so, we analyze optical systems using the *phase space* concept [22]. The phase space (PS) of an optical surface is described by the position and direction coordinates of all the rays that hit the surface [23]. We restrict ourselves to the two-dimensional case which is particularly relevant as it constitutes a good test case to demonstrate the performance of the new ray tracing method. For any rotational symmetric system the 2D case is very useful to study. Optical designers often start

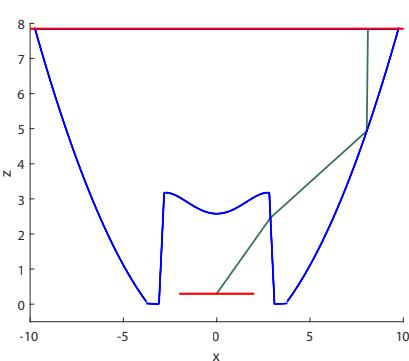


Figure 1.5: **A ray propagating inside an optical system.**

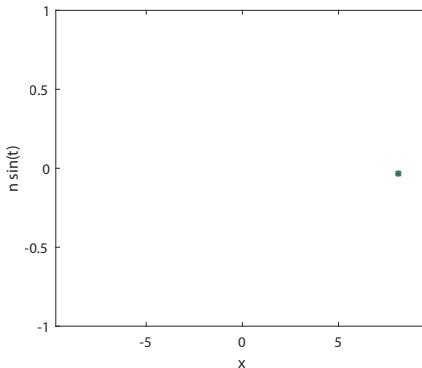


Figure 1.6: **Ray at the target PS of the system.** A ray in PS is a point with coordinates $(x, n \sin(t))$.

working in two dimensions where only the meridional plane is taken into account. In two dimensions we refer to *optical lines* instead of optical surfaces. The PS of a two-dimensional system is a two-dimensional space described by all the possible ray positions and directions.

Optical phenomena can be analyzed by using the PS and the photometric variables can be defined in PS [24]. Currently, some literature based on phase space optics has been published, showing that light propagation can be investigated using the phase space concept [25, 26, 27]. In particular, it has been proved that the radiance and irradiance distribution at the target can be analyzed by looking at the phase space transformations from the source to the target. Also, phase space optics might constitute an alternative approach for describing aberration phenomena [28, 29, 30]. In this thesis we introduce new methods based on PS providing a new way to calculate the light distribution at the target of optical systems. Our PS methods allows tracing less rays inside the system than existing procedures, i.e., MC and QMC ray tracing. This significantly reduces the computational time compared to conventional ray tracing. Next, the methods and results are briefly discussed.

1.2 Phase space methods and results

The new methods presented in this thesis are based on PS rays tracing. The PS of a line is a 2D space defined by all possible position and one direction coordinate of rays that hit the line. The position coordinate is given by one of the two coordinates of the intersection point between the rays and the optical line, the direction coordinate is the sine of the angle t that the ray forms with the normal of the line encountered measured counterclockwise multiplied by the index of refraction n of the material in which the line is located [31]. In Figure 1.5 we show a ray propagating inside a two-dimensional optical system in the (x, z) -plane. This ray is described by a unique point in target PS with corresponding coordinates $(x, n \sin(t))$ (see Figure 1.6). PS ray tracing takes into account the *path* followed by every ray, that is, the sequence of

the optical components encountered by every ray traced. Considering for every ray traced its corresponding path, we observe that the PS of the source and the target are partitioned into regions, each of them corresponds to rays that follow the same path. While the source PS is completely covered by these rays, the target PS is not.

The boundaries of the regions in target PS give important information about the output photometric variables because there the target luminance has a jump discontinuity from zero to a positive value. The positive luminance regions should be considered over all the possible directions to provide the profile of the output luminance and numerical integration gives the target intensity profile. Detecting the boundaries of the positive luminance regions, the photometric variables can be quickly computed without the need for tracing millions of rays, which is necessary for conventional procedures such as MC and QMC ray tracing. We focus on two different procedures: phase space ray tracing and backward ray mapping in phase space.

Phase space ray tracing is a forward method that employs the source and the target PS representation of the optical system. The key idea is to construct a nonuniform triangulation in source PS such that more triangles are defined close to the discontinuity of the luminance. The coordinates of the vertices of each triangle correspond to the coordinates of the rays traced that will be located close to the discontinuities of the luminance. In order to compute the target photometric variables, the boundaries of the positive luminance regions need to be calculated. Therefore, we provide two different approaches to obtain an approximation of the boundaries at the source PS.

The first technique is based on α -shapes. Given a triangulation in source PS (usually the Delaunay triangulation [32]), a parameter α is used to decide which triangles have to be included for the boundaries computation and which have to be removed. In general it is not easy to establish the value of α that gives the desired boundary computation [33]. We develop a procedure based on étendue conservation [34].

The second technique for the boundaries computation exploits the triangulation refinement explained above. The approximation of the boundaries in source PS is obtained by connecting the vertices on one side of the boundary between two regions, corresponding to rays that follow different paths. Etendue conservation is used to provide a stopping criterion for the triangulation refinement [35, 36].

Once the boundaries are calculated at the source PS, also the boundaries at the target are obtained (edge-ray principle, [37]). Assuming a Lambertian source, the output intensity is calculated by only tracing the rays located on these boundaries.

Numerical results are provided using both α -shapes and the triangulation refinement for several optical systems. To validate the method, the intensities found are compared to both MC and QMC simulations. The results demonstrate that using PS ray tracing allows tracing far less rays than MC ray tracing to obtain an accurate approximation of the intensity profile. However, when using α -shapes, the number of rays traced depends on the complexity of the design of the optical system. On the other hand, computing the boundaries employing the triangulation refinement, a speed of convergence proportional to the inverse of the number of rays traced is obtained for all the systems considered versus a speed of convergence proportional to the inverse of the square root of the number of rays traced for MC ray tracing. Numerical simulations show that PS ray tracing based on the triangulation refinement gives speed advantages also comparable with QMC ray tracing when applied to some

optical systems, while it is slower than QMC for some other systems. PS ray tracing is therefore further improved by introducing the backward ray mapping method based on a ray mapping reconstruction in PS.

Backward ray mapping in phase space is first developed for systems formed by straight line segments. In this case the phase spaces of *all* the lines that constitute the system are considered. We assume that the optical lines are designed such that they can both receive and emit light while the source can only emit light and the target only receive it. Both source and target PS of each line are computed and only one PS is implemented for the source and the target. Concatenating all the phase spaces with two different maps, we are able to construct an inverse map from the target to the source. This explains the name *concatenated backward ray mapping* for this method. Numerical results show that only the rays on the target PS that are located *exactly* at the boundaries of the positive luminance regions are traced. The output intensity is computed integrating the luminance at the target PS. A comparison between concatenated backward ray mapping and QMC ray tracing demonstrates that our method computes the *exact* intensity, reducing significantly the computational time.

In order to modify the method to systems including curved lines we use a different approach, which employs the PS representation of only the target of the optical system. Applying a bisection method combined with backward ray tracing we are able to construct the inverse map from the target to the source *directly*. This allows tracing only the rays at the boundaries of the positive luminance regions [38]. From these rays the output intensity is calculated. We show in simulations that *direct backward ray mapping* is more accurate and faster than QMC ray tracing.

Finally, direct backward ray mapping is modified for systems with Fresnel reflections. In this case every ray incident on a Fresnel lens is split into a reflected and a refracted ray, each of them carry a fraction of the energy of the incident ray. This leads to a multitude of possible paths. Numerical simulations show that direct backward ray mapping is able to detect *all* the possible paths and to determine the boundaries of *all* the positive luminance regions in target PS. Since, for Fresnel systems, the luminance is not constant at the target, a sample of rays inside those regions needs to be traced for calculating the luminance and the intensity profile.

1.3 Outline of this thesis

This work is organized in the following way.

In Chapter 2 an overview of the physics of illumination optics is provided. After a short introduction of radiometric variables, the photometric counterparts are defined. The reflection and refraction laws are described and total internal reflection is discussed. A description of Fresnel reflection is included. In this chapter we follow the literature reported in [39, 40, 41].

Chapter 3 includes a discussion on classical ray tracing. In particular, MC and QMC ray tracing are discussed. They are based on a combination of MC and QMC procedures with ray tracing methods. We explain how to calculate the target intensity using these techniques. Numerical results for a simple system show the convergence of the approximated intensities to the exact intensity by increasing the number of rays traced.

Chapter 4 introduces the PS concept for two-dimensional optical systems. We show that the PS provides a complete description of optical systems. We explain how to construct the triangulation refinement on which PS ray tracing is based. The PS representation of the source and the target of a simple system (the two-faceted cup) shows that the phase spaces are divided into several regions formed by rays that follow the same path when they propagates through the system. Two techniques for calculating the boundaries of these regions are provided next.

A method based on α -shapes is presented in Chapter 5. A technique based on étendue conservation is developed and numerical results for two different total internal reflection (TIR)-collimators are provided. For these systems also the target intensity in PS is calculated and it is compared with MC ray tracing. The chapter concludes with a discussion of the results obtained.

A different approach for the boundaries calculation, which employs a triangulation refinement in source PS, is presented in Chapter 6. The method is applied to three different systems: the two-faceted cup, a TIR-collimator and a parabolic reflector. Numerical results are compared with both MC and QMC ray tracing.

Next, a second method based on ray mapping reconstruction from the target to the source is developed. Chapter 7 includes the description of concatenated backward ray mapping for systems formed by straight and reflective line segments. The target intensity is computed for two different optical systems: the two-faceted cup and a multifaceted cup. Numerical results are compared to QMC ray tracing.

In Chapter 8 we present the direct backward ray mapping method which is a modification to systems including curved and refractive lines. A detailed explanation of the idea and the algorithm used are given. The results for the TIR-collimator and the parabolic reflector validate the method showing that it calculates the intensity correctly.

The research concludes with Chapter 9 which discusses direct backward ray mapping applied to systems with Fresnel reflection. The theoretical explanation of the method is followed by numerical results applied to a system formed by the source, the target and a simple convex lens. We show in simulations that the boundaries of the positive luminance regions are calculated correctly. Finally, the profile of the luminance is obtained considering a sample of rays inside these regions and the intensity is given by integrating the luminance over all the possible positions.

Chapter 10 summaries the finding and presents discussions and insights for future research.

Chapter 2

Illumination optics

This chapter provides some concepts of illumination optics used in this thesis. We start explaining the difference between radiometry and photometry. In particular, we focus on the photometric variables, defining them both in three and two dimensions. The reflection and refraction laws and the phenomenon of total internal reflection are explained next. The last paragraph of the chapter gives a brief introduction to Fresnel reflection.

2.1 Radiometric and photometric variables

Radiometry concerns with the measurement of electromagnetic radiation across the entire electromagnetic spectrum. Photometry is the subfield of radiometry that takes into account only part of the electromagnetic spectrum corresponding to the visible light [42]. Radiometry deals with radiometric quantities. An important radiometric quantity is the radiant flux Φ_r (unit watt W) which is the total energy emitted from a source or received by a target per unit time:

$$\Phi_r = \frac{dQ}{d\tau}, \quad (2.1.1)$$

where Q is the energy and τ the time.

In illumination optics the measurement of light is given in terms of the impression that it gives on the human eye. Therefore, illumination optics deals with photometric variables rather than with radiometric variables. The most important photometric variables are defined in the following using the notation adopted by Chaves in [43]. The luminous flux Φ (unit lumen, lm) is defined as the *perceived* power of light by the human eye. The radiant and the luminous flux are related by the luminous efficacy function y (unit lm/W) which defines how many lumen correspond to one Watt of power at a given wavelength. The luminous efficacy function reaches its maximum at a wavelength of 555 nm where it is equal to 683 lm/W. We may normalize the luminous efficacy function with its maximum value of 683. The normalized function $\bar{y}(\lambda)$ is the luminosity function shown in Figure 2.1 where λ is the wavelength. It is a *dimensionless* quantity with a range between 0 and 1 [44].

The luminous flux corresponding to one Watt of radiation power at any wavelength

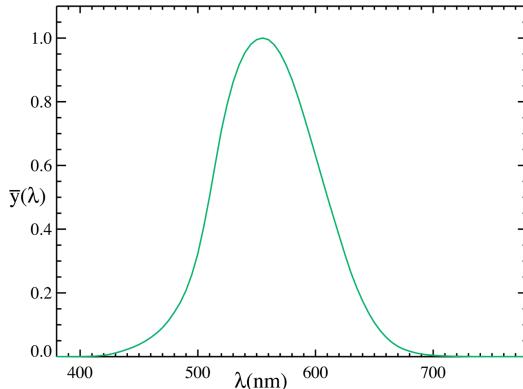


Figure 2.1: **Luminosity function** $\bar{y}(\lambda)$. Relation between the eye's sensitivity and the wavelength of light. The luminosity function is dimensionless [45].

is given by the product of 683 lm/W and the luminosity function at the same wavelength, i.e., $683 \bar{y}(\lambda)$. Hence, the total luminous flux Φ is defined as:

$$\Phi = 683 \int_0^\infty \Psi_r(\lambda) \bar{y}(\lambda) d\lambda, \quad (2.1.2)$$

where $\Psi_r(\lambda)$ is the spectral radiant flux, i.e., the power (in Watt) per unit wavelength (unit W/m). The corresponding photometric variable is the spectral luminous flux $\Psi(\lambda)$ (unit lm/m), i.e., the flux as perceived by the human eye as a function of the wavelength. Using the following relation between Ψ_r and Ψ :

$$683 \bar{y}(\lambda) = \frac{\Psi(\lambda)}{\Psi_r(\lambda)}, \quad (2.1.3)$$

equation (2.1.2) can be written as:

$$\Phi = \int_0^\infty \Psi(\lambda) d\lambda. \quad (2.1.4)$$

Geometric optics describes a beam of light as a collection of parallel light rays, where a light ray can be interpreted as a line or curve along which the electromagnetic energy travels. A ray is always direct perpendicular to the light's wavefront. The infinitesimal luminous flux $d\Phi$ incident on an infinitesimal surface dA is called illuminance E (unit lm/m²) and is defined as:

$$E = E(x) = \frac{d\Phi}{dA}. \quad (2.1.5)$$

The corresponding radiometric variable is called *irradiance*, we indicate it with P . The density of light emitted by a point source in a given direction depends on the solid angle.

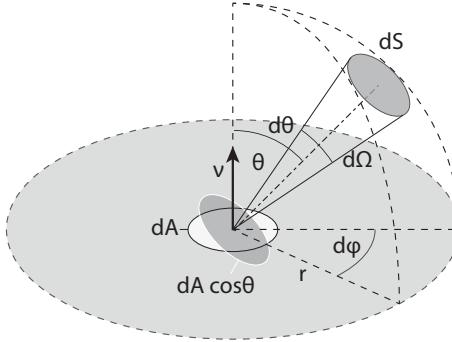


Figure 2.2: **Solid angle.** $d\Omega$ is in a given direction θ and φ .

The solid angle in a given direction is expressed by a cone of rays emitted in that particular direction by a point source located at the center of the unit sphere [6]. Let us now consider a finite source dA and let dS be the area on the unit sphere subtended by the cone, the infinitesimal solid angle $d\Omega$ is given by:

$$d\Omega = dS = \sin(\theta)d\theta d\varphi, \quad (2.1.6)$$

where θ and φ are the polar and the azimuthal angle, respectively, which the normal ν to the infinitesimal area dA makes with the direction of the central line of $d\Omega$ (see Figure 2.2). The solid angle on the entire unit sphere $\Omega = 4\pi$ and its unit is steradian (sr) [46]. The luminous intensity I (unit candela, $cd = lm/sr$) is defined as the luminous flux $d\Phi$ per solid angle $d\Omega$ and is given by:

$$I = I(\theta, \varphi) = \frac{d\Phi}{d\Omega}. \quad (2.1.7)$$

The luminance $L = L(\mathbf{x}, \theta, \varphi)$ (unit cd/m^2) depends both on the position and the direction, it is the luminous flux per unit solid angle $d\Omega$ and per unit projected area $\cos\theta dA$. L is given by:

$$L = L(\mathbf{x}, \theta, \varphi) = \frac{d\Phi}{\cos\theta dAd\Omega}. \quad (2.1.8)$$

Note that from (2.1.7) and (2.1.8) we can derive a relation between the intensity and the luminance. The intensity I emitted by the infinitesimal area dA is given by:

$$I = \frac{d\Phi}{d\Omega} = L(\mathbf{x}, \theta, \varphi) \cos\theta dA. \quad (2.1.9)$$

When the luminance is uniform over a finite area A , the luminous intensity emitted in the direction θ and φ is:

$$I(\theta, \varphi) = L(\theta, \varphi)A \cos\theta. \quad (2.1.10)$$

Further, when $L(\mathbf{x}, \theta, \varphi)$ does not depend on the position and the direction (i.e. $L(\mathbf{x}, \theta, \varphi) = L$), we obtain Lambert's cosine law:

$$I(\theta) = I_0 \cos\theta, \quad (2.1.11)$$

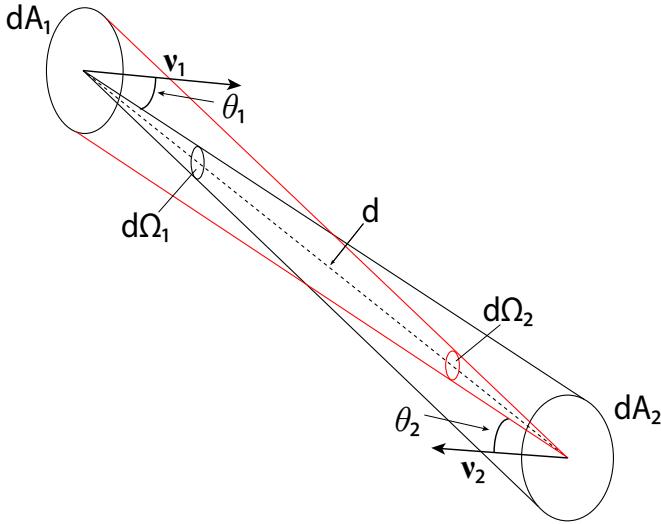


Figure 2.3: **Transfer of flux from the source dA_1 to the target dA_2 .** dA_1 and dA_2 are two surfaces with normals ν_1 and ν_2 , respectively. Their centers are located at a distance d . θ_1 and θ_2 are the angles made by the central ray with the normals ν_1 and ν_2 , respectively.

where $I_0 = I(\theta = 0) = LA$. Light sources emitting light with a constant luminance are called *Lambertian* sources.

Finally, we give a definition of the étendue U (unit $m^2\text{sr}$). Etendue is a french word that means extent or spread. In geometric optics, it is a quantity to describe how light is spread out in terms of area and solid angle [47, 48]. The quantity dU of a source dA is defined as:

$$dU = n^2 \frac{1}{L} d\Phi = n^2 \cos \theta dA d\Omega, \quad (2.1.12)$$

where n is the index of refraction of the medium in which dA is immersed. In phase space optics the étendue is considered to be a volume in phase space (or an area for two-dimensional systems). This concept will be clarified in Chapter 4 in which we treat the phase space in more detail. An important property is conservation of étendue within an optical system in the absence of absorption. In the following we show, using the approach of Chaves in [43], how conservation of étendue in a lossless system can be derived. Consider a light ray emitted from an infinitesimal area dA_1 to the area dA_2 . Suppose that the centers of dA_1 and dA_2 are located at a distance d from each other and that both the surfaces are located in the same medium, see Figure 2.3. We derive étendue conservation for the case in which dA_1 and dA_2 are located in the same medium (see [43, 6] for the general case). We indicate with ν_1 and ν_2 the normals on the surfaces dA_1 and dA_2 , and with θ_1 and θ_2 the angles that the ray connecting the centers of dA_1 and dA_2 forms with ν_1 and ν_2 . The infinitesimal solid angle $d\Omega_1$ subtended by dA_2 at the center of dA_1 and the flux $d\Phi_1$ passing

through dA_2 emitted from dA_1 are defined as:

$$d\Omega_1 = \frac{dA_2 \cos(\theta_2)}{d^2}, \quad (2.1.13a)$$

$$d\Phi_1 = L_1 \cos \theta_1 dA_1 d\Omega_1. \quad (2.1.13b)$$

Similarly, the infinitesimal solid angle $d\Omega_2$ subtended by dA_1 at the center of dA_2 and the flux $d\Phi_2$ passing through dA_1 emitted from dA_2 are equal to:

$$d\Omega_2 = \frac{dA_1 \cos \theta_1}{d^2}, \quad (2.1.14a)$$

$$d\Phi_2 = L_2 \cos \theta_2 dA_2 d\Omega_2. \quad (2.1.14b)$$

Then from equation (2.1.12) we obtain the following relations:

$$dU_1 = n^2 \cos \theta_1 dA_1 d\Omega_1 = \frac{n^2 \cos \theta_1 dA_1 dA_2 \cos \theta_2}{d^2}, \quad (2.1.15a)$$

$$dU_2 = n^2 \cos \theta_2 dA_2 d\Omega_2 = \frac{n^2 \cos \theta_2 dA_2 dA_1 \cos \theta_1}{d^2}, \quad (2.1.15b)$$

for dA_1 and dA_2 , respectively. From the previous equations we can conclude that $dU_1 = dU_2$ and therefore the étendue is conserved along a beam of light. Since the system is a lossless system energy is conserved from the source to the target ($d\Phi_1 = d\Phi_2$), therefore the étendue conservation implies:

$$L_1 = n^2 \frac{d\Phi_1}{dU_1} = n^2 \frac{d\Phi_2}{dU_2} = L_2, \quad (2.1.16)$$

where the first equality comes from (2.1.8) combined with (2.1.12).

In this thesis we consider two-dimensional optical systems. Hence, the definitions of the photometric parameters have to be adapted to two-dimensions. An infinitesimal line segment of length da emitting a ray that makes an angle θ with the normal ν are considered, see Figure 2.4. The two-dimensional illuminance (unit lm/m) denotes the luminous flux received by an infinitesimal line segment of length da and it is given by:

$$E = \frac{d\Phi}{da}. \quad (2.1.17)$$

The luminous intensity (unit [lm/rad]) is the luminous flux per angle $d\theta$:

$$I = \frac{d\Phi}{d\theta}. \quad (2.1.18)$$

The two-dimensional luminance (unit lm/(rad m)) is given by:

$$L = \frac{d\Phi}{\cos \theta da d\theta}. \quad (2.1.19)$$

Thus the following relation holds:

$$I = L(x, \theta) \cos \theta da, \quad (2.1.20)$$



Figure 2.4: **Ray emitted by an infinitesimal line segment.** da makes an angle θ with respect to the line normal ν .

Table 2.1: **Photometric variables**

Name	Symbol	Unit (3D)	Unit (2D)
Luminous flux	Φ	lm	lm
Illuminance/emittance	E	lm/m^2	lm/m
Intensity	I	$\text{lm}/\text{sr} = \text{cd}$	lm/rad
Luminance	L	cd/m^2	$\text{lm}/(\text{rad m})$
Etendue	U	$\text{m}^2 \text{sr}$	m rad

where x is a certain position at the light source da . Finally, the étendue dU (unit m rad) in two-dimensions is given by:

$$dU = n \cos \theta da d\theta. \quad (2.1.21)$$

An overview of the photometric variables used in this thesis is given in Table 2.1

In order to determine the light distribution on a surface and to compute the photometric variables on that surface, we need to understand how the light emitted from a source propagates. In the field of geometric optics the light propagation is described by light rays. The propagation of a light ray traveling through different media is determined by the reflection and refraction law. In the following we introduce these two laws and we explain the total internal reflection phenomenon.

2.2 Reflection and refraction law

A light ray is described by a position vector \mathbf{x} on a surface and a direction vector \mathbf{t} and can be parameterized by the arc length s . Light rays travel in a homogeneous medium along straight lines, once they hit a reflective surface their direction changes. Denoting with \mathbf{t}_i the direction of the incident ray and with ν the unit normal to the



Figure 2.5: **Propagation of a ray.** The ray travels through two materials with index of refraction n_i and n_t .

surface at the location of incidence, the direction t_r of the reflected ray is given by:

$$t_r = t_i - 2(t_i \cdot \nu)\nu, \quad (2.2.1)$$

where the vectors t_i and ν are unit vectors and $t_i \cdot \nu$ indicates the scalar product of t_i and ν . From the previous equation it follows that the vector t_r is a unit vector too, indeed considering the scalar product $t_r \cdot t_r$ we conclude:

$$t_r \cdot t_r = t_i \cdot t_i - 4(t_i \cdot \nu)(t_i \cdot \nu) + 4(t_i \cdot \nu)^2(\nu \cdot \nu) = 1. \quad (2.2.2)$$

Note, from (2.2.1) that the vectors t_i , t_r and ν are coplanar. Indicating with θ_i the incident angle and with θ_r the reflective angle such that $\theta_i, \theta_r \in [0, \pi/2)$, the reflection law states that $\theta_i = \theta_r$, see Figure 2.5.

When a ray propagates through two different media, its direction changes according to the law of refraction. Indicating with n_i the index of refraction of the medium in which the incident ray travels and with n_t the index of refraction of the medium of the transmitted ray, the direction of the transmitted ray is given by:

$$t_t = n_{i,t} t_i - \left[\sqrt{1 - n_{i,t}^2 + n_{i,t}^2(\nu \cdot t_i)^2} + n_{i,t}(\nu \cdot t_i) \right] \nu, \quad (2.2.3)$$

where $n_{i,t} = n_i/n_t$ [43]. While the direction of the normal ν to the surface is not relevant for the computation of the direction of the reflected ray, in fact:

$$t_r = t_i - 2(t_i \cdot \nu)\nu = t_i - 2(t_i \cdot (-\nu))(-\nu), \quad (2.2.4)$$

for computing the direction of the refracted ray, we need to specify the direction of ν which is usually chosen so that the scalar product between the incident ray and

the normal is negative $\mathbf{t}_i \cdot \boldsymbol{\nu} < 0$. Considering the cross product of both the terms in Equation (2.2.3) with the normal $\boldsymbol{\nu}$ of the incident surface we obtain:

$$\mathbf{t}_t \times \boldsymbol{\nu} = n_{i,t} (\mathbf{t}_i \times \boldsymbol{\nu}), \quad (2.2.5)$$

which leads to the Snell's law:

$$n_t \sin(\theta_t) = n_i \sin(\theta_i). \quad (2.2.6)$$

Note that (2.2.3) is only valid for

$$1 - n_{i,t}^2 + n_{i,t}^2 (\boldsymbol{\nu} \cdot \mathbf{t}_i)^2 \geq 0, \quad (2.2.7)$$

which implies that

$$\frac{n_t}{n_i} \geq \sqrt{1 - (\boldsymbol{\nu} \cdot \mathbf{t}_i)^2}, \quad (2.2.8)$$

from which we obtain:

$$n_t \geq n_i \sin \theta_i. \quad (2.2.9)$$

The angle θ_c for which the equality holds is

$$\theta_c = \arcsin \left(\frac{n_t}{n_i} \right), \quad (2.2.10)$$

and it is called the critical angle [43]. When the incident angle θ_i is exactly equal to the critical angle θ_c , the square root in (2.2.3) is zero and $\mathbf{t}_t \cdot \boldsymbol{\nu} = 0$, hence the transmitted ray propagates parallel to the refractive surface. When $\theta_i > \theta_c$ the light ray is no longer refracted but is only reflected by the surface. This phenomenon is called total internal reflection (TIR). When TIR occurs, 100% of the incident light is reflected and there is no refraction. Therefore, optical systems designed such that rays are reflected by TIR are very efficient.

In general, light that hits an ordinary refractive surface can be both reflected and refracted. Every incident ray generates two rays when interacting with a surface. Each of them carries a fraction of the total energy of the incident ray. Obviously, the sum of the reflected and transmitted energy equals the incident power. The amount of energy transported by the reflected and the refracted ray is determined by the Fresnel equations. In the next paragraph an overview of the Fresnel coefficients is given based on the theory in [39] Section 4.6.

2.3 Fresnel's coefficients for reflection and refraction

In order to derive Fresnel's equations we need to describe light as an electromagnetic wave. It is therefore useful to study light propagation from the perspective of electromagnetic theory which gives information about the incident, reflected and transmitted radiant flux density, denoted with P_i , P_r and P_t , respectively. The electric field \mathcal{E} can be written as:

$$\mathcal{E}(\mathbf{x}, \tau) = \mathcal{E}_0(\mathbf{x}) e^{i(k \cdot \mathbf{x} - \omega \tau)}, \quad (2.3.1)$$

where \mathbf{k} is the vector in the direction of the field propagation with modulus $|\mathbf{k}| = k$, \mathbf{x} is the position vector and τ is the time. The amplitude $\mathcal{E}_0(\mathbf{x})$ is constant in time and $\omega = ck/n$ is the value of the angular frequency with c the velocity of light in vacuum and n the index of refraction in which the wave is traveling. We recall that $n = c/v$ with v the speed of light in the material. Note that the angular frequency can be also written as $\omega = v k$ (in vacuum $n = 1$ and $\omega = ck$). The parameter $k = 2\pi/\lambda$ is the wave number in vacuum, with λ the wavelength.

Similarly, the magnetic field has the form:

$$\mathcal{B}(\mathbf{x}, \tau) = \mathcal{B}_0(\mathbf{x})e^{i(\mathbf{k}\cdot\mathbf{x}-\omega\tau)}, \quad (2.3.2)$$

where $\mathcal{B}_0(\mathbf{x})$ is the amplitude of the magnetic field. The electric and magnetic fields satisfy the following relations:

$$\frac{\mathbf{k}}{k} \times \mathcal{E} = v \mathcal{B}, \quad (2.3.3a)$$

$$\frac{\mathbf{k}}{k} \cdot \mathcal{E} = 0. \quad (2.3.3b)$$

Light can be considered as an electromagnetic wave, consisting of an electric field \mathcal{E} and a magnetic field \mathcal{B} , which propagates always perpendicular to \mathcal{E} and \mathcal{B} (see Equations (2.3.3)). By convention, the direction of the electric field \mathcal{E} with respect to the incident plane defines the *polarization* of an electromagnetic wave [41]. The direction of \mathcal{E} is given by the incident and reflected rays as is shown in Figure 2.6.

Fresnel's equations were introduced to describe the effect of an incident wave when encountering an interface located between two media having different indexes of refraction. In particular, the Fresnel coefficients determine the fractions of transmitted and reflected energy. In the following we provide the Fresnel coefficients and we briefly explain their physical interpretation. We refer the reader to [39, 40, 13] for more details. To derive Fresnel's equations the polarization of light must be taken into account.

Light is said to be *polarized* if the electric field oscillates in a single plane. Light is *unpolarized* when the direction of this electric field changes randomly in time. The light polarization can be classified into three different kinds of polarization:

- *Linear polarization*: The electric field is confined to a single plane perpendicular to the direction of propagation;
- *Circular polarization*: The electric field describes a circle around the direction of propagation;
- *Elliptic polarization*: The electric field describes an ellipse around the direction of propagation.

Any form of light can be defined by two orthogonal linear polarizations. Hence, the following two cases of light polarization are considered:

1. \mathcal{E} is perpendicular to the plane of incidence, and therefore \mathcal{B} is parallel to it (see Figure 2.7). In this case light is said to be *s-polarized* (from the German word *senkrecht*).



Figure 2.6: **Light ray that hits a mirror located on the reflecting plane.** The incident and the reflected ray are coplanar with the normal to the mirror. The plane of incidence is spanned by the reflected and the refracted rays. The plane of interface is perpendicular to the plane of incidence.

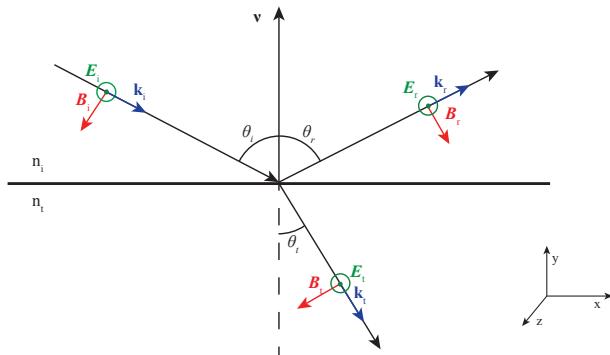


Figure 2.7: **Propagation of an electromagnetic wave for s-polarized light.** The components of \mathcal{E} are indicated with the green circles. The components of \mathcal{B} are indicated with red arrows.

2. \mathcal{E} is parallel to the plane of incidence and, therefore \mathcal{B} is perpendicular to it (see Figure 2.8). In this case light is said to be *p-polarized* (from the German word *parallel*).

The amplitude Fresnel coefficient r is defined by the ratio between the amplitudes of the reflected and the incident electric field. Similarly, t is the ratio between the amplitudes of the transmitted and the incident electric field. They are defined by:

$$r = \frac{|\mathcal{E}_{0r}|}{|\mathcal{E}_{0i}|}, \quad t = \frac{|\mathcal{E}_{0t}|}{|\mathcal{E}_{0i}|}. \quad (2.3.4)$$

The Fresnel equations provide the perpendicular r_s and t_s and parallel r_p and t_p versions of r and t , respectively. Energy conservation gives the boundary conditions of the electromagnetic field at the plane of the interface (perpendicular to the incident

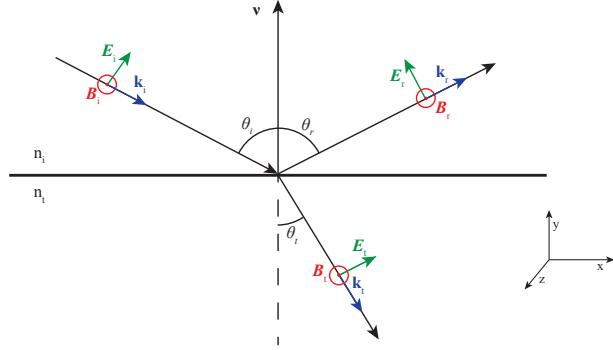


Figure 2.8: **Propagation of an electromagnetic wave for p-polarized light.** The components of \mathbf{B} are indicated with the red circle. The components of \mathbf{E} are indicated with green arrows.

plane), from which the Fresnel coefficients are derived. They are defined in both case 1 and case 2. Since the mathematical formulation is similar for the two cases, in the following we explain in detail the computation of the Fresnel coefficients only for s-polarized light (case 1).

For s-polarized light, the component tangential to the interface of \mathbf{E} and \mathbf{B}/μ are continuous across the boundary between the two different media, where μ is the permeability of the material in which the electromagnetic field is located. From now on, all quantities defined in the medium of the incident, reflective and transmitted light are indicated with the subscripts i, r and t, respectively. The continuity of the tangential component of \mathbf{E} leads to:

$$|\mathbf{E}_{0i}| + |\mathbf{E}_{0r}| = |\mathbf{E}_{0t}|, \quad (2.3.5)$$

while the continuity of the tangential component of \mathbf{B}/μ gives:

$$-\frac{|\mathbf{B}_{0i}|}{\mu_i} \cos \theta_i + \frac{|\mathbf{B}_{0r}|}{\mu_r} \cos \theta_r = -\frac{|\mathbf{B}_{0t}|}{\mu_t} \cos \theta_t, \quad (2.3.6)$$

where the sign convention is chosen as illustrated in Figure 2.7. Since $|\mathbf{B}| = |\mathbf{E}|/v$, (2.3.6) can be written as

$$\frac{1}{\mu_i v_i} (|\mathbf{E}_{0i}| - |\mathbf{E}_{0r}|) \cos \theta_i = \frac{1}{\mu_t v_t} |\mathbf{E}_{0t}| \cos \theta_t, \quad (2.3.7)$$

where we employed the fact that $v_i = v_r$, and $\theta_i = \theta_r$. Since $n = c/v$, the previous equation becomes:

$$\frac{n_i}{\mu_i} (|\mathbf{E}_{0i}| - |\mathbf{E}_{0r}|) \cos \theta_i = \frac{n_t}{\mu_i} |\mathbf{E}_{0t}| \cos \theta_t. \quad (2.3.8)$$

Most often used optical materials are non-magnetic, hence we assume $\mu_i = \mu_t = \mu_0$ [49]. Employing (2.3.5) we arriving at the Fresnel coefficients for s-polarized light:

$$\begin{aligned} r_s &= \frac{|\mathcal{E}_{0r}|_s}{|\mathcal{E}_{0i}|_s} = \frac{n_i \cos \theta_i - n_t \cos \theta_t}{n_i \cos \theta_i + n_t \cos \theta_t}, \\ t_s &= \frac{|\mathcal{E}_{0t}|_s}{|\mathcal{E}_{0i}|_s} = \frac{2 n_i \cos \theta_i}{n_i \cos \theta_i + n_t \cos \theta_t}, \end{aligned} \quad (2.3.9)$$

where the subscript s is used to remind the reader that we are considering s-polarized light. Using Snell's law (Equation (2.2.6)) the first pair of the Fresnel equations restricted to the case $\theta_i \leq \theta_c$ becomes:

$$\begin{aligned} r_s &= -\frac{\sin(\theta_i - \theta_t)}{\sin(\theta_i + \theta_t)} \\ t_s &= \frac{2 \sin \theta_t \cos \theta_i}{\sin(\theta_i + \theta_t)} \end{aligned} \quad (2.3.10)$$

A similar argument for p-polarized light leads to the calculation of the parallel versions r_p and t_p of r and t . The boundary conditions for the electric and magnetic field read:

$$|\mathcal{E}_{0i}| \cos \theta_i - |\mathcal{E}_{0r}| \cos \theta_r = |\mathcal{E}_{0t}| \cos \theta_t \quad (2.3.11a)$$

$$|\mathcal{B}_{0i}| + |\mathcal{B}_{0r}| = |\mathcal{B}_{0t}|, \quad (2.3.11b)$$

leading to the second pair of Fresnel's equations:

$$\begin{aligned} r_p &= \frac{|\mathcal{E}_{0r}|_p}{|\mathcal{E}_{0i}|_p} = \frac{n_t \cos \theta_i - n_i \cos \theta_t}{n_i \cos \theta_i + n_t \cos \theta_t}, \\ t_p &= \frac{|\mathcal{E}_{0t}|_p}{|\mathcal{E}_{0i}|_p} = \frac{2 n_i \cos \theta_i}{n_i \cos \theta_i + n_t \cos \theta_t}. \end{aligned} \quad (2.3.12)$$

Similarly to the s-polarized case, using Snell's law, the previous equations when $\theta_i \leq \theta_c$ reduce to:

$$\begin{aligned} r_p &= \frac{\tan(\theta_i - \theta_t)}{\tan(\theta_i + \theta_t)} \\ t_p &= \frac{2 \sin \theta_t \cos \theta_i}{\sin(\theta_i + \theta_t) \cos(\theta_i - \theta_t)} \end{aligned} \quad (2.3.13)$$

Furthermore, it can be checked that

$$t_s - r_s = 1 \quad (2.3.14)$$

holds for every θ_i , while

$$t_p + r_p = 1 \quad (2.3.15)$$

is valid only at normal incidence ($\theta_i = 0$).

The amplitude coefficients are shown in Figure 2.9 for the case in which light

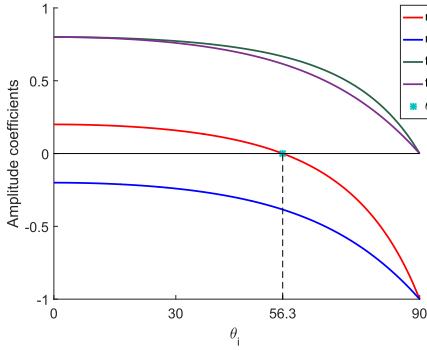


Figure 2.9: **Amplitude coefficients of reflection and transmission.** $n_i < n_t$ ($n_i = 1$ and $n_t = 1.5$). $\theta_p = 56.3^\circ$ is the polarization angle.



Figure 2.10: **Amplitude coefficients.** $n_i > n_t$ ($n_i = 1.5$ and $n_t = 1$). $\theta_{p'} = 33.7^\circ$ is the polarization angle and $\theta_c = 41.8^\circ$ is the critical angle.

travels from a less dense to a more dense medium, i.e., $n_i < n_t$ where $n_i = 1$ and $n_t = 1.5$, thus $i > \theta_t$. In Figure 2.10 the amplitude coefficients are shown for the case in which $n_i > n_t$ with $n_i = 1.5$ and $n_t = 1$ ($\theta_i < \theta_t$). Note, from Figure 2.9 that r_p approaches 0 when θ_i approaches θ_p and it gradually decreases, reaching -1 for an incident angle $\theta_i = 90^\circ$. The angle θ_p is called *Brewster's angle* or polarization angle as only the component perpendicular to the incident plane is reflected at that angle and therefore light is perfectly polarized. Similarly, Figure 2.10 shows that $r_p = 0$ for $\theta_i = \theta_{p'}$. It can be shown that $\theta_p + \theta_{p'} = 90^\circ$. Both r_p and r_s reach 1 when $\theta_i = \theta_c$. θ_c is called the critical angle. Light that hits the incident plane with an incident angle equal to or greater than θ_c is totally reflected back and no transmitted light is observed. This phenomenon is called total internal reflection.

To calculate the amount of energy reflected and transmitted, it is necessary to define the reflection and transmission coefficients in terms of the irradiance (or radiant flux density) P rather than in terms of amplitudes of the electric field. For a wave of amplitude $|\mathcal{E}_0|$ propagating in a non-magnetic medium with a refractive index n , the energy flux P is given by:

$$P = \frac{cn\varepsilon_0}{2} |\mathcal{E}_0|^2, \quad (2.3.16)$$

[39]. For a beam of light that hits a surface such that an area A is illuminated, P is the average energy that crosses in unit time a unit area A perpendicular to the direction of the energy flow. We indicate with P_i , P_r and P_t the incident, reflected and transmitted flux densities, respectively. The energy per unit time for the incident, reflected and transmitted beams are $P_i A \cos \theta_i$, $P_r A \cos \theta_r$ and $P_t A \cos \theta_t$, respectively. The reflectance \mathcal{R} is the ratio of the reflected power to the incident power:

$$\mathcal{R} = \frac{P_r \cos \theta_r}{P_i \cos \theta_i} = \frac{|\mathcal{E}_{0r}|^2}{|\mathcal{E}_{0i}|^2} = r^2, \quad (2.3.17)$$

where the second equality holds because $n_i = n_r$, $\varepsilon_i = \varepsilon_r$ and $\theta_i = \theta_r$. Similarly, the

transmittance \mathcal{T} is the ratio between the transmitted to the incident power:

$$\mathcal{T} = \frac{P_t \cos \theta_t}{P_i \cos \theta_i} = \frac{n_t \cos \theta_t}{n_i \cos \theta_i} \frac{|\mathcal{E}_{0t}|^2}{|\mathcal{E}_{0i}|^2} = \frac{n_t \cos \theta_t}{n_i \cos \theta_i} t^2. \quad (2.3.18)$$

Employing total energy conservation, that is:

$$P_i A \cos \theta_i = P_r A \cos \theta_r + P_t A \cos \theta_t, \quad (2.3.19)$$

we can easily prove that:

$$\mathcal{R} + \mathcal{T} = 1. \quad (2.3.20)$$

The perpendicular and parallel versions of \mathcal{R} and \mathcal{T} are given by:

$$\begin{aligned} \mathcal{R}_s &= r_s^2 \\ \mathcal{T}_s &= \frac{n_t \cos \theta_t}{n_i \cos \theta_i} t_s^2 \\ \mathcal{R}_p &= r_p^2 \\ \mathcal{T}_p &= \frac{n_t \cos \theta_t}{n_i \cos \theta_i} t_p^2 \end{aligned} \quad (2.3.21)$$

It can be shown that

$$\begin{aligned} \mathcal{R}_s + \mathcal{T}_s &= 1, \\ \mathcal{R}_p + \mathcal{T}_p &= 1. \end{aligned} \quad (2.3.22)$$

For normal incidence, i.e., $\theta_i = 0$, the incident plane is not defined and there is no distinction between the perpendicular and the parallel versions of \mathcal{R} and \mathcal{T} . As a consequence, (2.3.21) combined with (2.3.9) and (2.3.12) leads to:

$$\begin{aligned} \mathcal{R} &= \mathcal{R}_p = \mathcal{R}_s = \left(\frac{n_i - n_t}{n_t + n_i} \right)^2, \\ \mathcal{T} &= \mathcal{T}_p = \mathcal{T}_s = \frac{4n_i n_t}{(n_t + n_i)^2}. \end{aligned} \quad (2.3.23)$$

Many common light sources such as the sun, halogen lighting, LED spotlights, and incandescent bulbs produce unpolarized light. In this case, light can be represented by the sum of two orthogonal states. For unpolarized light, the convention is to take as reflectance and transmittance the averages:

$$\begin{aligned} \mathcal{R} &= \frac{\mathcal{R}_p + \mathcal{R}_s}{2} \\ \mathcal{T} &= \frac{\mathcal{T}_p + \mathcal{T}_s}{2} \end{aligned} \quad (2.3.24)$$

where \mathcal{R}_p , \mathcal{R}_s , \mathcal{T}_p and \mathcal{T}_s are obtained from (2.3.21). The reflectance \mathcal{R} and the transmittance \mathcal{T} give the fraction of the power of energy reflected and transmitted at every interaction of a light ray with a Fresnel surface.

With this overview we conclude this chapter. The notions given in Section 2.1 will be used in the entire thesis as our goal is to study the distribution of light at the target of some optical systems. In particular we will focus on the computation of the output intensity distribution. The reflection and refraction laws explained in Section 2.2 are needed to determine how the optical system changes the ray's direction every time that it hits a surface (or a line in the two-dimensional case). In Chapters 3-8 we restrict ourselves to pure reflection and refraction; the Fresnel reflection and refraction are ignored. Systems with Fresnel reflection is postponed until Chapter 9. The amount of reflected and transmitted light is calculated using (2.3.24) and (2.3.21) where r_s and t_s are obtained from (2.3.9) and r_p and t_p from (2.3.12). In our simulations we consider the general case of unpolarized light.

Chapter 3

Ray tracing

Ray tracing is a tool used in geometric optics to calculate the light distribution at the target of optical systems. Given an optical system and a set of rays at the source, ray tracing relates the emitted light with its output distribution.

Although the method can be implemented for two or three dimensions and for very complex system, here we consider the two-dimensional case only. We will thus refer to optical lines instead of optical surfaces. The two-dimensional case has some limitations. For example, it may not identify skew rays that is rays that do not propagate in the plane that contains both the optical axis and the point from which they are originated (*meridional plane*). As a consequence, the 2D analysis cannot guarantee a proper treatment of non-meridional rays in 3D. Nevertheless, the two-dimensional case is particularly relevant because it is a good test case to demonstrate the performance of new methods. Optical designers often start with 2D systems, where only the meridional plane is taken into account because it gives a good prediction of the target distribution in 3D (see [50], Chapter 4).

3.1 Ray tracing for two-dimensional optical systems

The purpose of ray tracing applied to non-imaging optical systems is to calculate the target rays distribution given an optical system and an initial distribution of the rays at the source. Light rays are straight lines and they are reflected or refracted by the optical components of a system. Every ray emitted from the source is followed until it reaches the target. The ray tracing procedure is constructed such that the position and the direction of the rays are calculated on every optical line that they cross.

Given a Cartesian coordinate system (x, z) , a two-dimensional optical system symmetric with respect to the z -axis is defined. Hence, we assume that the optical axis coincides with the z -axis. The optical system is formed by a source S , a target T and some optical components labeled with indices j where $j \in \{2, \dots, N_l - 1\}$ and N_l indicates the number of lines that form the system. S and T are indicated with the indices 1 and N_l , respectively. Every line separates two materials with different indices of refraction. We indicate with n_j the index of refraction of the medium in which the ray incident line j travels. Every ray emitted by S (line 1) can either go straight to the target T or hit some optical components $j \in \{2, \dots, N_l - 1\}$ before



Figure 3.1: **Shape of the two-faceted cup.** Each line of the system is labeled with a number. The source $S = [-2, 2]$ (line number 1) is located on the x -axis. The target $T = [-17, 17]$ (line 4) is parallel to the source and is located at a height $z = 40$. The left and right reflectors (line 2 and 3) connect the source with the target.

reaching T (line N_l). The intersection point of the rays with line j are $(x_j, z_j)_{j=1, \dots, N_l}$ and, $\mathbf{s}_j = (-\sin t_j, \cos t_j)$ indicates the direction vector of the rays that leave j , where t_j is the angle that the ray forms with respect to the optical axis, measured counter-clockwise. As we consider only forward rays, the angles $t_j \in (-\pi/2, \pi/2)$. Therefore, a ray segment between (x_j, z_j) and (x_k, z_k) with $k \neq j$ is parameterized in real space by:

$$\mathbf{r}(s) = \begin{pmatrix} x_j - s \sin t_j \\ z_j + s \cos t_j \end{pmatrix} \quad 0 < s \leq s_{\max}, \quad (3.1.1)$$

where s denotes the arc-length and s_{\max} is the maximum value that it can assume. Figure 3.1 shows an example where a single ray is traced inside a very simple optical system, the so-called two-faceted cup. The light source $S = [-a, a]$ (line 1) and the target $T = [-b, b]$ (line 4) are two segments normal to the z -axis, where $a = 2$ and $b = 17$. The left and right reflectors (line 2 and 3) are oblique segments that connect the source and the target. All the optical lines $j \in \{1, \dots, 4\}$ are located in air, thus the refractive index is $n_j = 1$ for every j .

In order to compute the target photometric variables, we need to know how the optical system influences the direction of the rays when they hit an optical line. Ray tracing relates the position coordinates (x_1, z_1) and the direction vector \mathbf{s}_1 of every ray at the source S with the corresponding position (x_{N_l}, z_{N_l}) and direction \mathbf{s}_{N_l} at the target T . In the following we will often use the target coordinates of the rays thus, to simplify the notation, we do not write the subscript N_l for the target coordinates. We rather write (x, z) instead of (x_{N_l}, z_{N_l}) , t instead of t_{N_l} and \mathbf{s} instead of \mathbf{s}_{N_l} for the target coordinates. The ray tracing algorithm can be outlined as follows:

1. Indicate with i the index of the line from which rays leave. Consider a ray that leaves the source S (line $i=1$);
2. Consider a ray with position coordinates (x_i, z_i) and direction $\mathbf{s}_i = (-\sin t_i, \cos t_i)$, use (3.1.1) to implement the ray parametrization $\mathbf{r}(s)$;

3. Compute the coordinates $(x_k, z_k)_{k=2, \dots, Nl}$ of the intersection points of the parameterized ray $\mathbf{r}(s)$ with all the lines that it crosses
 - a) if the shape of the lines is described by an explicit equation, the intersection points are determined analytically;
 - b) if there is no analytic description for the optical lines, the intersections need to be determined using iterative methods;
4. Determine the point (x_j, z_j) closest to (x_i, z_i) with $s > 0$ where s is the arc length of the corresponding ray.
5. If $j = Nl$, stop the procedure, the target ray's coordinates (x, z) and s are found.
6. If $j \neq Nl$, calculate the normal ν_j to line j at the point (x_j, z_j) ;
7. Compute the new ray direction \mathbf{s}_j of the ray that leaves line j at the point (x_j, z_j) :
 - a) if the line the ray hits is a reflective line, \mathbf{s}_j is given by (2.2.1);
 - b) if the line the ray hits is a refractive line, \mathbf{s}_j is given by (2.2.3);
8. Put $i = j$ and restart the procedure from 2.

The procedure explained above is repeated for every ray traced through the system [15]. Once the target position and the direction of every ray are computed, also the target photometric variables can be calculated using the definitions explained in the previous chapter, see Section 2.1.

There are different ways to implement the ray tracing procedure. The efficiency of ray tracing can be related to the distribution of the rays at the source. In Monte Carlo (MC) ray tracing the initial position and direction of the rays are chosen randomly. This is a very common method in non-imaging optics as it is very powerful and easy to implement. MC ray tracing will be explained in the next section. In Quasi-Monte Carlo (QMC) ray tracing the rays are chosen from a so-called low discrepancy sequence. This method is discussed in Section 3.3.

3.2 Monte Carlo (MC) ray tracing

MC *methods* rely on stochastic random samples to obtain numerical results for a deterministic problem. In particular, MC simulations can be a very easy way to solve physics problems based on numerical integration [51]. *Ray tracing* provides a light distribution at the target given an initial light distribution at the source and an optical system. MC *ray tracing* uses probabilistic random processes to implement ray tracing for illumination systems [52]. We can think of a light source as emitting a very large number of random rays and keep track of where they go during their propagation inside the optical system. Applying ray tracing to the initial random set of rays, a random rays distribution at the target of the system is obtained. From this distribution, we can calculate, for instance, the profile of the target intensity using MC approximation.

