RobinTrace

Poaky documentation

Thomas HOULLIER

v1.0 – 11 Dec 2022

Abstract

Poaky is a low-level software API layer for RobinTrace. Poaky is a reference implementation for ray-wise and element-wise operations in the forward simulation of sequential raytracing. The software is written in C++.

Keywords – Optical design, Sequential raytracing, C++

Revision History

Revision	Date	Author(s)	Description
1.0	11 Dec 2022	TH	creation

Contents

CC BY

1 Introduction

We document Poaky, which is a software component of RobinTrace. Poaky deals with basic ray-wise and element-wise sequential raytracing operations. Its goal is to provide a reference implementation and document clearly the forward simulation pass of sequential raytracing. The simulation is carried out within the context of geometrical optics.

We had similar endeavours in a previous work [8]. The present software is a rewrite of this work after putting more thought and research into the problem. Much of the present work is ultimately a rehash of the work of Welford [11], who was foundational in computerized optical design raytracing.

In its present version, only a handful of operations are implemented. The current development goal is to reach a minimal working example in RobinTrace.

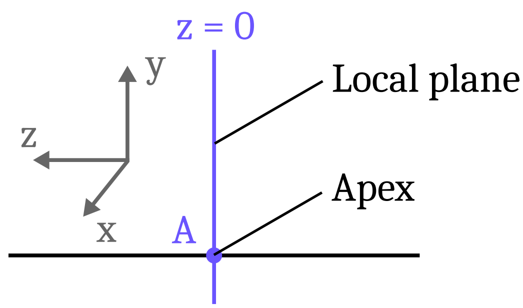

Figure 1: LCS diagram.

Whether Poaky will serve as the base layer for RobinTrace in its current state or merely as a reference is not yet decided.

2 Definitions and conventions

We outline the conventions we use. Some details may differ from the choices made in other tools.

2.1 Raytracing sequence

Poaky deals with the elementary operations in optical systems represented as a sequence of objects operating on rays of light. The objects are of two main types, optical surfaces and geometric propagators through some medium in between surfaces (which we call *transfer*). The sequence in which rays of light interact with either surfaces or transfers is known a priori. The simulation of the propagation of rays through such a sequence of objects is known as *sequential raytracing*. This discipline is linked with the more well known raytracing for rendering, though they seem to have developed more or less autonomously.

2.2 Local coordinate system

Each surface corresponds to an implicit Local Coordinate System (LCS). This coordinate system may be described as containing (Fig. 1):

- An apex A, which is the origin.
- A local $z = 0$ plane, which we often refer to as the *local plane*.
- A set of axes (implicit).

We call the LCS implicit because the actual meaning of the data expressed in it depends on the interplay between surface defition, ray transfer equations and ray operation conventions.

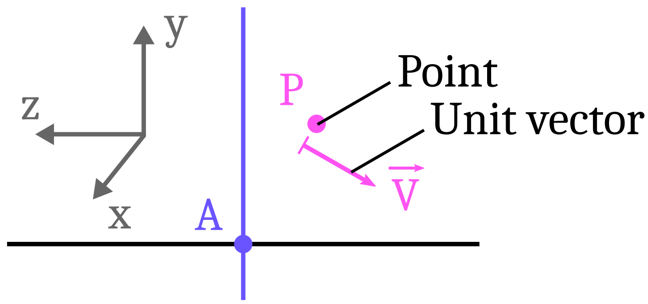

Figure 2: Ray definition in LCS.

2.3 Rays in local coordinate systems

Given a LCS, a ray may be defined by (Fig. 2):

- $P = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$, a point.
- $\overrightarrow{V} = \begin{bmatrix} l \\ m \\ n \end{bmatrix}$, a unit vector oriented by the light propagation.

The components of $\overrightarrow{V}$ are often called in optics by the name *direction cosines* [9]. Indeed, given the basis vectors of our LCS: $(\hat{x}, \hat{y}, \hat{z})$, the vector $\overrightarrow{V}$ can be described by the angles (α, β, γ) between itself and each basis vector respectively. These angles may be defined by (Eq. 1).

$$\begin{cases} \cos(\alpha) & = \frac{\overrightarrow{V} \cdot \hat{x}}{\left|\overrightarrow{V}\right|} \\ \cos(\beta) & = \frac{\overrightarrow{V} \cdot \hat{y}}{\left|\overrightarrow{V}\right|} \\ \cos(\gamma) & = \frac{\overrightarrow{V} \cdot \hat{z}}{\left|\overrightarrow{V}\right|} \end{cases} \tag{1}$$

The points (x_t, y_t, z_t) describing the ray trajectory through the use of a parameter t are described by (Eq. 2).

$$\begin{bmatrix} x_t \\ y_t \\ z_t \end{bmatrix} = \begin{bmatrix} x + t \cdot l \\ y + t \cdot m \\ z + t \cdot n \end{bmatrix} \tag{2}$$

3 Functional description

This section defines the program's objects and their associated operations. The style is minimal and close to the computations. For the rationale sustaining

Code	Meaning
0	Success, the ray is valid.
3	refract: Total Internal Reflection (TIR)
4	transfer: ray is parallel to the new local plane.
5	standard intersection: No intersection.

Table 1: Ray status codes.

the computation and complementary information, see the justification section (Section 4).

3.1 base

Some base types are useful throughout the program. These are detailed in this section.

Vec3 `Vec3` $\in \mathbb{R}^3$ are vectors in 3D space. They may represent points or directions. Depending on the context, they can be implicitely considered to have unit norm.

Mat3 `Mat3` $\in \mathbb{R}^{3\times 3}$ are 3D matrices. They are used to represent rotation matrices.

3.2 ray

`ray` objects are the centerpiece of the simulation. They must be lightweight objects. `ray` holds a position and a unit vector in the direction and orientation of the propagation of light:

- `Vec3 p`: A point.
- `Vec3 v`: A unit vector, oriented by light propagation.

The interpretation of the data contained in a `ray` is dependent on the context, as they are expressed in a given LCS.

In addition to their geometric definition, rays also hold a status code. This code signals whether raytracing operations were successful, and if not, which error case was encountered. There is no guarantee on the value of the ray point and vector when the status code signals an error.

- `int code`: Status code.

The status codes are defined in Tab. 1.

3.3 rop

`rop` are ray operations.

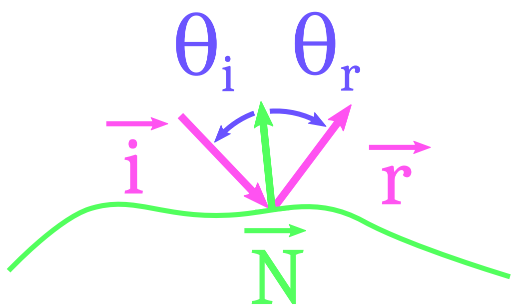

Figure 3: reflect operation quantities.

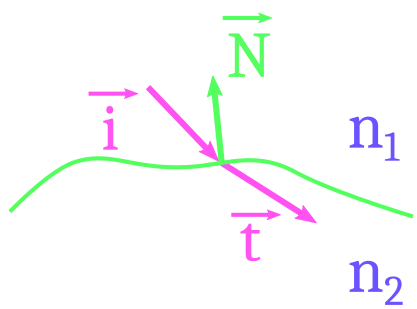

Figure 4: refract operation quantities.

3.3.1 reflect

`reflect` is a ray operation which applies the law of specular reflection [17]. The normal vector $\overrightarrow{N}$ is an input to the operation. There are no error cases. The operation is illustrated on Fig. 3.

$$\begin{bmatrix} l_r \\ m_r \\ n_r \end{bmatrix} = \begin{bmatrix} l \\ m \\ n \end{bmatrix} - 2 \cdot \overrightarrow{N} \cdot \left(\overrightarrow{N} \cdot \begin{bmatrix} l \\ m \\ n \end{bmatrix} \right) \tag{3}$$

3.3.2 refract

`refract` is a ray operation applying the Snell law of refraction [16]. We use Xavier Bec's formula ([6] p.105, [1]) for efficiency. The operation is illustrated on Fig. 4.

Let,

- n_1 the incident medium refraction index,
- n_2 the output medium refraction index,
- $n_r = \frac{n_1}{n_2}$,
- $\overrightarrow{i}$ the unit incident ray direction,

- $\overrightarrow{N}$ the unit surface normal vector,
- $\overrightarrow{t}$ the unit refracted ray direction.

$$\begin{aligned} c_1 &= -\overrightarrow{i} \cdot \overrightarrow{N} \\ w &= n_r \cdot c_1 \\ c_{2m} &= (w - n_r) \cdot (w + n_r) \end{aligned} \tag{4}$$

At this stage, if $c_{2m} < -1$, then we set the TIR ray error code and the computation stops. Otherwise we proceed with the computation of the refracted ray direction.

$$\overrightarrow{t} = n_r \cdot \overrightarrow{i} + (w - \sqrt{1 + c_{2m}}) \cdot \overrightarrow{N} \tag{5}$$

3.4 transfer

A transfer defines a new LCS to propagate the ray to. The transfer operation a change of basis and an intersection with the newly defined local plane.

Definition Let,

- LCS1 the starting LCS defined by its origin and basis vectors $(A_1, \hat{x_1}, \hat{y_1}, \hat{z_1})$,
- LCS2 the LCS defined by the transfer operation, $(A_2, \hat{x_2}, \hat{y_2}, \hat{z_2})$.

The transfer operation is characterized by

- `Mat3` B: A rotation matrix between the basis vectors of LCS1 and LCS2. B is orthogonal by definition, hence $B^{-1} = B^{\top}$ [15].
- `Vec3` $\overrightarrow{D}$: A translation vector between LCS1 and LCS2.

The coordinates of the origin and basis vectors of LCS2 are expressed in the original LCS1 coordinates by the following relations.

$$\begin{cases} A_2 = B \cdot \overrightarrow{D} \\ \hat{x_2} = B \cdot \hat{x_1} \\ \hat{y_2} = B \cdot \hat{y_1} \\ \hat{z_2} = B \cdot \hat{z_1} \end{cases} \tag{6}$$

Which is to say, LCS2 is obtained from LCS1 by first applying the rotation B and then translating the origin by $\overrightarrow{D}$, with $\overrightarrow{D}$ expressed in the rotated coordinates. The change of basis is illustrated on Fig. 5.