In the two-dimensional MC ray tracing, for every ray we need to choose one position coordinate x_1 and one angular coordinate t_1 , while the z_1 -coordinate of every

ray is always given (for instance, for the two-faceted cup in Figure 3.1, $z_1 = 0$). We indicate with x_1^i the x -coordinate of the i -th ray and with t_1^i its angular coordinate. A set of random variables $\{\mathbf{Y}_1, \dots, \mathbf{Y}_N\}$ is chosen such that they are uniformly distributed at the source, where $\mathbf{Y}_i = (x_1^i, t_1^i)$. Rays with those random initial coordinates are traced from S to T and, the corresponding random variables $\mathbf{Z}_i = (x^i, t^i)$ at the target are obtained. Note that the rays at the target are most likely not uniformly distributed. Once the target variables are computed using ray tracing, we can approximate the target intensity $I(t)$. Since the intensity only depends on the angular coordinate t we use MC integration in one dimension. The details are explained next.

The range of angular coordinates at the target is divided into intervals of equal length, the so-called bins and the number of rays that fall into each bin is determined. Thus, a partitioning $P_1 : -\pi/2 = t_0 < t_1 < \dots < t_{Nb} = \pi/2$ of the interval $[-\pi/2, \pi/2]$ is defined where Nb is the number of bins in P_1 . We remark that, with a slight abuse of notation, we indicated the angular coordinates of the rays at the target (line NL) with t_j instead of $t_{NL,j}$ for every $j \in \{0, \dots, Nb\}$. The normalized approximated intensity $\hat{I}_{MC}(t)$ is a piecewise constant function, whose value over the j -th bin is the ratio of the number of rays that fall into that bin, $Nr[t_{j-1}, t_j]$, and the total number of rays traced, $Nr[-\pi/2, \pi/2]$, i.e.:

$$\hat{I}_{MC}(t) = \frac{Nr[t_{j-1}, t_j]}{Nr[-\pi/2, \pi/2]} \quad \text{for } t \in [t_{j-1}, t_j]_{j=1, \dots, Nb}. \quad (3.2.1)$$

The output intensity is computed from the value of the intensity $\hat{I}_{MC}(t_{j-1/2})$ along the direction $t_{j-1/2} = (t_{j-1} + t_j)/2$ for every bin $[t_{j-1}, t_j]_{j=1, \dots, Nb}$. The intensity $\hat{I}_{MC}(t_{j-1/2})$ gives an estimate of the probability that a ray reaches the target with an angle in the j -th interval $[t_{j-1}, t_j]$. This probability $P_{j, \Delta t}$ is given by

$$P_{j, \Delta t} = \Pr(t_{j-1} \leq t < t_j) = \frac{\int_{t_{j-1}}^{t_j} I(t) dt}{\int_{-\pi/2}^{\pi/2} I(t) dt}, \quad (3.2.2)$$

where $I(t)$ is the output intensity (measured in lumen per radians, lm/rad). Note that $\sum_{j=1}^{Nb} P_{j, \Delta t} = 1$. From the mean value theorem applied to the function $I(t)$, which is continuous in $[t_{j-1}, t_j]$, there exists a value $\tau_j \in [t_{j-1}, t_j]$ for which:

$$\int_{t_{j-1}}^{t_j} I(t) dt = \Delta t I(\tau_j). \quad (3.2.3)$$

Hence, $P_{j, \Delta t}$ is proportional to the size $\Delta t = (t_{Nb} - t_0)/Nb$ of the bins and to $I(\tau_j)$. Although $I(\tau_j)$ does depend on the number of bins Nb, it is taken constant over $[t_{j-1}, t_j]$ as it is the value of the intensity for a given direction¹, so (3.2.3) indicates that $P_{j, \Delta t}$ is inversely proportional to the number of bins Nb of the partitioning P_1 . Indicating with $\Phi = \int_{-\pi/2}^{\pi/2} I(t) dt$ the total flux (measured in lumen, lm), the error between the intensity $I(t_{j-1/2})$ and the averaged MC intensity $\Phi \hat{I}_{MC}(t_{j-1/2})/\Delta t$ (not

¹This is possible since $I(\tau_j)$ is a continuous function in the closed interval $[t_{j-1}, t_j]$.

normalized) is given by

$$\begin{aligned} \left| I(t_{j-1/2}) - \frac{\Phi}{\Delta t} \hat{I}_{\text{MC}}(t_{j-1/2}) \right| &\leq \\ \left| I(t_{j-1/2}) - \frac{1}{\Delta t} \int_{t_{j-1}}^{t_j} I(t) dt \right| &+ \\ \frac{1}{\Delta t} \left| \int_{t_{j-1}}^{t_j} I(t) dt - \Phi \hat{I}_{\text{MC}}(t_{j-1/2}) \right|. \end{aligned} \quad (3.2.4)$$

The first term on the right hand side of (3.2.4) gives an estimate of how much the averaged intensity $\frac{1}{\Delta t} \int_{t_{j-1}}^{t_j} I(t) dt$ differs from the exact intensity $I(t_{j-1/2})$. This term is due to the discretization of the target and therefore depends on the number of bins Nb considered. Substituting the first order Taylor expansion of $I(t)$ around the point $t_{j-1/2}$ we obtain:

$$\left| I(t_{j-1/2}) - \frac{1}{\Delta t} \int_{t_{j-1}}^{t_j} I(t) dt \right| = C_1 / \text{Nb}^2 \quad (3.2.5)$$

with $C_1 > 0$ a certain constant.

The second part on the right hand side of inequality (3.2.4) gives an estimate of the statistical MC error and therefore depends also on the number of rays traced. In order to show how this term decreases as a function of the number of rays traced, we define the characteristic function $\chi_j(t)$ of the random variable t :

$$\chi_j(t) = \begin{cases} 1 & \text{if } t \in [t_{j-1}, t_j], \\ 0 & \text{otherwise.} \end{cases} \quad (3.2.6)$$

Since t is a random variable, $\chi_j(t)$ is still a random variable. The Bernoulli trial χ_j follows a binomial distribution $B(1, P_{j,\Delta t})$. Considering a sample of Nr rays, the variable $Y_j = \sum_{k=1}^{\text{Nr}} \chi_j(t^k)$ follows a binomial distribution $B(\text{Nr}, P_{j,\Delta t})$, where t^k is the angle that the k -th ray forms with the optical axis. Then, using the de Moivre-Laplace theorem, we conclude that, when a large number of rays is considered, the variable Y_j is approximated by a normal distribution with mean value and variance given by

$$E[Y_j] = \text{Nr}P_{j,\Delta t}, \quad (3.2.7a)$$

$$\sigma^2[Y_j] = \text{Nr}P_{j,\Delta t}(1 - P_{j,\Delta t}), \quad (3.2.7b)$$

(see [53, 54]). Thus, the normalized intensity along the direction $t_{j-1/2}$ is given by:

$$\hat{I}_{\text{MC}}(t_{j-1/2}) = \sum_{k=1}^{\text{Nr}} \chi_j(t^k) / \text{Nr}. \quad (3.2.8)$$

The corresponding expected value and variance are:

$$E[\hat{I}_{\text{MC}}(t_{j-1/2})] = P_{j,\Delta t}, \quad (3.2.9a)$$

$$\sigma^2[\hat{I}_{\text{MC}}(t_{j-1/2})] = P_{j,\Delta t}(1 - P_{j,\Delta t}) / \text{Nr}. \quad (3.2.9b)$$

Note that the standard deviation $\sigma_j := \sigma[\hat{I}_{\text{MC}}(t_{j-1/2})]$ is approximated by

$$\sigma_j = \sqrt{P_{j,\Delta t}(1 - P_{j,\Delta t})/\text{Nr}} \leq \frac{C_2}{\sqrt{\text{Nb}\text{Nr}}}, \quad (3.2.10)$$

as $P_{j,\Delta t} \leq C_2/\text{Nb}$ for some $C_2 > 0$. σ_j can be used to give an estimate of the difference between the intensity $\hat{I}_{\text{MC}}(t_{j-1/2})$ and its mean value $P_{j,\Delta t}$. Therefore, the second term of the right hand side of relation (3.2.4) becomes

$$\begin{aligned} \frac{1}{\Delta t} \left| \int_{t_{j-1}}^{t_j} I(t) dt - \Phi \hat{I}_{\text{MC}}(t_{j-1/2}) \right| &= \\ \frac{\Phi}{\Delta t} \left| P_{j,\Delta t} - \hat{I}_{\text{MC}}(t_{j-1/2}) \right| &\approx \\ \frac{\Phi}{\Delta t} \sigma_j &\leq C_3 \frac{\text{Nb}}{\sqrt{\text{Nb}\text{Nr}}} = C_3 \sqrt{\frac{\text{Nb}}{\text{Nr}}}, \end{aligned} \quad (3.2.11)$$

for some $C_3 > 0$, where the approximation holds because σ_j gives a measure for the error between $\hat{I}_{\text{MC}}(t_{j-1/2})$ and the probability $P_{j,\Delta t}$ [55]. The second approximation follows from (3.2.10). In summary, the MC error over the j -th bin is approximated by

$$\left| I(t_{j-1/2}) - \frac{\Phi}{\Delta t} \hat{I}_{\text{MC}}(t_{j-1/2}) \right| \approx \frac{C_1}{\text{Nb}^2} + C_4 \sqrt{\frac{\text{Nb}}{\text{Nr}}}, \quad (3.2.12)$$

for some $C_1 > 0$ and $C_4 > 0$. Considering a fixed number of bins, we observe that the minimal error is reached when $\text{Nr} \approx \text{Nb}^5$. Hence, if we double the number of bins we need to trace $2^5 = 32$ times more rays! We remark that the first term on the right hand side of (3.2.12) gives an estimation of the discretization error, while the second term on the right hand side of (3.2.12) provides an estimation of the MC error, which in this case also depends on the discretization (partitioning at the target).

We conclude this section implementing MC ray tracing for the two-faceted cup, the profile of which is depicted in Figure 3.1. Considering a set of $\text{Nr} = 10^3$ random rays at the source, we obtain an example of the ray distribution on the (x_1, t_1) -plane shown in Figure 3.2. Since the rays are chosen randomly, the distribution at the source could be different from the one shown in this figure.

For the intensity computation we consider a sample of 10^4 random rays and we divide the target target $T = [-b, b]$ into $\text{Nb} = 100$ bins. The profile of \hat{I}_{MC} is depicted in Figure 3.3 with the red line. The exact intensity is shown with the green line in the same figure.

MC ray tracing has the advantage of being very easy to implement and it does not require too much regularity of the function that has to be approximated. Furthermore, the error convergence does not depend on the dimension of the domain in which the function is defined. On the other hand, the MC method is time consuming as the error has a speed of convergence of order $\mathcal{O}(1/\sqrt{\text{Nr}})$ for increasing Nr while fixing the number of bins. Thus, to decrease the MC error by a factor 10 we need to increase the number of rays by a factor 100. Since MC ray tracing is a binning procedure, the error depends also on the number of bins in which the target is divided.

Finally we remark that the error bound is only a *probabilistic* error. This means, to calculate the value of the error, several simulations have to be repeated and the

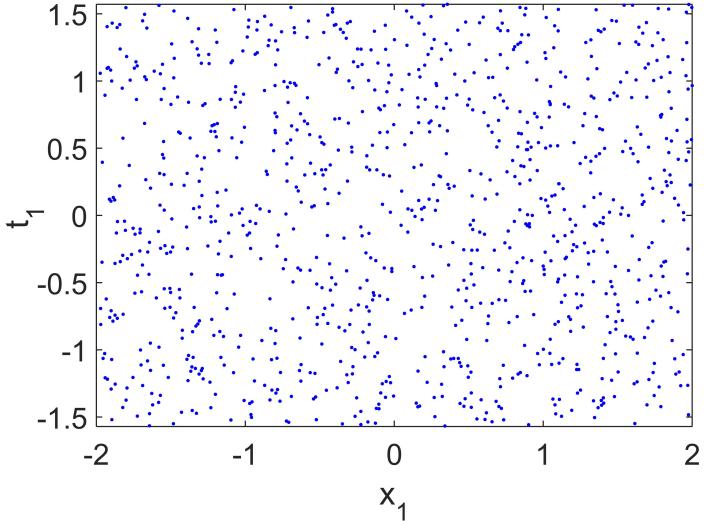


Figure 3.2: Rays at the source of the two-faceted cup with random position coordinate x and random angular coordinates t . 10^3 rays are depicted in this figure.

average of the errors obtained in every simulation has to be calculated.

MC noise can be reduced considering a different distribution of the initial rays set. Instead of considering random variables, the sample of rays can be defined such that they are regularly distributed. Methods based on this *deterministic approach* are called Quasi-Monte Carlo (QMC) methods. The ray tracing procedure that considers such rays distribution is called QMC ray tracing.

3.3 Quasi-Monte Carlo (QMC) ray tracing

In this section we first introduce QMC methods and then we briefly explain QMC ray tracing. QMC methods were proposed for the first time in the 1950s in order to speed up MC. Like MC methods, QMC procedures can be used to approximate the integral of a function f .

This section provides the basic notions about uniform distribution theory following Chapter 2 of [52]. We restrict ourselves to sets of the form

$$[\mathbf{a}, \mathbf{b}) = [a_1, b_1) \times [a_2, b_2) \subseteq [0, 1]^2,$$

the measure of which is

$$\lambda([\mathbf{a}, \mathbf{b})) = (b_1 - a_1) \times (b_2 - a_2).$$

Let $\{\mathbf{y}_i\}_{i=1,\dots,N}$ be a finite sequence of numbers, we define the set A as

$$A([\mathbf{a}, \mathbf{b}), N) = \{n \in \mathbb{N}_0 | 0 \leq n \leq N - 1 \text{ and } \mathbf{y}_n \in [\mathbf{a}, \mathbf{b})\}. \quad (3.3.1)$$

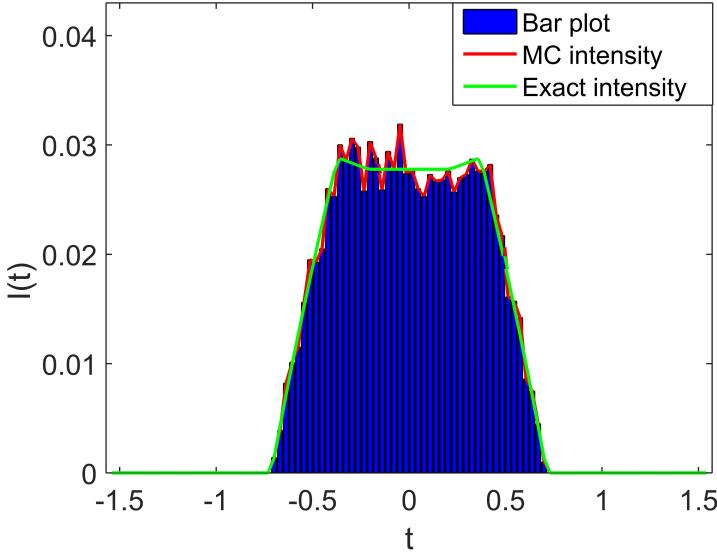


Figure 3.3: Comparison between the averaged normalized MC intensity and the normalized exact intensity. The MC intensity is computed using MC ray tracing with $N_r = 10^4$ and $N_b = 100$.

The cardinality $\text{card}(\mathbf{A}([\mathbf{a}, \mathbf{b}], N))$, i.e., the number of elements, of $\mathbf{A}([\mathbf{a}, \mathbf{b}], N)$, is the number of samples \mathbf{y}_n such that $\mathbf{y}_n \in [\mathbf{a}, \mathbf{b}]$. Let us now introduce the concept of sequences uniformly distributed modulo 1.

Definition 3.3.1. An infinite sequence $\{\mathbf{y}_n\}_{n \in \mathbb{N}_0} \in [0, 1]^2$ is said to be uniformly distributed modulo 1 (or equidistributed), if for every subset $[\mathbf{a}, \mathbf{b}] \subseteq [0, 1]^2$

$$\lim_{N \rightarrow \infty} \frac{\text{card}(\mathbf{A}([\mathbf{a}, \mathbf{b}], N))}{N} = \lambda([\mathbf{a}, \mathbf{b}]). \quad (3.3.2)$$

Given a finite sequence $\{\mathbf{y}_i\}_{i=1, \dots, N} \in [0, 1]^2$ uniformly distributed modulo 1 and a Riemann integrable function $f : [0, 1]^2 \rightarrow \mathbb{R}$, the integral of f can be approximated as the average of the values that f assumes on $\{\mathbf{y}_i\}$ for every $i = \{1, \dots, N\}$, that is:

$$\int_{[0,1]^2} f(\mathbf{y}) d\mathbf{y} = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{i=1}^N f(\mathbf{y}_i). \quad (3.3.3)$$

QMC method are, therefore, based on the same approximation of MC methods. The idea of QMC methods is to generate the set of points \mathbf{y}_i in $[\mathbf{a}, \mathbf{b}]$ such that they are not randomly distributed but also not exactly uniformly distributed. To measure how much the distribution of these sample points differs from a uniform distribution, the concept of discrepancy was introduced. Random sequences have a very high discrepancy, while uniformly distributed sequences have zero discrepancy. *Low-discrepancy*

sequences are sequences with a low discrepancy [56] where the discrepancy of a sequence is *low* if the fraction of points of the sequence belonging to an arbitrary set is close to the measure of that set. The definition of discrepancy in more mathematical terms is provided next.

Definition 3.3.2. *Given a set $Y = \{\mathbf{y}_1, \dots, \mathbf{y}_N\}$ of N points in $[0, 1]^2$. The discrepancy $D_N(Y)$ of Y is defined as*

$$D_N(Y) = \sup_{\mathbf{a}, \mathbf{b} \in [0, 1]^2} \left| \frac{\text{card}(A([\mathbf{a}, \mathbf{b}], N))}{N} - \lambda([\mathbf{a}, \mathbf{b}]) \right|. \quad (3.3.4)$$

Often, it is enough to consider the discrepancy in the subset $[\mathbf{a}, \mathbf{b}] \subseteq [0, 1]^2$ with $\mathbf{a} = 0$, in which case we talk about star discrepancy.

Definition 3.3.3. *Let $Y = \{\mathbf{y}_1, \dots, \mathbf{y}_N\}$ be a set of N points in $[0, 1]^2$. The star discrepancy $D_N^*(Y)$ of Y is defined as:*

$$D_N^*(Y) = \sup_{\mathbf{b} \in [0, 1]^2} \left| \frac{\text{card}(A([0, \mathbf{b}], N))}{N} - \lambda([0, \mathbf{b}]) \right|, \quad (3.3.5)$$

where $\lambda([0, \mathbf{b}]) = b_1 b_2$.

An important result is provided in the following

Theorem 3.3.4. *Using a low-discrepancy sequence $\{\mathbf{y}_i\}_{i=1, \dots, N}$, the absolute error of a 2D QMC quadrature rule, i.e.,:*

$$\text{err}(f, S_N) = \int_{[0, 1]^2} f(\mathbf{y}) d\mathbf{y} - \frac{1}{N} \sum_{i=1}^N f(\mathbf{y}_i) \quad (3.3.6)$$

can be bounded by the product of a term that depends on f and another term that depends on the star discrepancy of the set $\{\mathbf{y}_i\}_{i=1, \dots, N}$. This result follows the Koksma-Hlawka inequality, which gives the worst-case error estimation:

$$|\text{err}(f, S_N)| = \left| \int_{[0, 1]^2} f(\mathbf{y}) d\mathbf{y} - \frac{1}{N} \sum_{i=1}^N f(\mathbf{y}_i) \right| \leq V(f) D_N^*(Y), \quad (3.3.7)$$

where $V(f)$ is the so-called variation function of f in the sense of Hardy-Krause [57].

Usually both the star discrepancy $D_N^*(Y)$ and the variation function $V(f)$ are difficult to approximate. In particular, $V(f)$ in some cases is even infinite [58]. Hence, (3.3.7) is not good for predicting for which N the error will converge. For the functions we analyze in this thesis the corresponding variation function is always bounded and the convergence of QMC methods strongly depends on the low-discrepancy sequence that is used.

There are many ways to generate low-discrepancy sequences [59]. The most common QMC approach uses the so-called Sobol sequence. The algorithm for generating Sobol sequences is widely explained in the literature, (see for instance [60]). When using Sobol sequences, the absolute value of QMC error can be estimated by:

$$|\text{err}(f, S_N)| < C \frac{\log(N)^2}{N}, \quad (3.3.8)$$

for some $C > 0$. For higher dimensions ($d > 2$), the general relation holds

$$|\text{err}(f, S_N)| < C \frac{\log(N)^d}{N}. \quad (3.3.9)$$

Ray tracing based on QMC methods takes as position and angular coordinates of the rays at the source, the coordinates of the corresponding points of a low-discrepancy sequence. Therefore, to implement QMC ray tracing in two-dimensions we need to construct a 2D low-discrepancy sequence at the source. A set of $\text{Nr} = N$ rays with these initial coordinates is traced within the system and the target coordinates of all the rays traced are computed. We remark that the variables \mathbf{y}_i are not random variables but elements of the Sobol sequence in 2D. Because of this, QMC ray tracing is a *deterministic* method.

Similarly to MC ray tracing, the averaged and normalized QMC intensity \hat{I}_{QMC} is given by the approximation in (3.2.1), where now the variable \mathbf{y} is an element of the Sobol sequence instead of a random variable. Therefore, the target is divided into Nb bins and the number of rays that arrive at each bin is determined. The intensity is still a piecewise constant function, and its value over every bin $[t_{j-1}, t_j]_{j=1, \dots, \text{Nb}}$ is given by the intensity $\hat{I}_{\text{QMC}}(t_{j-1/2})$ calculated along the direction $t_{j-1/2} = (t_{j-1} + t_j)/2$ (middle point of the bin). The only difference between MC and QMC ray tracing is the choice of the initial ray set. Thus, we expect that QMC and MC errors have the same dependence on the number of bins Nb . More precisely, since the discretization error (first term on the right hand side of inequality (3.2.4)) does not depend on MC noise, we expect that it does not change for QMC ray tracing. Regarding the second term on the right hand side of (3.2.4), we showed in the previous section that it depends on the standard deviation of the approximated intensity and on the number of bins. Therefore, to estimate the error as a function of the number of bins Nb , the variance of QMC intensity needs to be calculated.

In Figure 3.4 we show the distribution of the position and direction coordinates of the rays at the source of the two-faceted cup from Figure 3.1. A set of 10^3 rays generated from a 2D Sobol sequence is considered, the coordinates (x_1, t_1) of every ray at the source are depicted with blue dots. We note that the rays have a regular distribution on the (x, t) -plane. We remark that for the system in Figure 3.1, $x_1 \in [-2, 2]$ and $t_1 \in [-\pi/2, \pi/2]$. Since Sobol sequences are defined inside intervals of the $[\mathbf{a}, \mathbf{b}] \subseteq [0, 1]^2$, we scaled the points of the sequence \mathbf{y}_i accordingly in order to take into account all the possible positions and directions that the rays can assume.

Dividing the target into $\text{Nb} = 100$ bins, we computed the target intensity. In Figure 3.5 we show the profile of the output intensity at the target of the two-faceted cup computed using QMC ray tracing with 10^4 rays. The QMC intensity is depicted with the red line. It is compared to the exact intensity shown in the same figure with the green dotted line.

A comparison between Figure 3.3 and 3.5 shows that for the two-faceted cup and for a set of $\text{Nr} = 10^4$ rays, QMC ray tracing performs better than MC ray tracing.

In order to compare MC and QMC ray tracing, we calculate the target intensity using both methods gradually increasing the number of rays traced. The errors between the approximated averaged normalized intensity \hat{I}_A ($A = \text{MC, QMC}$) and the exact normalized intensity \hat{I}_{exact} are calculated. The speed of convergence for MC is



Figure 3.4: 10^3 rays at the source of the two-faceted cup with position coordinate x_1 and angular coordinate t_1 . They are distributed as the points of a 2D Sobol sequence.

shown in Figure 3.6 with the green line, while the behavior of QMC ray tracing is depicted in the same picture with the blue line. The results shown for a simple optical system are indeed consistent with what we expected from the theoretical analysis.

Although QMC ray tracing is an improvement of MC ray tracing for small dimensions, it has still two main disadvantages. First, its convergence is strongly related with the dimension in which it is implemented. Second, like MC ray tracing, QMC ray tracing is a binning procedure, therefore the error still depends on the number of bins in which the target is divided and only the averaged value of the intensity over every bin is provided.

From the results presented in this chapter we can conclude that the choice of the initial ray set can have a big impact on the performance of the ray tracing procedure. Based on the idea of taking a smart choice of the initial ray set, we develop a new ray tracing method which is based on phase space. The phase space (PS) concept will be introduced in the next chapter. The new ray tracing method employs the PS of the source and the target of the optical systems. We will show that phase space ray tracing allows us to trace a relatively small number of rays inside the system to obtain the desired accuracy of the target intensity.

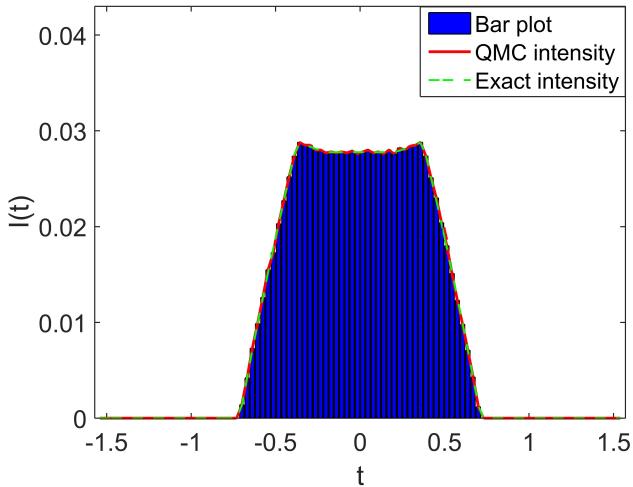


Figure 3.5: QMC intensity for the two-faceted cup obtained tracing $N_r = 10^4$ rays and dividing the target into $N_b = 100$ bins.



Figure 3.6: Error as function of the number of rays traced for fixed number of bins $N_b = 100$. MC ray tracing convergence is of the order $\mathcal{O}(1/\sqrt{N_r})$ and it is shown with the green line. QMC ray tracing convergence is of the order $\mathcal{O}(1/N_r)$ and it is depicted with the blue line.

Chapter 4

Ray tracing on phase space

Ray tracing on phase space is a method which employs the phase space (PS) of the source and the target of the optical system. Moreover, it takes into account the trajectory that every ray follows during its propagation. Before explaining the method, we introduce the PS concept.

4.1 Phase space

Every ray in three dimensions can be described by three position coordinates and three direction coordinates. The PS of an optical surface is a four-dimensional space characterized by two position and two direction coordinates. The position coordinates are two of the coordinates of the intersection point of the ray with the surface, while the direction coordinates are the momentum coordinates of the vector tangent to the ray projected on that optical surface [31].

In two dimensions, every ray parametrization is obtained from two position and two direction coordinates. The PS of an optical line is described by one position and one direction coordinate. Hence, for two-dimensional systems, every ray in PS is described by a point in a two-dimensional space. Given an optical line j , the ray position coordinate on PS is the x -coordinate of the intersection point between the ray and the line j . The direction coordinate is the sine of the angle that the ray forms with respect to the normal ν of line j multiplied by the index of refraction n . We choose ν always directed inside the same medium in which the incident ray travels. The PS is indicated with $S = Q \times P$, where Q is the set of the position coordinates q and P is the set of the direction coordinates $p = n \sin \theta$, with $\theta \in [-\pi/2, \pi/2]$ the angle between the ray segment inside the system and the normal measured counterclockwise. In the following, the PS is considered only for the source S and the target T and for no other line of the optical system. The coordinates of every ray on S and T are indicated with (q_1, p_1) and (q, p) , respectively.

As an example, in Figures 4.1 and 4.2 we show the source and target PS of the two-faceted cup (in Figure 3.1), sampled with 10^4 random rays. The coordinates of every point correspond to the position and direction coordinates of a ray which are calculated using the ray tracing procedure. Furthermore, we store the path Π that every ray follows, where we refer to a path as the sequence of lines encountered by

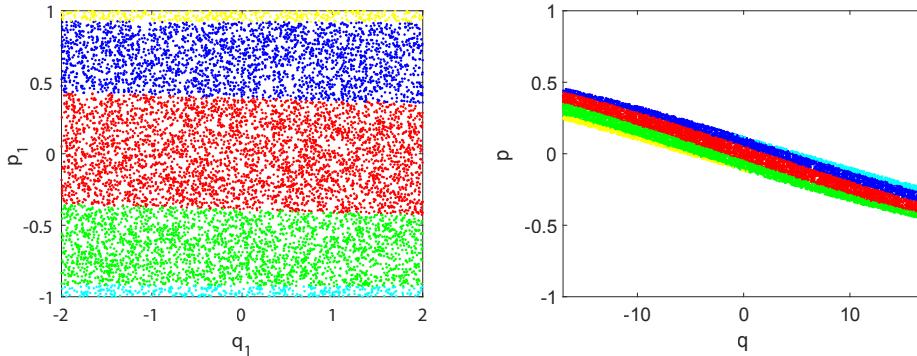


Figure 4.1: **Source PS.** Five different paths can occur for the two-faceted cup. Figure 4.2: **Target PS.** Five different paths can occur for the two-faceted cup.

the ray. In Figures 4.1 and 4.2 a color is associated to every path, hence all the rays that follow the same path are depicted with the same color. We note that the source and target phase spaces are partitioned into different regions according to the path Π followed by the rays. Given a path Π , the corresponding regions are indicated with $R_s(\Pi)$ and $R_t(\Pi)$ at the source and the target PS, respectively. Rays that propagate through the two-faceted cup can follow 5 different paths. Some rays are emitted from the source and arrive at the target without hitting any other line, they follow path $\Pi_1 = (1, 4)$. These rays are depicted in red in the PS pictures. Some other rays can hit the left or the right reflector (line 2 and 3, respectively) once, their corresponding paths are $\Pi_2 = (1, 2, 4)$ and $\Pi_3 = (1, 3, 4)$, respectively. These rays are the blue and green dots in PS. Finally, there is the possibility that the rays have two reflections before hitting the target. They follow either path $\Pi_4 = (1, 2, 3, 4)$ or path $\Pi_5 = (1, 3, 2, 4)$ and they are depicted with the yellow and cyan points.

For the two-faceted cup all light emitted by the source arrives at the target. In order to derive the photometric variables at the target we need to understand where light ends up, i.e., which parts of the target PS are illuminated by the source. Indeed, while the source PS is completely covered by rays, some parts of the target PS are not reached by any ray at all. This means that, while the luminance at the source PS is positive for any possible position and direction, the luminance at the target PS is positive only inside the regions $R_t(\Pi)$, for every path Π , and it is equal to 0 outside those regions. For this reason, from now on we will refer to $R_t(\Pi)$ as the *positive luminance regions*.

It is very important to remark that, although S and T have a different ray distribution, the area covered by the rays is conserved. This follows from étendue conservation. From (2.1.21) we rewrite the two-dimensional étendue as:

$$U = \int_{x^{\min}}^{x^{\max}} \int_{\theta^{\min}}^{\theta^{\max}} n \cos(\theta) dx d\theta = \int_P \int_Q dq dp. \quad (4.1.1)$$

where we indicated with x^{\min} and x^{\max} the minimum and maximum rays position coordinates and with θ^{\min} and θ^{\max} the minimum and maximum angles of the rays and

the normal to the target. The second equality holds since $dp = n \cos \theta d\theta$. Therefore, in two dimensions, étendue can be seen as an area in PS. Etendue conservation leads to the conservation of the areas of regions with positive luminance.

For the two-faceted cup in Figure 3.1, $S = [-2, 2] \times [-1, 1]$, thus, the étendue at the source is $U_s = 8$ (see Figure 4.1). Computing the total area covered by the positive luminance regions at the target using the trapezoidal rule, we obtain $U_t = 8$ which numerically proves étendue conservation for the two faceted-cup.

In the next section we provide a literature overview of a fundamental principle in non-imaging optics: "the edge-ray principle".

4.2 The edge-ray principle

The goal in non-imaging optics is to transfer all light from the source aperture to the output aperture. Systems that satisfy this property are referred to *ideal optical systems*. Several methods to design ideal optical systems are based on the edge-ray principle [61, 62]. Basically it states that all the light rays exiting the edges of the source will end at the edges of the target. This guarantees that all light emitted from the source will arrive at the receiver, see Figure 4.3.

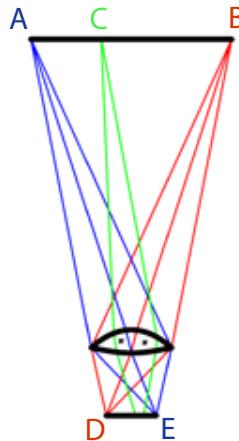


Figure 4.3: **A lens receiving light.** The lens redirects light emitted from the source AB to the receiver DE. Rays that leave the edges of the source hit the edges of the target (blue and red rays). Rays coming from the interior of the source will end at the interior of the target (green rays) [63].

In 1985 Miñano proved the principle by using the PS of the source and the target of an optical system [64, 65]. He proved the principle for systems in inhomogeneous media, where the index of refraction is a continuous function, so the map that connects the source and target phase spaces is a continuous map. Indicating with $M(P)$ the optical map of a point P , Miñano showed that if $M(\partial S) = \partial T$ then $M(S) = T$ and vice versa. The first version of the edge-ray principle [65] can be enunciated in two-dimensions as follows:

Lemma 4.2.1. *Edge-ray principle (version1)*

Suppose that:

- a) There are two regions R_s and R_t in source and target PS with the same area such that

$$M(\partial R_s) = M(\partial R_t);$$

- b) The refractive-index distribution n is a continuous function;

Then, the following relation holds:

$$M(R_s) = M(R_t).$$

The previous lemma claims that if there exist a map connecting the boundaries of two regions from the source to the target, then also the interior of those regions are connected using the same map. Note that the second assumption in the previous lemma implies that the optical map is continuous in PS. However, for some optical systems, as for instance the compound parabolic concentrator (CPC), the ray mapping in PS is not continuous. This is due to multiple reflections that rays can encounter with the reflectors and implies that some rays at the edge of the source could not be mapped into rays at the edges of the target [66].

In 1994 Ries and Rabl reformulated the edge-ray principle providing a version valid for all systems even if the ray map in PS is not continuous [37]. Suppose that $R_s(\Pi)$ and $R_t(\Pi)$ are the regions, corresponding to path Π , at the source and the target PS, respectively. They showed that, for a given path Π , if the boundaries $\partial R_s(\Pi)$ are mapped into the boundaries $\partial R_t(\Pi)$, then also the regions $R_s(\Pi)$ are mapped into the regions $R_t(\Pi)$. Then, to map S to T it is necessary and sufficient that the first version of the edge ray principle is observed for all part of S and T defined by the number of reflections [37].

Lemma 4.2.2. *Edge-ray principle (generalized version)*

Let $(\Pi_j)_{j=1, \dots, N_p}$ denote all possible paths N_p . Every possible path corresponds to a certain number of reflections or refractions. Let us denote with $R_s(\Pi_j)$ and $R_t(\Pi_j)$ the regions at S and T associated to path Π_j such that they are a partition of S and T, that is:

$$\begin{aligned} S &= \bigcup_{j=1}^{N_p} R_s(\Pi_j), \text{ with } R_s(\Pi_j) \cap R_s(\Pi_i) = \emptyset \text{ for } i \neq j \\ T &\supset \bigcup_{j=1}^{N_p} R_t(\Pi_j), \text{ with } R_t(\Pi_j) \cap R_t(\Pi_i) = \emptyset \text{ for } j \neq i. \end{aligned}$$

Then, to map a source region into a target, it is necessary and sufficient that the first version of the edge ray principle is observed for all parts of S and T:

$$M(\partial R_s(\Pi_j)) = \partial R_t(\Pi_j), \quad \forall j \in \{1, \dots, N_p\}.$$

Hence, the edge-ray principle constitutes a tool for designing ideal systems and, to this purpose, it is sufficient that the rays of $\partial R_s(\Pi)$ are transformed to the rays of $\partial R_t(\Pi)$ for every path Π [67].

Using the PS concept and the edge-ray principle we develop a new ray tracing method. A non-uniform distribution of the rays is provided by developing a triangulation refinement at the source PS which is explained in the next section. The triangulation refinement provides more rays close to the boundaries of the regions $R_s(\Pi)$ each of them is formed by the rays that follow the same path Π .

4.3 Phase space ray tracing

PS ray tracing takes advantage of the fact that there exists an optical map $M : S \rightarrow T$ such that

$$M(q_1, p_1) = (q, p), \quad (4.3.1)$$

for every $(q_1, p_1) \in S$. For very simple systems, like the two-faceted cup, it is possible to determine an analytic expression for M . This is not the case for most of the optical systems we deal with. In these cases it is necessary to implement ray tracing to calculate how light is distributed at the target. As mentioned in the previous section, for some optical systems M is not even continuous. Nevertheless, given a path Π , the restriction of M to R_s , i.e., $M(\Pi) : R_s(\Pi) \rightarrow R_t(\Pi)$ is a continuous and bijective map. The edge ray principle guarantees that $M(\Pi)$ maps $R_s(\Pi)$ onto $R_t(\Pi)$ preserving topological features. In particular, the boundary $\partial R_s(\Pi)$ is mapped onto the boundary $\partial R_t(\Pi)$. Employing the maps $M(\Pi)$ for all the possible paths Π , the output light distribution is determined. Therefore, the photometric variables at the target can be calculated.

The luminance $L(q, p)$ at the target PS is given by:

$$\begin{aligned} L(q, p) &> 0 \text{ for } (q, p) \in R_t(\Pi) \text{ for some path } \Pi, \\ L(q, p) &= 0 \text{ otherwise.} \end{aligned} \quad (4.3.2)$$

The target intensity along a given direction $p = \text{const}$ is computed through an integration of the target luminance $L(q, p)$ over q and it is defined in T by:

$$I_{PS}(p) = \int_Q L(q, p) dq. \quad (4.3.3)$$

Note that, while in the real space the intensity is defined as a function of the angular coordinate θ (see Chapter 2), in PS the intensity is defined as a function of the direction coordinate $p = n \sin(\theta)$. The previous equation implies that, assuming a Lambertian source, the problem of computing the target intensity is reduced to the problem of calculating the boundaries $\partial R_t(\Pi)$ for all possible paths Π . Hence, the intensity along the direction $p = \text{const.}$ is given by the sum of the interval lengths formed by the support of the luminance and the line $p = \text{const.}$ For example, if two intersection points between line $p = \text{const.}$ and the boundary $\partial R_t(\Pi)$ are found, indicating their position coordinates with $q^{\min}(\Pi, p)$ and $q^{\max}(\Pi, p)$, where $q^{\min}(\Pi, p) < q^{\max}(\Pi, p)$, and using equation (4.3.2), we obtain that equation (4.3.3) reduces to:

$$I_{PS}(p) = \sum_{\Pi} \int_{q^{\min}(\Pi, p)}^{q^{\max}(\Pi, p)} L(q, p) dq = \sum_{\Pi} (q^{\max}(\Pi, p) - q^{\min}(\Pi, p)), \quad (4.3.4)$$

where the sum is over all the possible paths and the second equation holds as we assume Lambertian source with $L = 1$ in $R_t(\Pi)$. In case more than two intersection points occur, a generalized equation needs to be used for calculating the intensity. Note that for every single ray only one path is possible as we are assuming that all the lines are reflective lines. Because of this, the regions $R_t(\Pi)$ do not overlap, i.e.

$$\bigcap_{\Pi} R_t(\Pi) = \emptyset, \quad (4.3.5)$$

where the intersection is over all possible paths.

From equation (4.3.4) we note that, using the PS structure, only the rays on the boundaries $\partial R_t(\Pi)$ are required for obtaining the target intensity profile. The aim is to construct a ray tracing procedure that allows us tracing less rays overall and more rays close to the discontinuity of the luminance, i.e., close to the boundaries $\partial R_t(\Pi)$. To this purpose, we start from a triangulation made by only two triangles, then a triangulation refinement at S is defined as explained in the following.

The procedure starts with coordinates $(q_1^k, p_1^k)_{k=1,\dots,4}$ of the four corner points of S. For each of them, the corresponding path $(\Pi^k)_{k=1,\dots,4}$ is calculated. Next, the grid is divided into two equal triangles joining two opposite vertices (in our simulation we always trace the diagonal north-west south-east to define the new triangles). For each triangle the rays located at its corners are traced. If the paths corresponding to those rays are not all equal, one or more boundaries $\partial R_t(\Pi)$ are expected to cross the triangle. In that case, the middle points $(q_1^k, p_1^k)_{k=5,6,7}$ of each side of the triangle are added and the three corresponding rays are traced (unless they were already traced in the previous steps). Each refinement step leads to four new triangles (see Figure 4.4).

When all the rays corresponding to the corners of each triangle have the same path, it is not necessary to refine the triangles anymore. Since the triangles very close to the boundaries are always crossed by at least a boundary, at least two different paths are found for the rays at the vertices of those triangles. Because of this, the procedure could continue infinitely, therefore, two parameters ε_q^{\min} and ε_p^{\min} are introduced to defined a stopping criterion. The algorithm stops when the length of the sides of the triangles is smaller than ε_q^{\min} and ε_p^{\min} in q and p direction.

We indicate all the possible paths with $(\Pi_j)_{j=1,\dots,N_p}$ where N_p is the maximum number of paths¹ ($N_p = 5$ for the two-faceted cup in Figure 3.1). If the size of the triangles is too big, it can happen that a region formed by rays that follow a path Π_j is located completely inside a triangle whose vertices are related to another path Π_i with $j \neq i$, see Figure 4.5. To avoid this, two parameters ε_q^{\max} and ε_p^{\max} are defined for the q_1 -axis and the p_1 -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. The values of the parameters ε_q^{\min} , ε_p^{\min} , ε_q^{\max} and ε_p^{\max} determine the number of rays traced. Thus, on the one hand, decreasing ε_q^{\min} and ε_p^{\min} more rays close to the boundaries are traced; on the other hand, decreasing the values of ε_q^{\max} and ε_p^{\max} more rays in the interior of the regions are traced.

The triangulation refinement is provided by Algorithm 1 which uses the two recurs-

¹We remind the reader that we indicate with $\Pi^k = \Pi(q_1^k, p_1^k)$ the path followed by rays with coordinates (q_1^k, p_1^k) in source PS. Note that it can happen that $\Pi^k = \Pi^h$ for $k \neq h$. With $(\Pi_j)_{j=1,\dots,N_p}$ we indicate all the possible N_p paths that can occur, therefore $\Pi_i \neq \Pi_j$ if $i \neq j$.

ive functions LEFT TRIANGLE and RIGHT TRIANGLE. The function LEFT TRIANGLE is defined in Algorithm 2 (see Figure 4.6). A similar procedure gives the function RIGHT TRIANGLE (see Figure 4.7).

Algorithm 1 Triangulation refinement algorithm

```

Initialize  $\varepsilon_q^{\min}, \varepsilon_q^{\max}, \varepsilon_p^{\min}$ , and  $\varepsilon_p^{\max}$ , Ray = [empty];
Define a structure that contains related data in fields.
▷  $\varepsilon_q^{\min}, \varepsilon_q^{\max}, \varepsilon_p^{\min}$ , and  $\varepsilon_p^{\max}$  are fixed parameters needed to stop the procedure
▷ Ray: structure that contains all the data of rays traced (i.e., position, direction and path).

1:  $(q_1^1, p_1^1) = (-a, -1)$                                 ▷ left bottom corner of source PS
2:  $(q_1^2, p_1^2) = (a, -1)$                                 ▷ right bottom corner of source PS
3:  $(q_1^3, p_1^3) = (a, 1)$                                 ▷ right upper corner of source PS
4:  $(q_1^4, p_1^4) = (-a, 1)$                                 ▷ left upper corner of source PS
5: for  $k = 1 \rightarrow 4$  do
6:   Trace the ray with initial coordinates  $(q_1^k, p_1^k)$  in S;
7:   Calculate the corresponding path  $\Pi^k$ ;
   ▷ Store the information found in the structure Ray;
8:   Ray.q = [Ray.q,  $q_1^k$ ];
9:   Ray.p = [Ray.p,  $p_1^k$ ];
10:  Ray.II = [Ray.II,  $\Pi^k$ ];
11: end for
12: VL = [1, 2, 4]                                         ▷ VL vertices of the left triangle
13: VR = [2, 3, 4]                                         ▷ VR vertices of the right triangle
14: LEFT TRIANGLE(VL, Ray,  $\varepsilon_{q_1}^{\min}, \varepsilon_{q_1}^{\max}, \varepsilon_{p_1}^{\min}, \varepsilon_{p_1}^{\max}$ )      ▷ Refine the left triangle
15: RIGHT TRIANGLE(VR, Ray,  $\varepsilon_{q_1}^{\min}, \varepsilon_{q_1}^{\max}, \varepsilon_{p_1}^{\min}, \varepsilon_{p_1}^{\max}$ )     ▷ Refine the right triangle
16: return Ray;
```

Figure 4.8 shows an example of a triangulation refinement at the source PS of the two-faceted cup in Figure 3.1. For this optical system, the width of the q_1 -axis in source PS is two times the width of the p_1 -axis. Thus, our choice is $\varepsilon_p^{\min} = \frac{1}{2}\varepsilon_q^{\min}$ and $\varepsilon_p^{\max} = \frac{1}{2}\varepsilon_q^{\max}$ with $\varepsilon_q^{\min} = 0.1$ and $\varepsilon_q^{\max} = 1$.

The triangulation refinement allows finding *all* the possible paths $(\Pi_j)_{j=1,\dots,N_p}$ and their corresponding regions $R_s(\Pi_j)_{j=1,\dots,N_p}$. Using the edge-ray principle, we conclude that also the regions $R_t(\Pi_j)_{j=1,\dots,N_p}$ at the target are determined and only the rays close to the boundaries ∂R_s need to be considered to obtain the target ray distribution.

4.4 Conclusions

In this chapter we introduced the phase space concept. We explained a new ray tracing method based on the source and the target PS representation. In PS every point corresponds to a unique ray. The coordinates of every point correspond to the initial ray position q_1 and the initial ray direction $p_1 = \sin \theta_1$ (as $n_1 = 1$) expressed with respect to the normal of the source. The method also takes into account the paths followed by every ray traced. Considering only reflection, every single ray follows only

one path and, therefore, the PS regions do not overlap.

As an example, we provided the source and the target PS representation of the two-faceted cup. The edge-ray principle guarantees that all the rays that follow the same path are located in the same regions in PS. If we know these regions at the source we can determine the corresponding regions at the target. It is sufficient to map the boundaries at the source $\partial R_s(\Pi)$ to obtain their corresponding target boundaries $\partial R_t(\Pi)$.

The boundaries $\partial R_t(\Pi)$ are particularly relevant because there the luminance jumps from 0 to a positive value. Assuming a Lambertian source, only the rays at the boundaries are needed to compute the target intensity. Based on this idea, a triangulation in S is constructed such that the rays closest to $\partial R_s(\Pi)$ are selected and more rays in their vicinity are created to get progressively better estimates of the boundaries.

In Figure 4.9 we show three different ray distributions on the source PS of the two-faceted cup. In Figure 4.9a, 10^3 random points are shown. MC ray tracing is based on this random distribution of the initial rays set. In Figure 4.9b, 10^3 points of a two-dimensional Sobol sequence are shown. Since Sobol sequences are defined in a unit square, we scaled it such that all the source PS $S = [-2, 2] \times [-1, 1]$ is covered by rays. Such regular distributions can lead to several advantages for the computation of the target intensity, see Section 3.3. Finally, Figure 4.9c shows a non-uniform distribution of rays at the source PS on which PS ray tracing is based. Such distribution is obtained from the triangulation refinement explained in the previous section. The procedure requires tracing more rays close to the boundaries $\partial R_s(\Pi)$ and only few rays in their interior of the regions in source PS. From the edge ray-principle, we obtain that these rays will be located close to the boundaries $\partial R_t(\Pi)$ of the regions at the target PS. The target PS intensity is calculated using only the rays that are located at the boundaries $\partial R_t(\Pi)$. Thus, in order to obtain the intensity profile at the target, the boundaries $\partial R_t(\Pi)$ need to be determined.

In the next chapter we provide two different approaches to find the boundaries $\partial R_t(\Pi)$ using a set of rays given by the triangulation refinement.

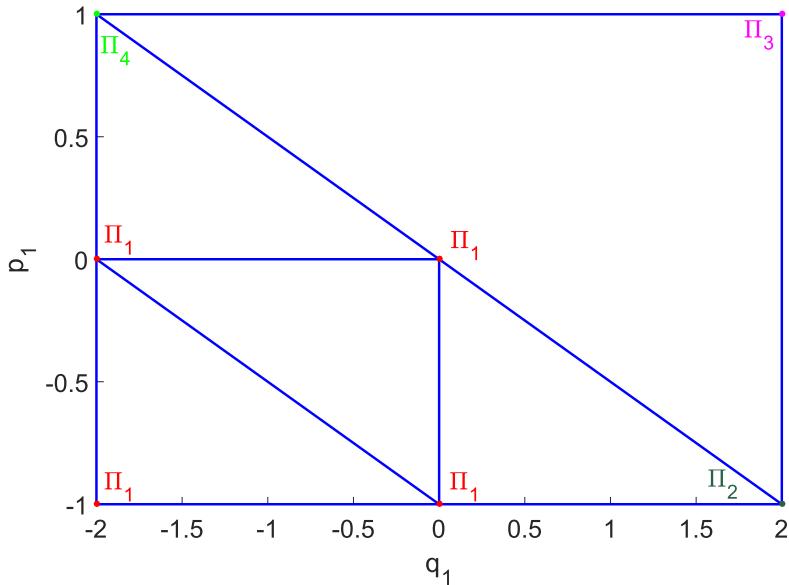


Figure 4.4: **Triangulation refinement.** If the rays related to the vertices of the triangles follow a different path a new refinement step is required. Each refinement step leads to four new triangles.

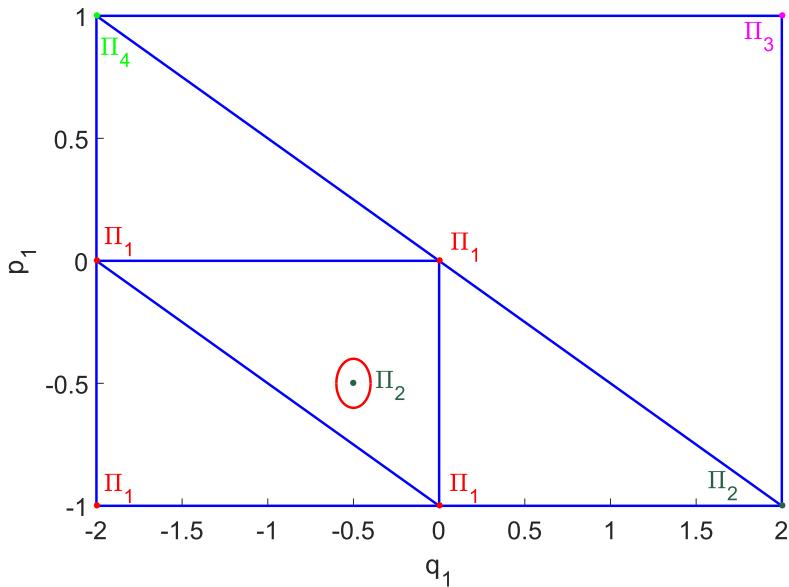


Figure 4.5: **Triangulation refinement.** The red line encloses a region of rays that follow the path Π_2 and is completely located inside a triangle. The algorithm is not able to detect that region and, a further refinement is required.