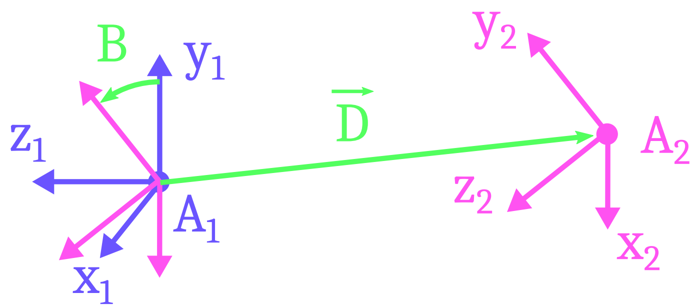

Figure 5: Definition of the transfer between LCS1 and LCS2.

Operation The `transfer` operation can be decomposed as two successive operations on the ray:

- A change of basis from LCS1 to LCS2.
- A ray intersection with the local plane of LCS2.

Let a ray expressed in LCS1 with starting point and direction $(P_1, \overrightarrow{V_1})$. We first operate a change of basis from LCS1 to LCS2, which gives the coordinates $(P_2, \overrightarrow{V_2})$.

$$\begin{cases} P_2 = \begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix} = B^{-1} \cdot P_1 - \overrightarrow{D} \\ \overrightarrow{V_2} = \begin{bmatrix} l_2 \\ m_2 \\ n_2 \end{bmatrix} = B^{-1} \cdot \overrightarrow{V_1} \end{cases} \tag{7}$$

We signal an error in the case $n_2 = 0$. This case corresponds to the ray being parallel to the local plane of LCS2. We operate the intersection of the ray with the LCS2 local plane next. The result ray is $(P_3, \overrightarrow{V_3})$. The operation is illustrated by Fig. 6.

$$\begin{cases} \overrightarrow{V_3} = \overrightarrow{V_2} \\ t = -\frac{z_2}{n_2} \\ P_3 = \begin{bmatrix} x_2 + t \cdot l_2 \\ y_2 + t \cdot m_2 \\ 0 \end{bmatrix} \end{cases} \tag{8}$$

3.5 shape

Shape is an abstract concept specifying two operations:

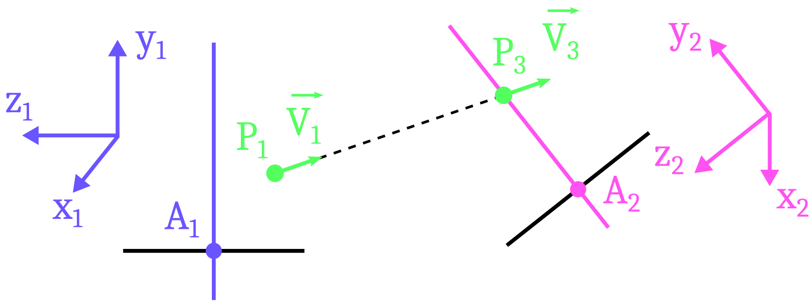

Figure 6: Illustration of the transfer operation on a ray.

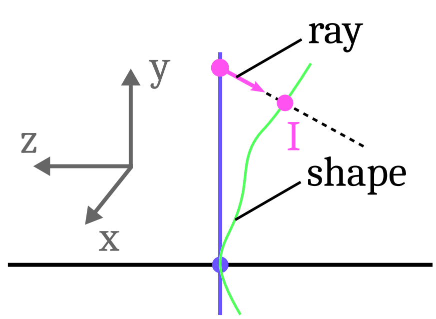

Figure 7: Illustration of the abtract intersect operation for a shape on a ray.

- `intersect`: Intersect a ray with the shape.
- `normal`: Provide a vector normal to the shape at the current ray position.

intersect The intersection operation takes a `ray` expressed in the current surface coordinate system with point on the local plane. It computes the intended intersection point between the ray and the shape. This operation is illustrated on Fig. 7.

normal The normal operation provides a normal vector at the ray's current position on the shape. The normal vector is expressed in the surface LCS. The normal vector is a unit vector. The normal vector is oriented with a $\hat{z}$ component of opposite sign to that of the ray's vector, *ie* the normal vector is in the opposite half-plane to the incident ray. The normal operation is illustrated on Fig. 8.

3.5.1 plane

Definition A `plane` is the local $z = 0$ plane in the current LCS. It is specified implicitely.

intersect The input ray is already on the local plane. We do *nothing* and cannot fail.

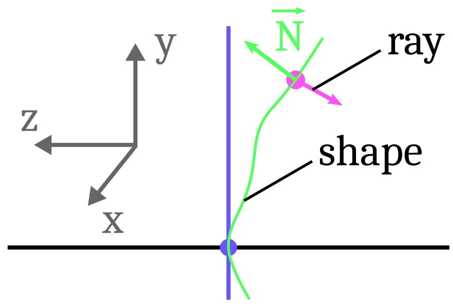

Figure 8: Illustration of the abstract normal operation for a shape and a ray.

normal The plane normal vector is trivial. It is $\overrightarrow{N} = (0, 0, -\text{sign}(n))$. There are no error cases.