Algorithm 2 Algorithm for the refinement of the left triangles

```

1: procedure LEFT TRIANGLE(VL, Ray,  $\varepsilon_q^{\min}, \varepsilon_q^{\max}, \varepsilon_p^{\min}, \varepsilon_p^{\max}$ )
2:   VL = [1, 2, 4]
3:    $q_1^1 = \text{Ray}.q(\text{VL}(1)), p_1^1 = \text{Ray}.p(\text{VL}(1))$ 
4:    $q_1^2 = \text{Ray}.q(\text{VL}(2)), p_1^2 = \text{Ray}.p(\text{VL}(2))$ 
5:    $q_1^3 = \text{Ray}.q(\text{VL}(3)), p_1^3 = \text{Ray}.p(\text{VL}(4))$ 
6:    $\text{dist}_q = |q_1^2 - q_1^1|$ 
7:    $\text{dist}_p = |p_1^3 - p_1^1|$ 
8:   RefineTriangle = false;
9:   DifferentPath = false;
10:  if  $\text{dist}_q > \varepsilon_q^{\max}$  or  $\text{dist}_p > \varepsilon_p^{\max}$  then
11:    RefineTriangle = true;
12:  end if
13:  for  $k = 1 \rightarrow 2$  do
14:    if  $\Pi^k \neq \Pi^{k+1}$  then
15:      DifferentPath = true;
16:    end if
17:  end for
18:  if  $\text{dist}_q > \varepsilon_q^{\min}$  or  $\text{dist}_p > \varepsilon_p^{\min}$  then
19:    RefineTriangle = DifferentPath;
20:  else
21:    if (DifferentPath is true) then
22:      Ray(VL).boundary = true;            $\triangleright$  A boundary crosses the triangle
23:    end if
24:  end if
25:  if (RefineTriangle is true) then
26:    Define the points at the middle of each side of the triangle
27:     $(q_1^5, p_1^5) = ((q_1^1 + q_1^2)/2, p_1^1)$ 
28:     $(q_1^6, p_1^6) = (q_1^5, (p_1^1 + p_1^2)/2)$ 
29:     $(q_1^7, p_1^7) = (q_1^1, p_1^6)$ 
30:    for  $k = 5 \rightarrow 7$  do
31:      if The ray with coordinates  $(q_1^k, p_1^k)$  is not traced yet then
32:        Trace the ray with initial coordinates:  $(q_1^k, p_1^k)$  in PS;
33:        Compute the corresponding path  $\Pi^k$ ;
34:        Store the ray's coordinates  $\text{Ray}.q = [\text{Ray}.q, q_1^k]$ ;
35:        Store the ray path  $\text{Ray}.\Pi = [\text{Ray}.\Pi, \Pi^k]$ ;
36:      end if
37:    end for
38:    return LEFT TRIANGLE([VL(1), 5, 7], Ray,  $\varepsilon_q^{\min}, \varepsilon_q^{\max}, \varepsilon_p^{\min}, \varepsilon_p^{\max}$ );
39:    return LEFT TRIANGLE([5, VL(2), 6], Ray,  $\varepsilon_q^{\min}, \varepsilon_q^{\max}, \varepsilon_p^{\min}, \varepsilon_p^{\max}$ );
40:    return LEFT TRIANGLE([7, 6, VL(3)], Ray,  $\varepsilon_q^{\min}, \varepsilon_q^{\max}, \varepsilon_p^{\min}, \varepsilon_p^{\max}$ );
41:    return RIGHT TRIANGLE([5, 6, 7], Ray,  $\varepsilon_q^{\min}, \varepsilon_q^{\max}, \varepsilon_p^{\min}, \varepsilon_p^{\max}$ );
42:  end if
43:  return Ray;
44: end procedure

```

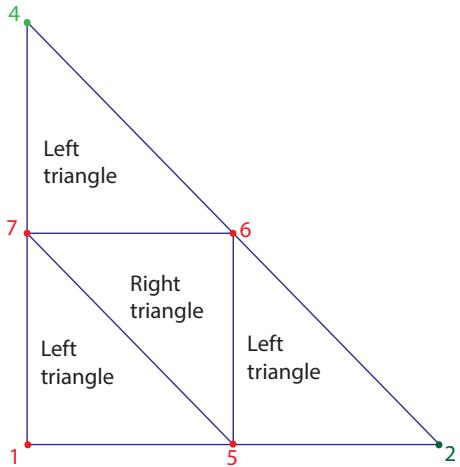


Figure 4.6: **Left triangulation refinement.** The algorithm used is the recursive function LEFT TRIANGLE.

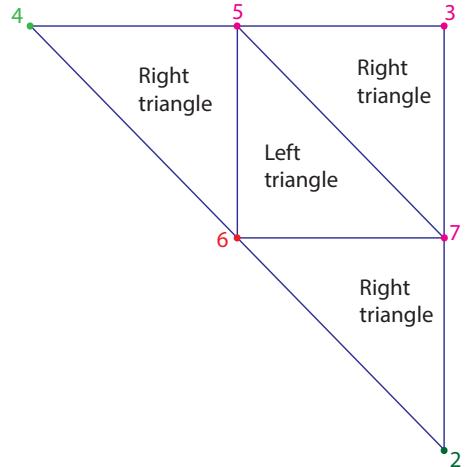


Figure 4.7: **Right triangulation refinement.** The algorithm used is the recursive function RIGHT TRIANGLE.

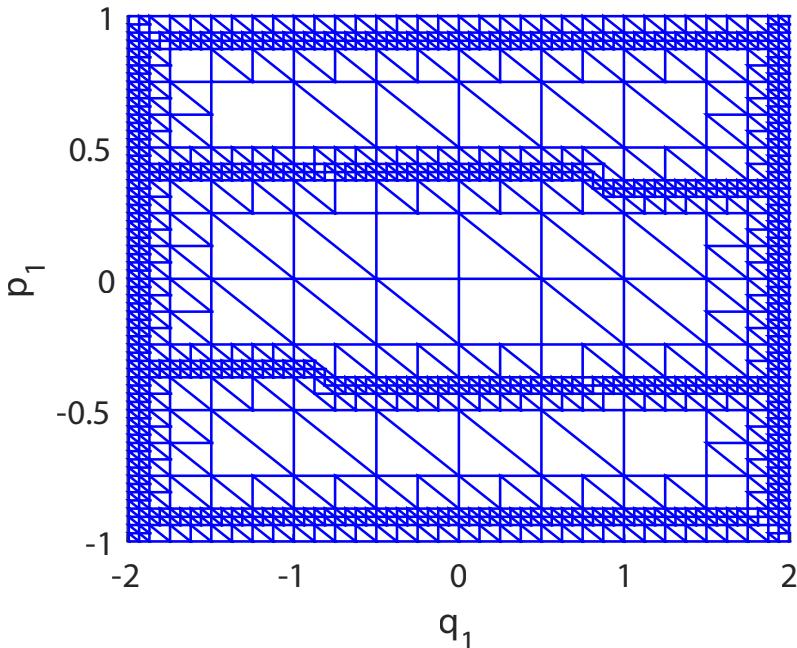
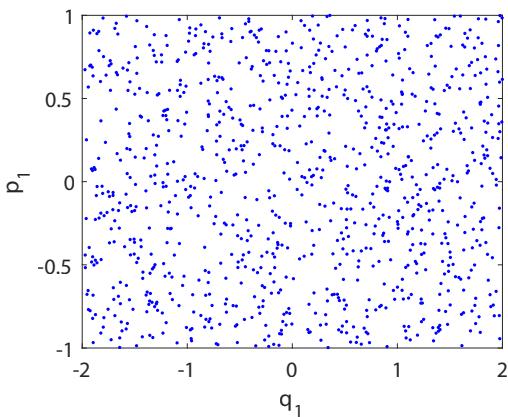
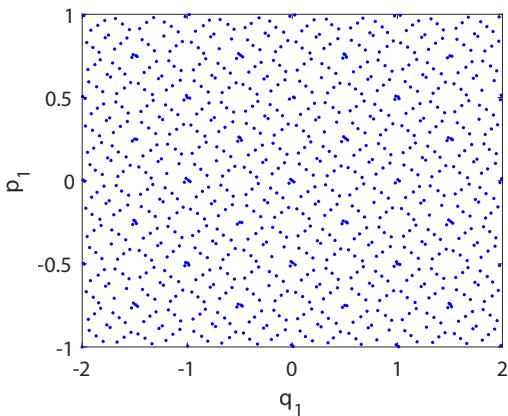


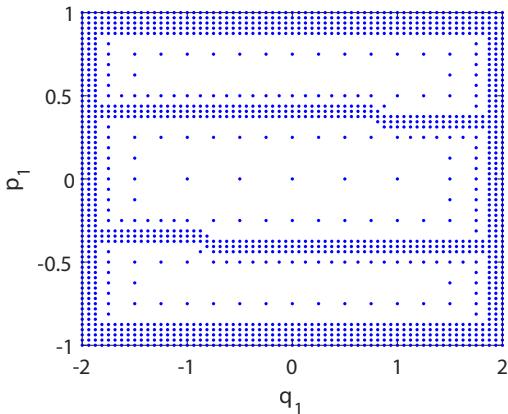
Figure 4.8: **Triangulation refinement of source phase space.** Near the boundaries more rays are traced. The values of the parameters are $\varepsilon_{q_1}^{\min} = 0.1$, $\varepsilon_{q_1}^{\max} = 1$, $\varepsilon_{p_1}^{\min} = \varepsilon_{q_1}^{\min}/2$ and $\varepsilon_{p_1}^{\max} = \varepsilon_{q_1}^{\max}/2$.



(a) **MC grid.** 10^3 rays randomly distributed (MC ray tracing).



(b) **QMC grid.** 10^3 rays distributed as the point of a Sobol sequence (QMC ray tracing).



(c) **PS grid.** $1.5 \cdot 10^3$ rays distributed using the triangulation refinement (PS ray tracing).

Figure 4.9: **Three different ray distributions.** Source PS of the two-faceted cup.

Chapter 5

The α -shapes approach

In the previous chapter we presented a new ray tracing approach based on PS. We explained that, in order to compute the target intensity, it is necessary to know the boundaries of the regions in target PS with positive luminance. Ray tracing in PS requires tracing only the rays close to these boundaries. The rays traced can be seen as a point cloud in PS. To detect the shape formed by those rays, the α -shapes approach is employed [68].

Methods based on α -shapes are widely used to reconstruct an unknown shape formed by a set of finite data points [69]. α -shapes is a very powerful tool to construct the shape of a point cloud. As the parameter α varies, we can obtain different α -shapes from the point set itself to the convex hull [70]. The disadvantage of such method is that it can be very hard to choose the appropriate value of the parameter α and, in most cases it can be selected only by trial-and-error.

We developed a technique based on α -shapes that gives a criterion to determine the value of the parameter α , for which the boundaries are approximated well [34].

This chapter is organized as follows. An overview of the-state-of-the-art about α -shape methods is provided in Section 5.1; the technique used for computing the α value is explained in Section 5.2; the results for two different kind of total internal reflection (TIR)-collimators are given in Section 5.3. Discussions and conclusions are provided in the last paragraph of this chapter.

5.1 α -shapes theory

Given a finite set $U = \{u_1, \dots, u_N\} \subset \mathbb{R}^2$ of points, α -shapes are geometrical objects that give us an approximation of the shape formed by the point cloud. For now we do not further specify the notion of shape. A more precise definition will be provided later.

Before giving a formal definition, we explain an intuitive and nice interpretation of α -shapes [71]. Let us think of a stracciatella ice-cream¹. If we desire to know the shape formed by the chocolate pieces we can start eating the ice cream using a spoon with a spherical scoop and try not to remove any piece of chocolate. We will obtain

¹Stracciatella ice cream is made with milk-based ice-cream and fine pieces of chocolate [72].

a shape formed by arcs and points. Straightening the arcs to line segments we obtain broken lines which constitute the boundary of the so-called α -shape of the point set U . A very small spoon will allow us to eat the entire ice cream without eating any piece of chocolate, while with a larger spoon we are not able to eat any chunk of the ice cream without chocolate pieces. In this example, the chocolates pieces are the points of set U and, the parameter α determines the radius of the carving spoon (the spherical spoon in two-dimension is simply a circle).

The formal definition of α -shape was first given by Edelsbrunner, Kirkpatrick and Seidel in 1983 [73]. They describe α -shape as a generalization of the convex hull of a finite set of points in the plane. Let α be a non negative number $0 \leq \alpha < \infty$. If $\alpha = 0$ the shape degenerates to the point set U . On the other hand, when $\alpha \rightarrow \infty$ the α -shape is simply the convex hull of U . If $0 < \alpha < \infty$ the α -shape is a polygon of U [74]. The construction of the α -shape is closely related to the Delaunay triangulation of U [75]. Therefore, a formal definition of triangulation and Delaunay triangulation is now required.

Given a set U of points not all aligned, let us consider the set E of all the straight-line segments whose endpoints are in U . A triangulation T of U is the subset of E with the maximum number of segments such that all the line segments of T intersect only at their endpoints [76].

Before giving a more formal definition of triangulation, let us define a partition of a set $X \subset \mathbb{R}^2$ as a collection of the subsets which divide X into non-overlapping regions so that any point in X is located in only one region.

Definition 5.1.1. *Let $P \subset \mathbb{R}^2$ be the convex hull of U and $T = \{T_1, \dots, T_h\}$ be a partition of P into closed triangles, that is triangles that include their edges. Suppose that the following properties hold:*

- a) $P = \bigcup_{i=1}^h T_i$,
- b) $\forall T_i, T_j \in T, T_i \neq T_j$, and

$$\text{int}(T_i) \cap \text{int}(T_j) = \emptyset,$$

where $\text{int}(T) = T - \partial T$,

then T is called a triangulation of P [77].

The Delaunay triangulation T' of the point set U has the property that the circumcircle of any triangle of T' does not contain any point of U . This is called the Delaunay property. A very commonly used algorithm to construct such triangulation is the following.

T' is constructed by modifying a general triangulation T such that every point satisfies the Delaunay property. Therefore, every triangle that does not satisfy such property is flipped such that the new edge is part of the triangulation, see Figure 5.1. Given, for example, an arbitrary triangulation T in two-dimensions, for each edge \overline{ab} in T which is not on the boundary of the convex hull the two triangles Δ_{abc} and Δ_{abd} with the common edge \overline{ab} are identified. Then, if either the circumcircle of triangle Δ_{abc} contains point d or the circumcircle of triangle Δ_{abd} contains point c , the edge \overline{ab} cannot be included in the Delaunay triangulation and, therefore, \overline{ab} is replaced by the edge \overline{cd} such that the triangles Δ_{acd} and Δ_{bcd} are constructed. The new edge \overline{cd}



(a) **Not acceptable triangle.** The point d is inside the circle circumscribing the triangle Δ_{abc} , therefore the edge \overline{ab} cannot be included in the Delaunay triangulation.

(b) **Acceptable triangle.** The flipped triangle Δ_{acd} satisfies the Delaunay property, thus it is included in the Delaunay triangulation.

Figure 5.1: Construction of the Delaunay triangulation in 2D.

locally satisfies the Delaunay property and the triangles Δ_{acd} and Δ_{bcd} are added to the Delaunay triangulation T' .

Several other algorithms have been developed to construct a Delaunay triangulation, see for example [78, 79]. Given a point set U and a triangulation T , it can be proved that the corresponding Delaunay triangulation T' is unique. Moreover, it maximizes the largest minimum angle among all possible triangulations of a point set U [80].

Alternatively, the Delaunay triangulation can be constructed as the dual of the Voronoi diagram [81]. Let $X \subset \mathbb{R}^2$ be a metric space endowed with the Euclidean distance $d(x, y)$ for $x, y \in X$. For *almost*² every point $x \in \mathbb{R}^2$, there is a unique point that is the closest point to x . The Voronoi cell of a point $u_i \in U$ contains all points in \mathbb{R}^2 that are closest to u_i , see Figure 5.2. The Voronoi diagram of $U \subset \mathbb{R}^2$ is defined as the set of all Voronoi cells [82]. A more formal definition of the Voronoi diagram is given in the following.

Definition 5.1.2. Let $U = \{u_1, \dots, u_N\}$ be a set of points in \mathbb{R}^2 . The Voronoi cell V_i associated to point u_i is defined as:

$$V_i = \{x \in \mathbb{R}^2 \mid |x - u_i| < |x - u_j| \quad \forall j \neq i\}. \quad (5.1.1)$$

The Voronoi diagram V is defined as

$$V = \bigcup_{i=1}^N V_i \quad (5.1.2)$$

where $V_i \cap V_j = \emptyset$ for $i \neq j$.

²Note the importance of the word *almost*. Some points can have the same distance with two or more points of U .

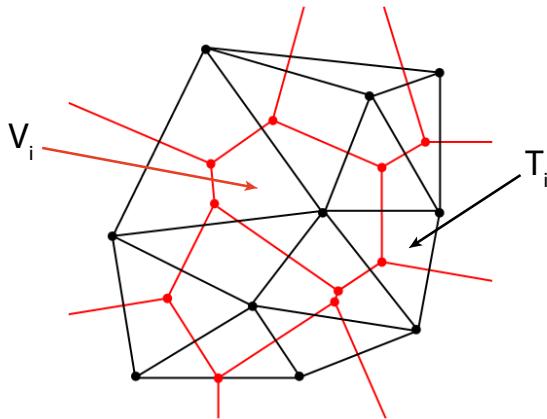


Figure 5.2: **Relationship between the Delaunay triangulation and the Voronoi Diagram** [84]. The black line segments are the boundaries of the Delaunay triangulation, the red line segments constitutes the boundaries of the Voronoi diagram.

For the definition of Voronoi diagram in higher dimensions see [83].

The Delaunay triangulation triangulates the convex hull of U and, therefore it does not constitute a suitable method for reconstructing the contour formed by a point cloud. The α -shape method was developed to solve such problem [85, 69]. Starting from the Delaunay triangulation T' of a point set U , the corresponding α -shape of V is formed by the only triangles of T' that satisfy the so-called " α -test" which is now briefly explained. For each triangle we calculate the circumradius, i.e., the radius of the circumcircle. If the radius is larger than α the triangle is removed from the shape. The choice of the parameter α is highly significant in the α -shapes procedure and, it has to be selected such that the desired approximation of the shape formed by the points of V is obtained.

To summarize, the α -shape construction can be outlined as follows:

1. Construct a Delaunay triangulation³ T' of the point cloud U ;
2. For every triangle $T'(i) \in T'$ calculate its circumradius $r(i)$;
3. If $r(i) \leq \alpha$ keep the triangle $T'(i)$ in the triangulation;
4. If $r(i) > \alpha$ remove the triangle from the triangulation;
5. Select from the new triangulation obtained those sides that belong to only one triangle, the so-called *free boundary edges*⁴. By definition, the free boundary edges are not a common edge of any two triangles.

α -shapes provide a nice mathematical definition of the *shape* of a set of points. In two dimensions, α -shapes gives the contour of the point cloud which is approximated by a

³In the simulations we present in this chapter the Matlab function *Delaunay* is used [86].

⁴In the simulations we present in this chapter the Matlab function *freeBoundary* is used [87].

family of line segments. Although they are a powerful tool for determining the shape of a point cloud, there exist shapes that are not described well by classical α -shapes. Indeed, for some point sets there is no value of α that gives a good approximation of the contour formed by the point cloud. Usually the parameter α is determined according to the density of the point cloud, therefore, it can be difficult to obtain a good approximation of a shape formed by a non-uniform point set. Furthermore, the α -shape method does not work well when the shape we need to approximate has a sharp turn or a joint, this case will be clarified in Section 5.3 with an example.

There are several ways to determine the value of α [88]; in the next section we provide a technique that exploits the conservation of étendue in PS.

5.2 Determination of α using étendue conservation

As mentioned in Section 4.1, in two-dimensions étendue can be seen as an area in PS. Therefore, given an optical system with a source $S = [-a, a]$, the étendue at the source coincides with the area of source PS, and it is given by:

$$U = 4n_1 a \sin(\theta_1^{\max}), \quad (5.2.1)$$

where a is the half length of the source, n_1 the index of refraction of the medium in which the S is located and θ_1^{\max} is the maximum value of the angle that the rays make with the normal ν_1 of the source.

For some optical systems, all the rays emitted by the source arrive at the target, for some others there are also rays that can end at other detectors which are located outside the system. Indicating with $R_1(\Pi)$ the regions in source PS formed by the rays that reach the target following path Π and with $R(\Pi)$ the corresponding regions at the target, the étendue U_1 at the source restricted to rays that arrive at the target is given by:

$$U(R_1(\Pi)) = \iint_{R_1(\Pi)} dq dp. \quad (5.2.2a)$$

$$U_1 = \sum_{\Pi} U(R_1(\Pi)), \quad (5.2.2b)$$

where $U(R_1(\Pi))$ is the contribution to the étendue given by the rays inside $R_1(\Pi)$ in source PS and the sum is over all possible paths Π from the source to the target. Similarly, the étendue at the target of the rays emitted by the source is:

$$U(R(\Pi)) = \iint_{R(\Pi)} dq dp. \quad (5.2.3a)$$

$$U_t = \sum_{\Pi} U(R(\Pi)), \quad (5.2.3b)$$

Note that, since both the source and the target are located in air ($n_1 = 1$), in (5.2.2a) and (5.2.3a), and from now on, we omit writing the index of refraction n_1 .

In order to determine the value of α in the α -shape procedure that approximates the boundaries $\partial R(\Pi)$ accurately, we use étendue conservation ($U_t = U_1$). The α -shapes method is applied to every region $R(\Pi)$ for a range of values of α ; for each value

an approximation of the boundaries $\partial R(\Pi)$ is obtained and the intersection points $q^{\max}(\Pi, p)$ and $q^{\min}(\Pi, p)$ between $\partial R(\Pi)$ and the horizontal lines $p = \text{const}$, with $p \in [-1, 1]$, are computed for every path Π . Therefore, Equation (5.2.3a) becomes:

$$U(R(\Pi)) = \int_{-1}^1 (q^{\max}(\Pi, p) - q^{\min}(\Pi, p)) dp. \quad (5.2.4)$$

In case more than two intersection points between the line $p = \text{const}$ and $\partial R(\Pi)$ occur, the previous equation needs to be generalized. Suppose that r intersection points $(q^i(\Pi, p), p)_{i=1,\dots,r}$ are found. Ordering their q -coordinates in ascending order, the target étendue is calculated by:

$$U(R(\Pi)) = \sum_{i=1}^m \int_{-1}^1 (q^{2i}(\Pi, p) - q^{2i-1}(\Pi, p)) dp, \quad (5.2.5)$$

where m is the integer part of $r/2$. The integrals in (5.2.4) and (5.2.5) are calculated discretizing the interval $[-1, 1]$ into $\text{Nb} = 100$ sub-intervals of equal length, the so-called bins, and using the trapezoidal rule.

Matching the étendue at the source U_1 with the étendue at the target U_t , a unique value α_c of α is determined. Implementing the α -shapes procedure with $\alpha = \alpha_c$, an approximation of the boundaries $\partial R(\Pi)$ is found and the intensity at the target can be calculated.

If two intersection points between $p = \text{const}$ and $\partial R(\Pi)$ are found the target intensity is calculated using (4.3.4). If more than two-intersection points are found we use the generalized equation:

$$I_{\text{PS}}(p) = \sum_{\Pi,i} \int_{q^{2i-1}(\Pi,p)}^{q^{2i}(\Pi,p)} L(q, p) dq = \sum_{\Pi,i} (q^{2i}(\Pi, p) - q^{2i-1}(\Pi, p)), \quad (5.2.6)$$

where $q^{2i}(\Pi, p) > q^{2i-1}(\Pi, p)$, the summation over Π is for all the paths Π for which the intersection $p = \text{const}$ and $R(\Pi)$ is not empty, and the summation over i is for $i = 1, 2, \dots, m$. The second equation holds as we assume $L(q, p) = 1$.

To clarify our idea we apply the method to two different optical systems, the results are presented next.

5.3 Results for a TIR-collimator

We apply the α -shapes method to the set of points in target PS obtained by using PS ray tracing. In this chapter the procedure is applied to two different kind of total internal reflection (TIR)-collimators.

Let us first describe the TIR-collimator depicted in Figure 5.3. It is an optical system symmetric with respect to the z -axis, it consists of a lens (central curve), two broken lines adjacent to the lens, two curved lines on each side and a top formed by a horizontal segment. The lens (line 2) and the broken lines, formed by a collection of three segments (lines 3, 4, and 5 and 9, 10 and 11), are refractive line segments while the curved lines (labeled with 6 and 8) are designed in such a way that light is totally internal reflected (which explains the name TIR). The light source S (line 1) and the

target T (line 12) are two straight line segments normal to the optical axis. The source $S = [-2, 2]$ is located at a height $z_1 = 0.3$ from the x -axis. The target $T = [-9.7, 9.7]$ is parallel to the source and is located at a height $z = 8.2$. Both S and T are located in air ($n_1 = 1$). The volume inside the collimator is filled with a material with index of refraction $n_2 = 1.5$ (e.g. glass). The collimator is surrounded by two vertical lines (lines 13 and 15) and two horizontal lines (12 and 14) that receive the light emitted from the source; among these the one at the top (line 12) is assumed to be the target, and it is located at a small distance from the top (line 7).

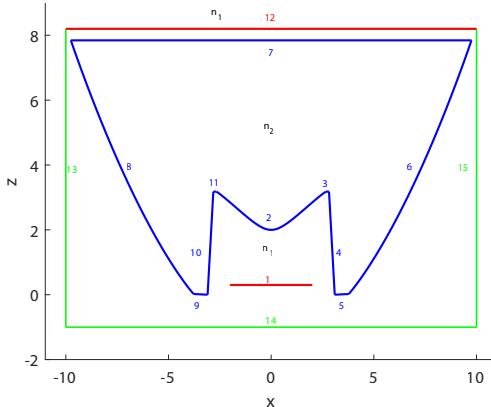


Figure 5.3: Shape of the TIR-collimator. Each line of the system is labeled with a number. The shape of the collimator is shown in blue. Three detectors depicted with green lines (surfaces 13, 14, and 15) are located at the left, the right and the bottom of the optical system. The source (line 1) and the target (line 12) are depicted in red.

Using PS ray tracing explained in Section 4.3 with parameters $\varepsilon_{q_1}^{\max} = 0.05/4$, $\varepsilon_{p_1}^{\max} = 0.05/8$, $\varepsilon_{q_1}^{\min} = 0.8/4$ and $\varepsilon_{p_1}^{\min} = 0.8/8$, around $1.67 \cdot 10^4$ rays are traced (see Table 5.1). The ray distribution at the source PS is shown in Figure 5.4, where we depicted the rays that follow the same path with the same color. Seven different paths are found. The yellow rays follow path $\Pi_1 = (1, 2, 7, 12)$; the red rays follow path $\Pi_2 = (1, 4, 6, 7, 12)$; the green rays follow path $\Pi_3 = (1, 10, 8, 7, 12)$; the blue rays follow path $\Pi_4 = (1, 3, 7, 12)$ and the magenta rays follow path $\Pi_5 = (1, 11, 7, 12)$. The rays located inside the white areas correspond to rays that do not reach the target, they follow either path $\Pi_6 = (1, 4, 7, 6, 15)$ or path $\Pi_7 = (1, 10, 7, 8, 13)$ and they do not give any contribution to the target intensity. Note that, given two adjacent paths, the corresponding regions in S have usually a common boundary. Since for this system not all the rays emitted by the source arrive at the target, U_t needs to be compared to the étendue U_1 at the source given by only those rays that reach the target (the rays that follow paths Π_6 and Π_7 are discarded). To this purpose U_1 is calculated by removing from the total area U of S those areas occupied by the regions formed by the rays that hit the left or the right detector (white regions in Figure 5.4). For the TIR collimator in Figure 5.3, U is obtained from (5.2.1), the source étendue

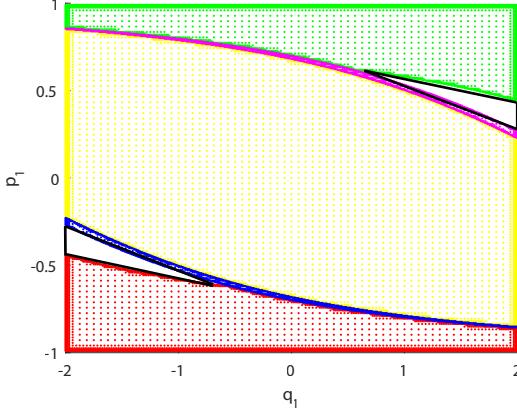


Figure 5.4: **Distribution of the rays on \mathbf{S} .** Around $1.67 \cdot 10^4$ rays are traced using the triangulation refinement with parameters: $\varepsilon_q^{\max} = 0.8/4$, $\varepsilon_p^{\max} = 0.08/8$, $\varepsilon_q^{\min} = 0.05/4$, $\varepsilon_p^{\min} = 0.05/8$. Rays that belong to the same region are depicted with the same color. The rays located inside the white areas do not reach the target. The boundaries of the two white regions are approximated by triangles depicted with black lines.

U_1 corresponding to the area covered by the rays that arrive at the target can be approximated by:

$$U_1 = U - 2A_T, \quad (5.3.1)$$

where $U = 8$ and A_T is the approximated area of the triangles shown in Figure 5.4 with black lines.

Next, U_t is calculated several times from (5.2.5) where every time the boundaries $\partial R(\Pi)$ are obtained by using α -shapes for a different value of α .

To clarify this concept, in Figure 5.5 we provide an example where the source étendue U_1 and target étendue U_t are computed from a set of around $1.67 \cdot 10^4$ rays. The approximated source étendue $U_1 \approx 7.77$ is depicted with the red line. The blue line shows how the étendue at the target changes as a function of α . The smallest difference $\Delta U = |U_1 - U_t|$ is obtained using $\alpha = \alpha_c = 0.08$. In Figure 5.6 we show the boundaries $\partial R(\Pi)$ in target PS with $\alpha_c = 0.08$ and tracing $1.67 \cdot 10^4$ rays. The target intensity $I_{PS}(p)$ for $p \in [-1, 1]$ is obtained from Equation (4.3.4).

To validate our method we compare the PS intensity with the QMC intensity. To this purpose a partitioning $P_2 : -1 = p_0 < p_1 < \dots < p_{Nb} = 1$ of the interval $[-1, 1]$ into $Nb = 100$ bins is considered. The averaged and normalized PS intensity \hat{I}_{PS} is calculated for every $(p^{h+1/2} = \frac{1}{2}(p^{h+1} + p^h))_{h=0, \dots, Nb-1}$ dividing the PS averaged intensity by the total étendue:

$$\hat{I}_{PS}(p^{h+1/2}) = \frac{1}{U_t} \int_{p_h}^{p_{h+1}} I_{PS}(p) dp. \quad (5.3.2)$$

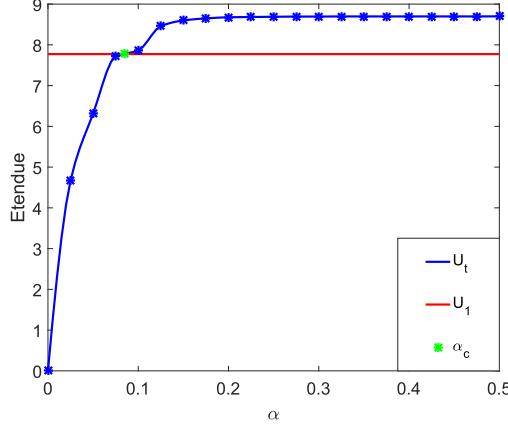


Figure 5.5: **Etendue for the TIR-collimator.** U_t is computed for a range of values for α . $U_1 \approx 7.77$ The green dot indicates the value of $\alpha_c = 0.08$ which gives a good approximation of the boundaries $\partial R(\Pi)$ at the target. Around $1.67 \cdot 10^4$ rays have been traced using PS ray tracing.

The averaged and normalized QMC intensity $(\hat{I}_{\text{QMC}}(p^{h+1/2}))_{h=0, \dots, \text{Nb}-1}$ is given by

$$\hat{I}_{\text{QMC}}(p^{h+1/2}) = \frac{\text{Nr}[p^h, p^{h+1}]}{\text{Nr}[-1, 1]} \quad \text{for } p \in [p^h, p^{h+1}). \quad (5.3.3)$$

Both approximate intensities $\hat{I}_A(A = \text{PS, QMC})$ are compared to an intensity \hat{I}_{ref} 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 QMC simulation with 10^7 rays is used to obtain the averaged normalized intensity \hat{I}_{ref} . The intensity profile \hat{I}_{PS} obtained using PS ray tracing with $8.3 \cdot 10^4$ rays and $\alpha = \alpha_c = 0.06$ is depicted in Figure 5.7 with a red line. \hat{I}_{PS} is hardly distinguishable from \hat{I}_{ref} which is indicated with the dashed and blue line in Figure 5.7.

Finally, we calculate the error between \hat{I}_A and \hat{I}_{ref} , defined as:

$$\text{error} = \frac{\sum_{h=1}^{\text{Nb}} |\hat{I}_A(p^{h+1/2}) - \hat{I}_{\text{ref}}(p^{h+1/2})|}{\text{Nb}}. \quad (5.3.4)$$

The QMC and PS intensities are calculated several times increasing the number of rays to improve the accuracy. Table 5.1 and 5.2 describe how the number of rays traced affects the error. In Table 5.1 the correlation between α_c and the number of rays is evident. Note that increasing the number of rays the value of α_c and the corresponding error decrease.

In Table 5.2 the numerical results of QMC ray tracing are reported. Increasing the number of rays traced, the error gradually decreases. In Figure 5.8, the results listed in Table 5.1 and Table 5.2 are shown. The red line depicts the convergence of the PS error and the blue line indicates the QMC error. We need to emphasize that the

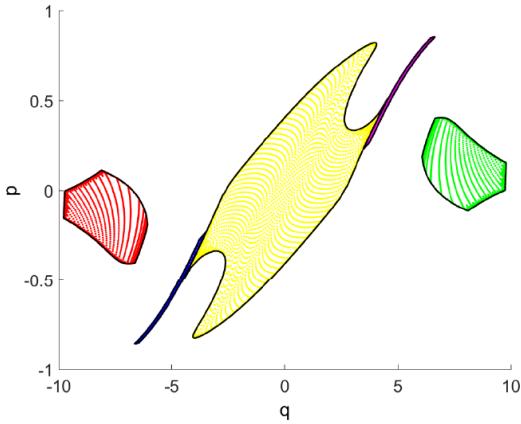


Figure 5.6: **Target PS representation.** A set of $1.67 \cdot 10^4$ rays are traced. Rays that follow the same path are depicted with the same color. The choice of the colors is consistent with Figure 5.4. The boundaries $\partial R(\Pi)$ are computed through the α -shapes method with $\alpha = \alpha_c = 0.08$.

Table 5.1: **Errors of the PS intensity**

Number of rays	ε_q^{\max}	ε_q^{\min}	ε_p^{\max}	ε_p^{\min}	α_c	PS error
3 339	0.8	0.05	0.8/2	0.05/2	0.14	$1.47 \cdot 10^{-3}$
7 567	0.8/2	0.05/2	0.8/4	0.05/4	0.10	$3.01 \cdot 10^{-4}$
16 755	0.8/4	0.05/4	0.8/8	0.05/8	0.08	$8.60 \cdot 10^{-5}$
83 005	0.8/16	0.05/16	0.8/32	0.05/32	0.06	$1.31 \cdot 10^{-5}$

convergence of the error of PS ray tracing for increasing Nr may change according to the design of the optical system. This is because the approximation of the boundaries in PS depends on the accuracy of the α -shapes method. The α -shapes procedure is unable to properly detect the boundaries of regions with a sharp turn if not enough points are given [33]. Indeed, on the one hand a low density requires a large value of α to accept the triangles in a region, on the other hand, choosing α too large, the shape of the region could be destroyed. Indeed, increasing α more triangles are kept and triangles inside the regions $R(\Pi)$ could be taken into account creating holes in $R(\Pi)$. Figure 5.9 clarifies this concept showing that the region formed by rays that hit the lens is hard to approximate when there is a small number of rays inside the region. Consequently either a region bigger than the area covered by the rays is obtained or some triangles which are not part of the boundaries are included in the triangulation. This results in an inaccurate intensity (either too high or to low). To obtain a good approximation of the boundaries of these kind of patches more rays have to be traced. The PS error decreases very fast increasing the number of rays (see Table 5.1 and Figure 5.8).

To show how the error plot changes according to the regularity of the shape of

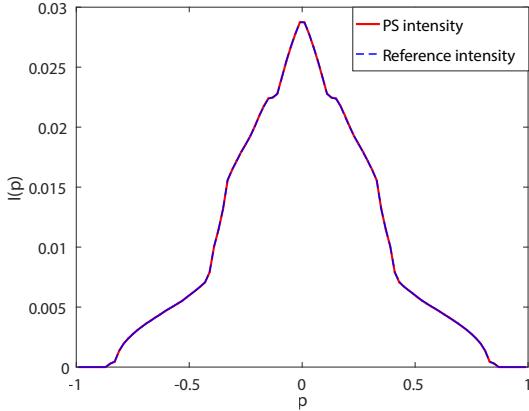


Figure 5.7: **Target intensity profile.** The exact intensity is computed using the QMC method for a set of 10^7 rays. For the PS intensity a set of $6.3 \cdot 10^4$ rays is considered and $\alpha_c = 0.06$ is chosen to compute the boundaries $\partial R(\Pi)$.

Table 5.2: **Error of the QMC intensity**

Number of rays	QMC error
10^3	$1.65 \cdot 10^{-3}$
10^4	$3.96 \cdot 10^{-4}$
10^5	$6.36 \cdot 10^{-4}$
10^6	$1.02 \cdot 10^{-5}$

the regions $\partial R(\Pi)$, we consider another example of a TIR-collimator. Figure 5.9 shows that the hardest region to approximate is given by those rays that follow path $\Pi_1 = (1, 2, 7, 12)$. We therefore consider a TIR-collimator with a flatter lens and with the target located at a smaller distance to the top (see Figure 5.10). The source $S = [-2, 2]$ (surface number 1) is located in air at a height $z_1 = 0.3$ from the x -axis. The target $T = [-9.7, 9.7]$ (surface 12) is parallel to the source and is located in air at a height $z = 7.85$. 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.

Tracing around $3 \cdot 10^3$ rays using PS ray tracing, we obtain the target rays distribution shown in Figure 5.11. Compared with the distribution in Figure 5.6, we note that the extremities at top and bottom of the region formed by the rays that hit the lens are less pronounced. Moreover a target located very close to the top makes the shape of that region less stretched along the q -axis. Therefore, it is expected that the α -shapes method performs better for a small number of rays.

PS and QMC ray tracing are implemented for the TIR-collimator in Figure 5.10. The approximated intensities \hat{I}_A ($A = \text{PS}, \text{QMC}$) are compared to the reference intensity \hat{I}_{ref} (QMC ray tracing with 10^7 rays). PS error is depicted with the red line and, QMC error is depicted with the blue line.

Numerical results show that, for the TIR-collimator in Figure 5.10, the computa-

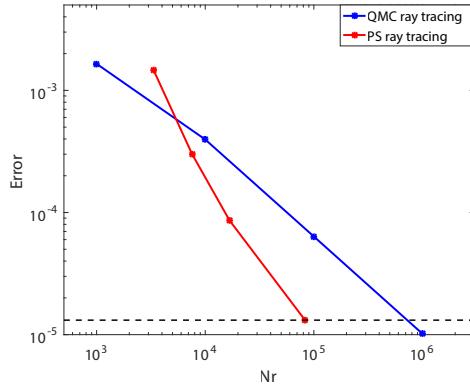


Figure 5.8: **PS and QMC errors as a function of the number of rays** The horizontal dotted line shows that an error equal to $1.31 \cdot 10^{-5}$ can be obtained tracing almost 10 times fewer rays in phase space.

tional time needed to achieve an error of the order of 10^{-5} is reduced using PS ray tracing compared with QMC ray tracing.

5.4 Conclusion

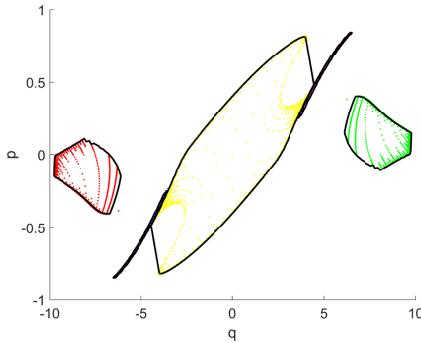
The aim of this chapter was using α -shapes to detect the boundaries of the regions formed by the rays traced.

First, we reported some theory about α -shapes which are commonly used to approximate the shape formed by a point cloud. These methods depend on a parameter α that in most cases can be determined only by simulations.

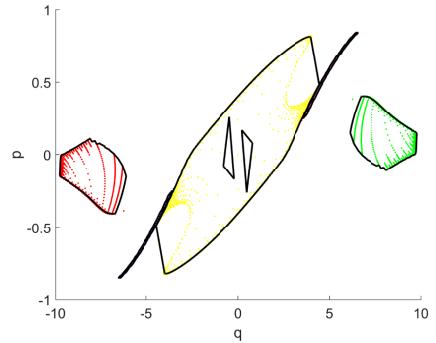
Using étendue conservation, we developed a new approach to detect the value of α that better approximates the boundaries in target PS. We applied α -shapes to two different kind of TIR-collimators. The target PS intensity was computed for both systems several times increasing every time the number of rays traced. Finally, the corresponding errors between the approximated intensities and a reference intensity was calculated. We observed that PS ray tracing allows tracing far less rays compared to QMC ray tracing. Numerical results show that using PS ray tracing the desired accuracy can be achieved reducing significantly the number of rays traced.

However, we observed that the error convergence for PS ray tracing strongly depends on the design of the optical system (shapes of the region in target PS). Indeed, the accuracy of the intensity is related to the precision of the α -shape, that is, to the choice of the parameter value of α . For more complicated shapes in PS, more rays need to be traced for a good boundaries reconstruction.

In order to remove the dependence of PS ray tracing on the parameter α , we will construct another procedure to detect the boundaries of the regions in target PS. The new technique is based on the triangulation refinement explained in Section 4.3. The details are explained in the next chapter and numerical results are reported for



(a) Boundaries approximation obtained using the α -shapes method with $\alpha_c = 0.3$ (black lines).



(b) Boundaries approximation obtained using the α -shapes method with $\alpha_c = 0.31$ (black lines).

Figure 5.9: **Approximated boundaries at the target PS.** Tracing 3339 rays and using α -shapes, the boundaries cannot be approximated well. A small change of the parameter α leads to a completely different approximation of the boundaries.

several optical systems.

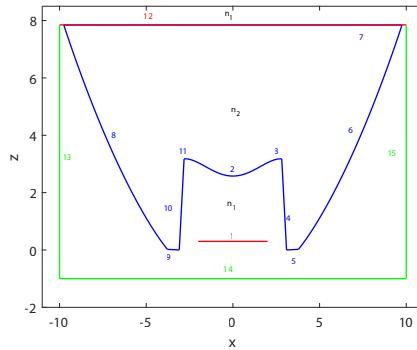


Figure 5.10: **Shape of the TIR-collimator.** Each line of the system is labeled with a number. $n_1 = 1$ is the refraction index of the medium (air) where the source and the target are located, and $n_2 = 1.5$ the refraction index of the medium (glass) inside the optical system.

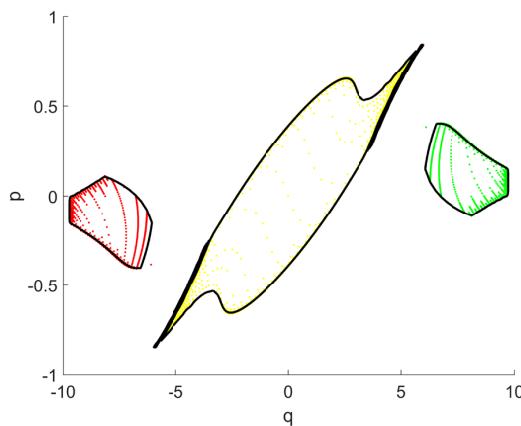


Figure 5.11: **Target phase space for the TIR-collimator depicted in Figure 5.10.** The black line depicts the best approximation of $\partial R(\Pi)$ for $3 \cdot 10^3$ rays. The α -shapes method gives an accurate approximation of the boundaries for $\alpha_c = 0.9$.

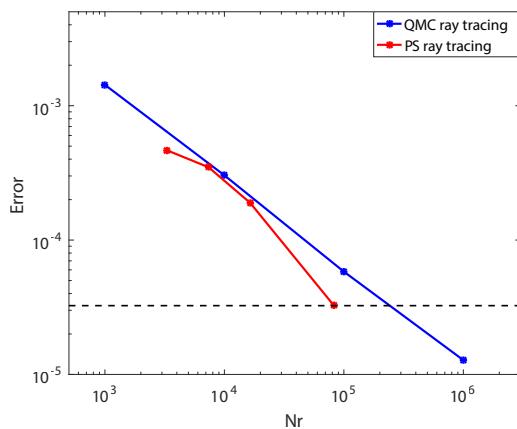


Figure 5.12: **PS and QMC errors.**

Chapter 6

Boundaries reconstruction based on the triangulation refinement

The purpose of this chapter is to provide an alternative approach to the α -shapes methods for determining the boundaries $\partial R(\Pi)$ of the regions with positive luminance in target PS. The idea of the method is based on the triangulation refinement of the source PS explained in Chapter 4. The boundaries $\partial R(\Pi)$ are approximated by connecting those vertices of the triangles that follow the path same Π . Numerical results are provided for three optical systems: the two-faceted cup, the TIR-collimator and a parabolic reflector.

The PS method is compared to both MC and QMC ray tracing. Discussion and results are provided in the last section of this chapter.

6.1 Reconstruction of the boundaries

In Chapter 4 we have seen that, using the triangulation refinement, more rays close to the boundaries are traced selecting increasingly smaller values for the parameters $\varepsilon_{q_1}^{\min}$ and $\varepsilon_{p_1}^{\min}$. Once the algorithm stops, only the triangles that are expected to be crossed by at least a boundary $\partial R(\Pi)$ are taken into account for the construction of the boundary. From now on we call these triangles the *boundary triangles*. Two triangles are neighbors if they have one side in common. For each boundary triangle its neighbor is found so that an ordered sequence of triangles is constructed. Given a path Π , the corresponding boundary $\partial R_1(\Pi)$ on S is approximated connecting the vertices of the boundary triangles which correspond to rays following path Π . The edge-ray principle is employed in order to define the corresponding boundaries $\partial R(\Pi)$ at the target. Thus, $\partial R(\Pi)$ at the target are given by

$$M(\partial R_1(\Pi)) : \partial R_1(\Pi) \rightarrow \partial R(\Pi), \quad (6.1.1)$$

where M is defined in (4.3.1) and $M(\partial R_1(\Pi))$ is the restriction of M to $\partial R_1(\Pi)$ for every path Π .

In this chapter we develop a criterion to establish the value of the parameters $\varepsilon_{q_1}^{\min}$, $\varepsilon_{q_1}^{\max}$, $\varepsilon_{p_1}^{\min}$ and $\varepsilon_{p_1}^{\max}$ which gives a good approximation of $\partial R(\Pi)$. Similar to the selection of α in the α -shapes procedure, the triangulation parameters are chosen using étendue conservation, i.e., conservation of area in PS. The core of our approach is the following.

The étendue U_1 at the source PS restricted to the rays that arrive at the target is calculated. If all the rays emitted by the source are received by the target, U_1 can be easily determined by (5.2.1). In case some rays do not arrive at the target and rather reach other detectors, we use (5.2.2a) and (5.2.2b).

The étendue U_t at the target PS is calculated using (5.2.3a) and (5.2.3b). To do so, the triangulation refinement method is applied to the regions $R(\Pi)$ for a range of values of $\varepsilon_{q_1}^{\max}$ and for a fixed value of $\varepsilon_{q_1}^{\min}$. The parameters along the p -axis are scaled by

$$\begin{aligned} w &= \frac{q_1^{\max} - q_1^{\min}}{p_1^{\max} - p_1^{\min}}, \\ \varepsilon_{p_1}^{\min} &= \frac{\varepsilon_{q_1}^{\min}}{w}, \\ \varepsilon_{p_1}^{\max} &= \frac{\varepsilon_{q_1}^{\max}}{w}, \end{aligned} \tag{6.1.2}$$

where p_1^{\min} and p_1^{\max} are the minimum and the maximum p -coordinate in S , and q_1^{\min} and q_1^{\max} are the minimum and the maximum q -coordinate in S . Every set of parameters gives a certain triangulation, for each of them an approximation of the boundaries $\partial R(\Pi)$ is obtained. Next, the intersection points $(q^i(\Pi, p), p)_{i=1,\dots,r}$ between $\partial R(\Pi)$ and the horizontal line $p = \text{constant}$ are calculated for every path Π , and for $p \in [-1, 1]$. Ordering their q -coordinates corresponding to each direction p in ascending order, the integral in Equation (5.2.5) is computed. Changing the values of the parameters, different approximations of $\partial R(\Pi)$ are found and, consequently, different values of U_t . By construction, U_t is always underestimated ($U_t < U_1$) because the approximated boundaries are found joining the vertices of the *boundary triangles* which are *inside* the regions $R(\Pi)$.

To use the parameters that give a good accuracy of the target photometric variables, the difference $\Delta U = U_1 - U_t$ is calculated for every value of U_t found. The values of the parameters that give a small ΔU provide a triangulation refinement from which a good approximation of the target photometric variables can be computed.

A similar method as described here is presented by Moore [89]. In Moore's method each ray leaves a point source at the same position while the angle coordinate changes. It starts considering three sampling rays and their corresponding paths are taken into account. In case the paths are equal the rays traced are representative of all the rays inside the polygon that they describe at the target, otherwise an interpolation is required to compute the illumination pattern. This interpolation can affect the efficiency of the method. Our method employs the distribution of the rays at the target PS and avoids using any interpolation. Moreover, a criterion to stop our algorithm is provided in such a way that no more rays than necessary are traced. This makes ray tracing in PS more accurate compared to Moore's procedure. Furthermore, Moore method is not suitable for systems in which the size and the spatial distribution of the source is important as it consider only point sources.

The triangulation refinement method is tested for several optical systems. The results are presented next.

6.2 The two-faceted cup

In this section we apply the triangulation refinement in PS to the two-faceted cup described in Chapter 3 and depicted in Figure 3.1. We start tracing rays inside the system using PS ray tracing as explained in Chapter 4. To avoid rays parallel to the source and rays emitted from the endpoints, we consider their initial position q_1 and initial direction p_1 such that

$$p_1 \in [-1 + 10^{-6}, 1 - 10^{-6}] = [p_1^{\min}, p_1^{\max}], \\ q_1 \in [-2 + 10^{-12}, 2 - 10^{-12}] = [q_1^{\min}, q_1^{\max}].$$

A stopping criterion for the triangulation is defined using étendue conservation. Since the two-faceted cup is formed by only reflective lines and its target is adjacent to the left and the right reflector (it is located exactly at the top of the system), all the rays emitted by the source arrive at the target. Thus, from (5.2.1) with $n_1 \sin(\theta_1^{\max}) = p_1^{\max}$ and $a = q_1^{\max}$ follows

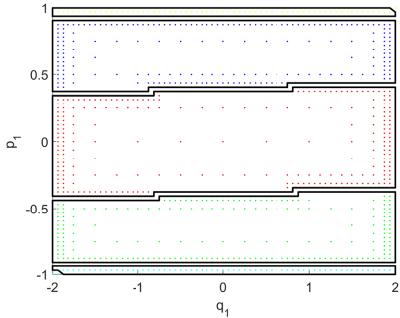
$$U_1 = U \approx 8. \quad (6.2.1)$$

Ray tracing in PS is implemented by varying the parameter $\varepsilon_{q_1}^{\min}$, and fixing $\varepsilon_{q_1}^{\max}$ (we choose $\varepsilon_{q_1}^{\max} = 1$), while the other two parameters are given by (6.1.2). Every set of parameters gives a different triangulation at the source PS. The approximated boundaries are computed for several triangulations joining the vertices of the triangles crossed by a boundary.

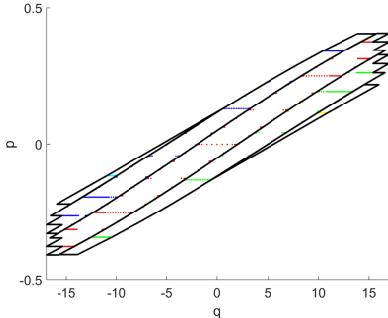
For example, if we consider $\varepsilon_{q_1}^{\min} = 0.1$, $\varepsilon_{q_1}^{\max} = 1$ and the corresponding parameters for the p -axis given by (6.1.2), a triangulation with around 1500 rays (vertices of the triangles) is found. The boundaries $\partial R_1(\Pi)$ and $\partial R(\Pi)$ are calculated using this triangulation refinement and are depicted in black in Figures 6.1a and 6.1b, respectively. For this set of rays we found $\Delta U \approx 0.53$. Next, we can decrease $\varepsilon_{q_1}^{\min}$ to obtain a more precise approximation of U_t . Choosing $\varepsilon_{q_1}^{\min} = 0.025$ and $\varepsilon_{q_1}^{\max} = 1$, a triangulation formed by around 7500 rays is obtained. The approximated boundaries $\partial R_1(\Pi)$ and $\partial R(\Pi)$ are depicted with black lines in Figures 6.1c and 6.1d, respectively. The approximation of the target étendue gives $\Delta U \approx 0.13$. Obviously, the boundaries computation obtained using $\varepsilon_{q_1}^{\min} = 0.025$ is more accurate. Note that, decreasing $\varepsilon_{q_1}^{\min}$, the number of rays increases.