3.5.2 standard

The so-called `standard` shape describes surfaces which belong to quadrics of revolution with axis z.

Definition The surface shape `standard` is defined using:

- c: the radius of curvature reciprocal, $c = \frac{1}{R}$.
- k: a scalar parameter which changes the type of the quadric.

The mathematical classification of the surface depends on the value of k,

- $k < -1$: One of the sheets of a hyperboloid of revolution of two sheets.
- $k = -1$: Circular paraboloid.
- $k > -1$: Spheroid, with the case $k = 0$ being a sphere.

Note that the case $c = 0$ describes a plane.

An explicit altitude formula may be given for part of the surface (in the case of spheroids, only the hemisphere containing the apex is described by this formula). Let $r^2 = x^2 + y^2$.

$$z = \frac{c \cdot r^2}{1 + \sqrt{1 - (k+1) \cdot c^2 \cdot r^2}} \tag{9}$$

The effective surface definition is given by the `intersect` operation. Please note that in the case of spheroids, intersections beyond the hemisphere closest to the apex are valid.

intersect The intersection point I is found with the following operations sequence.

$$\begin{cases} f = c \cdot ({x_P}^2 + {y_P}^2) \\ g = n - c \cdot (l \cdot x_P + m \cdot y_P) \\ h = g^2 - c \cdot f \cdot (1 + k \cdot n^2) \end{cases} \tag{10}$$

In the case $h \leq 0$, we signal a ray error of absence of intersection. Else we continue,

$$t = \frac{f}{g + \text{sign}(n) \cdot \sqrt{h}} \tag{11}$$

$$I = \begin{bmatrix} x_P \\ y_P \\ 0 \end{bmatrix} + t \cdot \begin{bmatrix} l \\ m \\ n \end{bmatrix} \tag{12}$$

normal The normal vector at the current point is given by,

$$\overrightarrow{N} = \begin{bmatrix} c \cdot x \\ c \cdot y \\ c \cdot (k+1) \cdot z - 1 \end{bmatrix} \cdot \frac{\text{sign}(n)}{\sqrt{1 - 2c \cdot k \cdot z + c^2 \cdot (k+1) \cdot k \cdot z^2}} \tag{13}$$

4 Proofs and justification

Some implementation details require further justification and explanations. The mathematical derivations are written with sufficient details so that they may be followed without pen and paper.

4.1 Ray operations (rop)

Various formulae derivations and checks are presented.

4.1.1 Reflection formula derivation

We derive the reflection formula. See the illustration Fig. 3. Let:

- The surface normal $\overrightarrow{N}$ at the ray intersection.
- The incident ray unit vector $\overrightarrow{i}$.
- The reflected ray unit vector $\overrightarrow{r}$.
- θ_i the (unsigned) angle between $\overrightarrow{N}$ and $\overrightarrow{i}$.
- θ_r the (unsigned) angle between $\overrightarrow{N}$ and $\overrightarrow{r}$.

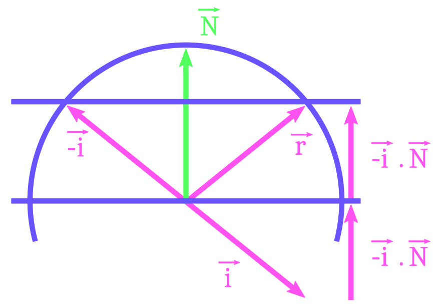

Figure 9: Illustration for the geometric construction of the reflect operation.

We want to compute $\overrightarrow{r}$ as a function of $\overrightarrow{i}$ and $\overrightarrow{N}$. We assume the laws of reflection, which essentially state that light behaves as a billiard ball in its geometrical interaction with the surface. We present several ways of looking at the problem. In our opinion, these are not so much derivations as different restatements of the laws of reflection.

Bounce derivation The effect of the light bouncing off the surface can be stated in the following fashion. The component of $\overrightarrow{i}$ colinear to $\overrightarrow{N}$ is inverted by the reflection. The other components of $\overrightarrow{i}$ stay the same. This immediately gives the tractable relation.

$$\overrightarrow{r} = \overrightarrow{i} - 2 \cdot (\overrightarrow{N} \cdot \overrightarrow{i}) \cdot \overrightarrow{N} \tag{14}$$

Geometric construction The ray reflection may be viewed as a geometric construction based on the laws of reflection (see [3] p.291 [2] p.335). We draw the geometric view of the problem on Fig. 9.

Algebraic derivation We produce a derivation similar to the one included in [3] (p.131).

- The reflected ray is in the plane spanned by $\overrightarrow{i}$ and $\overrightarrow{N}$ (*plane of incidence*).
- $\theta_i = \theta_r$
- Except in the degenerate case $\overrightarrow{i} = -\overrightarrow{N}$, $\overrightarrow{r} \neq -\overrightarrow{i}$. Ie the reflected ray does not retrace on the incident ray.

Translated algebraically, these statements suffice to determine a formula for $\overrightarrow{r}$. Let (α, β) both in $\mathbb{R}$. The reflected ray in the incidence plane must be expressed as:

$$\overrightarrow{r} = \alpha \cdot \overrightarrow{i} + \beta \cdot \overrightarrow{N} \tag{15}$$

We now add the remaining constraints on $\overrightarrow{r}$ in order to narrow down expressions for α and β.

$$\begin{aligned} & \theta_i = \theta_r \\ \implies & \cos(\theta_i) = \cos(\theta_r) \\ \implies & -\overrightarrow{i} \cdot \overrightarrow{N} = \overrightarrow{N} \cdot \overrightarrow{r} \end{aligned} \tag{16}$$

$$\begin{aligned} -\overrightarrow{i} \cdot \overrightarrow{N} &= \overrightarrow{N} \cdot (\alpha \cdot \overrightarrow{i} + \beta \cdot \overrightarrow{N}) \\ &= \alpha \cdot \overrightarrow{i} \cdot \overrightarrow{N} + \beta \cdot \overrightarrow{N} \cdot \overrightarrow{N} \\ &= \alpha \cdot \overrightarrow{N} \cdot \overrightarrow{i} + \beta \end{aligned} \tag{17}$$

Hence the following expression for β.

$$\beta = -(\alpha + 1) \cdot \overrightarrow{N} \cdot \overrightarrow{i} \tag{18}$$

Another constraint we exploit is that $\overrightarrow{r}$ is a unit vector.

$$\begin{aligned} & |\overrightarrow{r}| = 1 \\ \implies & \left|\alpha \cdot \overrightarrow{i} + \beta \cdot \overrightarrow{N}\right| = 1 \\ \implies & (\alpha \cdot i_x + \beta \cdot N_x)^2 + (\alpha \cdot i_y + \beta \cdot N_y)^2 + (\alpha \cdot i_z + \beta \cdot N_z)^2 = 1 \\ \implies & \alpha^2({i_x}^2 + {i_y}^2 + {i_z}^2) + \beta^2({N_x}^2 + {N_y}^2 + {N_z}^2) + \\ & 2 \cdot \alpha \cdot \beta(i_x \cdot N_x + i_y \cdot N_y + i_z \cdot N_z) = 1 \\ \implies & \alpha^2 + \beta^2 + 2 \cdot \alpha \cdot \beta \cdot \overrightarrow{N} \cdot \overrightarrow{i} = 1 \end{aligned} \tag{19}$$

Now we plug the expression for β (Eq. 18).

$$\begin{aligned} \implies & \alpha^2 + (\alpha+1)^2(\overrightarrow{N} \cdot \overrightarrow{i})^2 - 2 \cdot \alpha(\alpha+1) \cdot (\overrightarrow{N} \cdot \overrightarrow{i})^2 = 1 \\ \implies & \alpha^2(1 - 2(\overrightarrow{N} \cdot \overrightarrow{i})^2 + (\overrightarrow{N} \cdot \overrightarrow{i})^2) + \\ & \alpha(-2 \cdot (\overrightarrow{N} \cdot \overrightarrow{i})^2 + 2 \cdot (\overrightarrow{N} \cdot \overrightarrow{i})^2) + (\overrightarrow{N} \cdot \overrightarrow{i})^2 = 1 \\ \implies & \alpha^2 \cdot (1 - (\overrightarrow{N} \cdot \overrightarrow{i})^2) + (\overrightarrow{N} \cdot \overrightarrow{i})^2 = 1 \\ \implies & \alpha^2 = \frac{1 - (\overrightarrow{N} \cdot \overrightarrow{i})^2}{1 - (\overrightarrow{N} \cdot \overrightarrow{i})^2} \end{aligned} \tag{20}$$

The degenerate case $\overrightarrow{i} = -\overrightarrow{N}$, which leads to $1 - (\overrightarrow{N} \cdot \overrightarrow{i})^2 = 0$, is excluded. The answer in the degenerate case is $\overrightarrow{r} = -\overrightarrow{i}$. We continue the main derivation path with:

$$\alpha = \pm 1 \tag{21}$$

The case $\alpha = -1$ leads to $\beta = 0$ and $\overrightarrow{r} = -\overrightarrow{i}$. We excluded this case by hypothesis. The remaining answer is $\alpha = +1$.

Summarizing:

$$\begin{cases} \alpha = 1 \\ \beta = -2 \cdot \overrightarrow{N} \cdot \overrightarrow{i} \end{cases} \tag{22}$$

This leads us to the expression for $\overrightarrow{r}$.

$$\overrightarrow{r} = \overrightarrow{i} - 2 \cdot (\overrightarrow{N} \cdot \overrightarrow{i}) \cdot \overrightarrow{N} \tag{23}$$

□

4.1.2 Refraction

We derive the refraction formula vectorial form from the laws of refraction. Then we check the equivalence of Bec's formula with the derived refraction formula.