In Figure 6.2 we show with the blue line how the target étendue varies as a function of the parameter $\varepsilon_{q_1}^{\min}$. The exact étendue $U = 8$ is depicted with the red line and it is computed using Equation (5.2.5). By decreasing $\varepsilon_{q_1}^{\min}$ an increase of U_t is observed. In Figure 6.2 we show the approximation of U_t for at most around $1.2 \cdot 10^5$ rays traced using PS ray tracing with parameters $\varepsilon_{q_1}^{\min} = 0.8 \cdot 10^{-4}$, $\varepsilon_{q_1}^{\max} = 1$, $\varepsilon_{p_1}^{\min} = \varepsilon_{q_1}^{\min}/2$ and $\varepsilon_{p_1}^{\max} = \varepsilon_{q_1}^{\max}/2$. We expect that further decreasing $\varepsilon_{q_1}^{\min}$ the value of U_t becomes more precise.

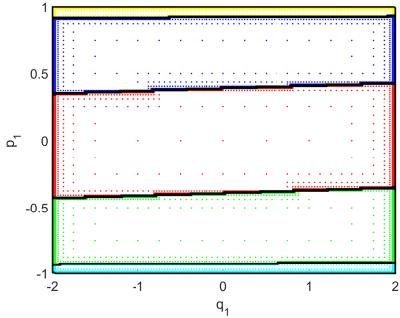
The PS intensity \hat{I}_{PS} with $1.2 \cdot 10^5$ rays is calculated from Equation (4.3.4). The intensity profile is shown in Figure 6.3 with the red line. In the same graph we show the reference intensity \hat{I}_{ref} with the dotted blue line. For the two-faceted cup the



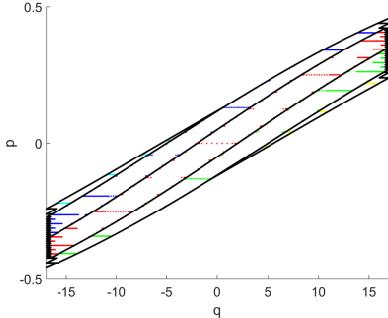
(a) The black lines are the boundaries at S. 1500 rays are traced using the triangulation refinement with $\varepsilon_{q_1}^{\min} = 0.1$.



(b) The black lines are the boundaries at T. 1500 rays are traced using the triangulation refinement with $\varepsilon_{q_1}^{\min} = 0.1$.



(c) The black lines are the boundaries at S. 7500 rays are traced using the triangulation refinement with $\varepsilon_{q_1}^{\min} = 0.025$.



(d) The black lines are the boundaries at T. 7500 rays are traced using the triangulation refinement with $\varepsilon_{q_1}^{\min} = 0.025$.

Figure 6.1: **Boundaries at S and T of the two-faceted cup.** The approximated boundaries are computed using the triangulation refinement with two different values of $\varepsilon_{q_1}^{\max}$.

reference intensity is actually the exact intensity ($\hat{I}_{\text{ref}} = \hat{I}_{\text{exact}}$).

Finally, we compare PS ray tracing with both MC and QMC ray tracing by computing the error between the approximated intensities \hat{I}_A ($A = \text{MC}, \text{QMC}, \text{PS}$) and the exact intensity \hat{I}_{ref} using (5.3.4). For the error calculation we use (5.3.4) with $Nb = 100$. The results are shown in Figure 6.4 where the MC, QMC and PS intensity are depicted with the green, blue and red line, respectively. The graph shows that using PS ray tracing in combination with the triangulation refinement admits tracing far less rays compared to MC ray tracing. A comparison between QMC ray tracing and PS ray tracing shows that more rays are needed in PS. Indeed, although the shapes of all the regions $R(\Pi)$ are very smooth, their boundaries at the edge of the target phase space T are difficult to approximate by triangles. With the triangulation refinement, the vertical straight lines at the edge of T are always approximated by a broken line. However, the two-faceted cup is a very simple system and QMC ray tracing does not require a large number of rays to obtain the desired accuracy.

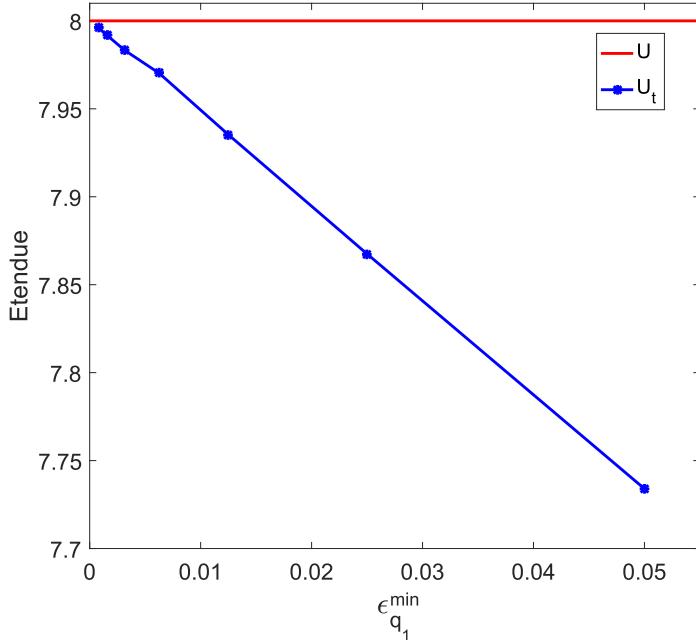


Figure 6.2: **Etendue for the two-faceted cup.** The total étendue as an area in PS is depicted with the red line. The approximated étendue for a range of values of $\epsilon_{q_1}^{\min}$ is shown with the blue line.

Nevertheless, PS ray tracing has a big advantage compared to QMC ray tracing. Indeed, as we have seen in Chapter 3, MC and QMC ray tracing are binning procedures. Therefore, the MC and QMC intensities are given by the average over every bin and the error also depends on the number of bins. PS ray tracing gives a pointwise intensity along all possible directions. In the simulations shown in this thesis we always compute the average PS intensity. This is needed to give a fair comparison of PS ray tracing versus MC and QMC ray tracing. It is very important to observe that no error related to the number of bins is involved in the PS procedure.

To investigate in more detail the performance of PS ray tracing, we test the method for more complicated systems. In the next paragraph we present the results for a TIR-collimator.

6.3 A TIR-collimator

In this section we provide the results of PS ray tracing for a TIR-collimator, using the triangulation refinement to compute the boundaries $\partial R(\Pi)$ in target PS. In particular, we consider the TIR-collimator depicted in Figure 5.10. Since this system is located in two different media (air and glass), also the refraction law plays a role in the ray tracing procedure. We run PS ray tracing for the TIR-collimator several times



Figure 6.3: **Intensity profile at the target of the two-faceted cup.** The reference intensity is the exact intensity. The PS intensity is computed using the triangulation refinement with $\varepsilon_{q_1}^{\min} = 0.8 \cdot 10^{-4}$, $\varepsilon_{q_1}^{\max} = 1$, $\varepsilon_{p_1}^{\min} = \varepsilon_{q_1}^{\min}/2$ and $\varepsilon_{p_1}^{\max} = \varepsilon_{q_1}^{\max}/2$. Around $1.2 \cdot 10^5$ rays are traced.

gradually increasing the number of rays, i.e., gradually decreasing the values of the parameters $\varepsilon_{q_1}^{\min}, \varepsilon_{q_1}^{\max}, \varepsilon_{p_1}^{\min}$ and $\varepsilon_{p_1}^{\max}$ in the triangulation. In order to trace more rays close to the boundaries, we decide to vary only the values of $\varepsilon_{q_1}^{\min}$ and $\varepsilon_{p_1}^{\min}$ while fixing the values of $\varepsilon_{q_1}^{\max}$ and $\varepsilon_{p_1}^{\max}$ as the last two are responsible of the number of rays inside the regions $R(\Pi)$. Every ray traced has initial position coordinate $q_1 \in [-a, a]$ with $a = 2$ and the initial direction coordinate $p_1 \in [-1, 1]$. Therefore, the source PS of the TIR-collimator is the rectangular domain $S = [-2, 2] \times [-1, 1]$. The parameters $\varepsilon_{q_1}^{\min}$ and $\varepsilon_{p_1}^{\max}$ are scaled as in Equation (6.1.2).

To determine the triangulation refinement that gives a good approximation of the target intensity we compare U_1 (source étendue) to U_t (target étendue) and use étendue conservation. In this case, not all light emitted by the source of the TIR-collimator arrives at the target. Indeed, using PS ray tracing, $N_p = 7$ different paths $(\Pi_j)_{j=1,\dots,N_p}$ are found but only five of them are paths from the source (line 1) to the target (line 12), see also Section 5.3. Thus, we need to remove from the total area of S those parts occupied by the rays that arrive at some others detectors and not at the target. Indicating with A_T the area of each of these parts (see Section 5.3), the source PS is given by:

$$U_1 = 8 - 2A_T \approx 7.77. \quad (6.3.1)$$

The target étendue U_t is obtained from Equation (5.2.4) for a range of values of $\varepsilon_{q_1}^{\min}$ and for $\varepsilon_{q_1}^{\max} = 1$ fixed. The boundaries $\partial R(\Pi)$ are found for every value of $\varepsilon_{q_1}^{\min}$ and, U_t is calculated for each of these boundaries. The results shown in Figure 6.5 give

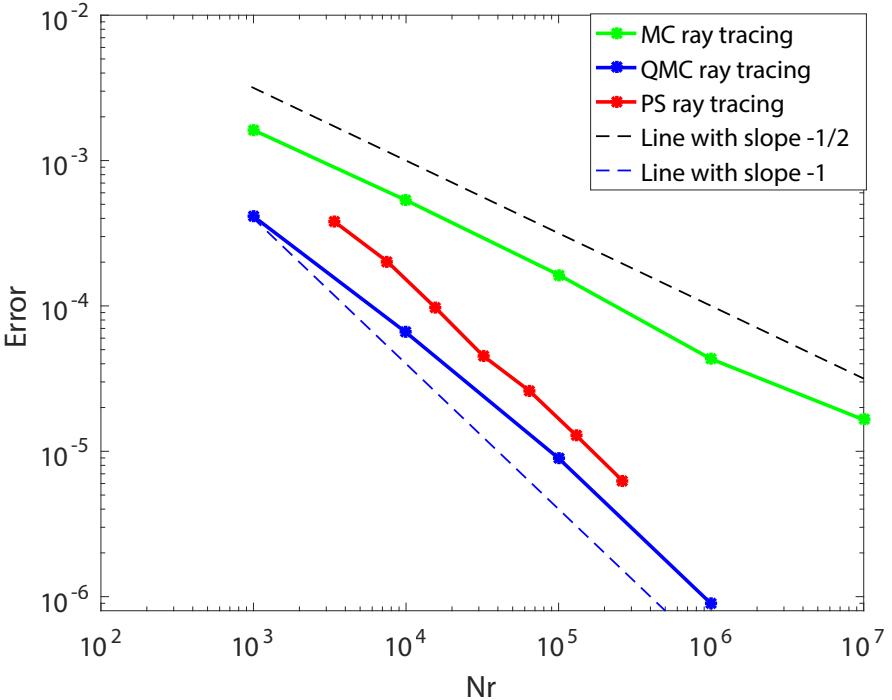


Figure 6.4: **Error plot for the two-faceted cup.** The errors between the approximated intensities \hat{I}_A ($A = MC, QMC, PS$) and the exact intensity \hat{I}_{exact} .

the étendue plot as a function of $\varepsilon_{q_1}^{\min}$.

The best approximation of U_t shown in the previous graph is obtained using $\varepsilon_{q_1}^{\min} = 1.6 \cdot 10^{-3}$, tracing around $1.62 \cdot 10^5$ rays. The boundaries $(\partial R_1(\Pi_j))_{j=1,\dots,5}$ and $(\partial R(\Pi_j))_{j=1,\dots,5}$ of the regions formed by these rays are shown in Figure 6.6 with red lines (see also Figure 5.11 for comparison).

The target PS intensity \hat{I}_{PS} is computed and is compared with a reference intensity \hat{I}_{ref} which is given by QMC ray tracing with 10^7 rays (as the exact intensity for the TIR-collimator is unknown). The profile of the two intensities is given in Figure 6.7.

To validate our method, PS ray tracing is compared to both MC and QMC ray tracing. The error between the approximated intensities \hat{I}_A ($A = QMC, MC, PS$) and the reference intensity \hat{I}_{ref} as a function of the number of rays traced is calculated. The error plot is shown in a logarithmic scale in Figure 6.8 where the MC, QMC and PS convergences are shown with the green, the blue and the red line, respectively. The black dotted line is a line with slope $-\frac{1}{2}$, the blue dotted line has slope -1 . The graph shows that MC ray tracing converges, for $N_r \rightarrow \infty$, with an order of $\mathcal{O}\left(\frac{1}{\sqrt{N_r}}\right)$, while both PS and QMC ray tracing have a speed of convergence of the order $\mathcal{O}\left(\frac{1}{N_r}\right)$. Note that PS ray tracing allows tracing 10^2 times less rays compared to MC ray tracing and almost 10 times less rays compared to QMC ray tracing to obtain an error

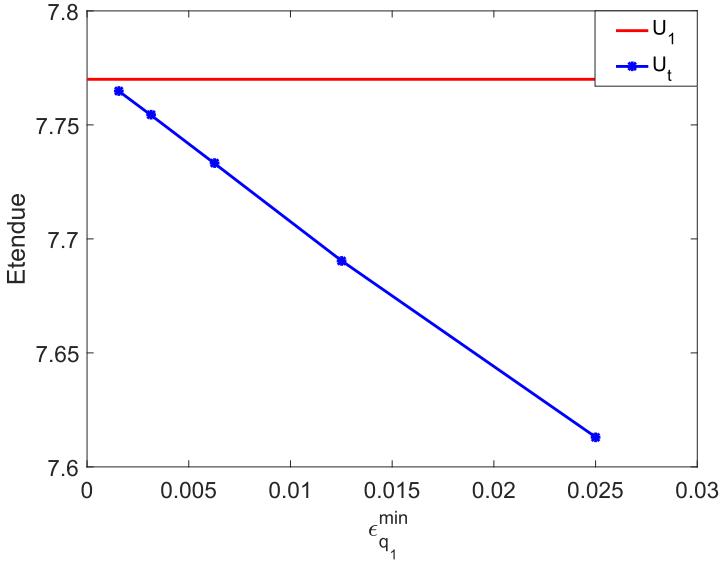


Figure 6.5: **Etendue of the TIR-collimator.** A comparison between U_1 and U_t shows that by decreasing the value of $\epsilon_{q_1}^{\min}$, $\Delta U = U_1 - U_t$ decreases.

Table 6.1: **Errors of the PS intensity for the TIR-collimator**

ϵ_g^{\max}	Nr	Etendue	PS error	PS CPU-time (sec.)
0.05	3 547	7.50	$1.75 \cdot 10^{-4}$	1.98
0.025	8 055	7.61	$1.49 \cdot 10^{-4}$	4.69
0.125	17 300	7.69	$8.68 \cdot 10^{-5}$	10.61
$6.3 \cdot 10^{-3}$	38 300	7.73	$4.43 \cdot 10^{-5}$	26.56
$3.1 \cdot 10^{-3}$	79 600	7.75	$2.27 \cdot 10^{-5}$	83.21
$1.6 \cdot 10^{-3}$	162 300	7.76	$1.20 \cdot 10^{-5}$	240.53

of 10^{-4} .

Finally, in order to show the advantages of PS ray tracing in terms of the computational time, we provide an error convergence as a function of the CPU-time for all the three methods (MC, QMC and PS raytracing) shown in Figure 6.9. The choice of the colors is consistent with Figure 6.8. We observe that PS ray tracing outperforms both MC and QMC ray tracing. PS ray tracing is approximately 10^2 times faster than MC ray tracing and 10 times faster than QMC ray tracing.

The results shown in Figures 6.8 and 6.9 are reported in Tables 6.1 and 6.2. Next, we show the result for a system in which more than 5 paths are possible. In particular we present the results for an optical system for which multiple reflections between rays and the mirrors occur.

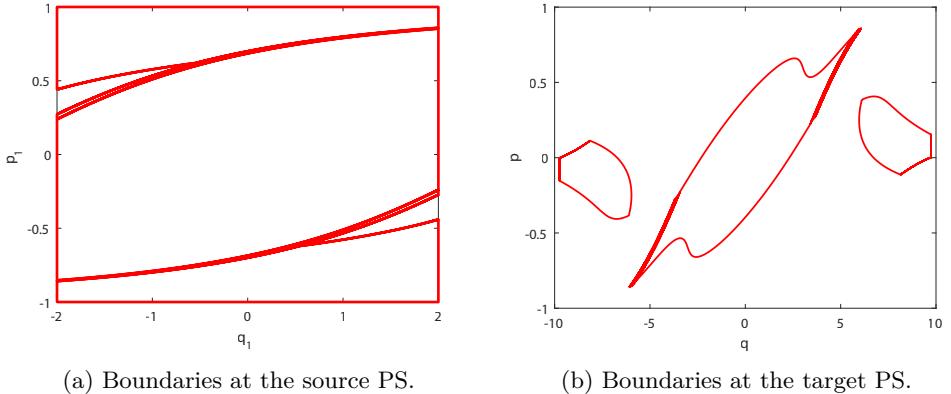


Figure 6.6: **Boundaries at S and T of the TIR-collimator.** The red lines show the boundaries found using $\varepsilon_{q_1}^{\min} = 1.6 \cdot 10^{-3}$, $\varepsilon_{q_1}^{\max} = 1$, $\varepsilon_{p_1}^{\min} = \varepsilon_{q_1}^{\min}/2$ and $\varepsilon_{p_1}^{\max} = \varepsilon_{q_1}^{\max}/2$.

Table 6.2: **Errors of the MC and QMC intensities for the TIR-collimator**

Nr	MC error	MC CPU-time (sec.)	QMC error	QMC CPU-time (sec.)
10^3	$2.09 \cdot 10^{-3}$	2.73	$1.43 \cdot 10^{-3}$	2.63
10^4	$6.42 \cdot 10^{-4}$	25.98	$3.03 \cdot 10^{-4}$	25.84
10^5	$1.92 \cdot 10^{-4}$	259.92	$5.82 \cdot 10^{-5}$	258.28
10^6	$7.45 \cdot 10^{-5}$	2585.83	$1.28 \cdot 10^{-5}$	2482.67

6.4 A Parabolic reflector

In this section we show an example of a parabolic reflector which is depicted in Figure 6.10. It consists of a source S (line 1), a target T (line 4) parallel to S and two reflectors (lines 2 and 3) which are arcs of the same parabola. The minimum of the parabola is located at the point with x -coordinate equal to 0. $S = [-a, a]$ (with $a = 2$) and $T = [-b, b]$ (with $b = 17$) are lines perpendicular to the optical axis (z -axis) and are located at $z = 0$ and $z = 40$, respectively. All the optical lines are located in air, therefore the index of refraction $n = 1$ for every line. The optical axis of the system in Figure 6.10 corresponds to the z -axis. We trace rays in PS with source direction coordinates $p_1 \in [-1, 1]$ and source position coordinates $q_1 \in [-a + \varepsilon, a - \varepsilon]$ where $\varepsilon > 0$ is a small number. In particular we take $\varepsilon = 10^{-12}$.

As an example we show the triangulation refinement obtained for the parameters

$$\varepsilon_{q_1}^{\min} = 0.025, \varepsilon_{q_1}^{\max} = 0.5, \varepsilon_{p_1}^{\min} = \varepsilon_{q_1}^{\min}/2, \text{ and } \varepsilon_{p_1}^{\max} = \varepsilon_{q_1}^{\max}/2,$$

for which around 8300 rays are traced in PS. Their distribution at S and T is shown in Figures 6.11a and 6.11b, respectively. The distribution of the rays in PS gives information about the paths they follow. We note that for the parabolic reflector many paths are found. Every path corresponds to a given number of reflections. Rays can have multiple reflections at lines 2 and 3 before arriving at the target. The parameters used in the triangulation refinement establish not only the number

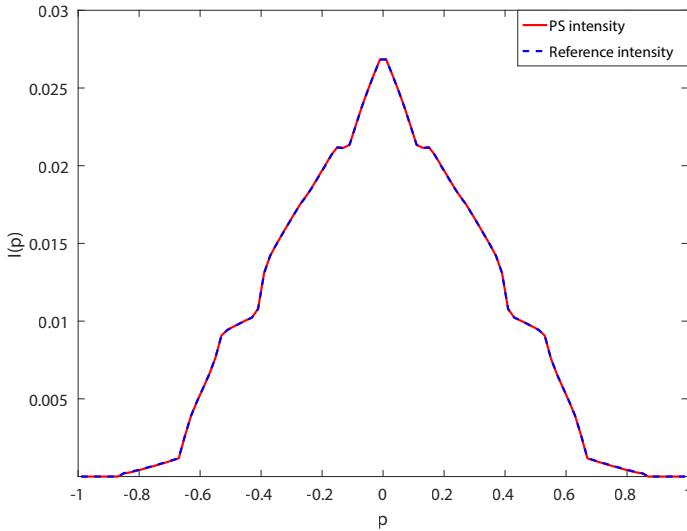


Figure 6.7: **Target intensity for the TIR-collimator.** The PS intensity \hat{I}_{PS} is computed using PS ray tracing with around $1.62 \cdot 10^5$ rays. The reference intensity \hat{I}_{ref} is obtained by QMC ray tracing with 10^7 rays.

of rays traced but also the number of paths detected. For instance, for the values of the parameters defined above, the triangulation refinement is able to detect 17 different paths. This means that up to 8 multiple reflections occur between the rays and the two mirrors. Counting the number of rays that follow a given path Π , we can calculate the fraction of rays for every path. For example, tracing around 8300 rays, the percentage of the rays that have 8 multiple reflections along one of the two reflectors is around 0.13%. Rays that reflect many times before reaching the target do not give a significant contribution to the target intensity. Decreasing the value of the parameter $\varepsilon_{q_1}^{\min}$, more paths can be found. The more reflections occur the smaller the correspond area in PS is. In order to find as many path as possible, also the parameter $\varepsilon_{q_1}^{\max}$ needs to be decreased. Increasing the number of reflections considered, the corresponding regions in PS become smaller and smaller, see Figure 6.11.

Like for the optical systems considered in the previous sections, a stopping criterion of the triangulation refinement is determined for the parabolic reflector. Etendue conservation is used in order to find the values of the parameters that give a good approximation of the boundaries of the regions with positive luminance in PS. For the parabolic reflector in Figure 6.10 all the rays that leave the source arrive at the target. Indeed, line 2 and 3 can only reflect rays (refraction law is not involved) and the target étendue is equal to the source étendue. From Equation (5.2.1) we obtain:

$$U = U_1 = 4(a - \varepsilon) \approx 8. \quad (6.4.1)$$

A range of values of $\varepsilon_{q_1}^{\min}$ and $\varepsilon_{q_1}^{\max}$ is considered (the triangulation parameters for the p -axis depend on the q -axis parameters according to Equation (6.1.2)). For each

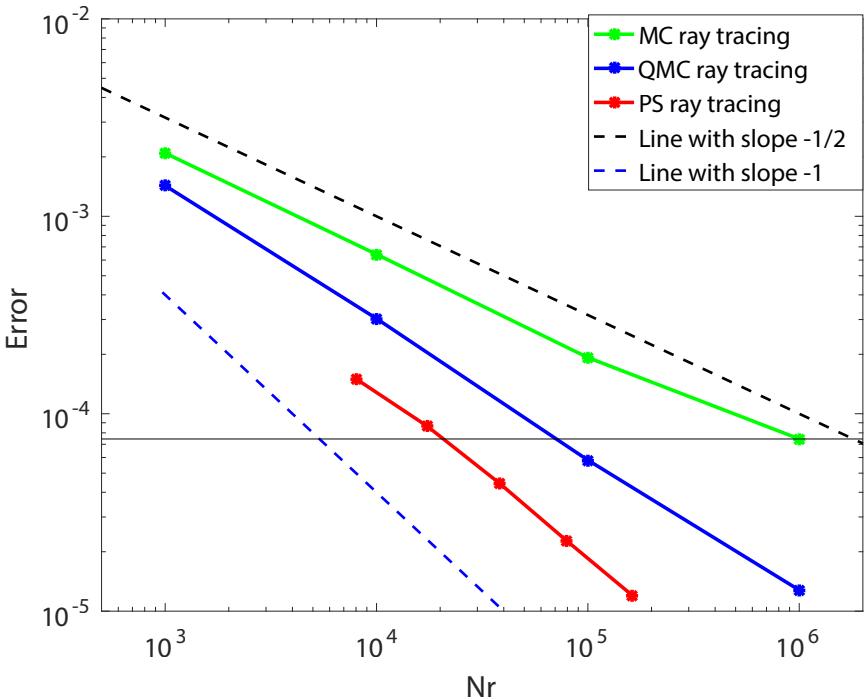


Figure 6.8: **Error as a function of the number of rays for the TIR-collimator.** The reference intensity \hat{I}_{ref} is obtained by QMC ray tracing with 10^7 rays.

couple of values, an approximation of the boundaries $\partial R(\Pi)$ is found for every path Π as explained. U_t is calculated for the approximated boundaries using Equation (5.2.4). In Table 6.3 we show how the number of rays traced, the paths found and the value for the target étendue depend on the triangulation parameters. We observe that, decreasing $\varepsilon_{q_1}^{\min}$ and $\varepsilon_{q_1}^{\max}$ the number of both the rays traced and the paths found increases. A maximum of 17 different paths are detected. Furthermore, the value of U_t gets closer and closer to the exact étendue U .

Using the triangulation refinement that gives the best étendue approximation, we calculate the target PS intensity \hat{I}_{PS} from Equation (4.3.3). The intensity profile is shown with the red line in Figure 6.12. The PS intensity is compared to a reference intensity \hat{I}_{ref} , computed using QMC ray tracing with 10^7 rays. \hat{I}_{ref} is depicted Figure 6.12 with the dotted blue line. The graph shows that the two intensities coincide.

Now, our method is compared to both MC and QMC ray tracing. The error between the approximate intensities \hat{I}_A ($A = \text{PS}, \text{MC}, \text{QMC}$) and the reference intensity is calculated. In Figure 6.13 the error as a function of the number of rays traced is shown for the three methods. The green line represents the MC error, the blue line pictures the QMC error and the red line depicts the PS error. The errors are shown in a logarithmic scale, the dotted black line has slope $-1/2$, the dotted blue line has slope -1 . Like for the other systems, MC ray tracing converges proportionally to

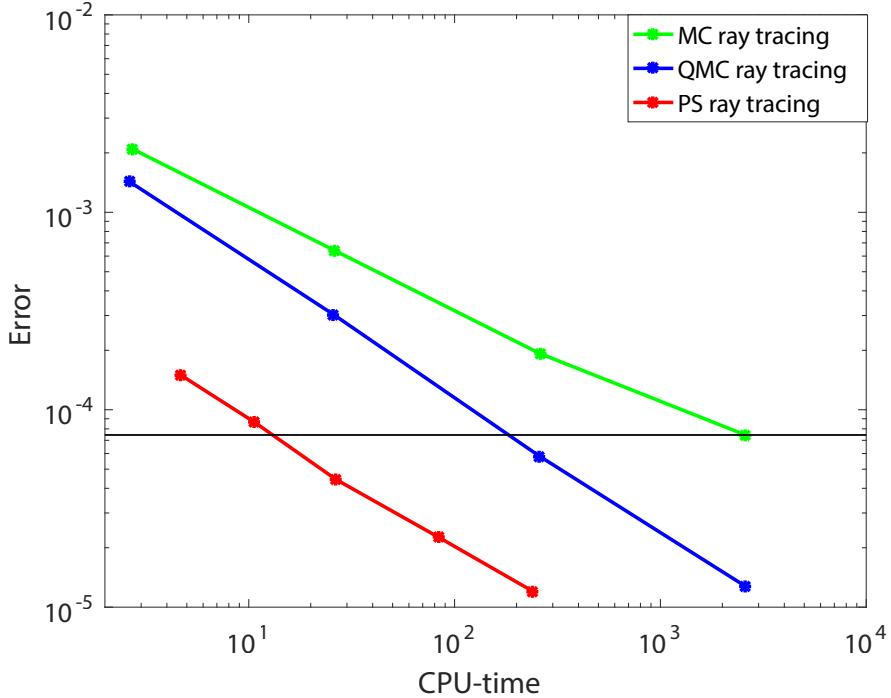


Figure 6.9: **Error as a function of the CPU-time for the TIR-collimator.** The reference intensity \hat{I}_{ref} is obtained by QMC ray tracing with 10^7 rays.

$1/\sqrt{\text{Nr}}$. QMC convergence is proportional to $1/\text{Nr}$ for $\text{Nr} \rightarrow \infty$. Less rays compared to MC ray tracing are needed in PS. In particular, to achieve an error of 10^{-4} , around 10 times less rays in PS are needed compared to MC and few rays less than QMC.

Finally, the error as a function of the CPU-time is shown in Figure 6.14. The MC, QMC and PS errors are depicted with the green, blue and red line, respectively. To obtain an error of the order of 10^{-5} , PS ray tracing is 10 times faster than MC ray tracing while becomes twice slower than QMC ray tracing. The detailed results of the numerical simulations are reported in Tables 6.5 and 6.6.

As explained above, MC and QMC errors also depend on the number of bins Nb. In the simulations shown in this chapter we have always considered $\text{Nb} = 100$. On the contrary, PS ray tracing calculates the intensity pointwise, nevertheless we considered $\text{Nb} = 100$ bins at the target and we calculate the averaged normalized PS intensity over every bin in order to have a precise comparison with the binning procedures. Because of this, we expect that increasing the number of bins, the average PS intensity becomes more accurate. To verify this conjecture, we implemented MC, QMC and PS ray tracing considering a partitioning of the interval $[-1, 1]$ into $\text{Nb} = 150$ bins. The number of rays considered for the reference intensity have to be increased $(1.5)^5$ times. The averaged normalized intensities \hat{I}_A ($A = \text{MC}, \text{QMC}, \text{PS}$) found considering $\text{Nb} = 150$ bins are compared with the reference intensity \hat{I}_{ref} (averaged and normal-

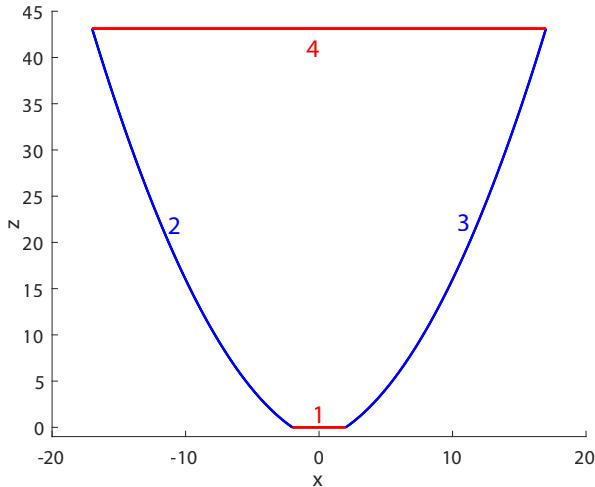


Figure 6.10: **A parabolic reflector.** Each line of the system is labeled with a number. The source $S = [2, 2]$ (line 1) is located on the x -axis. The target $T = [-17, 17]$ (line 4) is parallel to the source and is located at a height $z = 40$. The left and right reflectors (lines 2 and 3) are arcs of the same parabola.

Table 6.3: Results of the triangulation refinement.

ε_q^{\min}	ε_q^{\max}	Nr	Np	Etendue
0.2	1	643	11	5.71
0.1	1	1 573	15	7.23
0.025	0.5	8 357	17	7.65
0.025/2	0.5/2	18 613	17	7.82
0.025/4	0.5/4	40 465	17	7.82
0.025/8	0.5/8	86 529	17	7.96
0.025/16	0.5/16	185 581	17	7.98

ized) which is computed using QMC ray tracing with $7.6 \cdot 10^7$ rays. The error as a function of the number of rays is shown in Figure 6.15. The results show that, using PS ray tracing an error of the order of 10^{-5} is obtained tracing around 10^2 times less rays than MC ray tracing and twice less rays than QMC ray tracing. We conclude that, increasing the number of bins, PS error outperforms both MC and QMC ray tracing (see also Figure 6.13).

6.5 Discussion and conclusions

In this chapter we presented a method to calculate the boundaries of the regions with positive luminance in PS. This method does not depend on the parameter α needed for the α -shapes method presented in Chapter 5. Indeed, given a triangulation at the source PS, the boundaries are computed connecting the vertices of the boundary

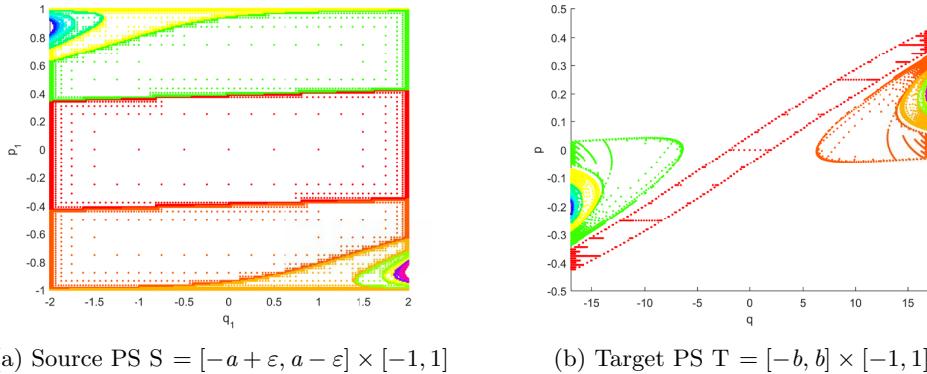


Figure 6.11: Rays distribution at S and T of the parabolic reflector. Around 8300 rays are traced using PS ray tracing with parameters $\varepsilon_{q_1}^{\min} = 0.025$, $\varepsilon_{q_1}^{\max} = 0.5$, $\varepsilon_{p_1}^{\max} = \varepsilon_{q_1}^{\min}/2$, and $\varepsilon_{p_1}^{\min} = \varepsilon_{q_1}^{\max}/2$. 17 different paths are found, each of them correspond to a certain number of reflections.

Table 6.4: Errors of the PS intensity for the parabolic reflector

Nr	PS error	CPU-time (sec.)
643	$5.18 \cdot 10^{-4}$	0.24
1 573	$8.99 \cdot 10^{-4}$	0.48
8 357	$2.87 \cdot 10^{-4}$	2.40
18 613	$1.38 \cdot 10^{-4}$	5.48
40 465	$5.80 \cdot 10^{-5}$	16.14
86 529	$2.90 \cdot 10^{-5}$	55.04
185 581	$1.66 \cdot 10^{-5}$	245.97

triangles, i.e., triangles crossed by a boundary, that follow the same path. Employing étendue conservation, a stopping criterion for the triangulation refinement was developed. We applied the method to three different optical systems: the two-faceted cup, a TIR-collimator, and a parabolic reflector. Numerical results show that PS ray tracing is faster and more accurate than MC ray tracing. Compared to QMC ray tracing we observed accuracy and speed advantages of an order of magnitude with our method for the TIR-collimator. For the two-faceted cup, PS ray tracing has a slower convergence compared to QMC ray tracing. For the parabolic reflector PS and QMC ray tracing display similar convergence. As an example, for this system we showed that increasing the number of bins the errors decrease. To conclude we state that QMC ray tracing performs better than PS ray tracing for very simple optical systems, but the PS approach is more suitable for complicated optical systems.

In order to further improve PS ray tracing we develop a new method which employs the PS of *all* the optical lines. This method is explained in the next chapter.

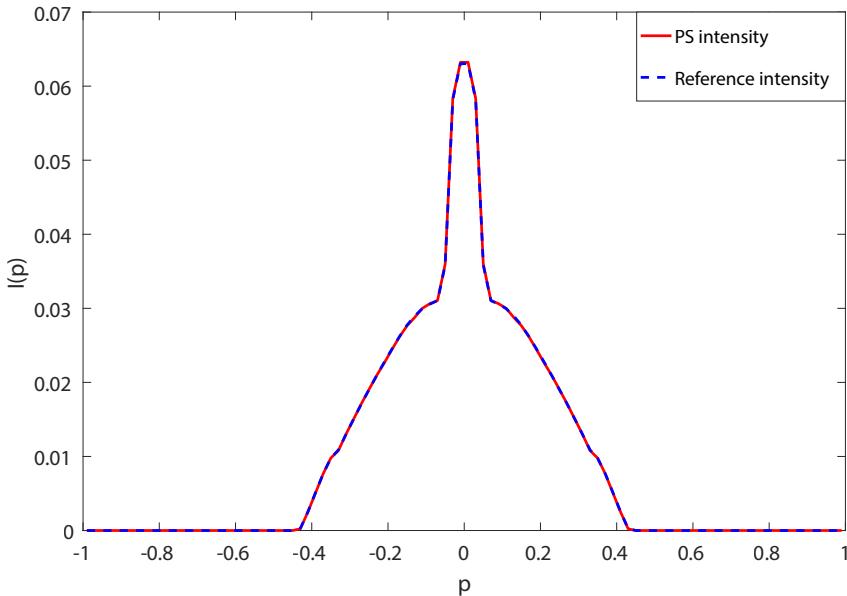


Figure 6.12: **Target intensity of the parabolic reflector.** For the PS intensity the parameters $\varepsilon_{q_1}^{\min} = 1.56 \cdot 10^{-3}$, $\varepsilon_{q_1}^{\max} = 3.13 \cdot 10^{-2}$, $\varepsilon_{p_1}^{\min} = \varepsilon_{q_1}^{\min}/2$, and $\varepsilon_{p_1}^{\max} = \varepsilon_{q_1}^{\max}/2$ are used. Around $8.15 \cdot 10^4$ rays are traced in PS. For the reference intensity QMC ray tracing with 10^7 rays is implemented.

Table 6.5: Errors of the MC intensity for the parabolic reflector

Nr	MC error	CPU-time (sec.)
10^3	$1.18 \cdot 10^{-3}$	0.39
10^4	$5.74 \cdot 10^{-4}$	3.43
10^5	$1.73 \cdot 10^{-4}$	33.13
10^6	$5.79 \cdot 10^{-5}$	328.96
10^7	$1.68 \cdot 10^{-5}$	3325.39

Table 6.6: Errors of the QMC intensity for the parabolic reflector

Nr	QMC error	CPU-time (sec.)
10^3	$1.36 \cdot 10^{-3}$	0.53
10^4	$2.05 \cdot 10^{-4}$	3.44
10^5	$2.89 \cdot 10^{-5}$	31.22
10^6	$6.96 \cdot 10^{-6}$	314.59

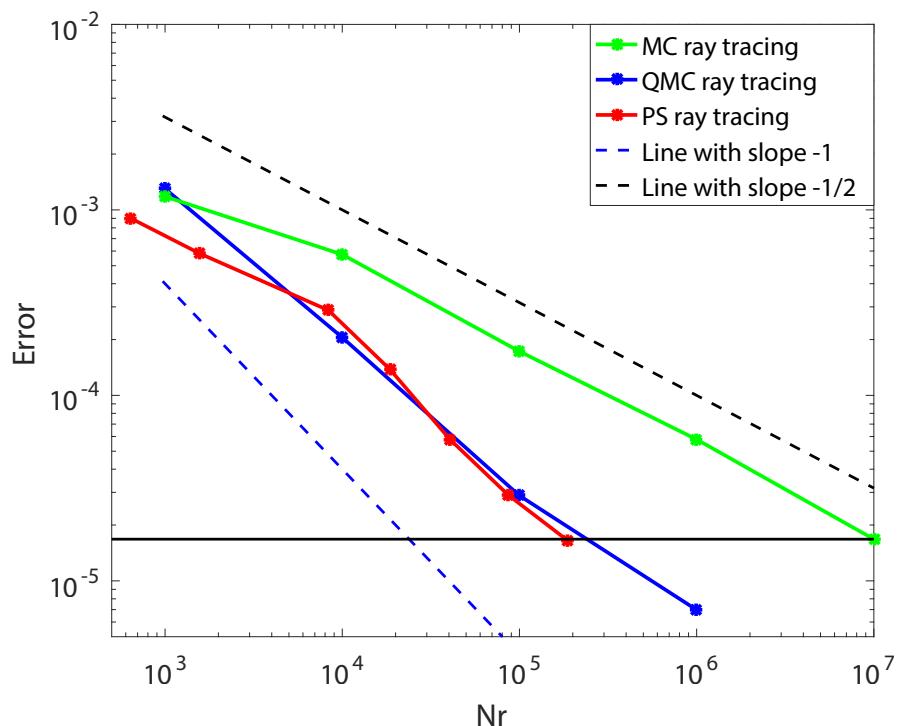


Figure 6.13: **Error as a function of the number of rays traced.** Less rays are needed using PS ray tracing compared to both MC and QMC ray tracing.

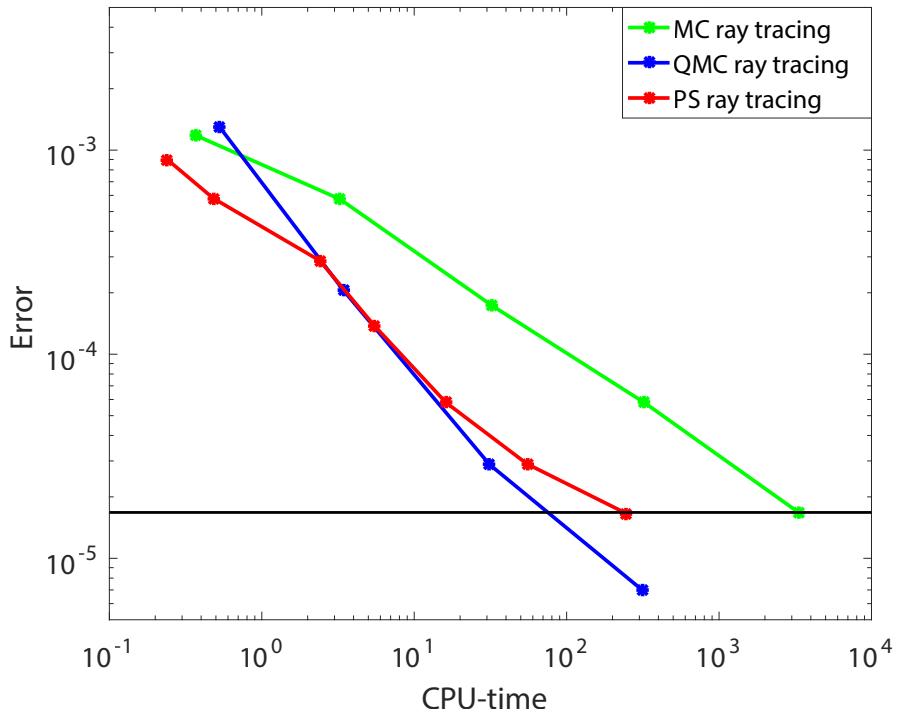


Figure 6.14: **Error as a function of the CPU-time.** PS ray tracing has significant advantages in terms of the CPU-time compared to MC ray tracing. For the parabolic reflector the computational time is comparable with QMC ray tracing.

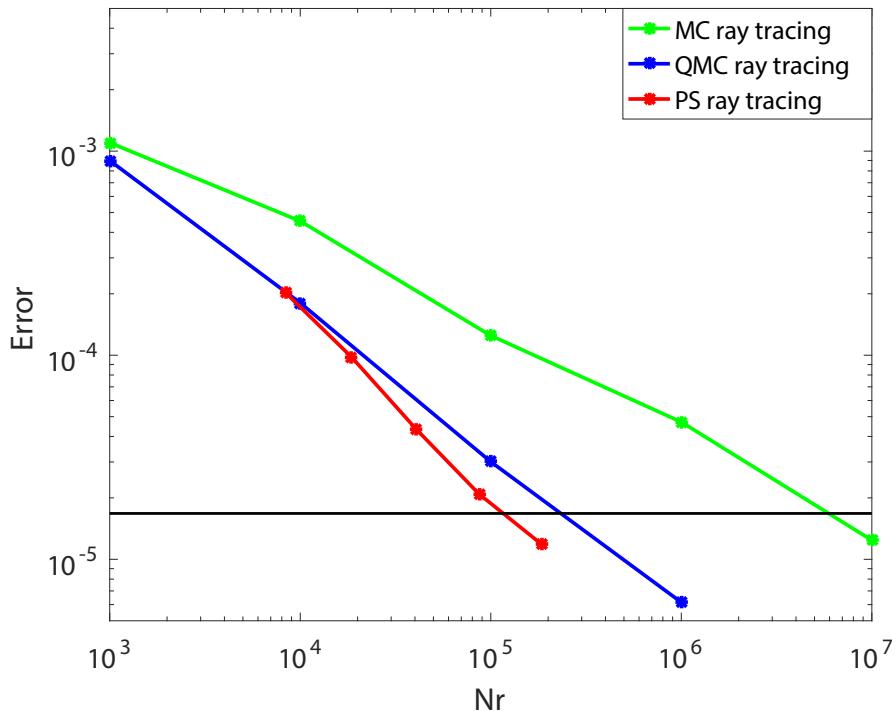


Figure 6.15: **Error plot for the parabolic reflector considering $N_b = 150$ bins.** Increasing the number of bins, the PS error decreases resulting in a better convergence compared to MC and QMC ray tracing.

Chapter 7

Concatenated backward ray mapping

In the previous chapter we have seen that PS ray tracing based on the source and the target PS constitutes an improvement of MC and QMC ray tracing. Now, a method that employs not only the source and the target PS but also the PS of *all* the other lines that constitute the optical system is introduced. In this chapter, we consider systems formed only by straight and reflective line segments. All lines can be modeled as receivers of the incident light and emitters of the reflected light, they constitute the target for incident rays and the source for reflected rays. Moreover, we assume that the source can only emit light and the target can only receive light. Therefore, two different PS are considered for the reflectors and one PS for S and T. All these phase spaces are connected through a map which relates the ray coordinates on every PS. This map can be written as the concatenation of many maps which can be classified as two different kind of maps, i.e., the map that connects the source and the target PS of two *different* lines and the map that connects the target and the source PS of the *same* line. Employing the inverses of these maps we are able to detect the parts of target PS illuminated by the source. All the PS considered are divided into several regions, the boundaries of which can be determined exactly for systems formed by straight lines. We make the assumption of a Lambertian source. As a consequence, the output intensity along a given direction is obtained from the total width of all the patches with positive luminance, measured along that direction.

In this chapter, two different optical systems are investigated: the two-faceted cup and the so-called multi-faceted cup. Next, the details of the procedure are explained for the two-faceted-cup.

7.1 Phase spaces of the two-faceted cup

A two-faceted cup is introduced in Chapter 3 and depicted in Figure 3.1. It is formed by a source S, a target T and two reflectors which are straight lines segments. Using the same notation of Chapter 4, we denote with $S = Q \times P$ the PS and with (q, p) the rays coordinates in S.

Let's now introduce some new notation. The source and the target PS of a line j are indicated with S_j and T_j , respectively. The initial rays coordinates in S are denoted with $(q_{s,1}, p_{s,1})$. The coordinates of every ray that reaches the line $j \in \{2, 3, 4\}$ are indicated with $(q_{t,j}, p_{t,j})$ on T_j . In the following, to simplify the notation, we indicate the target coordinates of the rays on T_4 with (q, p) instead of $(q_{t,4}, p_{t,4})$. After reflection, the ray leaves line $j \in \{2, 3\}$ at the same position but with a different direction, the new ray coordinates are indicated with $(q_{s,j}, p_{s,j})$ on S_j . Note that $q_{s,j} = q_{t,j}$, while $p_{s,j}$ is obtained applying the reflection law to the direction coordinate $p_{t,j}$ of the incident ray.

The phase spaces S_j and T_j of each line j are partitioned into different regions, $(S_{j,k})_{k=2,3,4}$ and $(T_{j,l})_{l=1,2,3}$, respectively, where $k \neq j$ is the index of the line that is illuminated by j and $l \neq j$ is the index of the line that illuminates j . Hence, $S_{j,k} \subset S_j$ is the part of S_j corresponding to rays that illuminate line k , and $T_{j,l} \subset T_j$ is the part of T_j corresponding to rays originating from the line l . Note that, due to the fact that the source only emits light, we do not define its target PS T_1 . Similarly, since the target only receives light, its source PS S_4 is not defined. For the two-faceted cup, six different phase spaces need to be considered which are given by the following expressions:

$$\begin{aligned} S_1 &= S_{1,2} \cup S_{1,3} \cup S_{1,4}, \\ S_2 &= S_{2,3} \cup S_{2,4}, \\ S_3 &= S_{3,2} \cup S_{3,4}, \\ T_2 &= T_{2,1} \cup T_{2,3}, \\ T_3 &= T_{3,1} \cup T_{3,2}, \\ T_4 &= T_{4,1} \cup T_{4,2} \cup T_{4,3}. \end{aligned} \tag{7.1.1}$$

Note that, as the source cannot receive light and the target cannot emit light, the regions $(S_{j,1})_{j=2,3}$ and $(T_{j,4})_{j=2,3}$ are not considered.

The boundaries $\partial S_{j,k}$ are mapped into the boundaries $\partial T_{k,j}$ for every $j = \{1, 2, 3\}$ and $k = \{2, 3, 4\}$ with $k \neq j$ (edge-ray principle). For the two-faceted cup, and for all systems formed by straight line segments, these boundaries can be determined analytically. Given two lines j and k with $j \neq k$, the boundaries S_j and T_k are determined as follows. Let $(x_{j,\ell}, z_{j,\ell})$ and $(x_{j,r}, z_{j,r})$ be the coordinates of the points located at the left and the right extreme of line j , respectively. Similarly, $(x_{k,\ell}, z_{k,\ell})$ and $(x_{k,r}, z_{k,r})$ are the coordinates of the points located at the left and the right extreme of line k , respectively. The boundaries $\partial S_{j,l}$ and $\partial T_{l,j}$ are formed by four different curves, two of them are given by all the rays that leave the end points of line j and hit line l and, the others two are given by the rays that leave the interior of line j and hit the end points of line l . They are given by:

$$\begin{aligned} \partial S_{j,l} &= \partial S_{j,l}^1 \cup \partial S_{j,l}^2 \cup \partial S_{j,l}^3 \cup \partial S_{j,l}^4, \\ \partial T_{l,j} &= \partial T_{l,j}^1 \cup \partial T_{l,j}^2 \cup \partial T_{l,j}^3 \cup \partial T_{l,j}^4. \end{aligned} \tag{7.1.2}$$

In the following we explain in more details the case for $j = 1$ and $l = 4$ (see Figure 7.1).

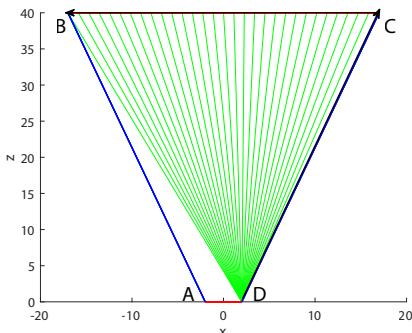
The boundaries $\partial S_{1,4}$ and $\partial T_{4,1}$ are given in Figures 7.2 and 7.3, respectively. $\partial S_{1,4}^1$ and $\partial T_{4,1}^1$ are obtained tracing out line 4 from $q_\ell = -b$ to $q_r = b$ by rays leaving



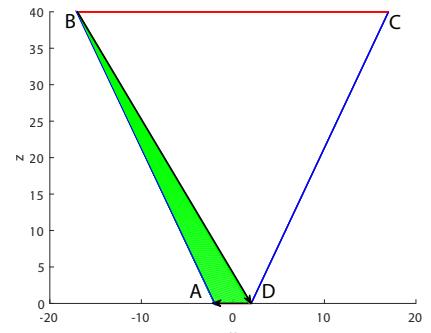
(a) Rays that leave the left end point of the source (line 1) and trace out the target (line 4).



(b) Rays that trace out the source (line 1) and hit the right end point of the target (line 4).



(c) Rays that leave the right end point of the source (line 1) and trace out the target (line 4).



(d) Rays that trace out the source (line 1) and hit the left end point of the target (line 4).

Figure 7.1: Rays located on the boundaries of the regions $\partial S_{1,4}$ and $\partial T_{4,1}$.
 $A = (x_{1,\ell}, z_{1,\ell}) = (-2, 0)$ and $D = (x_{1,r}, z_{1,r}) = (2, 0)$ are the left and right corner points (or end points) of S (line 1). $B = (x_{4,\ell}, z_{4,\ell}) = (-17, 40)$ and $C = (x_{4,r}, z_{4,r}) = (17, 40)$, are the left and right corner points of T (line 4).

$q_{1,\ell} = -a$ with varying p_1 , these rays are shown in Figure 7.1a, and the boundary segments $\partial S_{1,4}^1$ and $\partial T_{4,1}^1$ are the orange line segments labeled with c. $\partial S_{1,4}^2$ and $\partial T_{4,1}^2$ are given tracing out line 1 from $q_{1,\ell} = -a$ to $q_{1,r} = a$ with varying p_1 , such that all rays hit $q_r = b$, these rays are shown in Figure 7.1b, the boundary segments $\partial S_{1,4}^2$ and $\partial T_{4,1}^2$ are depicted in blue (lines segments labeled with d). Likewise, $\partial S_{1,4}^3$ and $\partial T_{4,1}^3$ are obtained tracing out line 4 from $q_r = b$ to $q_\ell = -b$ by rays leaving $q_{1,r} = x_{1,r} = a$ with varying p_1 . These rays are shown in Figure 7.1c, $\partial S_{1,4}^3$ and $\partial T_{4,1}^3$ are the red line segments labeled with e. Finally, $\partial S_{1,4}^4$ and $\partial T_{4,1}^4$ are given tracing out line 1 from $q_{1,r} = a$ to $q_{1,\ell} = -a$ with varying p_1 , such that all rays hit $q_\ell = -b$, these rays are shown in Figure 7.1d, $\partial S_{1,4}^4$ and $\partial T_{4,1}^4$ are the green lines segments labeled with f. We remind the reader that we use the notation (x, z) for the Cartesian coordinates of the optical system, while PS has (q, p) coordinates. It is worth noting that $q_{1,\ell} = x_{1,\ell}$, $q_{1,r} = x_{1,r}$, $q_\ell = x_{4,\ell}$ and $q_r = x_{4,r}$.

For the two-faceted cup there is an analytic expression for every line segment $\partial S_{j,k}^m$ and $\partial T_{k,j}^m$ in Equation (7.1.2) with $m \in \{1, \dots, 4\}$. For instance, the rays on the boundaries $\partial S_{j,k}^1$ and $\partial T_{k,j}^1$ are parameterized in the (x, z) -plane by

$$\mathbf{r}_{j,k}(t) = \begin{pmatrix} x_{k,\ell} - x_{j,\ell} + t(x_{k,r} - x_{k,\ell}) \\ z_{k,\ell} - z_{j,\ell} + t(z_{k,r} - z_{k,\ell}) \end{pmatrix} \quad 0 \leq t \leq 1. \quad (7.1.3)$$

These rays are located on a vertical line segment in S_j as only the p_j -coordinate changes and on a curved line in T_k as both the target position and direction vary. The analytic expressions for $\partial S_{j,k}^1$ and $\partial T_{k,j}^1$ are

$$\partial S_{j,k}^1(t) = \left\{ (q_j, p_j) = \left(q_{j,\ell} - q_{j,\ell} + t(q_{k,r} - q_{k,\ell}), |\boldsymbol{\nu}_j \times \hat{\mathbf{r}}_{j,k}(t)| \right) \right\}, \quad (7.1.4)$$

$$\partial T_{k,j}^1(t) = \left\{ (q_k, p_k) = \left(q_{k,\ell} - q_{j,\ell} + t(q_{k,r} - q_{k,\ell}), |\boldsymbol{\nu}_k \times \hat{\mathbf{r}}_{j,k}(t)| \right) \right\}, \quad (7.1.5)$$

where we have indicated with $\hat{\mathbf{r}}_{j,k}(t)$ the normalization of the ray in (7.1.3) and $\boldsymbol{\nu}_j$ and $\boldsymbol{\nu}_k$ are the normalized inward normals to lines j and k , respectively. Note that $\sin \tau_j = |\boldsymbol{\nu}_j \times \hat{\mathbf{r}}_{j,k}(t)|$ and $\sin \tau_k = |\boldsymbol{\nu}_k \times \hat{\mathbf{r}}_{j,k}(t)|$. Likewise, the boundaries $\partial S_{j,l}^m$ and $\partial T_{l,j}^m$ are calculated for every $m \in \{2, 3, 4\}$. Finally, $\partial S_{j,l}$ and $\partial T_{l,j}$ are determined using (7.1.2).

In Figures 7.4 – 7.9, $(\partial S_{j,k})_{j \neq k=2,3,4}$ and $(\partial T_{j,l})_{j \neq l=1,2,3}$ are depicted in blue and red, respectively. The source and target PS of lines 2 and 3 have some empty regions. These parts correspond to the regions formed by the rays that either go back to the source or are emitted from the target. These regions are not taken into account, see Equation (7.1.1). We observe that, because of the symmetry of the optical system, S_3 is the mirror image of S_2 after reflection in the central point $(q, p) = (-9.5, 0)$ followed by a translation $(q, p) \rightarrow (q + 19, p)$. Likewise T_3 is the mirror image of T_2 after the same reflection and translation.

7.1.1 Computation of the target photometric variables

In this section we explain how to compute the target photometric variables in PS. The intensity I along a given direction $p \in [-1, 1]$ in target phase space T_4 depends



Figure 7.2: **Source PS of line 1.**
Boundary of the region $S_{1,4}$.



Figure 7.3: **Target PS of line 4.**
Boundary of the region $T_{4,1}$.

on the luminance $L(q, p)$ defined as in Equation (4.3.3). For the two-faceted cup, it becomes:

$$I_{PS}(p) = \int_{-b}^b L(q, p) dq. \quad (7.1.6)$$

The parts of T_4 that are illuminated by the source S correspond to parts with positive luminance, for the other parts the luminance might be 0. Assuming positive luminance on S , the following relations hold:

$$\begin{aligned} L(q, p) &> 0 & \forall (q, p) \in T_{4,1}, \\ L(q, p) &\geq 0 & \forall (q, p) \in (T_{4,j})_{j=2,3}. \end{aligned} \quad (7.1.7)$$

Once a ray leaves the source S it can hit the reflectors several times before hitting the target T . To relate S and T , a map $M_{1,4}: S_1 \rightarrow T_4$ is introduced such that $M_{1,4}(q_{s,1}, p_{s,1}) = (q, p)$. As not all parts of T_4 are illuminated by the source S , the map $M_{1,4}$ is not surjective. Therefore, we need to determine the subsets of T_4 illuminated by S corresponding to the regions where the luminance is positive. To this purpose, we consider two different kinds of maps. The first map relates the coordinates of the source and the target PS of two *different* lines, we call it the *propagation map*. The second map relates the coordinates of the target and the source PS of the *same* line, we call it the *reflection map*. In particular, given two lines j and k with $j \neq k$, the propagation map $P_{j,k}: S_{j,k} \rightarrow T_{k,j}$ relates $S_{j,k}$ with $T_{k,j}$ and, it is defined as follows:

$$P_{j,k}(q_{s,j}, p_{s,j}) = (q_{t,k}, p_{t,k}), \quad (7.1.8)$$

where $q_{t,k}$ is given by the x -coordinate of the intersection point between the ray and line k , and $p_{t,k}$ is computed considering the direction of the incident ray with respect to the normal of line k . For one single line k , the reflection map $R_{k,l,h}: T_{k,l} \rightarrow S_{k,h}$ relates the regions $T_{k,l} \subset T_k$ and $S_{k,h} \subset S_k$. To simplify the notation, from now on we omit the dependence of $R_{k,l,h}$ from l and h , i.e., $R_{k,l,h} = R_k$. The reflection map is defined as:

$$R_k(q_{t,k}, p_{t,k}) = (q_{s,k}, p_{s,k}), \quad (7.1.9)$$

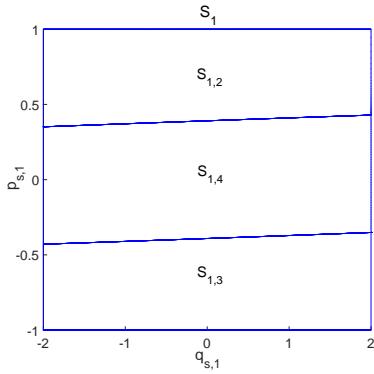


Figure 7.4: Source PS of line 1.
It is partitioned into regions $(S_{1,k})_{k=2,3,4}$ formed by rays that leave line 1 and hit line k .

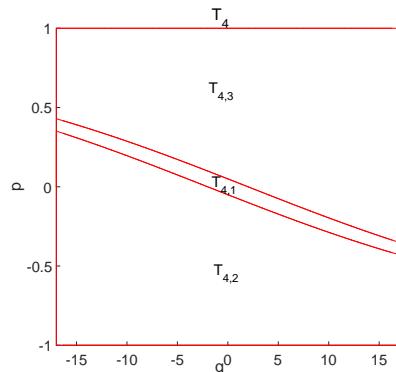


Figure 7.5: Target PS of line 4. It is partitioned into regions $(T_{4,l})_{l=1,2,3}$ formed by rays that leave line l and hit line 4.

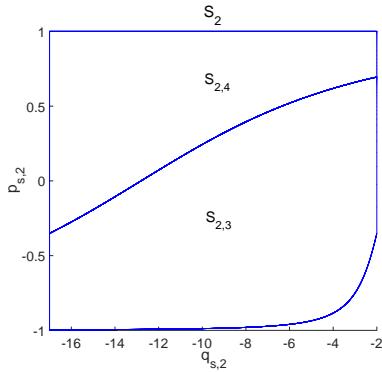


Figure 7.6: Source PS of line 2. It is partitioned into regions $(S_{2,k})_{k=3,4}$ formed by rays that leave line 2 and hit line k .

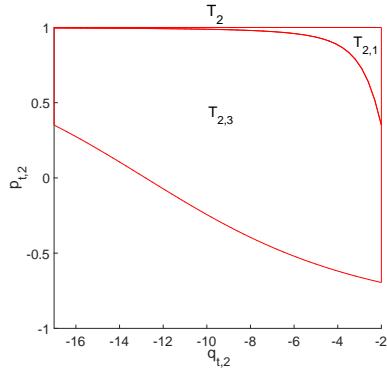


Figure 7.7: Target PS of line 2. It is partitioned into regions $(T_{2,l})_{l=1,3}$ formed by rays that leave line l and hit line 2.

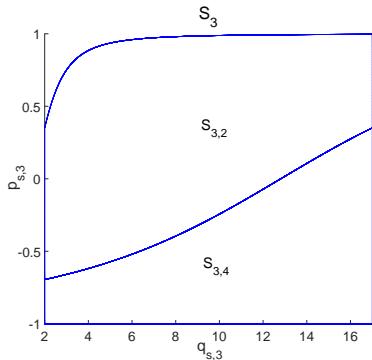


Figure 7.8: Source PS of line 3. It is partitioned into regions $(S_{3,k})_{k=2,4}$ formed by rays that leave line 3 and hit line k .

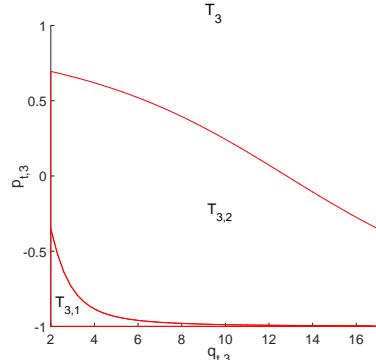


Figure 7.9: Target PS of line 3. It is partitioned into regions $(T_{3,l})_{l=1,2}$ formed by rays that leave line l and hit line 3.

where $p_{t,k}$ changes according to the reflection law and $q_{t,k} = q_{s,k}$ as R_k maps the target PS into the source PS of the same line k , that is T_k into S_k . Using a procedure similar to the ray transport matrices approach (see [39], Chapter 6), the map $M_{1,4}$ is described by the composition of $P_{j,k}$ and R_k defined in (7.1.8) and (7.1.9). This composition depends on the path Π followed by the rays. We indicate with $M_{1,4}(\Pi)$ the map $M_{1,4}$ restricted to the path Π and with $R(\Pi) \subset T_4$ the regions on T_4 formed by the rays that follow path Π . Considering all the possible paths Π from S to T , all the positive luminance regions $R(\Pi)$ on T_4 can be determined.

To clarify this concept, we provide the following example. Consider a ray that is emitted from the source (line 1), hits the left reflector (line 2) and finally reaches the target (line 4). The path Π followed by this ray is defined as $\Pi = (1, 2, 4)$ and the corresponding map $M_{1,4}(\Pi) : S_1 \rightarrow R(\Pi)$ that describes the propagation of all rays that follow path Π is defined by:

$$M_{1,4}(\Pi) : S_{1,2} \rightarrow T_{2,1} \rightarrow S_{2,4} \rightarrow T_{4,2}, \quad (7.1.10)$$

which can be written as:

$$M_{1,4}(\Pi) = P_{2,4} \circ R_2 \circ P_{1,2}. \quad (7.1.11)$$

In general, to construct the map $M_{1,4}(\Pi)$ we need to know its corresponding path Π . To determine all possible paths Π , instead of tracing the rays from S to T , we start considering the rays in T_4 . In particular, along a given direction $p \in [-1, 1]$ we consider the intersection points between the line $p = \text{const}$ and $(\partial T_{4,j})_{j=1,2,3}$. These points are traced back to line j from which they are emitted and their corresponding coordinates on S_j and T_j are computed. This is done applying sequentially the maps $P_{j,4}^{-1} : T_{4,j} \rightarrow S_{j,4}$ and $R_j^{-1} : S_j \rightarrow T_j$. Then the same procedure is repeated considering these new coordinates on T_j . The computation stops either when the points found are emitted from the source, that is when they are located on S_1 , or when they reach again the target, that is when they are located on T_4 . If a ray reaches S_1 , then a path Π from S to T is found. If a ray reaches again the target T_4 , then we conclude that it is not emitted by S and therefore, it is located inside the parts of T_4 with luminance equal to 0.

Finally, the inverse $M_{1,4}^{-1}(\Pi)$ of the map $M_{1,4}(\Pi)$ is constructed for every possible path Π . The map $M_{1,4}^{-1}(\Pi)$ is the composition of the inverses of the propagation and the reflection maps in reverse order according to the path Π . For instance, for path $\Pi = (1, 2, 4)$, $M_{1,4}^{-1}(\Pi)$ is given by:

$$M_{1,4}^{-1}(\Pi) = P_{1,2}^{-1} \circ R_2^{-1} \circ P_{2,4}^{-1}. \quad (7.1.12)$$

The steps of the procedure are shown in Figure 7.10 where the map in (7.1.12) is written in red.

Using the procedure explained above, given a ray with coordinates $(q, p) \in T_4$ we can establish whether it is located inside one of the positive luminance regions $R(\Pi)$ with or not. In case the ray is inside a region $R(\Pi)$, its corresponding coordinates $(q_{s,1}, p_{s,1}) \in S_1$ are obtained using $M_{1,4}^{-1}(\Pi)$, where Π is the path followed by this ray. The luminance in Equation (7.1.7) is, therefore, defined as in Equation (4.3.2), for some path Π connecting S and T . The target intensity is calculated from Equation

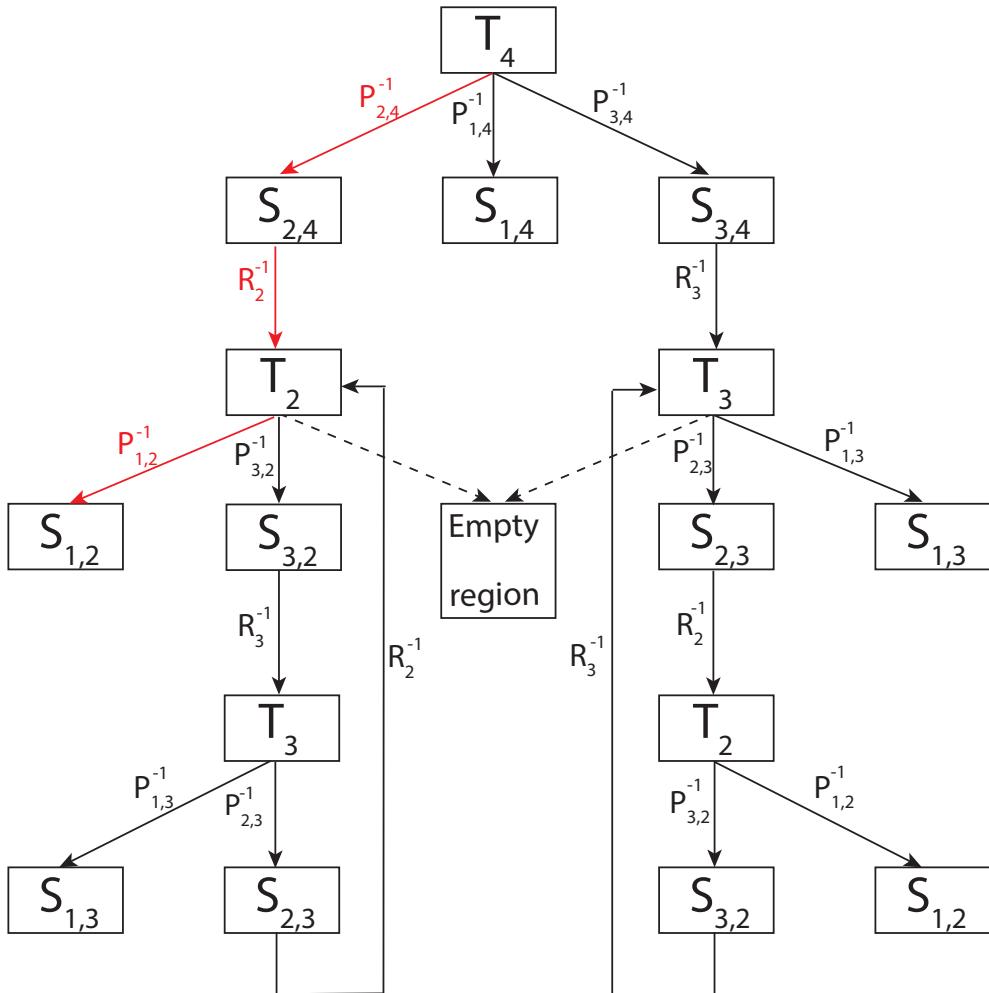


Figure 7.10: **Ray mapping tree.** It describes how to detect all the possible paths from S to T .

(7.1.6). Indicating with $q^{\min}(\Pi, p)$ and $q^{\max}(\Pi, p)$ the minimum and maximum position coordinates of the intersection points between the boundaries $\partial R(\Pi)$ and the line $p = \text{const}$, Equation (7.1.6) reduces to Equation (4.3.4), if only two intersection points are found, and to Equation (5.2.6) in case more than two intersection points occur. For the two-faceted cup there are only two intersection points between a line $p = \text{const}$ and $\partial R(\Pi)$, hence, in this chapter we use Equation (4.3.4). We remark that, for a given ray with corresponding coordinates (q, p) on T_4 , only one path is possible as we are assuming that all lines are reflective. Because of this, the regions $R(\Pi)$ do not overlap. Next, the details of the procedure to compute the coordinates $q^{\min}(\Pi, p)$ and $q^{\max}(\Pi, p)$ are explained.

7.1.2 The structure of the backward ray mapping algorithm

The goal is to determine the target intensity along a given direction $p = \text{const}$. Also in the ray mapping method, we assume a Lambertian source, therefore the intensity is equal to the sum of the lengths of the line segments given by the intersection of the line $p = \text{const}$ and the support of L (see Equation (4.3.4)). To determine these line segments, a recursive procedure is developed. The procedure starts on T_4 with a given direction $p = \text{const}$ and with the parallel rays corresponding to the end points $(q_\ell, p) = (-b, p)$ and $(q_r, p) = (b, p)$. We set the initial intensity $I(p) = 0$ along direction $p = \text{const}$. Considering the intersection between the line $p = \text{const}$ and the boundaries $(\partial T_{4,j})_{j=1,2,3}$ three intervals are found. Each interval corresponds to rays emitted by line j ($j \in \{1, 2, 3\}$). The rays corresponding to the end points of these intervals are traced back from T_4 to T_j where j is the line from which they are emitted. Then, another interval of parallel rays along the corresponding direction in T_j has to be considered and the intersection points between the line $p = p_{t,j}$ and $\partial T_{j,k}$ (with $j \neq 4$ and $j \neq k$) are calculated, where $p_{t,j}$ is the new direction of the rays traced back. The procedure continues recursively until the source is found.

Before explaining the details, let us introduce some new notation. The role of the variables we introduce will become clear later on. The coordinates in T_k of the rays traced back from a line $j \neq k$ to line k are indicated with $(q_{t,k}^1, p_{t,k})$ and $(q_{t,k}^2, p_{t,k})$. The minimum and the maximum position coordinates are $q_{t,k}^{\min} = \min\{q_{t,k}^1, q_{t,k}^2\}$ and $q_{t,k}^{\max} = \max\{q_{t,k}^1, q_{t,k}^2\}$, respectively. The coordinates of the intersection points of $p = p_{t,k}$ with boundaries $\partial T_{k,j}$ need to be determined for every $j \in \{1, 2, 3\}$ and $k \in \{2, 3, 4\}$ with $k \neq j$. They are indicated with $(u_{k,j}^{\min}, p_{t,k})$ and $(u_{k,j}^{\max}, p_{t,k})$ where $u_{k,j}^{\min} < u_{k,j}^{\max}$. Since not all the rays whose corresponding coordinates are located inside the segment $[q_{t,k}^{\min}, q_{t,k}^{\max}]$ with direction $p = p_{t,k}$ follow the same path, the intersection segment $[v_{k,j}^{\min}, v_{k,j}^{\max}] = [q_{t,k}^{\min}, q_{t,k}^{\max}] \cap [u_{k,j}^{\min}, u_{k,j}^{\max}]$ needs to be calculated. $(v_{k,j}^{\min}, p_{t,k})$ and $(v_{k,j}^{\max}, p_{t,k})$ are the coordinates of the rays that need to be traced back from line k to line j .

The method can be outlined as follows.

1. Calculate the intersection points $(u_{4,j}^{\min}, p)$ and $(u_{4,j}^{\max}, p)$ between line $p = \text{const}$ and $\partial T_{4,j}$ for every $j \in \{1, 2, 3\}$, where $u_{4,j}^{\min} < u_{4,j}^{\max}$. This can be done analytically because the exact expression of the boundaries $\partial T_{4,j}$ is found as explained in Section 7.1.

2. Calculate the intersection segment

$$[v_{4,j}^{\min}, v_{4,j}^{\max}] = [u_{4,j}^{\min}, u_{4,j}^{\max}] \cap [q^{\min}, q^{\max}].$$

3. If $j = 1$, the coordinates $(v_{4,1}^{\min}, p)$ and $(v_{4,1}^{\max}, p)$ equal the coordinates $(q^{\min}(\Pi, p), p)$ and $(q^{\max}(\Pi, p), p)$ of the rays located on the boundary $\partial R(\Pi)$ with $\Pi = (1, 4)$. All the parallel rays with direction coordinate p and q -position coordinate $u_{4,1}^{\min} \leq q \leq u_{4,1}^{\max}$ are emitted by the source and they directly hit the target. Update the intensity using (4.3.4):

$$I(p) = I(p) + q^{\max}(\Pi, p) - q^{\min}(\Pi, p).$$

4. If $j \neq 1$, continue with the following steps

5. Trace back $(v_{4,j}^{\min}, p)$ and $(v_{4,j}^{\max}, p)$ from line 4 to line j to find their corresponding coordinates on T_j :

$$\begin{aligned} (q_{t,j}^1, p_{t,j}) &= R_j^{-1} \circ P_{j,4}^{-1}(v_{4,j}^{\min}, p), \\ (q_{t,j}^2, p_{t,j}) &= R_j^{-1} \circ P_{j,4}^{-1}(v_{4,j}^{\max}, p). \end{aligned}$$

6. Update the path $\Pi = (j, 4)$

7. Determine $q_{t,j}^{\min} = \min\{q_{t,j}^1, q_{t,j}^2\}$ and $q_{t,j}^{\max} = \max\{q_{t,j}^1, q_{t,j}^2\}$

8. Calculate the intersection points $(u_{j,k}^{\min}, p)$ and $(u_{j,k}^{\max}, p)$ between the line $p = p_{t,j}$ and $\partial T_{j,k}$ for every $k \in \{1, 2, 3\}$ with $k \neq j$.

9. Since not all rays whose corresponding coordinates are located inside the segment $[q_{t,j}^{\min}, q_{t,j}^{\max}]$ follow the same path, compute the intersection segment

$$[v_{j,k}^{\min}, v_{j,k}^{\max}] = [u_{j,k}^{\min}, u_{j,k}^{\max}] \cap [q_{t,j}^{\min}, q_{t,j}^{\max}]$$

10. If $k \neq 1$

- a) Trace back $(v_{j,k}^{\min}, p_{t,j})$ and $(v_{j,k}^{\max}, p_{t,j})$ from j to k

$$\begin{aligned} (q_{t,k}^1, p_{t,k}) &= R_k^{-1} \circ P_{k,j}^{-1}(v_{j,k}^{\min}, p_{t,j}), \\ (q_{t,k}^2, p_{t,k}) &= R_k^{-1} \circ P_{k,j}^{-1}(v_{j,k}^{\max}, p_{t,j}). \end{aligned}$$

- b) Update the path: $\Pi = (k, \Pi)$

- c) Set $j = k$ and repeat the procedure from point 7.

Else if $k = 1$, the rays reached the source and a possible path $\Pi = (1, \dots, 4)$ is found.

- a) Trace back to source

$$\begin{aligned} (q_{s,1}^1, p_{s,1}) &= P_{1,j}^{-1}(v_{j,1}^{\min}, p_{t,j}), \\ (q_{s,1}^2, p_{s,1}) &= P_{1,j}^{-1}(v_{j,1}^{\max}, p_{t,j}). \end{aligned}$$

b) Apply the forward map

$$\begin{aligned} (q^1(\Pi, p), p) &= M_{1,4}(\Pi)(q_{s,1}^1, p_{s,1}), \\ (q^2(\Pi, p), p) &= M_{1,4}(\Pi)(q_{s,1}^2, p_{s,1}). \end{aligned}$$

c) Update the intensity:

$$I(p) = I(p) + q^{\max}(\Pi, p) - q^{\min}(\Pi, p)$$

$$\text{where } q^{\min}(\Pi, p) := q^{\min} = \min\{q^1(\Pi, p), q^2(\Pi, p)\} \\ \text{and } q^{\max}(\Pi, p) := q^{\max} = \max\{q^1(\Pi, p), q^2(\Pi, p)\}.$$

To clarify the method, we give an example that describes how the target intensity along direction $p = -0.2$ is calculated. From Figure 7.11 to Figure 7.18 the steps used in this example are shown. A detailed description of those figures is given in the following.

The procedure starts with the rays with direction $p = 0.2$ on T_4 , where $q_l = -b$ and $q_r = b$ are the left and the right end points of the target T , respectively. The intersection points $(u_{4,j}^{\min}, p)$ and $(u_{4,j}^{\max}, p)$ of the line $p = -0.2$ with boundaries $\partial T_{4,j}$ are computed for every $j \neq 4$.

We start from $j = 1$. Therefore the coordinates $(u_{4,1}^{\min}, p)$ and $(u_{4,1}^{\max}, p)$ of the intersection points between line $p = -0.2$ and the boundary $\partial T_{4,1}$ are computed and these points are depicted in Figure 7.11. The source is now reached because $j = 1$ and, one possible path is found. The points $(u_{4,1}^{\min}, p)$ and $(u_{4,1}^{\max}, p)$ are located on the boundaries of the region formed by the rays that leave the source and directly hit the target, that is the rays located on $\partial R(\Pi_1)$ with $\Pi_1 = (1, 4)$. Therefore, the contribution to the intensity formed by the rays that follow the path $\Pi_1 = (1, 4)$ is given by $u_{4,1}^{\max} - u_{4,1}^{\min}$.

We continue with $j = 2$. The boundary $\partial T_{4,2}$ is considered in order to find other paths. The intersection points $(u_{4,2}^{\min}, p)$ and $(u_{4,2}^{\max}, p)$ of line $p = -0.2$ with the boundary $\partial T_{4,2}$ are calculated. They are depicted in Figure 7.12 with the magenta dots. Also the intersection segment

$$[v_{4,2}^{\min}, v_{4,2}^{\max}] = [u_{4,2}^{\min}, u_{4,2}^{\max}] \cap [q^{\min}, q^{\max}] \quad (7.1.13)$$

is calculated. In T_4 $v_{4,2}^{\min} = u_{4,2}^{\min}$ and $v_{4,2}^{\max} = u_{4,2}^{\max}$ because $q^{\min} = -b$ and $q^{\max} = b$ always coincide with the end points of T_4 . Their corresponding position coordinates $q_{s,2}^1$ and $q_{s,2}^2$ on S_2 are obtained from:

$$\begin{aligned} P_{2,4}^{-1}(v_{4,2}^{\min}, p) &= (q_{s,2}^1, p_{s,2}^1), \\ P_{2,4}^{-1}(v_{4,2}^{\max}, p) &= (q_{s,2}^2, p_{s,2}^2). \end{aligned} \quad (7.1.14)$$

The directions $p_{s,2}^1$ and $p_{s,2}^2$ on S_2 are given considering the direction $p_{t,2} = p$ with respect to the normal ν_2 of line 2. Note that $p_{s,2}^1 = p_{s,2}^2$ because all the lines are straight lines, their normals do not depend on the position where it is computed. Thus, in the following we will omit the subscripts for the direction coordinates. Then, the corresponding direction $p_{t,2}^1 = p_{t,2}^2$ on T_2 is calculated from:

$$\begin{aligned} R_2^{-1}(q_{s,2}^1, p_{s,2}) &= (q_{t,2}^1, p_{t,2}), \\ R_2^{-1}(q_{s,2}^2, p_{s,2}) &= (q_{t,2}^2, p_{t,2}). \end{aligned} \quad (7.1.15)$$

Note that $q_{s,2}^1 = q_{t,2}^1$ and $q_{s,2}^2 = q_{t,2}^2$ since the reflection map does not change the position coordinates. Equations (7.1.14) and (7.1.15) lead to:

$$\begin{aligned} R_2^{-1} \circ P_{2,4}^{-1}(v_{4,2}^{\min}, p) &= (q_{t,2}^1, p_{t,2}), \\ R_2^{-1} \circ P_{2,4}^{-1}(v_{4,2}^{\max}, p) &= (q_{t,2}^2, p_{t,2}). \end{aligned} \quad (7.1.16)$$

The map $R_2^{-1} \circ P_{2,4}^{-1}$ is depicted in red in Figure 7.10. The minimum $q_{t,2}^{\min} = \min\{q_{t,2}^1, q_{t,2}^2\}$ and maximum $q_{t,2}^{\max} = \max\{q_{t,2}^1, q_{t,2}^2\}$ are calculated. The points with coordinates $(q_{t,2}^{\min}, p_{t,2})$ and $(q_{t,2}^{\max}, p_{t,2})$ are depicted in blue Figure 7.13, where $p_{t,2} = 0.82$. To verify whether the corresponding rays are illuminated or not by the source, the procedure used for T_4 is now applied to T_2 along direction $p_{t,2} = 0.82$.

Next, the intersection points $(u_{2,j}^{\min}, p_{t,2})$ and $(u_{2,j}^{\max}, p_{t,2})$ of line $p_{t,2} = 0.82$ with boundaries $\partial T_{2,j}$ are computed for every $j \in \{1, 3\}$. We start from the boundary $\partial T_{2,1}$ obtaining the points $(u_{2,1}^{\min}, p_{t,2})$ and $(u_{2,1}^{\max}, p_{t,2})$ shown in Figure 7.13. Now, the position coordinates $v_{2,1}^{\min} = \max\{q_{t,2}^{\min}, u_{2,1}^{\min}\}$ and $v_{2,1}^{\max} = \min\{q_{t,2}^{\max}, u_{2,1}^{\max}\}$ need to be determined. All the rays located inside the segment $[v_{2,1}^{\min}, v_{2,1}^{\max}]$ in T_2 and with direction $p_{t,2}$ follow path $\Pi_2 = (1, 2, 4)$. In particular, the rays corresponding to the coordinates $(v_{2,1}^{\min}, p_{t,2})$ and $(v_{2,1}^{\max}, p_{t,2})$ are located on the boundaries of the region $R(\Pi_2)$ on T_4 formed by all the rays that follow path Π_2 . Their corresponding coordinates $(q^1(\Pi_2, p), p)$ and $(q^2(\Pi_2, p), p)$ on T_4 are obtained from:¹

$$\begin{aligned} P_{2,4} \circ R_2(v_{2,1}^{\min}, p_{t,2}) &= (q^1, p), \\ P_{2,4} \circ R_2(v_{2,1}^{\max}, p_{t,2}) &= (q^2, p). \end{aligned} \quad (7.1.17)$$

The rays corresponding to the coordinates (q^1, p) and (q^2, p) are located on the boundary $\partial R(\Pi_2)$ along direction $p = -0.2$. Indicating with $q^{\min} = \min\{q^1, q^2\}$ and $q^{\max} = \max\{q^1, q^2\}$, the distance $q^{\max} - q^{\min}$ gives the contribution to the intensity $I(p)$ of the rays located in $R(\Pi_2)$ where $p = -0.2$.

T_2 can also be illuminated by line 3, therefore the intersection points $(u_{2,3}^{\min}, p_{t,2})$ and $(u_{2,3}^{\max}, p_{t,2})$ of line $p_{t,2} = 0.82$ and $\partial T_{2,3}$ are calculated, these points are depicted in Figure 7.14. The coordinates $(v_{2,3}^{\min}, p_{t,2})$ and $(v_{2,3}^{\max}, p_{t,2})$ are shown in the same figure. As the source is not reached yet ($j = 3$), the rays corresponding to $(v_{2,3}^{\min}, p_{t,2})$ and $(v_{2,3}^{\max}, p_{t,2})$ are followed back using the inverses of the propagation and the reflection maps. The coordinates on T_3 are shown with blue circles in Figure 7.15 and they are obtained from:

$$\begin{aligned} R_3^{-1} \circ P_{3,2}^{-1}(v_{2,3}^{\min}, p_{t,2}) &= (q_{t,3}^1, p_{t,3}), \\ R_3^{-1} \circ P_{3,2}^{-1}(v_{2,3}^{\max}, p_{t,2}) &= (q_{t,3}^2, p_{t,3}). \end{aligned} \quad (7.1.18)$$

The minimum and the maximum position coordinates are $q_{t,3}^{\min} = \min\{q_{t,3}^1, q_{t,3}^2\}$ and $q_{t,3}^{\max} = \max\{q_{t,3}^1, q_{t,3}^2\}$, respectively. We found that $v_{3,2}^{\max} \neq u_{3,2}^{\max}$ because $[q_{t,3}^{\min}, q_{t,3}^{\max}] \subset [u_{3,2}^{\min}, u_{3,2}^{\max}]$, this means that the rays with corresponding position coordinates inside the interval $[q_{t,3}^{\max}, u_{3,2}^{\max}]$ will follow a different path. The procedure continues recursively. It stops either when the ray encounters the source, i.e., when $j = 1$, or when no intersection points between the direction $p = p_{t,k}$ and the boundaries $\partial T_{k,j}$ are determined for any $j \in 1, 2, 3$ with $j \neq k$.

¹With a slight abuse of notation we indicate $q^1(\Pi, p)$ with q^1 and $q^2(\Pi, p)$ with q^2 .

If the source is reached, then a valid path $\Pi = (1, 3, 2, 4)$ is found. Using the inverse of the propagation map, we compute

$$\begin{aligned} P_{1,3}^{-1}(q_{t,3}^{\min}, p_{t,3}) &= (q_{s,1}^1, p_{s,1}), \\ P_{1,3}^{-1}(q_{t,3}^{\max}, p_{t,3}) &= (q_{s,1}^2, p_{s,1}). \end{aligned} \quad (7.1.19)$$

The forward map $M_{1,4}(\Pi): S_1 \rightarrow R(\Pi)$ restricted to path $\Pi = (1, 3, 2, 4)$, i.e.

$$M_{1,4} = P_{2,4} \circ R_2 \circ P_{3,2} \circ R_3 \circ P_{1,3} \quad (7.1.20)$$

is applied to the coordinates $(q_{s,1}^1, p_{s,1})$ and $(q_{s,1}^2, p_{s,1})$:

$$\begin{aligned} M_{1,4}(q_{s,1}^1, p_{s,1}) &= (q^1(\Pi, p), p), \\ M_{1,4}(q_{s,1}^2, p_{s,1}) &= (q^2(\Pi, p), p). \end{aligned} \quad (7.1.21)$$

The coordinates $(q^1, p) := (q^1(\Pi, p), p)$ and $(q^2, p) := (q^2(\Pi, p), p)$ located on $\partial R(\Pi)$ in T_4 are found. Indicating with $q^{\min} = \min\{q^1, q^2\}$ and $q^{\max} = \max\{q^1, q^2\}$, the contribution to the intensity due to the rays that follow path Π is given by:

$$I(p) = I(p) + q^{\max}(\Pi, p) - q^{\min}(\Pi, p). \quad (7.1.22)$$

If no intersection points are found, then the rays traced are not emitted by the source, therefore no contribution to the intensity needs to be added. This is, for instance, the case for rays with coordinates $(v_{2,3}^{\min}, 0.82)$ and $(v_{2,3}^{\max}, 0.82)$ on T_2 in Figure 7.14. Below we explain this case in detail.

In Figure 7.15, the coordinates $(q_{t,3}^{\min}, p_{t,3})$ and $(q_{t,3}^{\max}, p_{t,3})$ in T_3 with $p_{t,3} = -0.29$ are shown. They are obtained from:

$$\begin{aligned} R_3^{-1} \circ P_{3,2}^{-1}(v_{2,3}^{\min}, 0.82) &= (q_{t,3}^1, p_{t,3}), \\ R_3^{-1} \circ P_{3,2}^{-1}(v_{2,3}^{\max}, 0.82) &= (q_{t,3}^2, p_{t,3}). \end{aligned} \quad (7.1.23)$$

From Figure 7.15 we note that there are no intersection points of the line $p_{t,3} = -0.29$ with $\partial T_{3,1}$. So, only the coordinates of the intersections $(u_{3,2}^{\min}, -0.29)$ and $(u_{3,2}^{\max}, -0.29)$ between line $p_{t,3} = -0.29$ and $\partial T_{3,2}$ are calculated. Next, the intersection interval

$$[v_{3,2}^{\min}, v_{3,2}^{\max}] = [u_{3,2}^{\min}, u_{3,2}^{\max}] \cap [q_{t,3}^{\min}, q_{t,3}^{\max}], \quad (7.1.24)$$

formed by parallel rays with direction $p_{t,3} = -0.29$, is considered. Using

$$\begin{aligned} R_2^{-1} \circ P_{2,3}^{-1}(v_{3,2}^{\min}, -0.29) &= (q_{t,2}^{\min}, p_{t,2}), \\ R_2^{-1} \circ P_{2,3}^{-1}(v_{3,2}^{\max}, -0.29) &= (q_{t,2}^{\max}, p_{t,2}), \end{aligned} \quad (7.1.25)$$

the corresponding coordinates $(q_{t,2}^{\max}, p_{t,2})$ and $(q_{t,2}^{\min}, p_{t,2})$ on T_2 are found (see Figure 7.16) with $p_{t,2} = -0.41$. Now the procedure is repeated again for T_2 along the direction $p_{t,2}$. No intersection points between the line $p_{t,2} = -0.41$ and $\partial T_{2,1}$ occur. Only, the intersection points $(u_{2,3}^{\min}, p_{t,2})$ and $(u_{2,3}^{\max}, p_{t,2})$ of line $p_{t,2} = -0.41$ and $\partial T_{2,3}$ are found. The intersection segment

$$[v_{2,3}^{\min}, v_{2,3}^{\max}] = [u_{2,3}^{\min}, u_{2,3}^{\max}] \cap [q_{t,2}^{\min}, q_{t,2}^{\max}] \quad (7.1.26)$$

is calculated. The coordinates on T_3 corresponding to the end points of the intersection interval are found using:

$$\begin{aligned} R_3^{-1} \circ P_{3,2}^{-1}(v_{2,3}^{\min}, p_{t,2}) &= (q_{t,3}^{\min}, p_{t,3}), \\ R_3^{-1} \circ P_{3,2}^{-1}(v_{2,3}^{\max}, p_{t,2}) &= (q_{t,3}^{\max}, p_{t,3}), \end{aligned} \quad (7.1.27)$$

where $p_{t,3} = 0.91$ (see Figure 7.17).

Considering the PS T_3 and the direction $p_{t,3} = 0.91$, we note that there are no intersection points of line $p_{t,3} = 0.91$ with both $\partial T_{3,1}$ and $\partial T_{3,2}$. Indeed, the whole segment $[q_{t,3}^{\min}, q_{t,3}^{\max}]$ is outside both $T_{3,2}$ and $T_{3,1}$. Because of this, all the rays with q -coordinates inside the interval $[q_{t,3}^{\min}, q_{t,3}^{\max}]$ and with direction $p = p_{t,3}$ are not illuminated by the source and no new real path is found.

Finally, the recursive procedure is applied to $T_{4,3}$. The first step is depicted in Figure 7.18. We decided not to show all the steps for $T_{4,3}$ as they are similar to those used for $T_{4,2}$ and explained above.

Finally, to compute the intensity along another direction $p^h \in [-1, 1]$ on T_4 , the procedure explained for $p = -0.2$ is repeated for $p = p^h$. In this way we find all the possible paths Π and the positive luminance regions $R(\Pi)$ on T_4 . Furthermore, considering every time the coordinates located on the boundaries of the regions $T_{j,k}$ for every $k \neq j$, also the boundaries $\partial R(\Pi)$ are determined for a given path Π as well as the coordinates $q^{\min}(\Pi, p)$ and $q^{\max}(\Pi, p)$ for every $p \in [-1, 1]$. In Algorithm 3, the main steps to calculate the intensity $I(p)$ along a direction $p = p^h$ in T_4 are given, where for the first step we take $k = 4$.

In the next section we provide the numerical results for the two-faceted cup.

7.2 Numerical results for the two-faceted cup

To demonstrate the accuracy of the method, a comparison with MC and QMC ray tracing is provided. The MC and QMC intensities are computed as explained in Chapter 3. We consider here the same partitioning $P : -1 = p^0 < \dots < p^{Nb} = 1$ of the interval $[-1, 1]$ used for all the simulations presented in the previous chapters. The profile of the QMC intensity is obtained tracing 10^7 rays and taking $Nb = 100$.

The PS intensity is obtained from (4.3.4) where the rays on the boundaries are obtained applying backward ray mapping. We observe that the method is suitable for detecting all the possible paths Π that a ray can follow during the propagation through the system. According to the results obtained with PS ray tracing, 5 different paths are found for the two-faceted cup. Given a path Π , the coordinates $(q^{\min}(\Pi, p^h), p^h)$ and $(q^{\max}(\Pi, p^h), p^h)$ of the corresponding rays located on $\partial R(\Pi)$ are determined for every $p = (p^h)_{h=0, \dots, Nb}$ where the values p^h are chosen from the partitioning P used for QMC ray tracing. These rays are depicted in Figure 7.19, where all the rays that follow the same path are shown with the same color. For the two-faceted cup, given a direction p^h and a path Π , only two rays are located on the boundary $\partial R(\Pi)$ of the corresponding region along that direction. As a consequence, at most $2NpNb$ rays need to be traced from the target to the source, where $Np = 5$ is the number of paths. The averaged normalized PS intensity is given by Equation (5.3.2) where the integrals are calculated using the trapezoidal rule. The approximated intensities $\hat{I}_A(A = \text{MC}, \text{QMC}, \text{PS})$ are compared to the reference intensity \hat{I}_{ref} which in this case



Figure 7.11: Target PS of line 4. $q_{t,4}^{\min}$ and $q_{t,4}^{\max}$ are the x -coordinates of the end points of line 4. The intersection points between the line $p = -0.2$ and $\partial T_{4,1}$ are $(u_{4,1}^{\min}, p)$ and $(u_{4,1}^{\max}, p)$. $v_{4,1}^{\min} = \max\{q_{t,4}^{\min}, u_{4,1}^{\min}\}$ and $v_{4,1}^{\max} = \min\{q_{t,4}^{\max}, u_{4,1}^{\max}\}$.

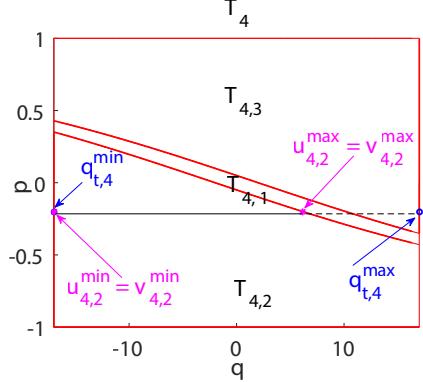


Figure 7.12: Target PS of line 4. The intersection points between the line $p = -0.2$ and $\partial T_{4,2}$ are $(u_{4,2}^{\min}, p)$ and $(u_{4,2}^{\max}, p)$. $v_{4,2}^{\min} = \max\{q_{t,4}^{\min}, u_{4,2}^{\min}\}$ and $v_{4,2}^{\max} = \min\{q_{t,4}^{\max}, u_{4,2}^{\max}\}$.

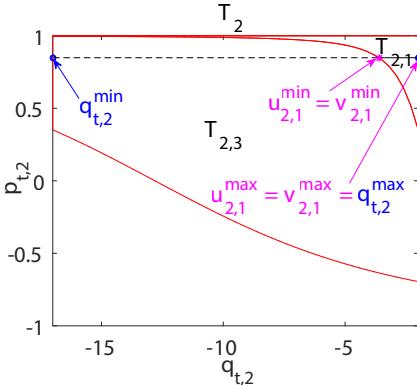


Figure 7.13: Target PS of line 2. The coordinates of the intersection points between the line $p_{t,2} = 0.82$ and $\partial T_{2,1}$ are $(u_{2,1}^{\min}, p_{t,2})$ and $(u_{2,1}^{\max}, p_{t,2})$. $v_{2,1}^{\min} = \max\{q_{t,2}^{\min}, u_{2,1}^{\min}\}$ and $v_{2,1}^{\max} = \min\{q_{t,2}^{\max}, u_{2,1}^{\max}\}$.

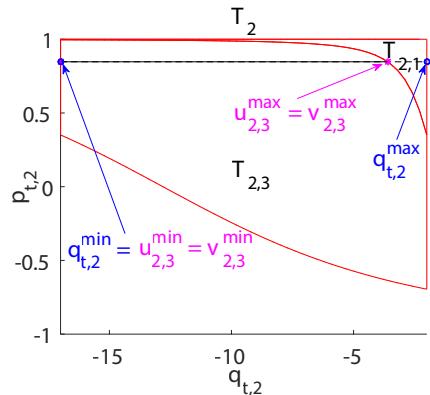


Figure 7.14: Target PS of line 2. The coordinates of the intersection points between the line $p_{t,2} = 0.82$ and $\partial T_{2,3}$ are $(u_{2,3}^{\min}, p_{t,2})$ and $(u_{2,3}^{\max}, p_{t,2})$. $v_{2,3}^{\min} = \max\{q_{t,2}^{\min}, u_{2,3}^{\min}\}$ and $v_{2,3}^{\max} = \min\{q_{t,2}^{\max}, u_{2,3}^{\max}\}$.



Figure 7.15: **Target PS of line 3.** The position coordinates of the intersection points between the line $p_{t,3} = -0.29$ and $\partial T_{3,2}$ are $u_{3,2}^{\min}$ and $u_{3,2}^{\max}$. $v_{3,2}^{\min} = \max\{q_{t,3}^{\min}, u_{3,2}^{\min}\}$ and $v_{3,2}^{\max} = \min\{q_{t,3}^{\max}, u_{3,2}^{\max}\}$.



Figure 7.16: **Target PS of line 2.** The intersection points between the line $p_t = p_{t,2}$ and $\partial T_{2,3}$ are $(u_{2,3}^{\min}, p_{t,2})$ and $(u_{2,3}^{\max}, p_{t,2})$. $v_{2,3}^{\min} = \max\{q_{t,2}^{\min}, u_{2,3}^{\min}\}$ and $v_{2,3}^{\max} = \min\{q_{t,2}^{\max}, u_{2,3}^{\max}\}$.

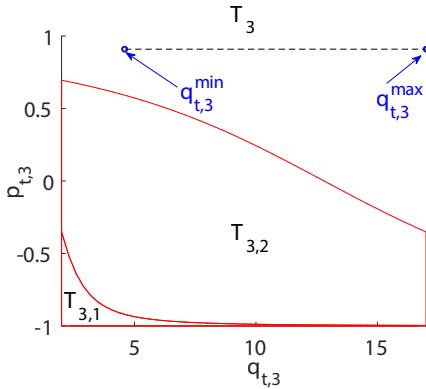


Figure 7.17: **Target PS of line 3.** There are no intersection points of the line $p_{3,2} = 0.91$ with the boundaries $\partial T_{3,2}$ and $\partial T_{3,1}$. The rays with coordinates inside the dotted segment hit again line 3 after some reflections and, therefore, are not emitted by the source.

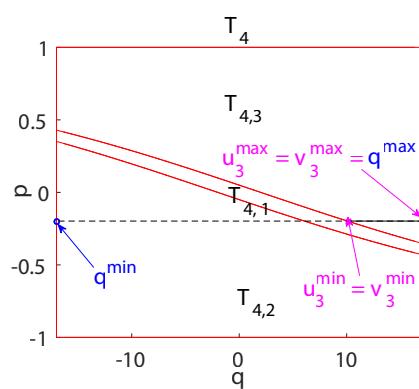


Figure 7.18: **Target PS of line 4.** $q_{t,4}^{\min} = -b$ and $q_{t,4}^{\max} = b$. The intersection points between the line $p = -0.2$ and $\partial T_{4,3}$ are $(u_{4,3}^{\min}, p)$ and $(u_{4,3}^{\max}, p)$. $v_{4,3}^{\min} = \max\{q_{t,4}^{\min}, u_{4,3}^{\min}\}$ and $v_{4,3}^{\max} = \min\{q_{t,4}^{\max}, u_{4,3}^{\max}\}$.

Algorithm 3 Recursive procedure for the intensity calculation

Initialize $k = 4$, $q_{t,4}^{\min} = q^{\min} = -b$, $q_{t,4}^{\max} = q^{\max} = b$, $p_{t,4} = p = \text{const}$, $\Pi = (4)$.

```

1: procedure INTENSITY COMPUTATION( $k$ ,  $q_{t,k}^{\min}$ ,  $q_{t,k}^{\max}$ ,  $p_{t,k}$ ,  $\Pi$ )
2:   for  $j = 1, 2, 3$  do
3:     if  $j \neq k$  then
4:       Compute the intersection points  $(u_{k,j}^{\min}, p_{t,k})$  and  $(u_{k,j}^{\max}, p_{t,k})$ 
5:        $\Pi = (j, \Pi)$ 
6:       Compute  $[v_{k,j}^{\min}, v_{k,j}^{\max}] = [u_{k,j}^{\min}, u_{k,j}^{\max}] \cap [q_{t,k}^{\min}, q_{t,k}^{\max}]$ 
7:       if  $(j \neq 1) \& (j \neq 4)$  then
8:         Trace back from  $T_k$  to  $T_j$ 

$$(q_{t,j}^1, p_{t,j}) = R_j^{-1} \circ P_{j,k}^{-1}(v_{k,j}^{\min}, p_{t,k})$$


$$(q_{t,j}^2, p_{t,j}) = R_j^{-1} \circ P_{j,k}^{-1}(v_{k,j}^{\max}, p_{t,k})$$

9:   Determine

$$q_{t,j}^{\min} = \min\{q_{t,j}^1, q_{t,j}^2\} \text{ and } q_{t,j}^{\max} = \max\{q_{t,j}^1, q_{t,j}^2\}$$

10:  return INTENSITY COMPUTATION( $j$ ,  $q_{t,j}^{\min}$ ,  $q_{t,j}^{\max}$ ,  $p_{t,j}$ ,  $\Pi$ )
11: else
12:   if  $j=1$  then
13:     if  $k \neq 4$  then
14:       Trace back from  $T_k$  to  $S_1$ , next apply the forward map

$$M_{1,4}(\Pi)$$


$$(q_{s,1}^1, p_{s,1}) = P_{1,k}^{-1}(v_{k,1}^{\min}, p_{t,k})$$


$$(q_{s,1}^2, p_{s,1}) = P_{1,k}^{-1}(v_{k,1}^{\max}, p_{t,k})$$


$$(q^1(\Pi, p), p) = M_{1,4}(\Pi)(q_{s,1}^1, p_{s,1})$$


$$(q^2(\Pi, p), p) = M_{1,4}(\Pi)(q_{s,1}^2, p_{s,1})$$

15:   Calculate

$$q^{\min}(\Pi, p) = \min\{q^1, q^2\}$$


$$q^{\max}(\Pi, p) = \max\{q^1, q^2\}$$

16:   where  $q^1 := q^1(\Pi, p)$  and  $q^2 := q^2(\Pi, p)$ .
17:   return  $I(p) = I(p) + q^{\max}(\Pi, p) - q^{\min}(\Pi, p)$ 
18:   else

$$q^{\min}(\Pi, p) = v_{4,k}^{\min}, q^{\max}(\Pi, p) = v_{4,k}^{\max}$$

19:   return  $I(p) = I(p) + q^{\max}(\Pi, p) - q^{\min}(\Pi, p)$ .
20:   end if
21:   else
22:     return  $I(p)$ 
23:   end if
24:   end if
25:   end if
26: end for
27: end procedure
```



Figure 7.19: **Target phase space of the two-faceted cup divided into 100 bins.** Five different paths are found. The rays with coordinates (q^{\min}, p) and (q^{\max}, p) in T_4 that are located at the boundaries $\partial R(\Pi)$ are depicted with dots, the color of the dots depends on the path Π followed by the rays. Using the ray mapping method, only these rays need to be traced from S to T for the intensity computation.

in the exact intensity ($\hat{I}_{\text{ref}} = \hat{I}_{\text{exact}}$). The results in Figure 7.20 show that our method computes the intensity correctly.