Algebraic derivation We work through an algebraic derivation of the vectorial expression for the refraction operation. Similar derivations may be found in [3] (p.288), originally by Whitted [12] and Heckbert [5].

The quantities involved in the problem are (illustrated on Fig. 10):

- n_1 the incident medium refraction index,
- n_2 the output medium refraction index,
- $\overrightarrow{i}$ the unit incident ray direction,
- $\overrightarrow{N}$ the unit surface normal vector,
- $\overrightarrow{t}$ the unit refracted ray direction,
- θ_i the acute angle between $-\overrightarrow{i}$ and $\overrightarrow{N}$,
- θ_t the angle between $-\overrightarrow{N}$ and $\overrightarrow{t}$.

Fig. 10

We take for granted a number of assumptions related to the law of refraction [16]:

- $\overrightarrow{t}$ is in the plane spanned by $\overrightarrow{i}$ and $\overrightarrow{N}$ (*plane of incidence*),
- Defining quadrants in the plane of incidence with respect to $\overrightarrow{N}$ and the origin, $\overrightarrow{t}$ is in the same quadrant as $\overrightarrow{i}$,
- $n_1 \cdot \sin(\theta_i) = n_2 \cdot \sin(\theta_t)$
- In the particular case where $-\overrightarrow{i} = \overrightarrow{N}$, the refracted ray is $\overrightarrow{t} = -\overrightarrow{N}$.

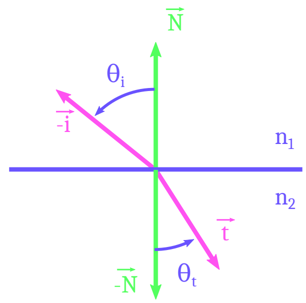

Figure 10: Quantities involved in the ray refraction operation.

We remind some relations and define some shorthands used for conciseness.

$$\begin{cases} -\overrightarrow{N} \cdot \overrightarrow{i} = \cos(\theta_i) = c_i \\ -\overrightarrow{N} \cdot \overrightarrow{t} = \cos(\theta_t) = c_t \\ \sin(\theta_i) = s_i \\ \sin(\theta_t) = s_t \\ \frac{n_1}{n_2} = n_r \end{cases} \tag{24}$$

Since $\overrightarrow{t}$ is in the incidence plane, it may be written as:

$$\overrightarrow{t} = \alpha \cdot \overrightarrow{i} + \beta \cdot \overrightarrow{N} \tag{25}$$

The other hypotheses provide the following constraints, which we will use in order to solve for α and β.

$$\begin{cases} n_1 \cdot s_i = n_2 \cdot s_t \\ \left|\overrightarrow{t}\right| = 1 \end{cases} \tag{26}$$

We may express a relation with the help of c_t:

$$\begin{aligned} c_t &= -\overrightarrow{N} \cdot \overrightarrow{t} \\ &= -\overrightarrow{N} \cdot (\alpha \cdot \overrightarrow{i} + \beta \cdot \overrightarrow{N}) \\ &= -\alpha \cdot \overrightarrow{N} \cdot \overrightarrow{i} - \beta \\ &= \alpha \cdot c_i - \beta \end{aligned} \tag{27}$$

Thus,

$$\beta = \alpha \cdot c_i - c_t \tag{28}$$

The $\overrightarrow{t}$ normalization constraint gives us another expression:

$$
\begin{aligned}
\left|\overrightarrow{t}\right| &= \left|\alpha \cdot \overrightarrow{i} + \beta \cdot \overrightarrow{N}\right| \\
&= (\alpha \cdot i_x + \beta \cdot N_x)^2 + (\alpha \cdot i_y + \beta \cdot N_y)^2 + (\alpha \cdot i_z + \beta \cdot N_z)^2 \\
&= \alpha^2 \cdot ({i_x}^2 + {i_y}^2 + {i_z}^2) + \beta^2 \cdot ({N_x}^2 + {N_y}^2 + {N_z}^2) + \\
&2 \cdot \alpha \cdot \beta \cdot (i_x \cdot N_x + i_y \cdot N_y + i_z \cdot N_z) \\
&= \alpha^2 + \beta^2 + 2 \cdot \alpha \cdot \beta \cdot \overrightarrow{i} \cdot \overrightarrow{N} \\
&= \alpha^2 + \beta^2 - 2 \cdot \alpha \cdot \beta \cdot c_i
\end{aligned}
\tag{29}
$$

Now we plug in the expression for β.

$$
\begin{aligned}
\left|\overrightarrow{t}\right| &= \alpha^2 + (\alpha \cdot c_i - c_t)^2 - 2 \cdot \alpha \cdot c_i \cdot (\alpha \cdot c_i - c_t) \\
&= \alpha^2 + \alpha^2 \cdot {c_i}^2 + {c_t}^2 - 2 \cdot \alpha \cdot c_i \cdot c_t - 2 \cdot \alpha^2 \cdot {c_i}^2 + 2 \cdot \alpha \cdot c_i \cdot c_t \\
&= \alpha^2 - \alpha^2 \cdot {c_i}^2 + {c_t}^2 \\
&= 1
\end{aligned}
\tag{30}
$$

Thus, making use of the Snell formula, and excluding the $c_i = 1$ case for which the refracted ray is just $-\overrightarrow{N}$,

$$
\begin{aligned}
\alpha^2 &= \frac{1 - {c_t}^2}{1 - {c_i}^2} = \frac{{s_t}^2}{{s_i}^2} = {n_r}^2 \\
&\iff \alpha = \pm n_r
\end{aligned}
\tag{31}
$$

By the $\overrightarrow{t}$ quadrant hypothesis, the component along $\overrightarrow{i}$ must be positive, hence $\alpha > 0 \implies \alpha = +n_r$.