To compare the speed of convergence of the three methods, we consider the error between the approximate intensities \hat{I}_A ($A = \text{MC}, \text{QMC}, \text{PS}$) and the exact intensity $\hat{I}_{\text{exact}} = \hat{I}_{\text{ref}}$. The three errors as a function of the CPU-time are depicted in a logarithmic scale in Figure 7.21. Numerical results show that MC ray tracing converges proportionally to the inverse of the square root of the number of rays traced, QMC error is proportional to the inverse of the number of rays, backward ray mapping is able to compute the output intensity of the two-faceted cup exactly. Also, it is much faster than MC ray tracing when an error smaller than 10^{-4} is required and it is faster than QMC ray tracing if an error smaller than around 10^{-5} is desired.

7.3 Extension of the method for the multi-faceted cup

The method can be generalized to more complicated optical systems. In particular, it can be used for all systems formed by straight line segments. The goal of this section is to show the generalization of the method to the multi-faced cup which is a system

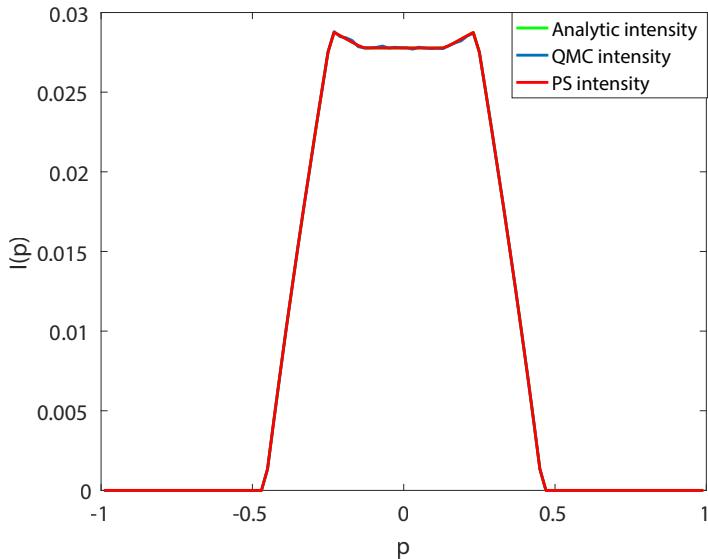


Figure 7.20: **Intensities for the two-faceted cup.** The intensities found with three different approaches are shown.

with many left and right segments as reflectors. The design of this system is explained below.

A multi-faceted cup is an optical system formed by a source, a target and $Nl - 2$ reflectors, where Nl is the number of optical line segments that form the system. Defining a Cartesian coordinate system (x, z) , the multi-faceted cup is symmetric with respect to the optical axis (z -axis). An example of this system is depicted in Figure 7.22 where all the lines are labeled with numbers. The source $S = [-a, a]$ (line 1) and the target $T = [-b, b]$ (line 22) are two segments both perpendicular to the optical axis, with $a = 2$ and $b = 17$. S is located at the height $z = 0$ while T has a height $z = 40$. Both sides of the system are divided into 10 segments which connect S with T . The ten adjacent segments at the left of the system (lines 2, \dots , 11) connect the left extreme of the source with the left extreme of the target. Similarly, ten adjacent segments at the right of the system (lines 12, \dots , 21) connect the right extreme of the source with the right extreme of the target. These segments are designed as follows. The intervals $[-b, -a]$ and $[a, b]$ are divided into ten sub-intervals of the same length $(b - a)/10$. The x -coordinates of the end points of the line segments 12, \dots , 21 are equal to the x -coordinates of the sub-intervals of $[a, b]$, while the x -coordinates of the end points of the line segments 2, \dots , 11 are equal to the x -coordinates of the sub-intervals of $[-a, -b]$. The z -coordinates of every end point of the line segments 2, \dots , 21 are given substituting their x -coordinates into the equation of the parabola whose symmetry axis is equal to the z -axis and that passes through the points $(-17, 40)$ and $(17, 40)$. The 20-faceted cup is now well defined and can be seen as an approximation of a parabolic reflector.

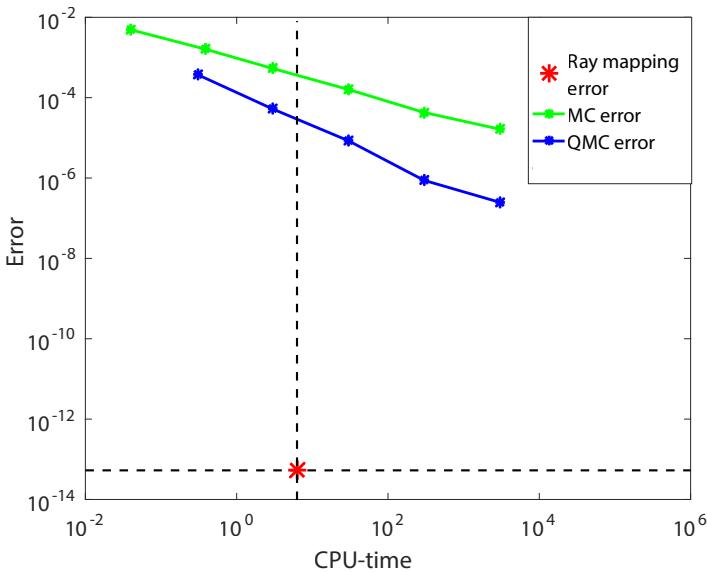


Figure 7.21: **Errors for the two-faceted cup.** The errors are depicted as a function of the CPU time (in seconds).

Similarly to the two-faceted cup, also for the multi-faceted cup we define the phase spaces of all the lines $j \in \{1, \dots, N_l\}$ (for the 20-faceted cup $N_l = 22$ which is also the index of the target). For the system in Figure 7.22, 42 different phase spaces need to be considered. In general, for a system formed by N_l straight line segments, $2N_l - 2$ phase spaces are considered. For all the systems formed by straight line segments, the boundaries $(\partial S_{j,k})_{j \neq k=2, \dots, N_l}$ and $(\partial T_{j,l})_{j \neq l=1, \dots, N_l-1}$ of the regions that form every PS are determined.

The boundaries $(\partial T_{N_l,l})_{l=1, \dots, N_l-1}$ for the 20-faceted cup are depicted in Figure 7.23 with red lines. All the possible paths that the rays can follow when propagating within the 20-faceted cup are determined using the same algorithm developed for the two-faceted cup and explained in Section 7.1.2. As the number of optical lines increases, the number of possible paths increases as well. Therefore, we have to construct a more complicated tree than the one in Figure 7.10. Despite this, the algorithm explained in the previous section still works fine and, also for the multi-faceted cup we are able to determine all the possible paths Π and all the positive luminance regions $R(\Pi)$ at target PS. For a given direction $p = \text{const}$ the position coordinates $q^{\min}(\Pi, p)$ and $q^{\max}(\Pi, p)$ of the intersection points between the boundaries $\partial R(\Pi)$ and the line $p = \text{const}$ are calculated for every possible path Π . Finally, the target intensity $\hat{I}_{PS}(p)$ along the direction p is obtained. Numerical results for a 20-faceted cup are given in the next section.

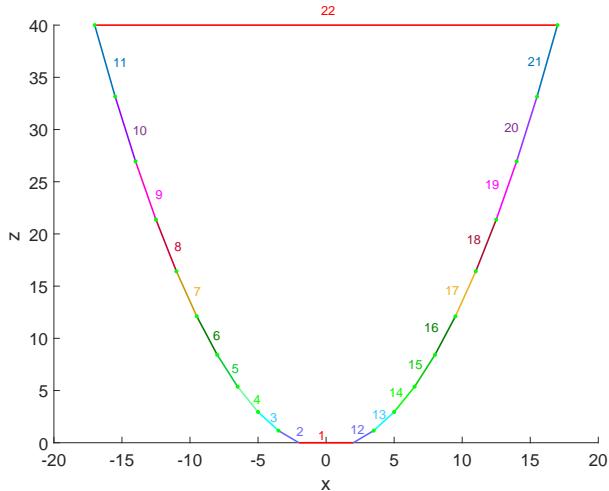


Figure 7.22: **The 20-faceted cup.** The system is formed by 22 different line segments: the source S , the target T , ten left reflectors and ten right reflectors. $S = [-2, 2]$ is located at $z = 0$. $T = [-17, 17]$ is parallel to the source and it is located at a height $z = 40$. All the lines are located in air.

7.4 Numerical results for the 20-faceted cup

In this section the results for the 20-faceted cup are presented. We compute the target intensity both with the backward ray mapping method and MC ray tracing. The same partitioning P of the interval $[-1, 1]$ used for the two-faceted cup is considered. A comparison between the reference intensity \hat{I}_{ref} and the ray mapping intensity \hat{I}_{PS} is shown in Figure 7.24, where \hat{I}_{ref} is obtained using QMC ray tracing with 10^8 rays.

Note that the intensity profile in Figure 7.24 is more concentrated around the direction $p = 0$ than the intensity of the two-faceted cup (see Figure 7.20). In particular, increasing the number of left and right reflectors the intensity profile becomes more and more peaked around the center approaching the profile of a parabolic reflector, (see Chapter 6). The error between the approximate intensities \hat{I}_A ($A = \text{QMC}, \text{PS}$) and the reference intensity \hat{I}_{ref} is shown in Figure 7.25. The PS intensity is calculated using (5.3.2) where the integral is approximated using the trapezoidal rule. Figure 7.25 shows that increasing the number of intervals in the trapezoidal rule, the PS error decreases. We computed the PS error four times considering 1, 10, 50, and 100 intervals in the trapezoidal rule. We remark that the PS method gives the value of the intensity pointwise, therefore we can compute the PS intensity without numerical integration. Nevertheless, we calculate the averaged intensity because we want to compare it with the QMC intensity \hat{I}_{QMC} .

The error convergence is depicted in Figure 7.25 with the red line. Since all the boundaries of the regions in PS are calculated exactly, our expectation is that the PS intensity is the exact intensity. From Figure 7.25 we observe that the minimum ray mapping error has an order of magnitude of 10^{-7} . This is due to the fact that for the 20-faceted cup the intensity cannot be computed exactly. Therefore, we took as



Figure 7.23: **Target PS of the 20-faceted cup.** The red lines are the boundaries $(\partial T_{22,l})_{l=1, \dots, 21}$ which are determined analytically. The numbers inside the regions $T_{22,l}$ indicate the value of the index l .



Figure 7.24: **Intensity for the 20-faceted cup.** Comparison between the reference intensity (QMC ray tracing with 10^8 rays) and the ray mapping intensity.

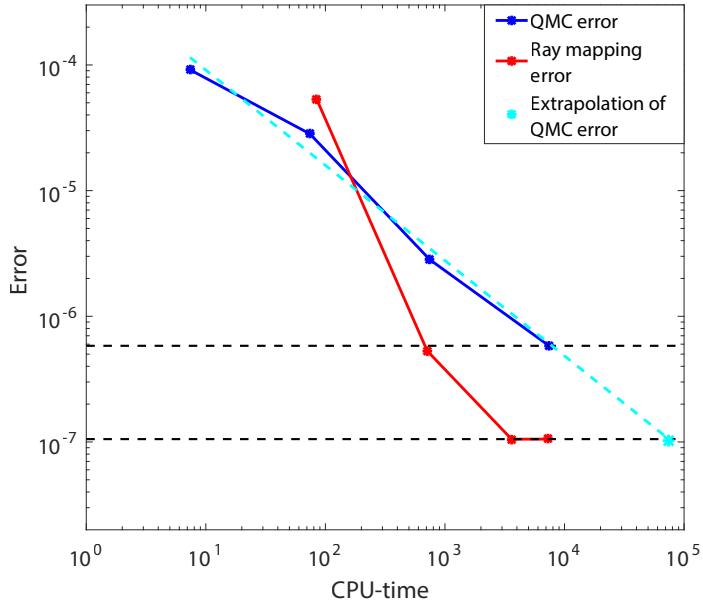


Figure 7.25: **Errors for the 20-faceted cup as a function of the CPU-time.** The ray mapping method is more accurate than QMC ray tracing and it is faster in case an error smaller than 10^{-5} is desired.

reference intensity \hat{I}_{ref} an intensity computed with QMC ray tracing using 10^8 rays which is not the exact intensity. The error between the normalized exact intensity \hat{I}_{exact} and the normalized approximate intensity \hat{I}_A is given by:

$$\frac{1}{Nb} \sum_{h=1}^{Nb} |\hat{I}_{\text{exact}}(p^h) - \hat{I}_A(p^h)| \leq \frac{1}{Nb} \left(\sum_{h=1}^{Nb} |\hat{I}_{\text{exact}}(p^h) - \hat{I}_{\text{ref}}(p^h)| + \sum_{h=1}^{Nb} |\hat{I}_{\text{ref}}(p^h) - \hat{I}_A(p^h)| \right). \quad (7.4.1)$$

To give an estimation of the first term on the right hand side of (7.4.1), we do a linear extrapolation from the errors values obtained. The extrapolated point at the time needed for computing the reference intensity (cyan dot in Figure 7.25) gives an estimation of the error between the reference intensity and the exact intensity. If the reference intensity is *exact* then we would expect an error *exactly* equal to 0. From the extrapolation we obtain

$$\sum_{h=1}^{Nb} |\hat{I}_{\text{exact}}(p^h) - \hat{I}_{\text{ref}}(p^h)| / Nb \approx 1.02 * 10^{-7}.$$

The results show that

$$\sum_{h=1}^{Nb} |\hat{I}_{\text{exact}}(p^h) - \hat{I}_{\text{ref}}(p^h)| / Nb \approx \sum_{h=1}^{Nb} |\hat{I}_{\text{ref}}(p^h) - \hat{I}_{\text{PS}}(p^h)| / Nb.$$

Since the accuracy of the reference intensity is comparable with the accuracy of the PS intensity, we claim that the error found with backward ray mapping is also due to the QMC error. We expect that considering a more accurate reference intensity, obtained for instance using QMC ray tracing with more rays, the PS error would decrease. We conclude that the backward ray mapping method performs very well also for more complicated systems. Compared to QMC ray tracing the new method is not only faster but also much more accurate.

7.5 Discussions

In this chapter, we presented an backward ray mapping method to compute the target intensity of a given optical system. The method employs the PS of *all* the lines that form the system. All these phase spaces are related to each other through two different kind of maps. A concatenation of these two maps gives a map that connects the coordinates of the rays at the source with those at the target. Employing the inverse of the concatenated map, all the possible paths that rays can follow during their propagation are found. Only the rays located on the boundaries of the positive luminance regions are traced, where every region is formed by rays that follow the same path during their propagation. From those rays the output intensity is calculated.

We presented numerical results for two optical systems: the two-faceted cup and the 20-faceted cup. The boundaries of the regions that form every PS are determined exactly. Numerical results show that the exact output intensity is obtained. We compared our method with MC and QMC ray tracing showing significant advantages in terms of the accuracy and the computational time. We conclude that the ray mapping method applied to systems formed by straight line segments calculates the *exact* intensity.

In the next chapter we present the method extended to systems formed by curved lines.

Chapter 8

Direct backward ray mapping

In Chapter 7 we introduced an inverse method based on ray mapping reconstruction in PS. The goal was to calculate the intensity distribution at the target of an optical system.

The idea was to construct a map from the target T to the source S using the PS of all the optical lines, which are divided into several regions. The method developed in the previous chapter requires that the boundaries of these regions can be determined exactly in every PS. Therefore, also the positive luminance regions were found analytically and the *exact* intensity could be determined. This is only possible for systems formed by straight line segments.

In this chapter we modify the method to systems formed by curved lines. In this case, the boundaries of the regions in PS cannot be determined exactly. Because of this, we need to apply a numerical procedure. In particular, we develop a method that employs only the PS of the target of the system. The boundaries are detected applying a bisection procedure in target PS in combination with backward ray tracing. The method is tested for two optical systems: the TIR-collimator and a parabolic reflector. The results are presented in Section 8.2 and 8.3, respectively.

8.1 Bisection method and backward ray tracing

The purpose of this section is to present the direct backward ray mapping method valid for systems formed by curved lines. Given a partition $P : -1 = p^0 < p^1 < \dots < p^{Nb} = 1$ of the interval $[-1, 1]$ with Nb the number of bins in the partitioning, the intensity in target PS is given by Equation (4.3.4) for every $p \in P$. Therefore, the problem reduces to calculating the coordinates $q^{\min}(\Pi, p)$ and $q^{\max}(\Pi, p)$ of the rays on $\partial R(\Pi)$ for every path Π .

We indicate with $(q^a, p) = (-b, p)$ and $(q^b, p) = (b, p)$ the coordinates of the end points of T along direction p . These points are associated to two rays in PS. Consider the corresponding position coordinate (x^a, z^a) and (x^b, z^b) and angular coordinates t^a and t^b of the ray in real space, where $x^a = q^a$, $x^b = q^b$, $z^a = z^b = h$ where h is the height of the target and $t^a = t^b = p$ refers to the emitted ray. Next, the rays parametrizations $r_a(s)$ and $r_b(s)$ are determined according to (3.1.1). We remind the reader that, in real space, the coordinates of each ray on line $j \neq Nl$ are

indicated with (x_j, z_j) and t_j where (x_j, z_j) are the coordinate of the intersection point between the ray and line j and t_j is the direction of the incident ray with respect to the *optical axis*. The procedure starts with intensity $I(p) = 0$ and the end points $(q^a, p) = (-b, p)$ and $(q^b, p) = (b, p)$. Since the boundaries of the regions in all the phase spaces are unknown, to determine from which line \mathbf{r}_a and \mathbf{r}_b are emitted we apply backward ray tracing. We denote with Nl the index of the target (line Nl) and with $k \in \{1, \dots, Nl - 1\}$ and $l \in \{1, \dots, Nl - 1\}$ the lines from which the rays with parametrization $\mathbf{r}_a(s)$ and $\mathbf{r}_b(s)$ are emitted, respectively. $\Pi^a = (k, Nl)$ and $\Pi^b = (l, Nl)$ are the last part of the paths followed by the two rays \mathbf{r}_a and \mathbf{r}_b , respectively. $(q^a, p) \in \partial R(\Pi^a)$ and $(q^b, p) \in \partial R(\Pi^b)$. At this stage we know whether the two rays are emitted from the same line or not.

First, assume $k = l$, then \mathbf{r}_a and \mathbf{r}_b hit the same line before reaching the target. In case $k = 1$ a possible path from the source to the target is found and, assuming a Lambertian source, the intensity is updated according to:

$$I(p) = I(p) + q^{\max}(\Pi^a, p) - q^{\min}(\Pi^a, p), \quad (8.1.1)$$

where $q^{\min}(\Pi^a, p) = q^{\min} = \min\{q^a, q^b\}$ and $q^{\max}(\Pi^a, p) = q^{\max} = \max\{q^a, q^b\}$. In case $k \neq 1$ the two rays \mathbf{r}_a and \mathbf{r}_b are traced back further using backward ray tracing.

Next, if $k \neq l$ the rays \mathbf{r}_a and \mathbf{r}_b are emitted from two different lines, hence $\Pi^a \neq \Pi^b$ and belong to different regions $R(\Pi^a)$ and $R(\Pi^b)$ in T . q^a in $R(\Pi^b)$, the other intersection points of line $p = \text{const}$ with the boundary $R(\Pi^b)$ are unknown. To determine the other coordinates of the rays on the boundary $\partial R(\Pi^a)$, the bisection method is applied to the segment $[q^a(\Pi^a, p), q^b(\Pi^b, p)]$ in target PS T along direction p . Thus, this interval is repeatedly halved until the position coordinate in target PS of the ray that follows the same path Π^a of \mathbf{r}_a is found (the corresponding direction coordinate p is fixed). The bisection procedure continues until the length of the segment considered becomes smaller than a fixed tolerance. Giving as input the coordinates $q^a(\Pi^a, p)$ and $q^b(\Pi^b, p)$ of the rays with parametrization $\mathbf{r}_a(s)$ and $\mathbf{r}_b(s)$, the path Π^a and the tolerance $\text{tol} = 10^{-12}$, the bisection method is implemented as in Algorithm 4. Similarly, others paths will be obtained later using the same procedure applied to another interval in target PS.

Once bisection stops, two points with coordinates (q^c, p) and (q^d, p) in T are found. The corresponding rays \mathbf{r}_c and \mathbf{r}_d follow path $\Pi^c = \Pi^a$ and $\Pi^d \neq \Pi^a$. All the rays with target coordinates (q, p) and $q^a \leq q \leq q^c$ follow path Π^a , while the rays with target coordinates (q, p) with $q^d \leq q \leq q^b$ follow another path $\Pi \neq \Pi^a$ (see Figure 8.1).

Now, if $k \neq 1$ the procedure applied to the interval $[q^a(p), q^b(p)]$ needs to be applied to $[q^a(p), q^c(p)]$ until the source is reached, i.e., until $k = 1$. If $k = 1$, the source is reached by the rays traced back from the target. This means that a possible path Π^a from S to T is found and the position coordinates $q^{\min}(\Pi^a, p)$ and $q^{\max}(\Pi^a, p)$ are determined and the intensity is updated according to (8.1.1).

Finally, to detect all possible paths that can occur along direction p the procedure explained above is applied also to the interval $[q^d(p), q^b(p)]$ along direction p continuing until the entire interval $[q^a(p), q^b(p)]$ is investigated. The main steps of the method are outlined in the following.

- Given a direction p , the end points (q^a, p) and (q^b, p) of the target PS T , where $q^a = -b$, $q^b = b$. Start from $j = Nl$.

Algorithm 4 Bisection($q^a(\Pi^a, p)$, $q^b(\Pi^b, p)$, $\mathbf{r}_a(s)$, $\mathbf{r}_b(s)$, tol, Π^a)

Initialize: step = 0, $j = Nl$

```

1: while  $|q^a - q^b| > \text{tol}$  do
2:    $x_j^m = q^m = (q^a + q^b)/2$ ,
3:    $z_j^m = z^a$ 
4:    $t_j^m = p^m = p$ 
5:    $\Pi^m = (Nl)$ 
6:   Consider the parametrization  $\mathbf{r}_m$  of the ray corresponding to  $(x_j^m, z_j^m)$  and  $t_j^m$ ,
7:   while step < length( $\Pi^a$ ) - 1 do
8:     Trace back the ray with parametrization  $\mathbf{r}_m$  from  $j$ 
9:     Find the line  $k$  that the ray hits
10:    Find the coordinates  $(x_k^m, z_k^m)$  on line  $k$ 
11:    Calculate the new direction  $t_j^m$  with respect to the optical axis
12:     $\Pi^m = (k, \Pi^m)$ .
13:    if  $k = 1$  or  $k = Nl$  then
14:      step = length( $\Pi^a$ )            $\triangleright$  If the source or the target are reached
15:       $\quad\quad\quad$   $\triangleright$  then exit from the while loop.
16:    else
17:      step = step + 1
18:    end if
19:  end while
20:  if  $\Pi^a = \Pi^m$  then
21:     $q^a = q^m$ 
22:     $\mathbf{r}_a = \mathbf{r}_m$ 
23:  else
24:     $q^b = q^m$ 
25:     $\Pi^b = \Pi^m$ 
26:  end if
27: end while
28:  $q^c = q^a, \Pi^c = \Pi^a$ .
29:  $q^d = q^b, \Pi^d = \Pi^b$ .
30: return  $(q^c, p), (q^d, p), \Pi^c$  and  $\Pi^d$ .
```

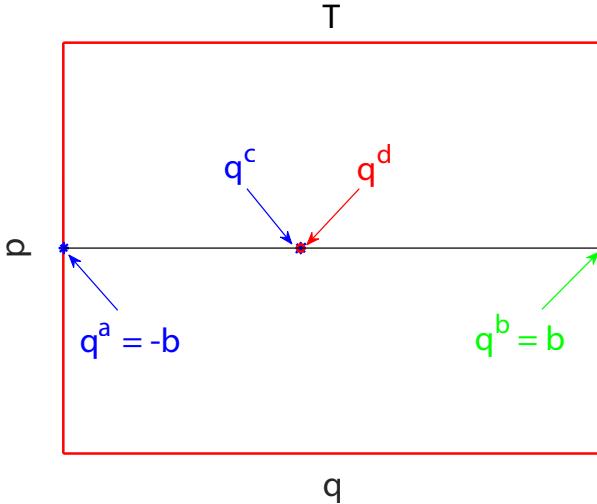


Figure 8.1: **Bisection in target PS T .** Algorithm 4 is run for the interval $[q^a, q^b]$ along direction $p = 0$. The coordinates q^c and q^d are found such that $|q^c - q^d| < \text{tol}$. $\Pi^c = \Pi^a$ and $\Pi^d \neq \Pi^a$.

2. Using backward ray tracing, trace back from line j the rays with parametrizations $r_a(s)$ and $r_b(s)$ corresponding to the position coordinates (x_j^a, z_j^a) and (x_j^b, z_j^b) and the angular coordinates t_j^a and t_j^b , respectively.
3. Determine indices $k \neq j$ and $l \neq j$ of the lines from which the rays with parametrizations $r_a(s)$ and $r_b(s)$ originated.
4. Consider the new rays parametrization $r_a(s)$ and $r_b(s)$ corresponding to the coordinates (x_k^a, z_k^a) , t_k^a and (x_l^a, z_l^b) , t_l^b , respectively.
5. Update the paths Π^a and Π^b : $\Pi^a = (k, \Pi^a)$ and $\Pi^b = (l, \Pi^b)$
6. If $k = l \neq 1$ and $k = l \neq Nl$
 - Set $j = k$
 - Restart the procedure from point 2
7. If $k = l = 1$
 - A relevant path $\Pi^a = \Pi^b$ is found.
 - Determine

$$q^{\min}(\Pi^a, p) = \min\{q^a(\Pi^a, p), q^b(\Pi^b, p)\}$$

$$q^{\max}(\Pi^a, p) = \max\{q^a(\Pi^a, p), q^b(\Pi^b, p)\}.$$
 - Update the intensity

$$I(p) = I(p) + q^{\max}(\Pi^a, p) - q^{\min}(\Pi^a, p)$$

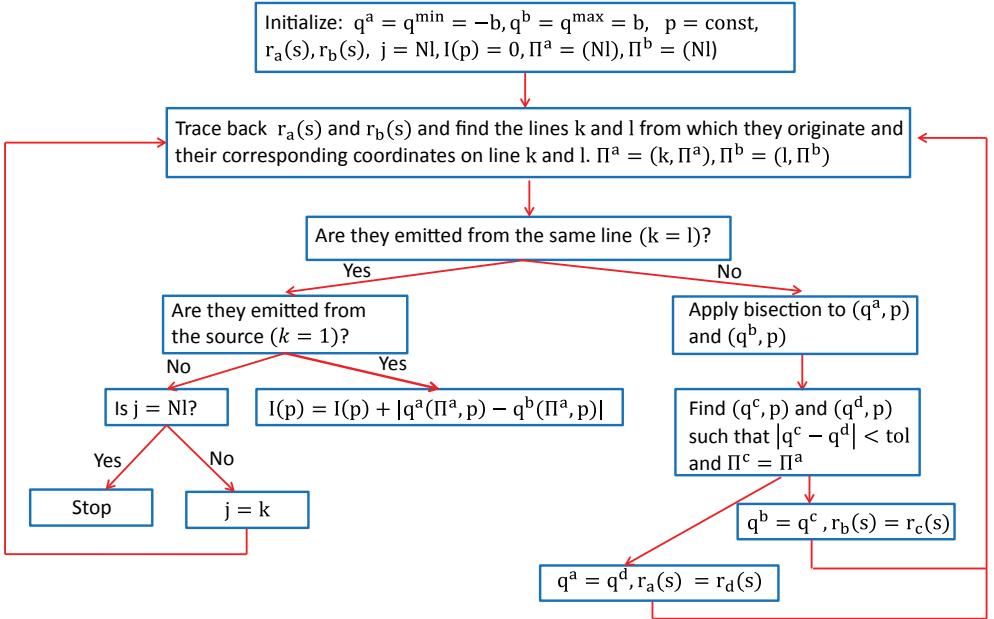


Figure 8.2: Main steps of direct backward ray mapping for systems with curved lines.

8. If $k \neq l$

- Apply the bisection method to the interval $[q^a, q^b]$ along direction p , given the points (q^c, p) and (q^d, p) in target PS T such that $|q^c - q^d| < \text{tol}$.
- If $k \neq Nl$
 - Set $q^b = q^c$,
 - Set $(x_k^b, z_k^b) = (x_k^c, z_k^b)$ and $t_k^b = t_k^c$
 - Set $j = k$
 - Restart from 2 with the updated coordinates,
- Update $q^a = q^d$,
- $\Pi^a = \Pi^c$
- Update $(x_k^a, z_k^a) = (x_k^d, z_k^d)$ and $t_k^a = t_k^d$
- Restart from 2

Giving as input $I(p) = 0$ for every direction p and the tolerance $\text{tol} = 10^{-12}$, the method is defined in the recursive Algorithm 5.

The procedure is able to determine all the possible paths that the rays can follow during their propagation from S to T. Also, rays located on the boundaries of the regions with positive luminance in target PS T are found. The method is summarized by the flowchart in Figure 8.2

Next, the method is applied to two optical systems formed by curved lines. In the

Algorithm 5 Recursive function for direct backward ray mapping

Initialize: $j = Nl$, $q^a = x_j^a = -b$, $q^b = x_j^b = b$, $p = t_j^a = t_j^b = \text{const}$, $z_j^a = z_j^b = h$, $\Pi^a = (Nl)$.

```

1: procedure INTENSITY COMPUTATION( $q^a, q^b, x_j^a, x_j^b, z_j^a, z_j^b, p, t_j^a, t_j^b, \Pi^a, j$ )
2:   Apply backward ray tracing to  $(x_j^a, z_j^a), t_j^a$  and  $(x_j^b, z_j^b), t_j^b$ 
3:   Determine the lines  $k \neq j$  and  $l \neq j$  from which  $r_a(s)$  and  $r_b(s)$  are emitted.
4:   Update path  $\Pi^a = (k, \Pi^a)$ 
5:   Calculate the position coordinates  $(x_k^a, z_k^a)$ ,  $(z_k^b, p_k^b)$  and  $t_k^a, t_k^b$ .
6:   if  $k = l$  then
7:     if  $k \neq 1$  then
8:       return INTENSITY COMPUTATION( $q^a, q^b, x_k^a, x_k^b, z_k^a, z_k^b, p, t_k^a, t_k^b, \Pi^a,$ 
k)
9:     else
10:      Calculate  $q^{\min} = \min\{q^a, q^b\}$ ,  $q^{\max} = \max\{q^a, q^b\}$ 
11:      Assume Lambertian source
12:       $I(p) = I(p) + q^{\max}(\Pi^a, p) - q^{\min}(\Pi^a, p)$ 
13:    end if
14:  else
15:    Apply bisection to the segment  $[q^a(\Pi^a, p), q^b(\Pi^b, p)]$ 
16:    Find the target coordinates  $(q^c, p)$  and  $(q^d, p)$  of the rays with parameterization  $r_c$  and  $r_d$ , such that

```

$$|q^c - q^d| < \text{tol}$$

```

17:  if  $k \neq Nl$  then
18:    return INTENSITY COMPUTATION( $q^a, q^c, x_k^a, x_k^c, z_k^a, z_k^c, p, t_k^a, t_k^c, \Pi^a, k$ )
19:  end if
20:  return INTENSITY COMPUTATION( $q^d, q^b, x_k^d, x_k^b, z_k^d, z_k^b, p, t_k^d, t_k^b, \Pi^d, k$ )
21: end if
22: end procedure

```

next section we show the results for the TIR-collimator and in Section 8.3 we provide numerical results for the parabolic reflector.

8.2 Results for the TIR-collimator

In this section we apply direct backward ray mapping to the TIR-collimator presented in Chapter 5 and depicted in Figure 5.10. The target PS of this system is the rectangular $T = [-b, b] \times [-1, 1]$ with $b = 9.7$. The aim is to detect all the possible path Π and the rays located on the boundaries $\partial R(\Pi)$ of the corresponding regions in target PS.

In Chapter 5, we found five different paths for the TIR-collimator (see Figure 5.11). The boundaries of the corresponding regions in target PS T are in general difficult to approximate. Furthermore, along one direction p more than two points can be located on the boundary $\partial R(\Pi) \cap R(\Pi)$ corresponding to a certain path Π . To determine properly all the boundaries $\partial R(\Pi)$, we need to divide the interval $[-b, b]$ in T into intervals of the same length. Hence, we consider a partitioning $Q = -b = q^0 < q^1 < \dots < q^{N_i} = b$ of $[-b, b]$ where N_i is the total number of sub-intervals along the q -axis. For each direction $p \in [-1, 1]$ the procedure explained in Section 8.1 is repeated for every sub-interval $[q^k(p), q^{k+1}(p)] \subset [q^a(p), q^b(p)]$ with $k = 0, \dots, N_i - 1$ and $q^a(p) = -b$ and $q^b(p) = b$.

To establish in how many sub-intervals N_i we need to divide the target, we exploit étendue conservation. We use the same idea applied to determine the value of α for the α -shapes method and to provide a stopping criterion for the triangulation refinement (see Chapters 5 and 6). The source étendue U_1 is calculated from (5.3.1), obtaining roughly $U_1 \approx 7.7$. The target étendue U_t is given by Equation (5.2.5). U_t is calculated several times considering every time a different partitioning Q for the q -axis of the target PS. Next, the absolute value of the difference between the source and target étendue is obtained from

$$\Delta U = |U_1 - U_t|. \quad (8.2.1)$$

If a small value of ΔU is obtained, then a good approximation of U_t is found and therefore, the partition Q used for the computation of U_t is suitable for detecting correctly the boundaries $\partial R(\Pi)$. In Figure 8.3 we show how ΔU decreases by increasing the number of sub-intervals N_i in the partitioning Q . Note that when $N_i = 100$ the étendue difference is roughly $5 \cdot 10^{-3}$. This can be related to the fact that both the source étendue U_1 and the target étendue U_t are approximated. In Figure 8.4 we show the distribution of the rays traced using the backward ray mapping method with $N_i = 30$ and $N_b = 100$. In this case, backward ray mapping detects 11 different paths from the source to the target. We observe that only 5 of them are the paths that we expect from PS ray tracing, which are:

$$\begin{aligned} \Pi_1 &= (1, 2, 7, 12), \\ \Pi_2 &= (1, 4, 6, 7, 12), \quad \Pi_3 = (1, 10, 8, 7, 12), \\ \Pi_4 &= (1, 3, 7, 12), \quad \Pi_5 = (1, 11, 7, 12). \end{aligned} \quad (8.2.2)$$



Figure 8.3: Difference between the source and the target étendue for the TIR-collimator. U_t is computed four times increasing every time the number of bins N_i where $N_i \in \{5, 10, 30, 100\}$.

(see Figure 5.10 for the of the line numbering of the TIR-collimator). Backward ray mapping also detects the spurious paths:

$$\begin{aligned} \Pi_6 &= (1, 2, 9, 8, 7, 12), & \Pi_7 &= (1, 2, 5, 6, 7, 12), \\ \Pi_8 &= (1, 2, 2, 7, 12), & \Pi_9 &= (1, 7, 12), \\ \Pi_{10} &= (1, 2, 4, 6, 7, 12), & \Pi_{11} &= (1, 2, 10, 8, 7, 12). \end{aligned} \quad (8.2.3)$$

These paths are due to numerical errors, the corresponding rays are rimmed in blue in Figure 8.4. Note that some rays inside different circles might correspond to the same path. The numerical error can be related to the precision of the bisection method and the numerical computation of the intersection points when backward ray tracing is used. We remark that in backward ray tracing, the intersection between the ray and the lens (line 2) is computed using the Newton-Raphson procedure. To detect only the boundaries of the regions formed by the rays that follow a *real* path Π_j , with $j \in \{1, \dots, 5\}$, we check the index of refraction that every ray has once it arrives at the source. If this is equal to the same index of S ($n = 1$ for the TIR-collimator), then the ray follows a physical path, otherwise it follows one of the paths in (8.2.3) and, therefore, it is not considered for the intensity calculation. This gives the ray distribution at the target PS shown in Figure 8.5. We observe that, discarding those rays, 5 different paths are found. These are the same paths we obtained using PS ray tracing. Figure 8.5 shows that the rays on the boundaries $\partial R(\Pi_j)$ are determined for every path Π_j with $j \in \{1, \dots, 5\}$.

The ability of the backward ray mapping method to recognize spurious paths makes it suitable to detect ghost stray light, which is unwanted light that reduce the performance of optical systems [90]. Optical designers are interested in developing methods for minimizing stray light intensity [91]. Backward ray mapping could be an

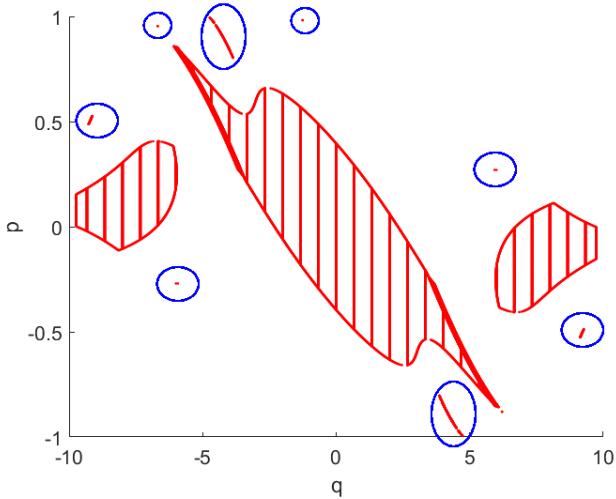


Figure 8.4: **Ray distribution at target PS of the TIR-collimator.** The q -axis is divided into $N_i = 30$ bins, the p -axis is divided into $N_b = 100$ bins. Approximately $6 \cdot 10^3$ rays are traced from the target to the source (red dots). Because of numerical errors, a few rays outside the region with positive luminance are found (rays rimmed in blue).

alternative approach for such purpose.

Note that some rays in the interior of the regions are still traced because we divided the target PS along the q -axis into $N_i = 30$ bins. As a consequence, also the rays located at the end points of every bin are traced.

The target PS intensity \hat{I}_{PS} is calculated using Equation (5.2.6). The profile of \hat{I}_{PS} with $N_i = 30$ is depicted in Figure 8.6 with the red line. It is compared with the reference intensity (blue line) which is given by QMC ray tracing with 10^7 rays. The picture shows that the backward ray mapping method calculates the intensity correctly.

Finally, we compare backward ray mapping with QMC ray tracing. The errors are obtained from (5.3.4) and are shown in a logarithmic scale in Figure 8.7 as a function of the CPU-time. The approximation of the PS intensity \hat{I}_{PS} is improved by increasing the number of bins N_i in the partitioning Q . The approximated QMC intensity \hat{I}_{QMC} is calculated several times gradually increasing the number of rays N_r . Both intensities are computed using the same number of bins, $N_b = 100$ in the partitioning P of the p -axis. The minimum ray mapping error is obtained with $N_i = 100$ bins, while the minimum QMC error is achieved tracing 10^6 rays. We observe that the minimal error obtained using the backward ray mapping is of an order of magnitude of 10^{-6} , while the minimum QMC error is of the order of 10^{-5} . Furthermore, an extrapolation of the QMC error shows that the backward ray mapping is more than 300 times faster compared to QMC! In Table 8.1 the numerical results obtained for backward ray mapping are reported. The error values for QMC ray tracing were already reported



Figure 8.5: **Ray distribution at target PS of the TIR-collimator.** The q -axis is divided into $N_i = 30$ bins, the p -axis is divided into $N_b = 100$ bins. Considering only the rays that arrive to the source with the correct index of refraction ($n = 1$), the regions with positive luminance are computed correctly.

Table 8.1: **Errors of the PS intensity for the TIR-collimator**

N_i	$ \Delta U $	PS error	CPU-time (sec.)
5	$1.6 \cdot 10^{-1}$	$5.19 \cdot 10^{-4}$	269
10	$2.8 \cdot 10^{-2}$	$2.09 \cdot 10^{-4}$	284
30	$5.7 \cdot 10^{-3}$	$3.15 \cdot 10^{-6}$	313
100	$5.1 \cdot 10^{-3}$	$2.52 \cdot 10^{-6}$	359

in Chapter 6 (Table 6.2).

In the next section we present the method for a parabolic reflector.

8.3 Results for the parabolic reflector

In this section we provide the results for the parabolic reflector in Figure 6.10. This is a very challenging example. Indeed, the rays that propagate through such a system can reflect many times along the left or the right mirror. As we have seen using PS ray tracing, this leads to many different paths. Every path corresponds to a given number of reflections with one of the two reflectors. In Chapter 6 we found 17 different paths for this parabolic reflector. Here, we apply the backward ray mapping method to detect all these paths.

The target PS of the parabolic reflector is the rectangular domain $T = [-b, b] \times [-1, 1]$ where $b = 17$. Like for the TIR-collimator, we divide the interval $[-b, b]$ in target PS into sub-intervals of the same length. Considering the partitioning



Figure 8.6: **Profile of the intensity for the TIR-collimator.** The PS intensity is calculated dividing the q -axis into $N_i = 30$ bins. The reference intensity is obtained from QMC ray tracing with 10^7 rays.

Q : $-b = q^0 < q^1 < \dots < q^{N_i} = b$ of $[-b, b]$ and a direction $p \in [-1, 1]$, the backward ray mapping explained in Section 8.1 is applied to every sub-interval $[q^k, q^{k+1}] \subset [-b, b]$ with $k = 0, \dots, N_i - 1$ and for every direction $p \in [-1, 1]$. To determine how many bins N_i are needed for a good approximation of the target photometric variables, we employ the same idea of the TIR-collimator. The source étendue U_1 is compared to several approximations of the étendue U_t at the target, each of them is given by a different partitioning Q of $[-b, b]$. For the parabolic reflector all the rays emitted from the source arrive at the target. Therefore, the exact étendue as an area in PS is computed from (5.2.1) obtaining $U = U_1 = 8$. The approximated target étendue U_t is given by Equation (5.2.3a). In Figure 8.8 we show the comparison between U_1 and several approximations of U_t by gradually increasing the number of bins N_i in the partitioning Q while fixing the maximum number of multiple reflections to 30. Increasing the number of bins N_i , U_t increases approaching the exact value $U_1 = 8$. After the division into $N_i = 4$ bins the improvement is slightly visible. Therefore, we conclude that $N_i = 4$ bins are enough to detect 30 multiple reflections.

In Figure 8.9 we show the ray distribution at the target PS obtained using backward ray mapping with $N_i = 4$ and at most 30 multiple reflections. The rays traced from the target to the source are depicted with the red dots. Most of the rays traced are located on the boundaries $\partial R(\Pi)$ of the regions with positive luminance. Only few rays are traced inside those regions. These are the rays located at the end points of every sub-interval $[q^k, q^{k+1}]$ with $k = 0, \dots, N_i - 1$.

The backward ray mapping method is able to detect 61 different paths. Indeed up

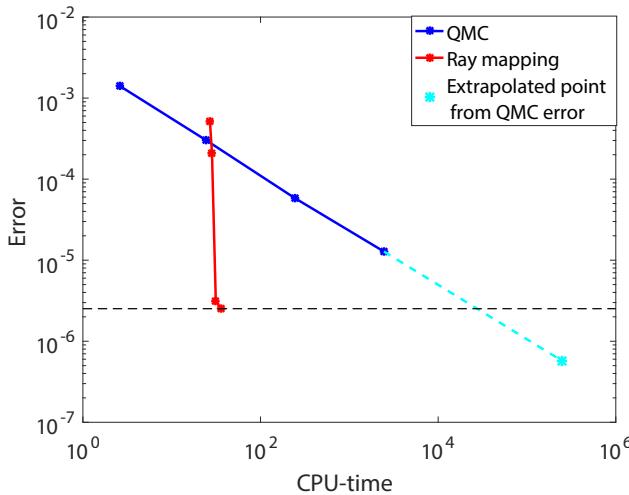


Figure 8.7: **Errors of ray mapping and QMC ray tracing for the TIR-collimator.** The direct backward ray mapping method is faster and more accurate than QMC ray tracing.

to 30 multiple reflections can occur at the left reflector and the right reflector. The path that goes directly from the source to the target (no reflections) has to be added. Using PS ray tracing we found at most 17 paths for the same parabolic reflector. Hence, we claim that ray mapping is much more accurate than PS ray tracing. Also, we observe the procedure can be stopped later to detect even more than 30 reflections. The more reflections are considered the better the accuracy obtained. Again, to define a stopping criterion we use étendue conservation. Fixing the number of bins $N_i = 4$ and $N_b = 100$ and gradually increasing the number of multiple reflections we see that the approximated target étendue U_t changes as the blue line in Figure 8.10. The horizontal red line represents the exact intensity $U = U_1 = 8$. We note that the more reflections are considered the smaller the value of $\Delta U = |U_1 - U_t|$ (see also Table 8.2). We observe that after around 30 multiple reflections there is no significant improvement in the computation of U_t . This is due to two main reasons. First, since only few rays follow multiple reflections, they do not give a significant contribution to the total étendue, the regions in PS formed by those rays are very small compared to the entire PS. Second, when more paths are considered also more bins N_b and sub-intervals N_i should be taken into account to obtain a more precise approximation of the étendue. From this we conclude that backward ray mapping has a good accuracy when around 30 multiple reflections and $N_i = 4$ bins are taken into account. Once a stopping criterion is established, direct backward ray mapping is run and the rays on the boundaries are determined. Finally, the target intensity is calculated from Equation (4.3.4).

In Figure 8.11 both the PS intensity (red line) and the reference intensity (dotted blue line) are shown. The PS intensity \hat{I}_{PS} is obtained considering at most 30 multiple



Figure 8.8: **Comparison between the exact étendue and the approximated target étendue by increasing Ni .** At most 30 multiple reflections are considered. Increasing Ni , the target étendue gets closer to the exact value $U_1 = 8$.

reflections, $Ni = 4$ and $Nb = 100$. The reference intensity \hat{I}_{ref} is given by QMC ray tracing with 10^8 rays and $Nb = 100$. The two intensities coincide.

To conclude, we calculate the errors between the approximated intensity \hat{I}_A ($A = \text{PS}, \text{QMC}$) and the reference intensity \hat{I}_{ref} . From the results in Figure 8.12 we observe that the PS error (red line) converges faster than the QMC error (blue line) as long as an error of an order of 10^{-6} is desired. Ray mapping results to be around 300 times faster than QMC ray tracing! Furthermore, it is much more accurate than QMC. Our method is able to detect *all* the possible paths that can occur. The procedure is stopped when 200 multiple reflections are reached. Our expectation is that, increasing the number of multiple reflections and the number of bins Ni , the accuracy can be improved even more. In Tables 8.2 and 8.3 these numerical results are reported.

8.4 Conclusions

In this chapter we extended the concatenated ray mapping method to systems formed by curved lines. Employing backward ray tracing and a bisection procedure in target PS, an inverse map from the target to the source was constructed such that all the possible paths that the rays can follow are determined. The numerical backward ray mapping method is able to detect the rays located on the boundaries of the regions formed by rays that follow the same path. From these rays the target intensity is calculated.

We presented two examples of optical systems: the TIR-collimator and the parabolic reflector. In both cases the target PS is divided into bins and the procedure is applied to each bin. A stopping criterion based on étendue conservation is developed

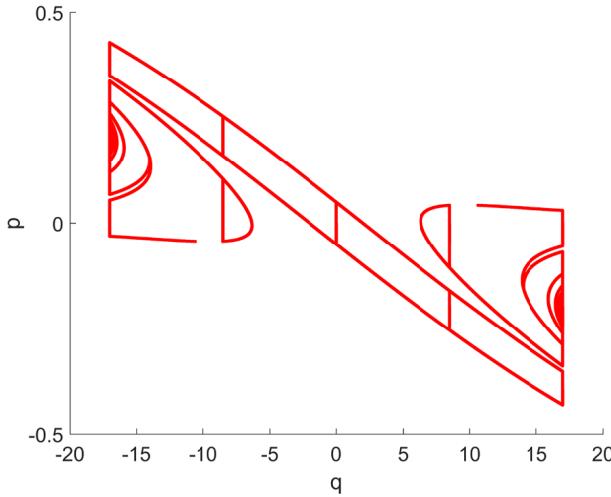


Figure 8.9: **Rays on the boundaries of the regions with positive luminance in target PS.** At most 30 multiple reflections are considered and $N_i = 4$ and $N_b = 100$. Only the rays on the boundaries and on the end points of each interval are traced.

to determine the number of bins needed to obtain a good accuracy. For the TIR-collimator we noticed that the method is able to detect rays that follow a spurious path due to numerical error. This gives the expectation that backward ray mapping can be used for detecting stray light. For the parabolic reflector, many paths can occur along the reflectors. Etendue conservation is used again to determine the number of multiple reflections to be considered. The target intensity is computed for both systems and is compared with a reference intensity given by QMC ray tracing with a large number of rays. The results show that the method is able to detect all the possible paths tracing a relatively small number of rays, typically around 10^3 . Comparing our method to QMC ray tracing, significant advantages are observed in accuracy and computational time for both optical systems.

In the next chapter we explain how to apply the method to systems with Fresnel reflection. We present the method for a system formed by the source, an ideal lens and the target.

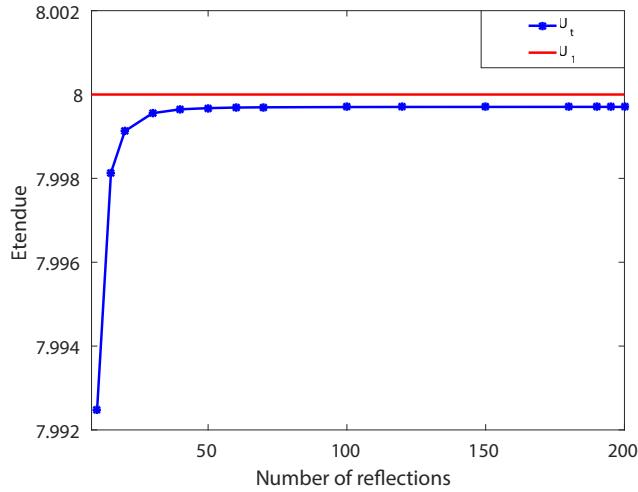


Figure 8.10: **Comparison between the exact étendue and the approximated target étendue by increasing the number of multiple reflections.** Fixing the number of bins along the q -axis, $N_i = 4$, and increasing the number of reflections considered, the étendue increases approaching to the exact value $U_1 = 8$.

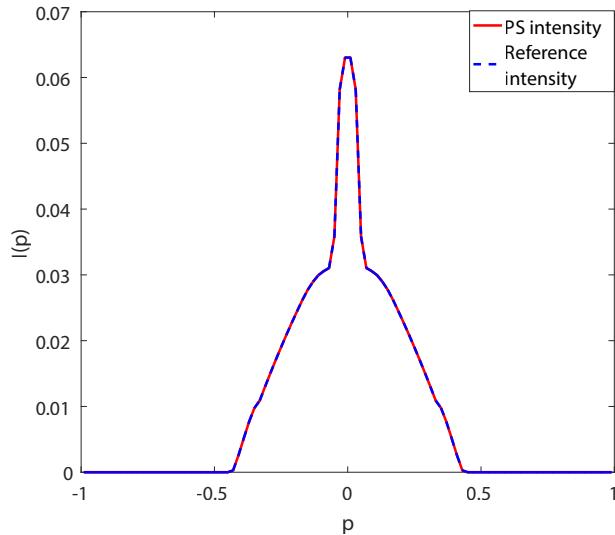


Figure 8.11: **Profile of the intensity for the parabolic reflector.** The ray mapping error is calculated considering $N_i = 4$ and at most 30 multiple reflections. The reference intensity is obtained by running QMC ray tracing with 10^8 rays.



Figure 8.12: **Errors of ray mapping and QMC ray tracing for the parabolic reflector.** The ray mapping error decreases by increasing the number of reflections considered. The QMC error reduces by tracing more rays. The extended backward ray mapping method is significantly faster and more accurate than QMC ray tracing.