We have now solved for the scalar magnitudes of $\overrightarrow{t}$.

$$
\begin{cases}
\alpha = n_r \\
\beta = n_r \cdot c_i - c_t
\end{cases}
\tag{32}
$$

And $\overrightarrow{t}$ may be expressed as:

$$
\overrightarrow{t} = n_r \cdot \overrightarrow{i} + (n_r \cdot c_i - c_t) \cdot \overrightarrow{N}
\tag{33}
$$

All that is left is to express c_t with respect to input problem quantities. The case of TIR appears when ${n_r}^2 \cdot {s_i}^2 > 1$ and is excluded from the remainder of the derivation.

$$
\begin{aligned}
c_t &= \sqrt{1 - {s_t}^2} \\
&= \sqrt{1 - {n_r}^2 \cdot {s_i}^2} \\
&= \sqrt{1 - {n_r}^2 \cdot (1 - {c_i}^2)}
\end{aligned}
\tag{34}
$$

Thus the expression for $\overrightarrow{t}$ with respect to the problem's input quantities is:

$$\overrightarrow{t} = n_r \cdot \overrightarrow{i} - \left(n_r \cdot \overrightarrow{N} \cdot \overrightarrow{i} + \sqrt{1 - {n_r}^2 \cdot \left(1 - \left(\overrightarrow{N} \cdot \overrightarrow{i} \right)^2 \right)} \right) \cdot \overrightarrow{N} \tag{35}$$

□

Bec's formula validation We perform a check of Bec's formula's (Eq. 5) validity with respect to the derived refraction formula (Eq. 35). Bec's formula is expressed as:

$$\overrightarrow{t} = n_r \cdot \overrightarrow{i} + (w - \sqrt{1 + c_{2m}}) \cdot \overrightarrow{N} \tag{36}$$

With:

$$\begin{cases} w & = -n_r \cdot \overrightarrow{i} \cdot \overrightarrow{N} \\ c_{2m} & = (w - n_r) \cdot (w + n_r) \end{cases} \tag{37}$$

While the formula from our derivation is expressed as:

$$\overrightarrow{t} = n_r \cdot \overrightarrow{i} - \left(n_r \cdot \overrightarrow{N} \cdot \overrightarrow{i} + \sqrt{1 - {n_r}^2 \cdot \left(1 - \left(\overrightarrow{N} \cdot \overrightarrow{i} \right)^2 \right)} \right) \cdot \overrightarrow{N} \tag{38}$$

We may re-express it as:

$$\overrightarrow{t} = n_r \cdot \overrightarrow{i} + \left(-n_r \cdot \overrightarrow{N} \cdot \overrightarrow{i} - \sqrt{1 + \gamma} \right) \cdot \overrightarrow{N} \tag{39}$$

We may check Bec's formula by proving that $\gamma = c_{2m}$.

$$\begin{aligned} c_{2m} &= (w - n_r) \cdot (w + n_r) = w^2 - {n_r}^2 \\ &= {n_r}^2 \cdot \left(\overrightarrow{i} \cdot \overrightarrow{N} \right)^2 - {n_r}^2 \\ &= -{n_r}^2 \cdot \left(1 - \left(\overrightarrow{i} \cdot \overrightarrow{N} \right)^2 \right) \\ &= \gamma \end{aligned} \tag{40}$$

Thus we find Bec's formula to be mathematically equivalent to the refraction formula we derived.

4.2 transfer

We derive the transfer operation formulae, and explain the assumptions around this operation.

4.2.1 transfer operation

We work through the formula for applying the transfer operation to a ray. We reuse the notations in the definition (Section 3.4). The transfer operation is the composition of a change of basis and an intersection with the new local plane.

Change of basis The initial ray $(P_1, \overrightarrow{V_1})$ is expressed in LCS1 coordinates. We express it in LCS2 with $(P_2, \overrightarrow{V_2})$. The LCS2 basis vectors are obtained by a rotation of the LCS1 basis vectors. We express everything in LCS1 coordinates up to the computation of $(P_2, \overrightarrow{V_2})$.

$$\begin{cases} \hat{x_2} = B \cdot \hat{x_1} \\ \hat{y_2} = B \cdot \hat{y_1} \\ \hat{z_2} = B \cdot \hat{z_1} \end{cases} \tag{41}$$

The origin A_2 of LCS2 is obtained by a translation of A_1 along the new, rotated, coordinates.