Table 8.2: **Errors of the PS intensity for the parabolic reflector**

Number of reflections	$ \Delta U $	PS error	CPU-time (sec.)
5	$1.74 \cdot 10^{-1}$	$3.222 \cdot 10^{-4}$	3.28
10	$7.52 \cdot 10^{-3}$	$1.555 \cdot 10^{-5}$	5.11
20	$9.00 \cdot 10^{-4}$	$2.059 \cdot 10^{-6}$	6.83
30	$5.00 \cdot 10^{-4}$	$1.269 \cdot 10^{-6}$	8.38
100	$2.99 \cdot 10^{-4}$	$1.038 \cdot 10^{-6}$	17.49
150	$2.96 \cdot 10^{-4}$	$1.039 \cdot 10^{-6}$	26.38
200	$2.95 \cdot 10^{-4}$	$1.039 \cdot 10^{-6}$	32.21

Table 8.3: **Errors of the QMC intensity for the parabolic reflector**

Nr	QMC error	CPU-time (sec.)
10^4	$2.05 \cdot 10^{-4}$	2.81
10^5	$2.87 \cdot 10^{-5}$	25.81
10^6	$7.18 \cdot 10^{-6}$	257.54
10^7	$1.15 \cdot 10^{-6}$	2491.32

Chapter 9

Direct backward ray mapping for systems with Fresnel reflection

In this chapter we present the backward ray mapping method for systems consisting of optical lines at which Fresnel reflections can occur. In the following we consider unpolarized light and we do not take into account scattering phenomena. In particular, we explain how to calculate the boundaries of the regions of positive luminance in target PS. We provide numerical results for our system formed by the source, a lens and the target. The chapter starts with a general introduction of the method, then the details are explained for a simple system, finally we show the results obtained.

9.1 Introduction

Including Fresnel reflections, every time that a ray hits an optical line, it splits into a reflected and a refracted ray, both of them carry a part of the incident energy. In case a ray propagating from a more dense to a less dense medium hits the line with an angle greater than the critical angle, it can be only reflected and TIR occurs (see Section 2.3). A part of this case, a ray emitted from the source generates two rays after interaction with the optical line (the reflected ray and the transmitted one) each of them has a fraction of the original power given by the reflectance \mathcal{R} and the transmittance \mathcal{T} obtained from (2.3.21). At the next intersection of the two rays with a line, each incident ray is split again. This process is called *ray splitting* and results in tracing a rapidly increasing number of rays, including those containing very little power. Many different paths can occur and the number of paths is determined considering all the possible combinations that a ray emanating from the source can generate. The number of possibilities increases with the number of optical lines that form the system. For example, for a system formed by a source, one single line and a target, two possibilities occur since a ray emitted from the source can either refract reaching the target or reflect going back to the source. If the system is formed by a source, two lines and a target much more paths are possible since the ray can reflect

forward and backward between the two optical lines before arriving either at the target or at the source again. For systems where multiple reflections can occur the number of paths can increase dramatically. Ray splitting is a powerful and accurate ray tracing method to calculate *all* physical paths. The disadvantage is that it generates more and more rays, many of which are not significant for computing the target intensity. This makes methods based on ray splitting time consuming. A stopping criterion is needed. Ray generation can be controlled either neglecting rays with a small flux or limiting the total number of interactions that a ray can undergo during its propagation.

A different ray tracing approach could be to decide for each ray if its reflected or refracted part has to be considered every time that an optical line is encountered. Such methods are called *one-ray-in one-ray-out* because the number of rays emitted from the source equals the number of rays traced through the system, [6]. MC and QMC ray tracing are examples of these kind of processes. They select a single ray every time that it hits an optical line. At every intersection between the ray and the line the transmittance T and reflectance R are determined. Next, a randomly generated number between 0 and 1 establishes which path the rays will continue to follow. Usually the assumption is to consider the reflected ray when R is greater than the random number, and the transmitted one otherwise. Therefore, for each ray traced a unique path is possible and the number of rays emitted from the source equals the total number of rays traced. Compared to methods that consider all possible paths, MC and QMC are easy to implement. However, to achieve a very accurate luminance or intensity distribution at the target, many rays need to be traced.

To improve existing methods, we apply direct backward ray mapping to systems formed by Fresnel lines following the ray splitting approach. The rays are traced back from the target and, for every incident ray, we take into account both the reflected and the refracted ray. Therefore, for a given position and direction of a ray emitted by the source at least two paths are allowed. As explained above, the number of paths depends on the Fresnel lines encountered by the rays. As the backward ray mapping traces the rays backwards, not all of these paths are *physical* paths, i.e., paths from the source to the target. Indeed, some rays traced back from the target could be reflected such that they reach again the target never arriving at the source. Moreover, it can happen that for the same ray coordinates at the target, more than one physical path is permitted. A point in target PS can correspond to two or more points in source PS. Therefore, the regions in target PS formed by the rays that follow the same path overlap. To calculate the boundaries of these regions correctly, we run direct backward ray mapping for each physical paths one by one. Every time that direct backward ray tracing is applied, we consider either the reflected or the refracted ray depending on the path of which we want to determine the boundary. This allows determining the boundaries of the corresponding regions in target PS. The power of every ray at each intersection with a line is calculated. Hence, once the rays on the boundaries are determined, their corresponding luminance at the target is also calculated. Note that the output luminance cannot be constant along a given direction and it depends on both the position q and the direction p in target PS. To determine the luminance of the rays that follow a certain path, the luminance needs to be interpolated along a line $p = \text{const.}$. The procedure is repeated for all the paths or at least for those needed to obtain a good accuracy. The partial luminance for every path is computed. The total luminance is given by the sum of all partial luminances

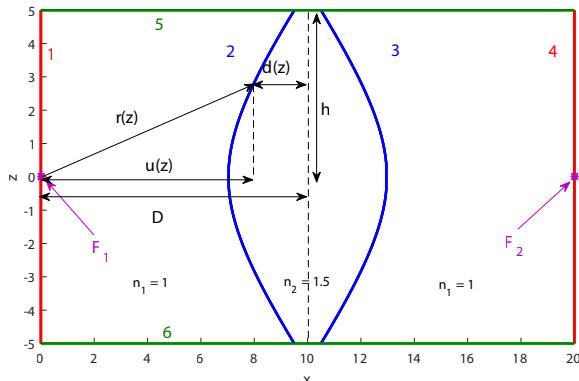


Figure 9.1: **An ideal lens.** The source (line 1) and the target (line 4) are located in air ($n = 1$), the material between lines 2 and 3 has index of refraction $n = 1.5$. Two detectors (lines 5 and 6) are located at the top and the bottom of the lens to detect light exiting from the system. $F_1 = (0, 0)$, $F_2 = (20, 0)$, $D = 10$, $h = 5$, $d(h) = 0.5$.

calculated. Finally, the intensity is given by an integration of the luminance along all the possible positions.

The method is presented in detail in the next section for a simple system.

9.2 Backward ray mapping for an ideal lens

In this section we present direct backward ray mapping for a simple optical system formed by the source, the target and a lens formed by two optical lines. First we describe the geometry of the system, then we explain the method for such system.

9.2.1 The geometry of the system

Let us consider the optical system depicted in Figure 9.1. It is formed by the source (line 1), a lens formed by two optical lines (line 2 and 3), the target (line 4) and two detectors (line 5 and 6) at the top and the bottom of the system. The source S is a vertical line segment located at $x = 0$ with $z \in [-h, h]$ where $h = 5$. The target T is a line segment of the same length, parallel to S and located at a distance $x = a$ from S with $a = 20$. The lens is formed by two refractive lines (simple lens), and it is convex. The lens is symmetric with respect to the axis $x = 10$. The optical axis is the line $z = 0$. The point $F_1 = (0, 0)$ and $F_2 = (a, 0)$ are the focal points of line 2 and 3, i.e., the points where a collimated beam of light will converge after the interaction with line 2 and 3, respectively [92].

The optical path length is defined as the product of the geometric ray path and the refractive index in which the ray travels, where the geometric ray path is the distance between two intersection points of the ray with two optical lines or the source. The theorem of Malus-Dupin guarantees that the optical path length is constant [93]. Employing Fermat principle, we derive the equations of the lens lines. In the following

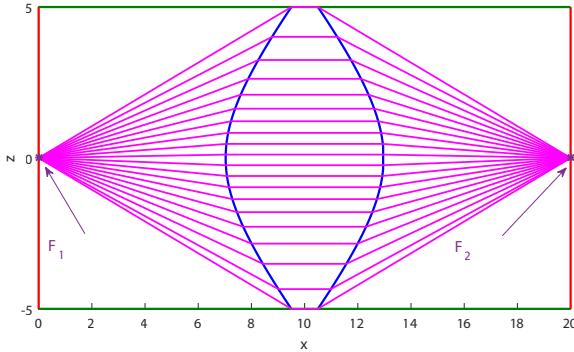


Figure 9.2: **An ideal lens.** All the rays exiting from F_1 arrive at F_2 .

we explain how the equation for the left lens (line 2) is obtained. We indicate with $d(z)$ half of the lens thickness at high z , with $\mathbf{r} = \mathbf{r}(z)$ the parametrization of the ray emitted from F_1 that reaches line 2 at height z , and $u(z)$ the projection of $\mathbf{r}(z)$ on the optical axis. The following relations hold:

$$\begin{aligned} |\mathbf{r}(z)|^2 &= u(z)^2 + z^2, \\ u(z) &= D - d(z), \end{aligned} \quad (9.2.1)$$

The optical path length of a ray emitted from F_1 and arriving at a certain point of the axis $x = 10$ is given by:

$$n_1 |\mathbf{r}(z)| + n_2 d(z) = OPL, \quad (9.2.2)$$

where $OPL > 0$ is a constant obtained from the ray passing through top of the lens

$$OPL = u(h) + n_2 d(h). \quad (9.2.3)$$

Substituting relations (9.2.1) in the previous equation, we obtain:

$$a u(z)^2 + b u(z) + c + z^2 = 0, \quad (9.2.4)$$

for every $z \in [-h, h]$, where

$$\begin{aligned} a &= 1 - n_2^2, \\ b &= 2n_2(n_2 D - OPL), \\ c &= -(OPL - n_2 D OPL)^2. \end{aligned} \quad (9.2.5)$$

Equation (9.2.4) leads to two real solutions, the negative one gives the expression for the left lens. The right lens is given by a reflection with respect to the axis $x = 0$ and a translation $(x, z) \rightarrow (x + 20, z)$.

The lens described above has the property that every ray that hits each curved line is bent towards the optical axis. The rays passing through it connect the points F_1 and F_2 (see Figure 9.2). Therefore, all the rays diverging from F_1 converge to F_2 . In the next section we explain how to apply direct backward ray mapping to this system.

9.2.2 Computation of the boundaries of the positive luminance regions

Our goal is to show, using direct backward ray mapping, that we can detect *all* the possible paths without tracing a huge number of rays. Furthermore, we are able to determine the rays that are located at the boundaries of regions corresponding to a certain physical path. First, let us analyze the backward ray tracing process for the system in Figure 9.1. Every ray is traced backward from the target T with an angle $\theta_1 \in [-\frac{\pi}{2}, \frac{\pi}{2}]$, i.e., with a direction coordinate $p_1 \in [-1, 1]$. Every time that a ray hits a lens, i.e., either line 2 or 3, the ray is split in two rays, the reflected ray and the refracted one. The reflectance \mathcal{R} and the transmittance \mathcal{T} are calculated. The reflected ray will continue to propagate within the system with the fraction of the power given by \mathcal{R} . Similarly, the transmitted ray will travel inside the system with the power given by \mathcal{T} . The trajectory of each ray is stopped either when it arrives at the source S or when it reaches again the target T . This leads to many different paths for a single ray with a given position and direction at the target.

Consider the target PS of the system in Figure 9.1. Since the source and the target are vertical lines (the x -coordinates are fixed while the z -coordinates vary), the PS coordinates at the target PS are (q, p) , where p is the ray direction coordinate defined as for the previous systems and q is the z -coordinate of the intersection point of the ray with the optical lines. To have an idea about the rays distribution at the target we represent the target coordinates on PS of 10^5 rays traced using MC ray tracing. MC ray tracing decides randomly which part of the ray has to be considered at every interaction with a line. This part propagates with the corresponding energy. Running MC ray tracing for the lens in Figure 9.1 with 10^5 rays and storing the corresponding paths, the following three paths are found:

$$\begin{aligned}\Pi_1 &= (1, 2, 3, 4), \\ \Pi_2 &= (1, 2, 3, 2, 3, 4), \\ \Pi_3 &= (1, 2, 3, 2, 3, 2, 3, 4).\end{aligned}\tag{9.2.6}$$

In Figure 9.3 we provide the target PS of the lens where we depicted rays that follow the same path with the same color. The rays in magenta are rays that follow path Π_1 , the cyan rays have two reflections before reaching the target, following path Π_2 , the black rays have four reflections inside the lens, these rays follow path Π_3 (see Equation (9.2.6)). Note that for increasing number of reflections the corresponding rays occur less frequently in PS. Furthermore, there are some white parts in target PS inside the colored regions. This can be due to either TIR or a different choice of rays (reflected or refracted) at the intersections with Fresnel lines. In the first case, the rays will be located outside the region $R(\Pi_1)$, otherwise they can reach the interior of $R(\Pi_1)$. Therefore, rays that originate from two close positions at the target with close direction coordinates can follow different paths. Thus, given two different paths Π_i and Π_j with $i \neq j$, the corresponding regions $R(\Pi_i)$ and $R(\Pi_j)$ can overlap in target PS:

$$\bigcap_{\Pi} R(\Pi) \neq \emptyset,\tag{9.2.7}$$

where the intersection is over all the possible paths. Because of this, we need to slightly modify direct backward ray mapping.

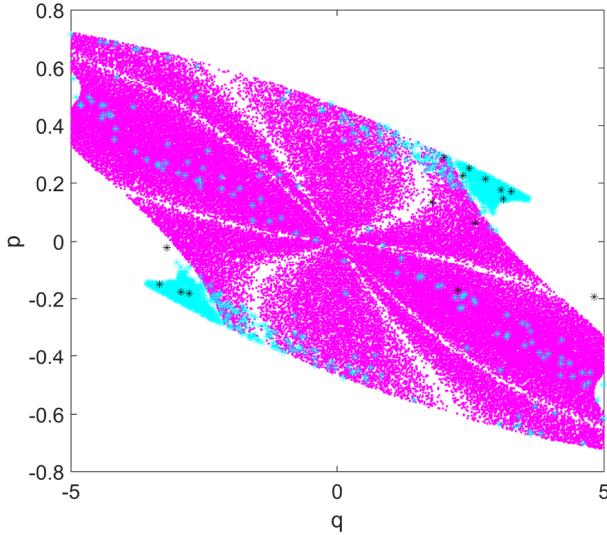


Figure 9.3: **Target PS of the ideal lens.** The magenta rays follow path Π_1 , the cyan rays follow path Π_2 and the black rays follow path Π_3 .

The key idea is to apply the bisection method of backward ray mapping for every single path Π separately. All the possible paths from the target to the source can be visualized in a tree, see Figure 9.4. R and T indicate if the reflected or the transmitted part of the ray is considered at every intersection point. Many possibilities occur, we indicate with C the sequence of the choices that are made at every intersection with the lens. For example, the path Π_1 in reverse order, that is $\Pi'_1 = (4, 3, 2, 1)$ corresponds to the choice $C_1 = (T, T)$ from the target to the source (red path in Figure 9.4). This means that the rays that follow path Π_1 are transmitted on both line 2 and 3. To find the intersection point of the boundary $\partial R(\Pi_1)$ with a line $p = \text{const}$, we start tracing back the rays with corresponding coordinates (q^a, p) and (q^b, p) located at the end points of the target PS, where $q^a = -h$ and $q^b = h$. Then the bisection method and backward ray tracing are applied as explained in the previous chapter. However, the difference with the previous method is that when the ray hits a line it is split in two rays and *either* the reflected *or* the refracted ray has to be selected according to the choice $C_1 = (T, T)$. The corresponding energy transported by the ray is calculated at every split. This allows us to determine the coordinates of the intersection points between a line $p = \text{const}$ and the boundary $\partial R(\Pi_1)$ which are denoted as $(q^i(\Pi_1, p), p)_{i=1,\dots,r}$ in increasing order where r is the number of intersections point found. In case $r = 2$ we indicate the coordinates of the two rays on the boundaries with $(q^{\min}(\Pi_1, p), p)$ and $(q^{\max}(\Pi_1, p), p)$.

Note that, including Fresnel reflection, the luminance at the target cannot be constant because its value depends on the Fresnel coefficients. Hence, along every direction $p \in [-1, 1]$, we sample points in $R(\Pi_1)$ by tracing back rays with direction coordinate p and position coordinate $q \in [q^{2i-1}(\Pi_1, p), q^{2i}(\Pi_1, p)]$ for every

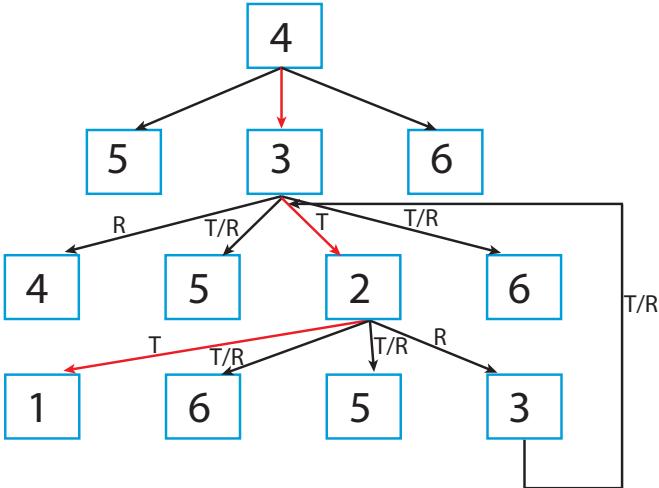


Figure 9.4: Tree with all the possible paths. Considering Fresnel reflections the rays can reflect many times between line 2 and 3 before reaching either the target, one of the two detectors or the source again. The inverse of path $\Pi_1 = (1, 2, 3, 4)$ that is $\Pi'_1 = (4, 3, 2, 1)$, is depicted in red. It corresponds to the choice $C_1 = (T, T)$.

$i = \{1, \dots, m\}$ where m is the integer part of $r/2$. At every interaction of each ray with a lens, either the reflected or the transmitted ray is traced further according to the choices sequence C_1 . The energy carried by each ray is calculated. Therefore the partial luminance $L_{\Pi_1}(q, p)$ at the target corresponding to path Π_1 is found. Repeating the procedure for all the possible directions $p \in [-1, 1]$, the profile of the partial luminance L_{Π_1} at the target given by all the rays that follow path Π_1 is computed for every $(q, p) \in T$.

Next, another path Π and the corresponding sequence of choices C are considered. We remark that every path is associated to a unique choice but the reverse is not always true. For example, the path $\Pi_2 = (1, 2, 3, 2, 3, 4)$ considered in reverse order from T to S , i.e., $\Pi'_2 = (4, 3, 2, 3, 2, 1)$ can only be associated to sequence $C_2 = (T, R, R, T)$. However, this sequence also corresponds with the two inverse paths $\Pi'_5 = (4, 3, 2, 3, 2, 5)$ and $\Pi'_6 = (4, 3, 2, 3, 2, 6)$. The last two paths Π'_5 and Π'_6 are not realistic paths from the target to the source, therefore they are discarded for the intensity calculation. Given a sequence C , many paths are possible but only one is the physical path. In the flowchart in Figure 9.5, we show all the steps needed to calculate the partial luminance $L_{\Pi_1}(q, p)$ along a given direction p and related to path $\Pi_1 = (1, 2, 3, 4)$ which corresponds to the sequence of choices $C = (T, T)$. The procedure to determine the other paths is similar.

Finally, the total luminance $L(q, p)$ is given by:

$$L(q, p) = \sum_{\Pi} L_{\Pi}(q, p), \quad (9.2.8)$$

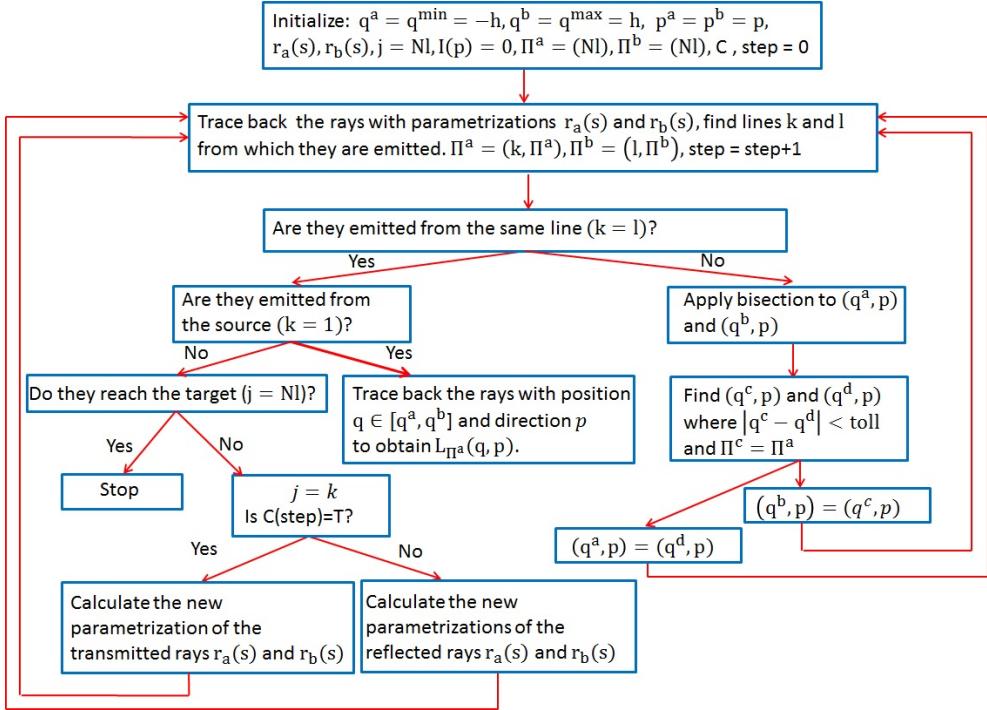


Figure 9.5: Main steps of backward ray mapping extended to systems with Fresnel reflection.

where the summation is over all the possible paths. From an integration of $L(q, p)$ the intensity at the target is given by:

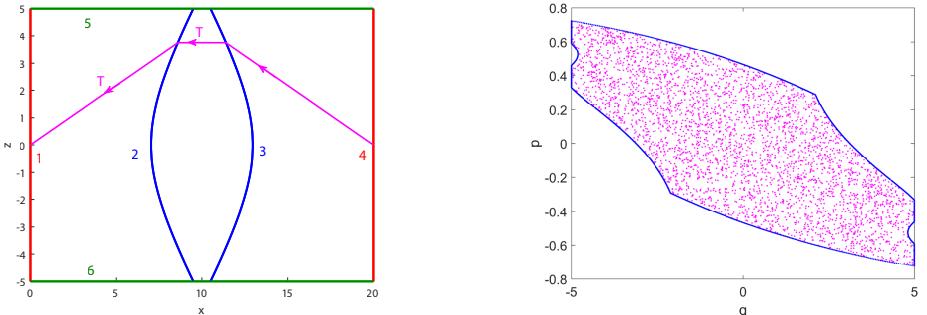
$$I(p) = \int_Q L(q, p) dq. \quad (9.2.9)$$

In the next section numerical results are shown.

9.3 Numerical results

In this section the numerical results for the lens are presented. We detect the boundaries of the regions formed by the rays that follow the same path. To this purpose we decide a priori the sequence of choices that has to be made at every intersection. This sequence determines whether the reflected or the transmitted part of the incident ray has to be considered when the rays traced back hit a line.

We start with the sequence of choices $C_1 = (T, T)$ which corresponds to the physical path $\Pi_1 = (1, 2, 3, 4)$. Indeed, every ray traced back hits the right curved line (line 3) and is split in two rays, at this point the reflectance \mathcal{R} and transmittance \mathcal{T} are calculated and, according to the first component of C_1 , the transmitted ray is



(a) **Ray traced back from the target to the source.** The ray follows inverse path $\Pi'_1 = (4, 3, 2, 1)$ corresponding to the choice $C_1 = (T, T)$ from the target to the source. The percentage of power of the ray at the source is approximately 55% of the initial power.

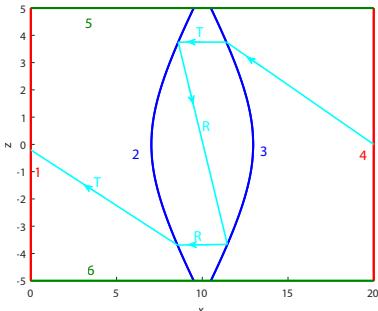
(b) **Boundary $\partial R(\Pi_1)$.** The boundary is depicted with the blue line, the red rays are traced using MC ray tracing and considering always the transmitted ray.

Figure 9.6: Computation of the boundary $\partial R(\Pi_1)$.

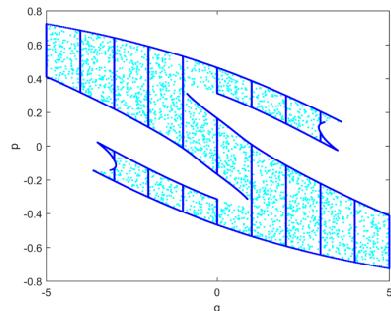
considered and traced back further with the corresponding power \mathcal{T} . This ray continues to propagate through the system until it hits either the detectors (line 5 and 6) or the curved line 2. In case it hits the detectors the procedure is stopped, otherwise the ray is split again into two more rays. Now, according to the second component of C_1 , the transmitted ray is considered and it continues its trajectory reaching either one of the detectors or the source (line 1). For example, in Figure 9.6a we draw in red a ray that follows the inverse path $\Pi'_1 = (4, 3, 2, 1)$. It is traced back from the target and has target PS coordinates $(q, p) = (0, -0.4)$. It arrives at the source with 55% of the initial power.

Using direct backward ray mapping and taking into account the choice C_1 , the boundary $\partial R(\Pi_1)$ is found which is the blue line in Figure 9.6b. To verify whether this boundary is correct we trace around 10^4 rays using MC ray tracing imposing that at every interaction of the ray with lines 2 and 3 always the transmitted part is considered. These rays are displayed in red in Figure 9.6b. The picture shows that all the rays traced are inside the boundary $\partial R(\Pi_1)$. So we conclude that the boundary is calculated correctly.

To detect the other paths we need to consider all possible sequences. We continue with $C_2 = (T, R, R, T)$. This means that if a ray traced back from the target hits line 3, the transmitted part is considered according to the first component of C_2 . Next, it is traced back further with its corresponding power given by \mathcal{T} . If the transmitted ray does not hit one of the detectors, it hits line 2 where it is split again. Now the reflected part of the ray is considered according to the second component of C_2 . The power of the reflected ray is given by the product $\mathcal{R}\mathcal{T}$. Note that, in the case of unpolarized light, the product $\mathcal{R}\mathcal{T}$ depends on the angle of incidence of the ray (see Equations (2.3.21) and (2.3.24)). The ray continues to propagate inside the system, if it hits line 3 it is reflected according to the third component of C_2 and, in case it hits line 2



(a) **Ray traced back from the target to the source.** The ray follows inverse path $\Pi'_2 = (4, 3, 2, 3, 2, 1)$ corresponding to the choice $C_2 = (T, R, R, T)$ from the target to the source. The percentage of power of the ray at the source is 7.2% of the initial energy.



(b) **Boundary $\partial R(\Pi_2)$.** The boundary is depicted with the blue line, the cyan rays are traced using MC ray tracing with 10^4 and considering two reflections between the lenses.

Figure 9.7: Computation of the boundary $\partial R(\Pi_2)$.

next, it is finally transmitted according to the last component of C_2 . If the ray finally reaches the source, it has followed the inverse path $\Pi'_2 = (4, 3, 2, 3, 2, 1)$. In Figure 9.7a we show in cyan a ray traced back from the target to the source that follows path Π_2 . The ray with target PS coordinates $(q, p) = (0, -0.37)$ is traced back and after two reflections inside the lens it arrives at the source with 7.2% of the initial energy. Direct backward ray mapping combined with sequence C_2 provides the boundary of the region $\partial R(\Pi_2)$ which is depicted in blue in Figure 9.7b. To properly detect the boundary we divided the target PS into $N_t = 10$ bins as explained in Section 8.2. Like for the boundary $\partial R(\Pi_1)$, to prove that the method computes the boundary correctly we traced 10^4 rays using MC ray tracing in combination with sequence C_2 . These rays are shown in cyan in Figure 9.7b.

Since more than two reflections can occur between line 2 and 3, the procedure continues considering the sequence $C_3 = (T, R, R, R, R, T)$ leading to four reflections between line 2 and 3. Every ray traced back from the target is first transmitted, then reflected between 2 and 3 four times and finally transmitted again. If the ray hits the source then it has followed the inverse path $\Pi'_3 = (4, 3, 2, 3, 2, 3, 2, 1)$ corresponding to sequence C_3 . An example of a ray that follows path Π_3 is depicted in black in Figure 9.8a. The ray is traced back from the target with target PS coordinates $(q, p) = (0, -0.39)$. It arrives at the source with a power equal to 0.0073% of the initial energy. The backward ray mapping with the sequence C_3 gives the boundary $\partial R(\Pi_3)$ depicted in blue in Figure 9.8b. This boundary is obtained dividing the target PS into $N_t = 10$ bins and applying the backward ray mapping to each bin. The black dots correspond to the rays in target PS obtained using MC ray tracing in combination with sequence C_3 with 10^4 rays.

Direct backward ray mapping is able to detect *all* the boundaries of *all* regions of positive luminance in target PS. Since rays with multiple reflections hardly contribute



(a) **Ray traced back from the target to the source.** The ray follows inverse path $\Pi_3 = (4, 3, 2, 1, 2, 3, 2, 1)$ corresponding to the sequence of choices $C_3 = (T, R, R, R, R, T)$ from the target to the source. The percentage of power of the ray at the source is 0.0073% of the initial energy.

(b) **Boundary $\partial R(\Pi_3)$.** The boundary is depicted with the blue line, the black rays are traced using MC ray tracing with 10^4 and considering four reflections between the lenses.

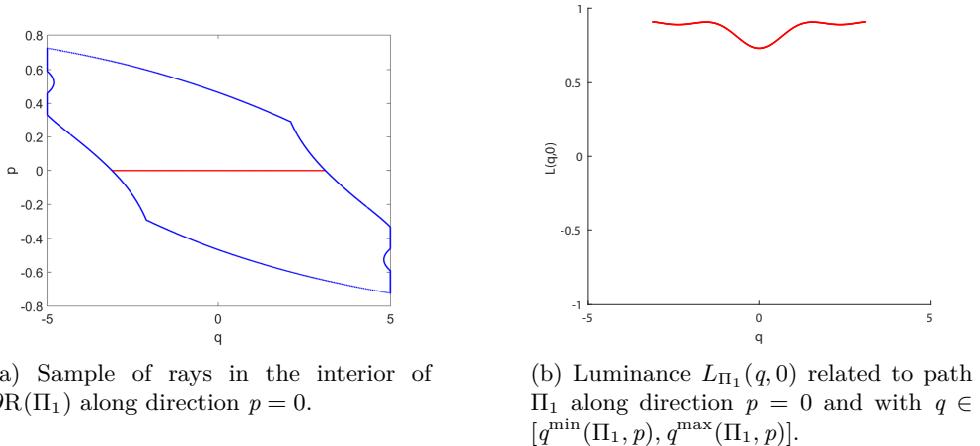
Figure 9.8: Computation of the boundary $\partial R(\Pi_3)$.

to the total power, we discard rays with more than four reflections. The procedure can be stopped according to the desired accuracy. The more reflections are considered, the better the accuracy.

Note that the luminance at the target cannot be constant because every ray carries a certain amount of energy that depends on the Fresnel coefficients. Because of this, a sampling in the interior of the boundaries is needed to compute the profile of the luminance at the target. In Figure 9.9a we show the procedure to compute the luminance related to path Π_1 along direction $p = 0$. A sample of rays with corresponding coordinates (q, p) where $q \in [q^{\min}(\Pi_1, p), q^{\max}(\Pi_1, p)]$ and $p = 0$ are traced from the target to the source using the backward ray tracing and taking into account sequence C_1 . The rays traced are depicted in red in Figure 9.9a. The luminance $L_{\Pi_1}(q, p)$ is calculated for every $q \in [q^{\min}(\Pi_1, p), q^{\max}(\Pi_1, p)]$ and $p = 0$. Its profile is depicted in Figure 9.9b.

Repeating the procedure explained above along all possible directions $p \in [-1, 1]$, the partial luminance $L_{\Pi_1}(q, p)$ is found for every $q \in [q^{\min}(\Pi_1, p), q^{\max}(\Pi_1, p)]$ and $p \in [-1, 1]$. The same procedure is applied to all the paths considered and the luminance corresponding to each path is calculated. The total luminance $L(q, p)$ is given by the sum in (9.2.8). Finally, the intensity is computed using (9.2.9).

We do not show here the numerical results of the total luminance and intensity because the work is still in progress. Our expectation is that direct backward ray mapping is suitable also for systems with Fresnel reflection. Indeed, the preliminary results show that the boundaries of the regions with positive luminance are calculated correctly. Once the boundaries are found also the luminance and the intensity can be computed applying an interpolation between the rays at the boundaries along every possible direction. Furthermore we expect that the method is much more accurate



(a) Sample of rays in the interior of $\partial R(\Pi_1)$ along direction $p = 0$.

(b) Luminance $L_{\Pi_1}(q, 0)$ related to path Π_1 along direction $p = 0$ and with $q \in [q^{\min}(\Pi_1, p), q^{\max}(\Pi_1, p)]$.

Figure 9.9: **Determination of the partial luminance.** $L_{\Pi_1}(q, 0)$ is related to path Π_1 along direction $p = 0$.

and faster than both MC and QMC ray tracing. This because we can analyze every single path independently.

9.4 Conclusion and outlook

In this chapter we extend direct backward ray mapping to optical systems where Fresnel reflections play a role. We observed that a ray emitted from the source follows many different paths as it is split into two rays every time that it hits a line. Thus, a unique point at the source PS will correspond to several points in target PS. This results in an overlap of the regions with positive luminance in target PS. The purpose of the method is to detect the boundaries of *all* these regions. This can be done fixing a priori which part of the ray has to be considered at every intersection of the ray with a Fresnel line.

Direct backward ray mapping is extended in this chapter such that only one boundary is computed at each run. The procedure is run as many times as the number of sequences of choices we want to consider. We presented preliminary numerical results for a system formed by the source, the target and a lens formed by two curved lines. We have shown that the method is able to determine the boundaries of all the regions with positive luminance correctly. We noticed that, including Fresnel reflections, the luminance cannot be constant. An interpolation between the rays on the boundaries is needed to obtain the luminance profile. We provided an example of such interpolation for a given path and along a certain direction. Furthermore, we explained the theory for the luminance and the intensity computation.

Future work might regard the calculation of the luminance and the intensity and a comparison with both MC and QMC ray tracing. Numerical results on the intensity computation should be provided.

Chapter 10

Conclusions and Recommendations

10.1 Summary

In this thesis we investigated new methods for computing the target photometric variables based on *phase space ray tracing*. The aim was to understand how light propagates through non-imaging optical systems in order to calculate the target photometric variables, e.g., luminance and intensity. The core of this work was to use the *phase space* (PS) which provides a full description of geometric optics. In this thesis we restricted ourselves to two-dimensional optical systems, the PS of which is a two-dimensional space. For every ray traced inside the system its path can be considered where a path is the sequence of the optical lines that it encounters. The PS representation of the optical system shows that all the rays that follow the same path are located inside the same region in PS which is therefore divided into regions. Those parts of the target PS that are illuminated by the source are the positive luminance regions, on the others parts the luminance is equal to zero. Our idea was to determine the boundaries of the positive luminance regions to reduce the number of rays needed for obtaining the photometric variables. In particular, assuming a Lambertian source, the coordinates of the rays located on the boundaries give all the information needed to compute the output luminance and therefore the intensity. To this purpose, we developed two methods: PS ray tracing and backward ray mapping in PS. The goal of both is to trace only the rays close to the boundaries reducing the total number of rays traced compared to existing methods, for example Monte Carlo (MC) and Quasi-Monte Carlo (QMC) ray tracing.

Phase space ray tracing exploits the PS of the source and the target of the optical system. We introduced an accurate procedure to construct a triangulation on the source PS which allows tracing most of the rays close to the boundaries of the positive luminance regions. The boundaries of those regions were approximated using two different approaches: the α -shapes method and a technique based on triangulation refinement.

The α -shapes method relies on a parameter α which establishes which triangles have to be kept in the PS triangulation and which have to be removed to approximate the boundaries correctly. We developed a procedure based on étendue conservation to determine the value of α that gives a good approximation of the boundaries. Numerical results were provided for two different kinds of TIR-collimators showing that PS ray tracing using α -shapes is much faster and more accurate than MC ray tracing. However, we observed that the speed of convergence depends on the smoothness of the shape of the regions in target PS and, therefore, on the optical system.

To eliminate the parameter α from the calculation of the boundaries, we developed a new approach for the boundaries computation based on triangulation refinement. This technique is able to determine the boundary triangles (triangles crossed by at least one boundary) of a given triangulation. Connecting the vertices of the boundary triangles corresponding to rays that follow the same path, a good approximation of all the boundaries is obtained. Tracing more rays leads to smaller triangles resulting in a better accuracy of the boundary computation. The stopping criterion employs étendue conservation. The method was applied to several optical systems with reflective and refractive optical lines. The results show that the boundaries of all the regions with positive luminance in target PS are calculated correctly even for complicated systems such as the parabolic reflector for which multiple reflections of the rays with the mirrors can occur.

Assuming a Lambertian source, the intensity was computed from the coordinates of the rays on the boundaries. The intensity profile obtained using PS ray tracing based on the triangulation refinement is compared to the two intensities found with MC and QMC ray tracing. Phase space ray tracing allows tracing far less rays compared to MC ray tracing resulting in a significant reduction of the computational time. Our method has an order of convergence proportional to the reciprocal of the number of rays traced versus an error convergence proportional to the inverse of the square root of the number of rays traced for MC ray tracing. The results showed that PS ray tracing and QMC ray tracing are comparable in terms of the computational time, indeed the corresponding convergence errors are proportional to the inverse of the number of rays traced. For the TIR-collimator, for example, PS ray tracing outperforms also QMC ray tracing while it is slower than QMC for some other systems as, for example, the parabolic reflector. However, we demonstrated in simulations that PS ray tracing is binning free while MC and QMC errors depends on the number of bins in which the target is divided. In order to further improve PS ray tracing we developed a second method which allows tracing only the rays located exactly on the boundaries of the regions with positive luminance.

The key idea of backward ray mapping is to construct an inverse map from the target to the source connecting the coordinates of the rays on the PS of each optical line encountered.

We presented concatenated backward ray mapping applicable to systems formed by straight line segments. It considers *all* the lines that form the system. We showed that the boundaries of the regions in every PS can be calculated *analytically*. Therefore, assuming a Lambertian source, concatenated backward ray mapping calculates the intensity *exactly*. Compared to QMC ray tracing the method is much more accurate and also faster (the exact intensity was found in less time than QMC).

Next, we introduced direct backward ray mapping which is a modification of concatenated backward ray mapping to systems formed by curved lines. In this case the boundaries of the positive luminance regions in all the phase spaces cannot be calculated analytically, therefore a bisection procedure combined with backward ray tracing is developed. As a consequence the *exact* target intensity cannot be obtained for systems formed by curved lines. Nevertheless, the method remains very accurate and numerical results showed that it is also able to detect *unphysical* paths due to numerical error, where we referred to physical paths as those from the source to the target. Direct backward ray mapping provides a more accurate intensity distribution in less time compared to QMC ray tracing. To achieve an error of around 10^{-6} , direct backward ray mapping is approximately $3 \cdot 10^2$ times faster than QMC ray tracing for the TIR-collimator and 10^2 times faster for the parabolic reflector. In Figure 10.1 we show an error comparison between all the methods applied to the TIR-collimator (left figure) and the parabolic reflector (right figure). We remark that while MC and QMC ray tracing are binning procedures which compute the averaged intensity, PS ray tracing and direct backward ray mapping compute the intensity pointwise. To compare the methods we calculated the averaged intensities also for PS ray tracing and direct backward ray mapping. Therefore, also for the PS procedures the target was divided into bins and the mean value of the intensity over every bin was computed. This mean value was given by the integral of the intensity over every bin divided by the size of the bin. In the simulations showed in this thesis, the integral was approximated by the trapezoidal rule discretizing the bin into 10 sub-intervals of equal length. The CPU-times of the PS methods shown in Figure 10.1, is divided by 10 to obtain the real CPU-time when the PS methods are not compared with binning procedures.

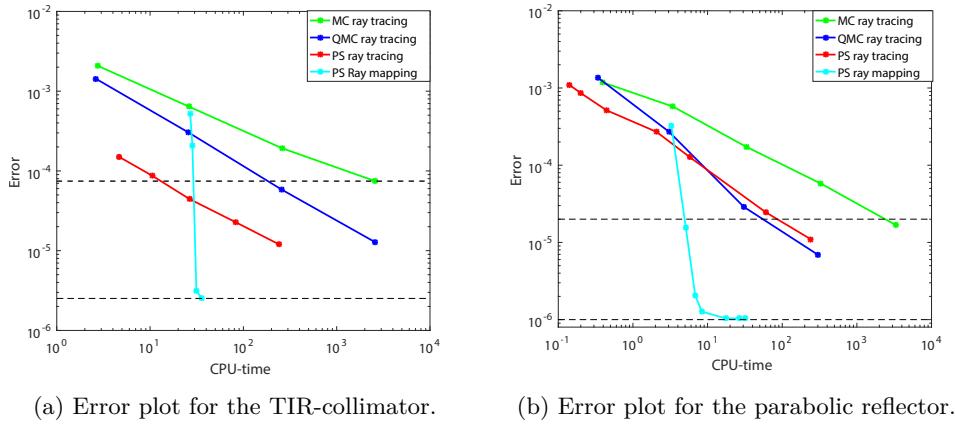


Figure 10.1: Comparison between MC, QMC, PS ray tracing and direct backward ray mapping.

Finally, we investigated systems where also Fresnel reflections are involved. Fresnel reflection leads to multiple paths due to the fact that, at every interaction with a line, each ray is split in two rays (the reflected and the transmitted) each of them carries a fraction of the energy transported by the incident ray. Direct backward ray

mapping is able to detect *all* the possible paths that can occur. Moreover, we showed that only the rays located on the boundaries of the positive luminance regions in target PS related to the *physical* paths are traced from the target to the source. To validate our method we traced forward a set of rays using MC ray tracing and we showed that the boundaries found with direct backward ray mapping enclose all the rays traced. The power associated to each ray on the boundary is calculated. For Fresnel systems the output luminance is not constant as it depends on the angles of incidence of every line and on the path followed. Therefore a sample of rays inside the positive luminance regions needs to be traced back to compute the luminance and the intensity.

To conclude we claim that PS methods might constitute alternative approaches to conventional ray tracing. The advantages are that far less rays are needed for computing the target photometric variables resulting in a reduction of the computational time. In particular, PS ray tracing is easy to implement and faster than MC ray tracing. For some systems it outperforms also QMC ray tracing while for some others the boundaries of the positive luminance regions could be difficult to approximate and more rays are required. Because of this, in some cases PS ray tracing can be slightly slower than QMC ray tracing. Direct backward ray mapping can be seen as an improvement of PS ray tracing as it is much more accurate. It allows tracing far less rays compared to MC, QMC and PS ray tracing, directly determining the rays on the boundaries of the positive luminance regions. Direct backward ray mapping is a very elegant method and, compared to MC and QMC ray tracing it is faster and much more accurate. This method could be used also to detect and minimize ghost stray light.

10.2 Recommendations

This work is far from finished. In the future, it might be useful to investigate in more details the two-dimensional case. The two-dimensional case is particularly relevant because it is a good test for new methods. Moreover it gives a complete analysis of three-dimensional rotationally symmetric systems as it fully describes of the meridional plane.

Regarding PS ray tracing, it could be interesting to analyze systems with a non-Lambertian source. Our proposal is to calculate the boundaries as we have done for a Lambertian source. Next, the profile of the luminance can be obtained by tracing a sample of rays with corresponding coordinates in the interior of the boundaries found. The intensity can be obtained by merely integrating the luminance over all the possible positions.

Regarding direct backward ray mapping, we are interested in providing simulations for systems with Fresnel reflection. The results shown in Chapter 9 give the expectation that the direct backward ray mapping method is suitable also for such systems and that it is much more precise and faster than both MC and QMC ray tracing. Scattering phenomena could be described by generalizing direct backward ray mapping for Fresnel systems. More paths would occur as at every intersection each ray can be split in more than two rays as it scatters in multiple directions. The range of possible directions can be discretized and a path can be associated to each

direction. We expect that the same algorithm used for Fresnel's reflections can be applied to every single path in case of scattering.

Future research should address the three-dimensional case. The first step could be to consider rotationally symmetric optical systems, i.e., systems invariant under rotations around the optical axis. Such systems are often used in illumination optics as they are easy to manufacture. They can be described by only considering the meridional rays, that is rays that propagate inside the plane containing the optical axis. This reduces the three-dimensional case to the two-dimensional one. For rotationally symmetric systems PS ray tracing might constitute design tool for optical designers, greatly reducing the time to design the optical systems.

Next, it can be useful to analyze asymmetric optical systems [94]. Every ray is described by three position and two direction coordinates. The corresponding PS is therefore a four dimensional space described by two of the position coordinates q_1 and q_2 of the intersection point between the ray and the optical surface and two direction coordinates p_1 and p_2 , expressed with respect to the normal of the surface. The target luminance in PS is a function of all these coordinates, while the intensity only depends on the direction coordinates and is given by a two-dimensional integral of the luminance over all the position coordinates q_1 and q_2 . Therefore, for fixed directions p_1 and p_2 , we need to compute the boundaries of the positive luminance regions in the (q_1, q_2) -plane.

To clarify our idea, we report a picture of the structure of the target of a three-dimensional system showed in [50] by Winston, Miñano and Benítez. In particular, they show the target of a CPC seen from above (constant direction). For example, Figure 10.2 shows the regions in the (q_1, q_2) -plane at the target of rays that leave the source with given angular coordinates. The regions labeled 0, 1, 2,... correspond to the regions that arrive at the target after 0, 1, 2, ... reflections; $F2, F3, \dots$ indicate the regions of the rays that begin to turn back after two, three, ... reflections. The blank regions are formed by the rays that still be traveling toward the exit aperture after five reflections.

We expect that both methods presented in this thesis can be extended to the three-dimensional case. PS ray tracing will be based on a triangulation reconstruction on the (q_1, q_2) -plane. The same procedure applied for the two-dimensional case in the (q_1, p_1) -plane can be applied to the (q_1, q_2) -plane for the three-dimensional case. This will allow to detect the boundaries of the positive luminance regions along the fixed directions p_1 and p_2 . Repeating the triangulation refinement procedure for all the possible directions, all the boundaries can be determined. Also, for the direct backward ray mapping in three-dimensions, we can to applied the same bisection procedure combined with the backward ray mapping used for the two-dimensional case. The only difference is that in 2D we detected the boundaries of the positive luminance regions in the (q_1, p_1) -plane, while in 3D we need to compute those boundaries in the (q_1, q_2) -plane and repeat the procedure for every p_1 and p_2 .

Although, the results showed for the two-dimensional case are very promising, we cannot predict the speed of convergence for the three-dimensional asymmetric optical systems. We expect that this would depend on the complexity of optical devices and of the corresponding regions in target PS. More research should be oriented on this topic.

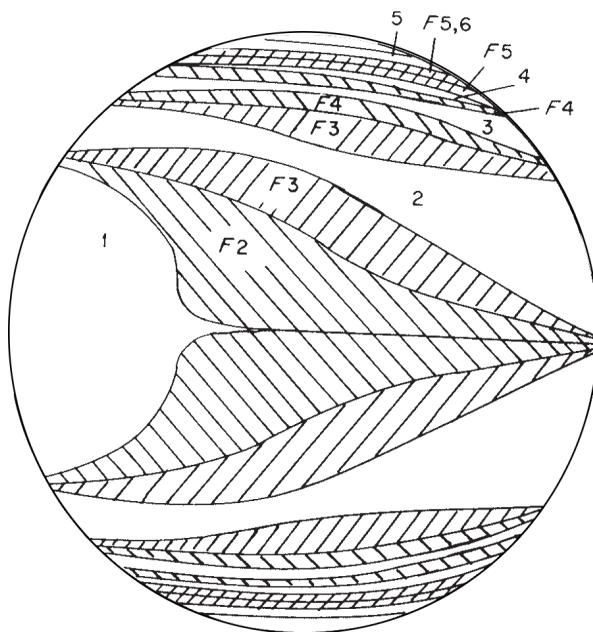


Figure 10.2: Regions at the target of a CPC of rays that leave the source with a fixed direction found using ray tracing. [50].

Appendix A

Monte Carlo integration versus Monte Carlo ray tracing

Although Monte Carlo integration (MC) and MC ray tracing are two different methods, they are based on the same idea. We think to fully comprehend MC ray tracing technique it is necessary to first understand the basics of MC methods. The aim of this appendix is to point out the similarities and the differences between the two methods. In the next section we give an introduction to MC integration for the two-dimensional case.

A.1 Monte Carlo integration

Let us consider a set $D = [\mathbf{a}, \mathbf{b}]$ with $\mathbf{a} = (a_1, a_2)$ and $\mathbf{b} = (b_1, b_2)$ elements of \mathbb{R}^2 such that $[\mathbf{a}, \mathbf{b}] = [a_1, b_1] \times [a_2, b_2]$. Consider a function $f : [\mathbf{a}, \mathbf{b}] \subset \mathbb{R}^2 \rightarrow \mathbb{R}$ and a random variable \mathbf{Y} with values in D and probability density function $\rho(\mathbf{y})$, where \mathbf{y} are the values of \mathbf{Y} in D . Note that we indicate the random variables with capital letters and the corresponding deterministic values with lowercase letters. The expected value of $f(\mathbf{Y})$ with respect of $\rho(\mathbf{Y})$ is:

$$\mathbb{E}[f(\mathbf{Y})] = \int_D f(\mathbf{y})\rho(\mathbf{y})d\mathbf{y}. \quad (\text{A.1.1})$$

If ρ is a uniform probability density function, the expected value is given by:

$$\mathbb{E}[f(\mathbf{Y})] = \frac{1}{\lambda([\mathbf{a}, \mathbf{b}])} \int_D f(\mathbf{y})d\mathbf{y}, \quad (\text{A.1.2})$$

where $\lambda([\mathbf{a}, \mathbf{b}]) = (b_1 - a_1) \cdot (b_2 - a_2)$. Let $\{\mathbf{Y}_i\}_{i=1,\dots,N}$ be independent samples of the probability density function ρ of \mathbf{Y} with values in D . Indicating with $S_N[f(\mathbf{Y})]$ the sum:

$$S_N[f(\mathbf{Y})] = \frac{1}{N} \sum_{i=1}^N f(\mathbf{Y}_i), \quad (\text{A.1.3})$$

and with $\Pr(E)$ the probability that the event E happens, the strong law of large numbers states that

$$\Pr\left(\lim_{N \rightarrow \infty} S_N[f(\mathbf{Y})] = \mathbb{E}[f(\mathbf{Y})]\right) = 1 \quad (\text{A.1.4})$$

[?]. The empirical mean is an unbiased estimator of the expectation value:

$$\mathbb{E}[f(\mathbf{Y})] \approx \frac{1}{N} \sum_{i=1}^N f(\mathbf{Y}_i). \quad (\text{A.1.5})$$

From the linearity of the expected value (see [?]) it follows

$$\mathbb{E}[S_N[f(\mathbf{Y})]] = \frac{1}{N} \sum_{i=1}^N \mathbb{E}[f(\mathbf{Y}_i)] = \mathbb{E}[f(\mathbf{Y})], \quad (\text{A.1.6})$$

where the second equality holds as $f(\mathbf{Y}_i)_{i=1,\dots,N}$ are still random variables with the same probability density function $\rho(\mathbf{y})$ because $(\mathbf{Y}_i)_{i=1,\dots,N}$ are independent samples with probability density function $\rho(\mathbf{y})$. This implies that $\mathbb{E}[f(\mathbf{Y}_i)] = \mathbb{E}[f(\mathbf{Y})]$ for every $i = 1, \dots, N$. Indicating with $\text{Var}[f(\mathbf{Y})]$ the variance of $f(\mathbf{Y})$:

$$\text{Var}[f(\mathbf{Y})] = \mathbb{E}[(f(\mathbf{Y}) - \mathbb{E}[f(\mathbf{Y})])^2] = \sigma^2[f(\mathbf{Y})], \quad (\text{A.1.7})$$

the Bienaym  formula ([?]) and the fact that $\text{Var}[f(\mathbf{Y}_i)] = \text{Var}[f(\mathbf{Y})]$ lead to

$$\text{Var}[S_N[f(\mathbf{Y})]] = \text{Var}\left[\frac{1}{N} \sum_{i=1}^N f(\mathbf{Y}_i)\right] = \frac{1}{N^2} \sum_{i=1}^N \text{Var}[f(\mathbf{Y}_i)] = \frac{1}{N} \text{Var}[f(\mathbf{Y})], \quad (\text{A.1.8})$$

which can be applied because the random variables $\{\mathbf{Y}_i\}_{i=1,\dots,N}$ are independent [?]. Equation (A.1.6) gives

$$\begin{aligned} \text{Var}[S_N[f(\mathbf{Y})]] &= \mathbb{E}[(S_N[f(\mathbf{Y})] - \mathbb{E}[S_N[f(\mathbf{Y})]])^2] \\ &= \mathbb{E}[(S_N[f(\mathbf{Y})] - \mathbb{E}[f(\mathbf{Y})])^2]. \end{aligned} \quad (\text{A.1.9})$$

Combining (A.1.8) with (A.1.9) we obtain:

$$\mathbb{E}[(S_N[f(\mathbf{Y})] - \mathbb{E}[f(\mathbf{Y})])^2] = \sigma^2[f(\mathbf{Y})]/N. \quad (\text{A.1.10})$$

Let us denote the integration error with:

$$\text{err}(f, S_N[f(\mathbf{Y})]) = \int_{\mathcal{D}} f(\mathbf{y}) \rho(\mathbf{y}) d\mathbf{y} - S_N[f] = \mathbb{E}[f(\mathbf{Y})] - S_N[f(\mathbf{Y})]. \quad (\text{A.1.11})$$

Considering a convex function $g : x \mapsto x^2$ we can write:

$$\begin{aligned} \mathbb{E}[|\text{err}(f, S_N)|] &= \sqrt{g(\mathbb{E}[|\text{err}(f, S_N)|])} \\ &\leq \sqrt{\mathbb{E}[g(|\text{err}(f, S_N)|)]} = \sqrt{\mathbb{E}[\text{err}^2(f, S_N)]}, \end{aligned} \quad (\text{A.1.12})$$

where the inequality follows from the Jensen's inequality ([?] Chapter 6). Using the previous relation and Equations (A.1.10) and (A.1.11), we obtain:

$$\mathbb{E}[|\text{err}(f, S_N)|] \leq \sqrt{\mathbb{E}[\text{err}^2(f, S_N)]} = \frac{\sigma[f]}{\sqrt{N}}. \quad (\text{A.1.13})$$

Hence, the absolute value of the integration error is, on average, bounded by $\sigma[f(\mathbf{Y})]/\sqrt{N}$, where $\sigma[f(\mathbf{Y})]$ is the standard deviation of f [52]. It is important to note that $\text{err}(f, S_N)$ does not depend on the dimension of the set D.

A.2 Monte Carlo ray tracing

The MC technique can be combined with ray tracing in order to compute the intensity distribution at the target of an optical system. In MC ray tracing the position and the direction of every ray are chosen randomly at the source.

In the two-dimensional case, for every ray we need to choose one position coordinate x_1 and one angular coordinate t_1 . These coordinates are chosen randomly and such that they are uniformly distributed at the source. Using ray tracing, the rays with those random initial coordinates are traced from S to T and, the corresponding random variables $\mathbf{Z}_i = (x^i, t^i)$ at the target are obtained, where x^i and t^i are the position and direction coordinates of the i -th ray at the target. Note that the rays at the target are most likely non-uniformly distributed. Once the target variables are computed, we can apply the idea of MC simulation to approximate the target intensity $I(t)$. Since the intensity only depends on the angular coordinate t we use MC approximation in one dimension. Hence, we approximate the expected value $\mathbb{E}[I]$ by a sum as described in Equation (A.1.5).

To clarify the connection between the MC integration and MC ray tracing, in Table A.1 we report a correspondence of the functions and the variables used in the previous section and in Chapter 3.

Table A.1: Comparison between the MC integration and MC ray tracing

MC method	MC ray tracing
$D = [\mathbf{a}, \mathbf{b}] \subset \mathbb{R}^2$	$[t_{j-1}, t_j) \subset \mathbb{R}$
$f : D \rightarrow \mathbb{R}^2$	$\chi_j : [t_{j-1}, t_j) \rightarrow \{0, 1\}$
\mathbf{Y}	t
N	Nr
$\{\mathbf{Y}_j\}_{j=1, \dots, N}$	$\{t^k\}_{k=1, \dots, \text{Nr}}$
$S_N[f(\mathbf{Y})] = \sum_{i=1}^N \frac{f(\mathbf{Y}_i)}{N}$	$\hat{I}_{\text{MC}} = \frac{\text{Nr}[t_{j-1}, t_j]}{\text{Nr}[-\pi/2, \pi/2]} = \sum_{k=1}^{\text{Nr}} \frac{\chi_j(t^k)}{\text{Nr}}$
$\mathbb{E}[S_N[f(\mathbf{Y})]] = \mathbb{E}[f(\mathbf{Y})]$	$\mathbb{E}[\hat{I}_{\text{MC}}] = P_{j, \Delta t}$
$\text{Var}[S_N[f(\mathbf{Y})]] = \frac{\text{Var}[f(\mathbf{Y})]}{N} = \frac{\sigma^2}{N}$	$\text{Var}[\hat{I}_{\text{MC}}] = \frac{P_{j, \Delta t}(1 - P_{j, \Delta t})}{\text{Nr}} = \sigma_j^2$

The main difference between the two methods is that in MC integration we know the probability density distribution ρ of the random variables \mathbf{Y}_i which is the uniform

distribution. Because of this, the expected value of the function f can be approximated by (A.1.5). On the contrary, in MC ray tracing we consider random variables equally distributed with a uniform distribution ρ at the source but, applying ray tracing to these variables, we do not know the target probability distribution anymore. Therefore the expected value of the *target* MC intensity \hat{I}_{MC} cannot be approximated by the sum in (A.1.5). Thus, the target is divided into intervals (for the one dimensional case) of the same length, i.e., bins, and the intensity is approximated by the sum of the characteristic function χ_j of the j -th interval as explained in Chapter 3 (see Equation 3.2.8). Because of this, the error convergence of MC integration only depends on the number of sample points (A.1.13), while it is also related to the number of bins for MC ray tracing (3.2.12).

Appendix B

Two examples of low discrepancy sequences

Low discrepancy sequences are widely used in QMC methods. Here, we provide some details on two types of those sequences: The Van der Corput sequence and the Sobol sequence. In the following we give a brief outline of the theory reported in [?, 52].

B.1 Van der Corput sequences

Van der Corput sequences take the name by its inventor Van der Corput who introduced it for the first time in 1935. In the following we show how to construct such sequence in one dimension, $d = 1$. This kind of sequences are particularly relevant because many other kind of sequences in higher dimensions are based on this one-dimensional case. Below we give the definition of radical inverse function.

Let $b \geq 2$ be an integer number. Any natural number $n \in \mathbb{N}_0$ has the following unique decomposition in base b :

$$n = \sum_{i=0}^{\infty} d_i b^i, \quad (\text{B.1.1})$$

where $d_i \in \{0, 1, \dots, b-1\}$ are called the digit numbers.

Definition B.1.1. *The radical inverse function $\phi_b : \mathbb{N}_0 \rightarrow [0, 1)$ in base b for a given number $n \in \mathbb{N}_0$ is defined as:*

$$\phi_b(n) = \sum_{i=1}^{\infty} \frac{d_{i-1}}{b^i}. \quad (\text{B.1.2})$$

As an example we calculate the radical inverse function $\phi_b(5)$ in base $b = 2$. The digit expansion in base b of $n = 5$ is:

$$5 = 1 \cdot 2^0 + 0 \cdot 2^1 + 1 \cdot 2^2. \quad (\text{B.1.3})$$

Therefore, $d_0 = 1$, $d_1 = 0$ and $d_2 = 1$. The radical inverse function $\phi_2(5)$ is:

$$\phi_2(5) = \frac{1}{2} + \frac{0}{4} + \frac{1}{8} = \frac{5}{8}. \quad (\text{B.1.4})$$

Definition B.1.2. The Van der Corput sequence in base b is defined as $\{\phi_b(n)\}_{n \in \mathbb{N}_0}$.

To clarify this concept we provide an example of a finite Van der Corput sequence. Suppose we have the finite sequence of numbers $n \in \{0, 1, \dots, 8\}$ the corresponding Van der Corput sequence $\{\phi_b(n)\}_{n \in \{0, 1, \dots, 8\}}$ in base $b = 2$ is:

$$\{\phi_2(n)\}_{n \in \{0, 1, \dots, 8\}} = \left\{ 0, \frac{1}{2}, \frac{1}{4}, \frac{3}{4}, \frac{1}{8}, \frac{5}{8}, \frac{3}{8}, \frac{7}{8}, \frac{1}{16} \right\}. \quad (\text{B.1.5})$$

It can be proved that the Van der Corput sequence in base b is uniformly distributed modulo one [52]. The van der Corput sequence has been extended to higher dimensions. Sobol sequences can be seen as extended Van der Corput sequences in base $b = 2$ for every dimension $d \geq 2$. Its construction is briefly explained below.

B.2 Sobol sequences

The aim is to generate a low-discrepancy sequence in the hypercube $[0, 1]^d$. The construction of a Sobol sequence can be schematized as follows.

Let us start from the simplest case of $d = 1$. Let P_j be a primitive polynomial in the field $\mathbb{Z}_2[x]$ that is a polynomial monic and irreducible over $\mathbb{Z}_2[x]$. The polynomial of degree s_j has the form:

$$P_j : x^{s_j} + a_{1,j}x^{s_j-1} + \dots + a_{s_j-1,j}x + 1, \quad (\text{B.2.1})$$

where the coefficients $\{a_{i,j}\}_{i=1, \dots, s_j-1}$ are either 0 or 1 and are chosen arbitrary[?]. The next step is to select odd positive numbers $m_{k,j}$ such that $m_{k,j} < 2^k$ for $1 \leq k \leq s_j$ while, for $k > s_j$, $m_{k,j}$ are defined recursively by:

$$m_{k,j} := 2a_{1,j}m_{k-1,j} \oplus 2^2a_{2,j}m_{k-2,j} \oplus \dots \oplus 2^{s_j-1}a_{s_j-1,j}m_{k-s_j+1,j} \oplus 2^{s_j}m_{k-s_j,j} \oplus m_{k-s_j,j}, \quad (\text{B.2.2})$$

where we have indicated with \oplus the bit-by-bit exclusive or operator which operates on two bit patterns giving as result 1 if two bits are different or 0 if both bits are equal. Now, the so-called direction numbers are defined by:

$$v_{k,j} = \frac{m_{k,j}}{2^k}. \quad (\text{B.2.3})$$

Then, for $n \in \mathbb{N}_0$ with finite base 2 expansion:

$$n = \sum_{i=0}^r n_i 2^i \quad (\text{B.2.4})$$

the sequence $\{x_{i,j}\}$ is given by

$$x_{n,j} = n_0 v_{1,j} \oplus n_1 v_{2,j} \oplus \dots \oplus n_{r-1} v_{r,j}. \quad (\text{B.2.5})$$

$x_{n,j}$ is the j -th component of the n -th points of a Sobol sequence. Sobol sequence is the sequence of points $(x_n)_{n \in \mathbb{N}_0}$. In the following we show with an example how to derive the first few points of a Sobol sequence for $d = 1$.

Let consider the primitive polynomial $Q_j : x^3 + x^2 + 1$ of degree $s_j = 3$, where $a_{1,j} = 0$ and $a_{2,j} = 1$. We start from the first three coefficients $m_{1,j} = 1$, $m_{2,j} = 3$, and $m_{3,j} = 7$ (note that other choices are possible). They lead to the following direction numbers:

$$v_{1,j} = \frac{1}{2}, \quad v_{2,j} = \frac{3}{4}, \quad v_{3,j} = \frac{7}{8}, \quad (\text{B.2.6})$$

that in binary notation are:

$$v_{1,j} = (0.1)_2 \quad v_{1,j} = (0.11)_2, \quad v_{1,j} = (0.111)_2. \quad (\text{B.2.7})$$

For the polynomial Q_j , Equation (B.2.2) becomes:

$$m_{k,j} := 2a_{1,j}m_{k-1,j} \oplus 2^2a_{2,j}m_{k-2,j} \oplus 2^3m_{k-3,j} \oplus m_{k-3,j} = 4m_{k-2,j} \oplus 8m_{k-3,j} \oplus m_{k-3,j}. \quad (\text{B.2.8})$$

Therefore,

$$\begin{aligned} m_{4,j} &= 4m_{2,j} \oplus 8m_{1,j} \oplus m_{1,j} \\ &= 12 \oplus 8 \oplus 1 \\ &= (1100)_2 \oplus (1000)_2 \oplus (0001)_2 \\ &= (0101)_2 = 5 \\ m_{5,j} &= 4m_{3,j} \oplus 8m_{2,j} \oplus m_{2,j} \\ &= 28 \oplus 24 \oplus 3 \\ &= (11100)_2 \oplus (11000)_2 \oplus (00011)_2 \\ &= (00111)_2 = 7, \end{aligned}$$

and so on. The corresponding direction are vectors:

$$v_{4,j} = \frac{5}{16} = (0.0101)_2 \quad v_{5,j} = \frac{7}{32} = (0.00111)_2. \quad (\text{B.2.10})$$

From (B.2.5) we find the j -th component of the first five points:

$$\begin{array}{lll} 0 = (0)_2 & x_{0,j} & = 0 \\ 1 = (1)_2 & x_{1,j} & = (0.1)_2 = 0.5 \\ 2 = (10)_2 & x_{2,j} & = (0.11)_2 = 0.75 \\ 3 = (11)_2 & x_{3,j} & = (0.1)_2 \oplus (0.11)_2 = (0.01)_2 = 0.25 \\ 4 = (100)_2 & x_{4,j} & = (0.111)_2 = 0.875 \\ 5 = (101)_2 & x_{5,j} & = (0.1)_2 \oplus (0.111)_2 = (0.011)_2 = 0.375. \end{array}$$

The generalization of Sobol sequences to higher dimensions $d > 1$ is calculated considering a sequence where the n -th point has the form:

$$q_n = (x_{n,1}, x_{n,2}, \dots, x_{n,d}), \quad (\text{B.2.11})$$

where the second index of the variables $x_{n,j}$ refers to the polynomial P_j (with corresponding degree s_j) which is considered for calculating the direction numbers. Therefore, d different sets of direction numbers are generated from a given polynomial P_j using Equation (B.2.3) and each component $x_{n,j}$ is computed using the corresponding direction vector.

Appendix C

Exact intensity for the two-faceted cup

For very simple optical systems, the *exact* target intensity of light can be calculated. Here we explain how this is done for the two-faceted cup formed by a Lambertian source described in Chapter 3. We remind the reader that, for this system, the intensity in a given direction p in PS is defined as:

$$I_{\text{PS}}(p) = \sum_{\Pi} \int_{q^{\min}(\Pi, p)}^{q^{\max}(\Pi, p)} L(q, p) dq = \sum_{\Pi} (q^{\max}(\Pi, p) - q^{\min}(\Pi, p)), \quad (\text{C.0.1})$$

where the sum is over all the possible paths, $q^{\min}(\Pi, p)$ and $q^{\max}(\Pi, p)$ are the minimum and maximum position coordinates of the rays on $\partial R_t(\Pi)$ along direction p , and the second equation holds as we assume $L = 1$ in $R_t(\Pi)$. Therefore, if we are able to provide an analytic expression for the boundaries $\partial R_t(\Pi)$ we can calculate the position coordinates $q^{\min}(\Pi, p)$ and $q^{\max}(\Pi, p)$ of the intersection points between the line $p = \text{const}$ and $\partial R_t(\Pi)$ analytically and the exact intensity for every direction p is obtained using (C.0.1). The procedure used for such purpose is explained next.

C.1 Analytic approach

The idea is to rotate the cup to determine the maximum number of reflections between the rays and the optical lines before reaching the target. The rays are considered to be straight lines instead of broken lines. Hence it is sufficient to find only one intersection point between the ray and a line segment (also in the case where more than one reflection occurs). Finally transforming back these points we obtain the corresponding coordinates at the target.

The two-faceted cup is defined in the (x, z) -plane as in Chapter 3. Let $\gamma \in [0, \pi/2]$ be the angle that the left and right reflector make with the normal to the source. The maximum z -coordinate that the two-faceted cup can reach during the rotation is

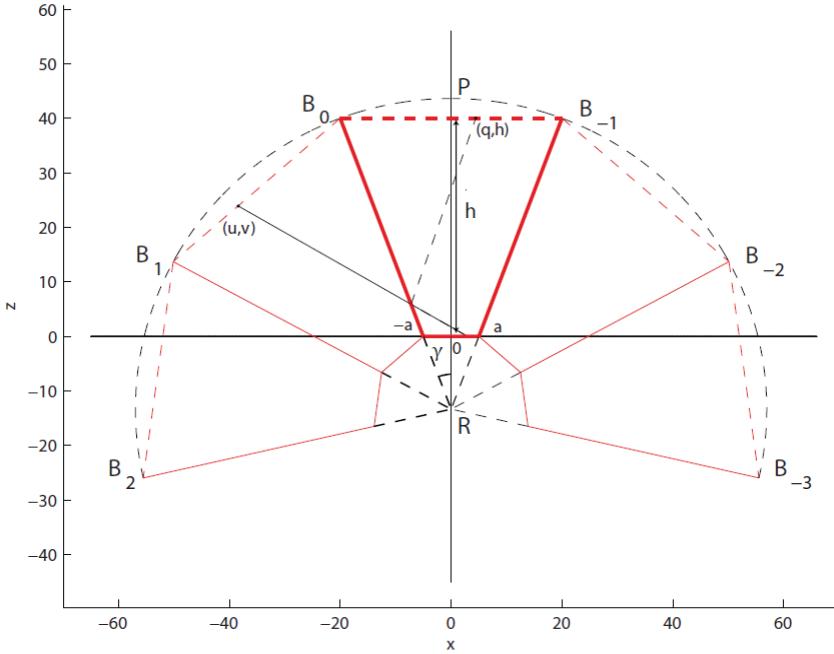


Figure C.1: **The two-faceted cup rotated twice on both sides.** The line segment with end points B_{k-1} and B_k is the $|k|$ times rotated target. The coordinates (q, h) on the target $B_{-1}B_0$ are obtained by transforming the coordinates (u, v) of the intersection point between a ray and the segment B_0B_1 . The point $R = (0, -\frac{a}{\tan \gamma})$ is the center of the circle described by rotating the cup (dashed line).

defined by:

$$\begin{aligned} Z &= \left(h + \frac{a}{\tan \gamma} \right) \frac{1}{\cos \gamma} - \frac{a}{\tan \gamma} \\ &= \frac{h}{\cos \gamma} + \frac{a(1-\cos \gamma)}{\sin \gamma}, \end{aligned} \quad (\text{C.1.1})$$

and $R = (0, -\frac{a}{\tan \gamma})$ is the rotation point. We define B_k as the clockwise ($k < 0$) or counterclockwise ($k \geq 0$) rotation image around the point R over an angle $\alpha_k = (2k+1)\gamma$, with k an integer number (Figure C.1 is illustrative). The position coordinates of points $B_k = (B_{k,x}, B_{k,z})$ are given by:

$$\begin{pmatrix} B_{k,x} \\ B_{k,z} \end{pmatrix} = - \begin{pmatrix} 0 \\ \frac{a}{\tan \gamma} \end{pmatrix} + \begin{pmatrix} \cos \alpha_k & -\sin \alpha_k \\ \sin \alpha_k & \cos \alpha_k \end{pmatrix} \cdot \begin{pmatrix} 0 \\ Z + \frac{a}{\tan \gamma} \end{pmatrix}, \quad (\text{C.1.2})$$

The maximum number of reflections r_{\max} a ray can undergo before arriving at the target is:

$$r_{\max} = \max\{k \in \mathbb{N} \mid B_{k-1,z} \geq 0\} \quad (\text{C.1.3})$$

because rays cannot reach $z < 0$. For example, for the two-faceted cup depicted in Figure 3.1, we found $r_{\max} = 2$.

Given the coordinates (x_1, z_1) and the angular coordinate t_1 of a ray at the source we can calculate the corresponding position (x, z) and direction coordinate t at the target as explained in the following.

We compute the coordinates (u, v) of the intersection point between the ray parametrization and the $|k|$ times rotated or reflected target $B_{k-1}B_k$ for which the intersection with the forward ray is not empty where $k = -r_{\max} - 1, \dots, r_{\max}$. Next, if k is even, the corresponding coordinates (x, z) at the target are found by rotating back the coordinates (u, v) , otherwise a reflection is applied. Therefore, the ray coordinates (x, z) at the target are given by:

$$\begin{pmatrix} x \\ z \end{pmatrix} = \begin{pmatrix} (-1)^k & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \cos(-2k\gamma) & -\sin(-2k\gamma) \\ \sin(-2k\gamma) & \cos(-2k\gamma) \end{pmatrix} \left(v + \frac{u}{\tan(\gamma)} \right) - \begin{pmatrix} 0 \\ \frac{a}{\tan\gamma} \end{pmatrix}. \quad (\text{C.1.4})$$

We observe that the sign depends on the parity of k . When $k = 0$, i.e., the ray does not reflect, the first two matrices become the identity matrix and the cup is not rotated nor reflected. When k is even, the determinant of the matrix given by the product between the first and the second matrix (C.1.4) is equal to 1 and we obtained a rotation matrix, while when k is odd the determinant of the product matrix is -1 and we have a reflection matrix.

The method of transforming the cup instead of the rays allows us to determine the positive luminance regions $R_1(\Pi_j)$ and $R(\Pi_j)$ in source and target PS, where every path $(\Pi_j)_{j=1, \dots, 2r_{\max}+1}$ corresponds to $|k|$ of reflections. The corresponding boundaries only consist of rays that either leave the extremes of the source or hit one of the points B_k .

Rays that leave the interior of S and hit B_k have as position coordinates in source PS $q_1 = x_1 \in (-a, a)$, the corresponding target PS coordinates are $q = x = B_{k,x}$. The direction coordinates of these rays at the source PS are $p_1 = \sin(t_1)$ where t_1 is given by:

$$t_1 = \arctan \left(\frac{x_1 - B_{k,x}}{B_{k,z}} \right). \quad (\text{C.1.5})$$

The corresponding direction coordinates at the target PS are $p = \sin(t)$ where t is given by:

$$t = (-1)^k(t_1 - 2k\gamma). \quad (\text{C.1.6})$$

Rays emitted from the end points of the source have a constant position coordinate $q_1 = x_1 = \text{const}$ in source PS while varying the direction coordinate $p_1 = \sin(t_1) \in [-1, 1]$ where $t_1 \in [-\pi/2, \pi/2]$. The corresponding target position coordinates x is obtained from (C.1.4) while the direction coordinates in the target PS are $p = \sin(t)$, where t is given by Equation (C.1.6) and $t_1 \in [-\pi/2, \pi/2]$. Note that the rays emitted from the end points of the source form vertical lines in source PS as $q_1 = \text{const}$ while varying $p_1 \in [-1, 1]$. On the other hand, rays that hit points B_k form vertical lines in target PS as $q = \text{const}$ while varying $p \in [-1, 1]$.

The boundaries at the source and target PS are shown in red in Figures C.2 and C.3, respectively. Once the boundaries at the target are determined, the coordinates of the intersection points between line $p = \text{const}$ and $\partial R(\Pi_j)$ are found for every path $(\Pi_j)_{j=1, \dots, 2k+1}$ along every direction p . The target intensity is computed using

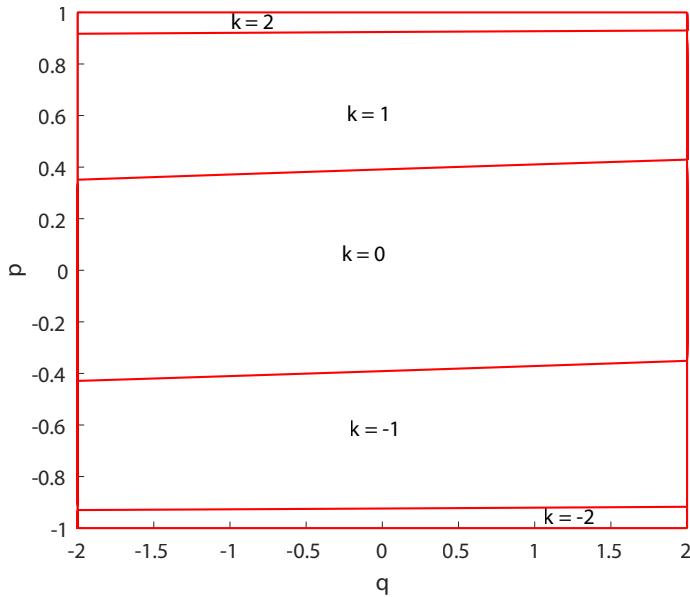


Figure C.2: Regions in source PS formed by rays that reflect $|k|$ times, for the two-faceted cup in Figure 3.1.

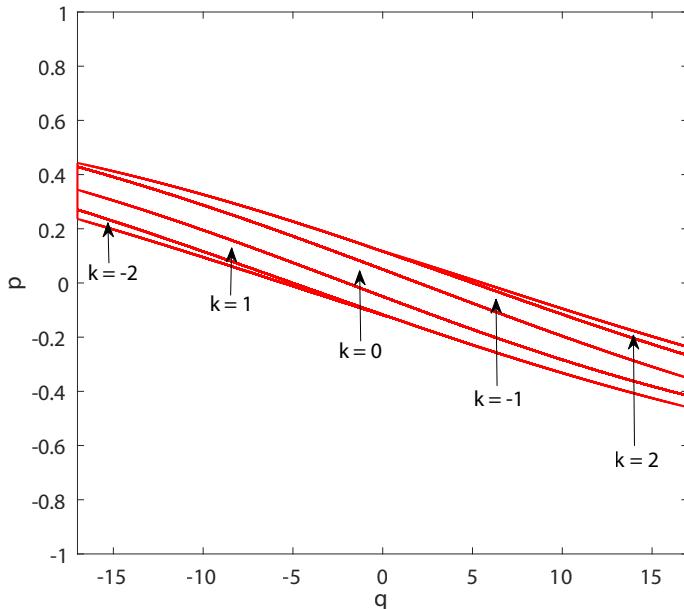


Figure C.3: Regions in target PS formed by rays that reflect $|k|$ times, for the two-faceted cup in Figure 3.1.

Equation (C.0.1). The intensity profile is depicted in Figure C.4. Since the boundaries

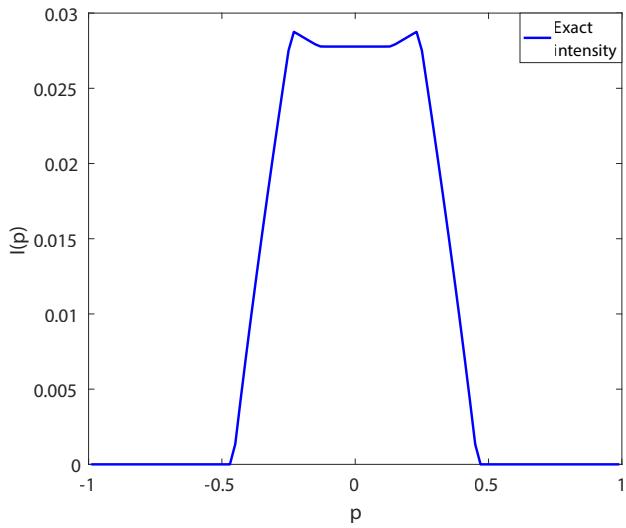


Figure C.4: Profile of the exact intensity at the target of the two-faceted cup.

are calculated analytically, the intensity $I(p)$ found is the exact intensity.

The exact intensity found as described above was taken as reference intensity in Chapters 6 and 7.

Bibliography

- [1] “How much electricity is used for lighting in the united states?” <https://www.eia.gov/tools/faqs/faq.php?id=99&t=3>.
- [2] “Global energy statistical year book 2017.” <https://yearbook.enerdata.net/total-energy/world-consumption-statistics.html>.
- [3] J. D. Bergesen, L. Tähkämö, T. Gibon, and S. Suh, “Potential long-term global environmental implications of efficient light-source technologies,” *Journal of Industrial Ecology*, vol. 20, no. 2, pp. 263–275, 2016.
- [4] <https://www.nasa.gov/sites/default/files/thumbnails/image/naples.jpg>.
- [5] C. J. Humphreys, “Solid-state lighting,” *MRS bulletin*, vol. 33, no. 4, pp. 459–470, 2008.
- [6] R. J. Koshel, *Illumination Engineering: design with nonimaging optics*. John Wiley & Sons, 2012.
- [7] I. Moreno and C.-C. Sun, “Modeling the radiation pattern of leds,” *Optics express*, vol. 16, no. 3, pp. 1808–1819, 2008.
- [8] T. Taguchi, “Present status of energy saving technologies and future prospect in white led lighting,” *IEEJ Transactions on Electrical and Electronic Engineering*, vol. 3, no. 1, pp. 21–26, 2008.
- [9] R. Haitz and J. Y. Tsao, “Solid-state lighting:‘the case’10 years after and future prospects,” *physica status solidi (a)*, vol. 208, no. 1, pp. 17–29, 2011.
- [10] B. Schweber, “Led luminaire design change may hamper repairability.” https://www.eetimes.com/author.asp?doc_id=1326332.
- [11] C. Prins, J. ten Thije Boonkkamp, T. Tukker, and W. IJzerman, “An inverse method for color uniformity in white led spotlights,” *Journal of Mathematics in Industry*, vol. 4, no. 1, p. 5, 2014.
- [12] D. Schreuder *et al.*, *Outdoor lighting: physics, vision and perception*. Springer, 2008.
- [13] M. Born and E. Wolf, *Principles of optics: electromagnetic theory of propagation, interference and diffraction of light*. Elsevier, 2013.

- [14] A. S. Glassner, *An introduction to ray tracing*. Elsevier, 1989.
- [15] H. Gross, *Handbook of the Optical Systems*, vol. 1. Wiley-VCH, 2005.
- [16] Z. Liu, K. Wang, X. Luo, and S. Liu, “Precise optical modeling of blue light-emitting diodes by monte carlo ray-tracing,” *Optics express*, vol. 18, no. 9, pp. 9398–9412, 2010.
- [17] D. Z. Ting and T. C. McGill Jr, “Monte Carlo simulation of light-emitting diode light-extraction characteristics,” *Optical Engineering*, vol. 34, no. 12, pp. 3545–3553, 1995.
- [18] G. Levy, “An introduction to quasi-random numbers,” *Numerical Algorithms Group Ltd.*, http://www.nag.co.uk/IndustryArticles/introduction_to_quasi_random_numbers.pdf (last accessed in April 10, 2012), p. 143, 2002.
- [19] R. Ohbuchi and M. Aono, “Quasi-monte carlo rendering with adaptive sampling,” *IBM Tokyo Research Laboratory*, vol. 1, 1996.
- [20] R. E. Caflisch, “Monte carlo and quasi-monte carlo methods,” *Acta numerica*, vol. 7, pp. 1–49, 1998.
- [21] B. Tuffin, “Randomization of quasi-monte carlo methods for error estimation: survey and normal approximation,” *Monte Carlo Methods and Applications mcma*, vol. 10, no. 3-4, pp. 617–628, 2004.
- [22] A. Torre, *Linear ray and wave optics in phase space: Bridging ray and wave optics via the Wigner phase-space picture*. Elsevier, 2005.
- [23] M. Testorf, B. Hennelly, and J. Ojeda-Castañeda, *Phase-Space Optics: Fundamentals and Applications: Fundamentals and Applications*. McGraw Hill Professional, 2009.
- [24] D. Rausch and A. Herkommer, “Illumination optics in phase space,” in *Fringe 2013*, pp. 373–376, Springer, 2014.
- [25] D. Rausch and A. M. Herkommer, “Phase space approach to the use of integrator rods and optical arrays in illumination systems,” *Advanced Optical Technologies*, vol. 1, no. 1-2, pp. 69–78, 2012.
- [26] D. Rausch and A. Herkommer, “Phase space imaging in optical design,” in *Novel Optical Systems Design and Optimization XVII*, vol. 9193, p. 919305, International Society for Optics and Photonics, 2014.
- [27] A. M. Herkommer and D. Rausch, “Phase space optics: an engineering tool for illumination design,” in *Optical Modelling and Design II*, vol. 8429, p. 84290C, International Society for Optics and Photonics, 2012.
- [28] A. M. Herkommer, “Phase space optics: an alternate approach to freeform optical systems,” *Optical Engineering*, vol. 53, no. 3, p. 031304, 2013.

- [29] J. Babington, “Freeform aberrations in phase space: an example,” *JOSA A*, vol. 34, no. 6, pp. 1045–1053, 2017.
- [30] K. B. Wolf, “Relativistic aberration of optical phase space,” *JOSA A*, vol. 10, no. 9, pp. 1925–1934, 1993.
- [31] K. B. Wolf, *Geometric optics on phase space*. Springer Science & Business Media, 2004.
- [32] J. Marsden, L. Sirovich, S. Antman, G. Iooss, P. Holmes, D. Barkley, and M. Dellnitz, “Texts in applied mathematics 44,” 2003.
- [33] M. Teichmann and M. Capps, “Surface reconstruction with anisotropic density-scaled alpha shapes,” in *Visualization'98. Proceedings*, pp. 67–72, IEEE, 1998.
- [34] C. Filosa, J. ten Thije Boonkkamp, and W. IJzerman, “A new ray tracing method in phase space using alpha-shapes,” 2015.
- [35] C. Filosa, J. ten Thije Boonkkamp, and W. IJzerman, “Ray tracing method in phase space for two-dimensional optical systems,” *Applied optics*, vol. 55, no. 13, pp. 3599–3606, 2016.
- [36] C. Filosa, J. ten Thije Boonkkamp, and W. IJzerman, “Phase space ray tracing for a two-dimensional parabolic reflector,” 2017.
- [37] H. Ries and A. Rabl, “Edge-ray principle of nonimaging optics,” *JOSA A*, vol. 11, no. 10, pp. 2627–2632, 1994.
- [38] C. Filosa, W. L. IJzerman, and J. ten Thije Boonkkamp, “Inverse ray mapping on phase space for two-dimensional optical systems,” in *Optical Fabrication and Testing*, pp. JW1B–5, Optical Society of America, 2017.
- [39] E. Hecht, *Optics*. Parson Addison-Wesley, 2002.
- [40] R. P. Feynman, R. B. Leighton, and M. Sands, *The Feynman lectures on physics, Vol. I: The new millennium edition: mainly mechanics, radiation, and heat*, vol. 1. Basic books, 2011.
- [41] R. P. Feynman, “Feynman lectures on physics. volume 2: Mainly electromagnetism and matter,” *Reading, Ma.: Addison-Wesley, 1964, edited by Feynman, Richard P.; Leighton, Robert B.; Sands, Matthew*, 1964.
- [42] E. F. Zalewski, “Radiometry and photometry,” *Handbook of optics*, vol. 2, pp. 24–1, 1995.
- [43] J. Chaves, *Introduction to nonimaging optics*. CRC press, 2015.
- [44] E. F. Schubert, T. Gessmann, and J. K. Kim, *Light emitting diodes*. Wiley Online Library, 2005.
- [45] <https://en.wikipedia.org/wiki/NonimagingopticsEdgerayprinciple>.
- [46] A. V. Arecchi, R. J. Koshel, and T. Messadi, “Field guide to illumination,” SPIE, 2007.

- [47] S. A. Lerner and B. Dahlgren, “Etendue and optical system design,” in *Non-imaging Optics and Efficient Illumination Systems III*, vol. 6338, p. 633801, International Society for Optics and Photonics, 2006.
- [48] H. Zhu and P. Blackborow, “Etendue and optical throughput calculations,” *En-ergetiq Technology, Inc., Woburn, MA*, 2011.
- [49] A. I. Lvovsky, “Fresnel equations,” *Encyclopedia of Optical Engineering*, pp. 1–6, 2013.
- [50] R. Winston, J. C. Miñano, and P. Benitez, *Nonimaging optics*. Academic Press, 2005.
- [51] H. W. Jensen, J. Arvo, P. Dutre, A. Keller, A. Owen, M. Pharr, and P. Shirley, “Monte carlo ray tracing,” in *ACM SIGGRAPH*, pp. 27–31, 2003.
- [52] G. Leobacher and F. Pillichshammer, *Introduction to quasi-Monte Carlo integration and applications*. Springer, 2014.
- [53] V. M. Zolotarev, *Modern theory of summation of random variables*. Walter de Gruyter, 1997.
- [54] R. Y. Rubinstein and D. P. Kroese, *Simulation and the Monte Carlo method*, vol. 10. John Wiley & Sons, 2016.
- [55] D. M. Diez, C. D. Barr, and M. Cetinkaya-Rundel, *OpenIntro statistics*. CreateSpace, 2012.
- [56] A. B. Owen, “Quasi-monte carlo sampling,” *Monte Carlo Ray Tracing: Siggraph*, vol. 1, pp. 69–88, 2003.
- [57] L. Brandolini, L. Colzani, G. Gigante, and G. Travaglini, “A koksma–hlawka inequality for simplices,” in *Trends in harmonic analysis*, pp. 33–46, Springer, 2013.
- [58] X. Wang and I. H. Sloan, “Low discrepancy sequences in high dimensions: How well are their projections distributed?,” *Journal of Computational and Applied Mathematics*, vol. 213, no. 2, pp. 366–386, 2008.
- [59] I. L. Dalal, D. Stefan, and J. Harwayne-Gidansky, “Low discrepancy sequences for monte carlo simulations on reconfigurable platforms,” in *Application-Specific Systems, Architectures and Processors, 2008. ASAP 2008. International Conference on*, pp. 108–113, IEEE, 2008.
- [60] P. Bratley and B. L. Fox, “Algorithm 659: Implementing sobol’s quasirandom sequence generator,” *ACM Transactions on Mathematical Software (TOMS)*, vol. 14, no. 1, pp. 88–100, 1988.
- [61] W. Welford and R. Winston, “On the problem of ideal flux concentrators,” *JOSA*, vol. 68, no. 4, pp. 531–534, 1978.

- [62] P. Benitez, R. Mohedano, J. C. Miñano, R. Garcia, and J. C. Gonzalez, “Design of cpc-like reflectors within the simultaneous multiple surface design method,” in *Proc. SPIE*, vol. 3139, pp. 19–28, 1997.
- [63] <https://commons.wikimedia.org/wiki/File:CIE1931Luminosity.png>
- [64] J. C. Miñano, “Two-dimensional nonimaging concentrators with inhomogeneous media: a new look,” *JOSA A*, vol. 2, no. 11, pp. 1826–1831, 1985.
- [65] J. C. Miñano, “Design of three-dimensional nonimaging concentrators with inhomogeneous media,” *JOSA A*, vol. 3, no. 9, pp. 1345–1353, 1986.
- [66] P. Davies, “Edge-ray principle of nonimaging optics,” *JOSA A*, vol. 11, no. 4, pp. 1256–1259, 1994.
- [67] J. C. Miñano and J. C. Gonzalez, “New method of design of nonimaging concentrators,” *Applied optics*, vol. 31, no. 16, pp. 3051–3060, 1992.
- [68] J. Portegies and P. Lighting, “Fast ray tracing in phase space for optical design,” 2013.
- [69] B. Guo, J. Menon, and B. Willette, “Surface reconstruction using alpha shapes,” in *Computer Graphics Forum*, vol. 16, pp. 177–190, Wiley Online Library, 1997.
- [70] X. Xu and K. Harada, “Automatic surface reconstruction with alpha-shape method,” *The visual computer*, vol. 19, no. 7, pp. 431–443, 2003.
- [71] A. Lucieer and M.-J. Kraak, “Alpha-shapes for visualizing irregular-shaped class clusters in 3 d feature space for classification of remotely sensed imagery,” in *Proceedings of SPIE*, vol. 5295, pp. 201–211, 2004.
- [72] “Stracciatella (ice cream).” [https://en.wikipedia.org/wiki/Stracciatella_\(icecream\)](https://en.wikipedia.org/wiki/Stracciatella_(icecream)).
- [73] H. Edelsbrunner, D. Kirkpatrick, and R. Seidel, “On the shape of a set of points in the plane,” *IEEE Transactions on information theory*, vol. 29, no. 4, pp. 551–559, 1983.
- [74] H. Edelsbrunner and E. P. Mücke, “Three-dimensional alpha shapes,” *ACM Transactions on Graphics (TOG)*, vol. 13, no. 1, pp. 43–72, 1994.
- [75] E. P. Mücke, “Shapes and implementations in three-dimensional geometry,” 1993.
- [76] E. L. Lloyd, “On triangulations of a set of points in the plane,” in *Foundations of Computer Science, 1977., 18th Annual Symposium on*, pp. 228–240, IEEE, 1977.
- [77] P. Knabner and L. Angermann, *Numerical Methods for Elliptic and Parabolic Partial Differential Equations*. Springer Science & Business Media, 2003.
- [78] D.-T. Lee and B. J. Schachter, “Two algorithms for constructing a delaunay triangulation,” *International Journal of Computer & Information Sciences*, vol. 9, no. 3, pp. 219–242, 1980.

- [79] R. J. Renka, “Algorithm 772: Stripack: Delaunay triangulation and voronoi diagram on the surface of a sphere,” *ACM Transactions on Mathematical Software (TOMS)*, vol. 23, no. 3, pp. 416–434, 1997.
- [80] W. H. Press, *Numerical recipes 3rd edition: The art of scientific computing*. Cambridge university press, 2007.
- [81] S. Fortune, “Voronoi diagrams and delaunay triangulations,” *Computing in Euclidean geometry*, vol. 1, no. 193-233, p. 2, 1992.
- [82] F. Cazals, J. Giesen, M. Pauly, and A. Zomorodian, “Conformal alpha shapes,” in *Point-Based Graphics, 2005. Eurographics/IEEE VGTC Symposium Proceedings*, pp. 55–61, IEEE, 2005.
- [83] K. Q. Brown, “Voronoi diagrams from convex hulls,” *Information Processing Letters*, vol. 9, no. 5, pp. 223–228, 1979.
- [84] “Delaunay voronoi.png.” <https://commons.wikimedia.org/wiki/File:DelaunayVoronoi.png>.
- [85] H. Edelsbrunner, “Alpha shapes - a survey,” *Tessellations in the Sciences*, vol. 27, pp. 1–25, 2010.
- [86] <https://nl.mathworks.com/help/matlab/ref/delaunay.html>.
- [87] <https://nl.mathworks.com/help/matlab/ref/triangulation.freeboundary.html>.
- [88] D. P. Mandal and C. Murthy, “Selection of alpha for alpha-hull in \mathbb{R}^2 ,” *Pattern Recognition*, vol. 30, no. 10, pp. 1759–1767, 1997.
- [89] K. Moore, “Methods and associated systems for simulating illumination patterns,” Dec.8 8 Dec. 2015. US Patent 9,208,603.
- [90] R. P. Breault, “Control of stray light,” *Handbook of Optics*, vol. 1, pp. 38–1, 1995.
- [91] S. Grabarnik, “Optical design method for minimization of ghost stray light intensity,” *Applied optics*, vol. 54, no. 10, pp. 3083–3089, 2015.
- [92] R. Leutz and A. Suzuki, *Nonimaging Fresnel lenses: design and performance of solar concentrators*, vol. 83. Springer, 2012.
- [93] J. E. Greivenkamp, *Field guide to geometrical optics*, vol. 1. SPIE Press Bellingham, WA, 2004.
- [94] H. Ries, N. Shatz, J. Bortz, and W. Spirk, “Performance limitations of rotationally symmetric nonimaging devices,” *JOSA A*, vol. 14, no. 10, pp. 2855–2862, 1997.

Description of the research

In this thesis we studied the light propagation within optical systems. Optical engineers are interested in design systems in such a way the desired output distribution is obtained. The goal in illumination optics is to obtain the desired output distribution of light. To this purpose the ray tracing procedure is widely used. Ray tracing is a forward method where a set of rays is traced within the system from the source to the target. The propagation of light is determined computing the position and the direction of every ray for all the optical surfaces that it encounters. There are many ways to implement the ray tracing process. Monte Carlo (MC) ray tracing is often used in non-imaging optics. Rays are randomly traced from the source to the target and each time that a ray hits an optical surface the coordinates of the intersection point of the ray with the surface and the new ray direction are calculated. The output variables are computed dividing the target into intervals, the so-called bins, and counting the rays that fall into each bin. To obtain the desired accuracy, millions of rays are required, therefore the method is extremely computationally expensive and it converges as the inverse of the square root of the number of rays traced.

MC ray tracing can be improved using as sample of points a low discrepancy sequence instead of random points. Discrepancy can be interpreted as a measure of how much the sample distribution differs from a uniformly distributed sample. The discrepancy is therefore zero for uniformly distributed points. A low discrepancy sequence gives a sample of points which are regularly distributed but not exactly uniformly distributed. Quasi Monte Carlo (QMC) method considers these kind of sequences as sample of points. Therefore, QMC ray tracing is implemented tracing a set of rays whose position and direction are given by the coordinates of a low discrepancy sequence of points. The main advantage of QMC method is its rate of convergence, it is faster than MC for low dimensional problems. Nevertheless, it has some disadvantages. First, it is not easy to give an error estimation for QMC method. Second, for high dimensional spaces the QMC can become very slow. Third, it is still a binning procedure. Hence, the accuracy depends both on the number of rays traced and on the umber of bins.

In order to improve the existing methods, the phase space (PS) of the optical system is considered in this thesis. The PS of an optical surface gives information about the position and the direction of every ray on that surface where the direction is expressed with respect to the normal of the surface. In PS, the ray's direction is given by the sine of the angle that the ray forms with respect to the normal of the surface multiplied by the index of refraction of the medium in which the ray is located. In two dimensions, the PS is a two-dimensional space where the coordinates of every

ray are specified by one position coordinate and one angular coordinate. For three dimensional systems the PS is a four dimensional space because every ray is specified by two position and two angular coordinates. Our idea is to use the structure of PS to trace only the rays close to the discontinuities of the luminance at the target PS. Two new approaches based on PS are presented in this work. They are tested for two-dimensional systems.

The first method is called ray tracing on PS and it is based on the source and the target PS representation of the optical system. It takes into account the sequence of optical lines that each ray hits when it propagates inside the system, that is the ray path. We note that the source and target phase spaces are partitioned into different regions each of them is formed by the rays that follow the same path. The idea is to use the edge-ray principle proved by Ries and Rabl (1994) which states that the area of these regions is conserved: all rays that are neighbors at the source PS remain close to each other at the target PS. To this purpose, a nonuniform triangulation of the source PS is constructed in such a way that new triangles are added to the triangulation only where boundaries occur. Assuming constant brightness, we only need to compute the boundaries of the regions in target PS to obtain the output photometric variables. We test the method for optical systems where both reflection and refraction laws are involved. Numerical results show that ray tracing on PS is faster and more accurate compared to MC ray tracing.

The second method employs not only the source and the target PS, but also the PS of *all* the other lines that constitute the system. All lines can be modeled as detectors of the incident light and emitters of the reflected light. Moreover, we assume that the source can only emit light and the target can only receive light. Therefore, one PS is taken into account for the source and one for the target. For the other surfaces both the source and target PS are considered. Furthermore, instead of starting from the source, the new method starts tracing back rays from target PS. In order to determine the coordinates of these rays, an inverse map from the target to the source PS is constructed as a concatenation of the maps that relate the PS of two different lines. Employing this map we are able to detect the rays that in target PS are located on the boundaries of the regions with positive luminance. First, we implement the method for systems formed by straight and reflective lines. In this particular case, the boundaries of the regions that form every PS can be computed analytically. This allows us to obtain an analytic target intensity distribution. The results are shown for a two-faceted cup and a multi-faceted cup. In both cases we note significant advantages both in terms of the accuracy and the computational time. Second, the method is developed for systems formed by curved lines. In this case the boundaries cannot be determined analytically and therefore a numerical procedure is involved. In particular, we apply a bisection method on target PS. Also in this case we compare our method to MC ray tracing and we observe significant advantages using the PS method. Finally, the ray mapping method in PS is applied to systems where also Fresnel reflection is taken into account. We obtain relevant results also in the last case.

Curriculum Vitae

Carmela Filosa was born on November 28, 1985 in Torre del greco, Italy. She finished the high school in 2003 at Liceo Scientifico Statale "G. Marconi", Colleferro. She obtained a bachelor (2008) and Master (2013) degree in Mathematics at the University of Rome "La Sapienza", Italy. In March 2014, she moved in Eindhoven (the Netherlands) to start a PhD project at the Eindhoven University of Technology in the department of Mathematics and Computer Science. The PhD project was under the supervision of Wilbert IJzerman and Jan ten Thije Boonkkamp. The research conducted in her doctoral studies was funded by Technologiestichting STW and, the daily work took place at the Centre for Analysis, Scientific computing and Applications (CASA) of TU/e and at the department of Philips Lighting of the High Tech Campus in Eindhoven. The results of her research are presented in this thesis.

Acknowledgments

Looking back at the journey to pursue my PhD, I realize that many things happened in only four years and that it would not be possible without the help of many people who I would thank here.

First of all, my gratitude goes to my supervisors for leading me during this path. I would never be able to write this thesis without their help.

Wilbert, thank you for your constant support and patience. Every meeting and discussion with you contributed to this work and to my professional growth. You have listened all my doubt and replied to my questions at any time of the day. I learned a lot from you and I am impressed about the breadth of your knowledge and the passion you have for your work.

Jan, thanks for being extremely available and for your daily encouragement. The door of your office was always open when I had a question. I really appreciated your meticulousness in correcting the text of my papers and of this thesis, they helped me a lot to improve my writing skills.

I am thankful to Teus Tukker who was one of my supervisors during the first year of my PhD. It is a pity we collaborated for such a short time.

I want to express my gratitude to all the members of my doctorate committee for having agreed to referee my thesis and for their fruitful comments.

I wish to thank Jorg for providing me his master thesis from which this work started and for discussing it with me.

Thank you for your helpful discussions on Chapter 3 and for the cover of my thesis.

A big thanks goes to all the CASA members for creating a nice and international work atmosphere. This would never be possible without the help of Enna, Marese and Jolijn who organized very nice events within CASA group. Thanks Enna also for helping me with the university bureaucracy.

Giovanni and Deepesh thanks for hosting me at your place when I wasn't able to walk. I will never forget the kindness and thoughtfulness you had for me.

Letizia and Gonzalo

Carla for organizing me a great birthday party with my colleagues and for cooking such a lot and good Italian food for all of us.

Thanks to my current and former officemates: Nitin, Thomas, Mohammed, Arshit, Behnaz, Deepesh, Yuri and Pranav for helping me to keep the concentration every day. A special thanks goes to Deepesh and Behnaz for supporting me and for the nice and pleasure time we spent together. I will be always grateful of your friendship.

Laura, thank you for your cute and encouragement words during my difficult time of writing the thesis.

Misa, Ale G., Ghislaine, Chiara, Manouk, Tsvetan and Jacopo, I will never regret to have shared our house with you and you never made me feel lonely at home. I discovered the beauty and the difficulty to meet up with people of different cultures.

I wish to thanks my mathematicians friends back in Italy: Federica, Ida, Stefania, Daniela and Arianna. It is always nice when we manage to meet all together after long time.

Italian friends in Eindhoven: Ale G., Ale Z., Carlo, Enrico, Fabio, Lorenzo, Marta and Misa.

My climbing friends to motivated me and to have shared this passion with them. Most of all Ale Z for being a nice teacher when I started and Carlo for making me discover how wonderful is to climb on the real rock.

Dad, you have been always a mystery for me. I am not able to explain all my emotions for you, but I can say for sure that the strongest feelings I have for you are love and endearment. I like thinking you have been on my side for all this years, and that now you are proud of me.

My mother for her support. Mamma, grazie per avermi supportato in ogni momento e non aver mai ostacolato le mie scelte nonostante a volte abbiano causato sofferenza e nostalgia per la distanza che ci separa. Il tuo amore e la tua fiducia in me sono stati fondamentali per il raggiungimenti di questo traguardo. Ti voglio bene.

My sister Carla for having been part of my *entire* life. Sister, I cannot put into words how grateful I am and how much I owe to you. I would never be able to pursue none of my successes without you. My admiration and esteem for you, made you my point of reference since I was child. I miss sharing the daily life with you but I feel so lucky when I realize that the distance did change nothing.

I desire to thank Guido for having organized very beatiful excursions around the south of Germany and the nice time we spent in the Netherlands with Carla and Carlo. I have a good memory of all the trips we made together. You are a nice and warm-hearther person.

Last, but not least, I thank you Carlo for your love. Since you are part of my life I This thesis would not have been possible without you. I cannot imagine my future without you at my at my side. I hope you will letting me know at your side for the