$$D = \begin{bmatrix} D_x \\ D_y \\ D_z \end{bmatrix} \tag{42}$$

$$\begin{aligned} A_2 &= A_1 + D_x \cdot \hat{x_2} + D_y \cdot \hat{y_2} + D_z \cdot \hat{z_2} \\ &= D_x \cdot \hat{x_2} + D_y \cdot \hat{y_2} + D_z \cdot \hat{z_2} \end{aligned} \tag{43}$$

Let us remember the columns of B are the basis vectors of LCS2 expressed in LCS1.

$$B = [\hat{x_2}, \hat{y_2}, \hat{z_2}] \tag{44}$$

Thus, we can simplify the expression of A_2.

$$A_2 = B \cdot D \tag{45}$$

The coordinates of P_2 are expressed thanks to a dot product with the basis vectors of LCS2.

$$\begin{cases} x_2 = (P - A_2) \cdot \hat{x_2} \\ y_2 = (P - A_2) \cdot \hat{y_2} \\ z_2 = (P - A_2) \cdot \hat{z_2} \end{cases} \tag{46}$$

Again, this can be viewed as the following matrix-vector multiplication.

$$\begin{aligned} P_2 &= B^\top \cdot (P - A_2) \\ &= B^\top \cdot (P - B \cdot D) \\ &= B^\top \cdot P - D \end{aligned} \tag{47}$$

Similarly, $\overrightarrow{V_2}$ may be expressed in LCS2.

$$\begin{cases} l_2 = \overrightarrow{V_1} \cdot \hat{x_2} \\ m_2 = \overrightarrow{V_1} \cdot \hat{y_2} \\ n_2 = \overrightarrow{V_1} \cdot \hat{z_2} \end{cases} \tag{48}$$

Which can be simplified as the following relation.

$$\overrightarrow{V_2} = B^\top \cdot \overrightarrow{V_1} \tag{49}$$

Intersection with the new local plane The ray characterized by $(P_2, \overrightarrow{V_2})$ is intersected with the LCS2 $z = 0$ plane. Every operation happens in LCS2 coordinates. The intersection condition between the ray parametrized by t and the plane is expressed as follows.

$$z_2 + t \cdot n_2 = 0 \tag{50}$$

We treat the case $n_2 = 0$ as an error case. It corresponds to the ray being parallel to the plane.

$$t = -\frac{z_2}{n_2} \tag{51}$$

We simply propagate point P_2 up to the intersection P_3.

$$P_3 = \begin{bmatrix} x_2 + t \cdot l_2 \\ y_2 + t \cdot m_2 \\ 0 \end{bmatrix} \tag{52}$$

The ray vector $\overrightarrow{V_2}$ does not undergo any operation, hence $\overrightarrow{V_3} = \overrightarrow{V_2}$.

4.2.2 Ray conventions

The transfer objects take rays defined by a point located anywhere and propagate them to rays resting on the local plane of the next surface's LCS. Surfaces operate under the assumption that the input rays have a $z = 0$ coordinate, and may leave the output rays with any position. These assumptions are put in place for the management of complex ray/surface intersections (which are not implemented in the present Poaky version).

4.3 General expression for a shape normal vector

Given either an explicit formula ($z = f(x, y)$), or an implicit formula ($F(x, y, z) = 0$) representing a surface, we show how to compute the unit normal vector at a given point. Additionally we orient the normal vector in the opposite half-plane to an incident ray with a vector component n.

From an explicit formula Given a shape in its LCS defined by an equation $z = f(x, y)$, we can compute the unit normal vector at point (x, y) using (Eq. 53) [10].

$$\overrightarrow{N} = \frac{\text{sign}(n)}{\sqrt{1 + \left(\frac{\partial f}{\partial x}(x,y)\right)^2 + \left(\frac{\partial f}{\partial y}(x,y)\right)^2}} \cdot \begin{bmatrix} \frac{\partial f}{\partial x}(x,y) \\ \frac{\partial f}{\partial y}(x,y) \\ -1 \end{bmatrix} \tag{53}$$

From an implicit formula A normal vector may also be defined using an implicit surface definition of the form $F(x, y, z) = 0$ ([13], [11] section 4.6). A normal vector at a point on the surface is simply given by the gradient ∇F evaluated at this point.

$$\nabla F = \begin{bmatrix} \frac{\partial F}{\partial x} \\ \frac{\partial F}{\partial y} \\ \frac{\partial F}{\partial z} \end{bmatrix} \tag{54}$$

Thus the unit normal vector oriented opposite to n is given by:

$$\overrightarrow{N} = -\frac{\nabla F(x,y,z)}{|\nabla F(x,y,z)|} \cdot \text{sign}(n) \cdot \text{sign}\left(\frac{\partial F}{\partial z}(x,y,z)\right) \tag{55}$$

4.4 shape: standard

The `standard` shape is partly defined with the formula in (Eq. 9). This formula may be found in [11] (section 4.7) and in [4][1]. We explore its link with quadrics and try to provide a rationale for how it was defined from quadrics.

Note we freely convert between R and c, and r and (x, y) in the derivations.

4.4.1 Range of definition

The range in r for which the `standard` shape is defined depends on k. The shape is defined when,

$$\begin{aligned} 1 - (k+1) \cdot c^2 \cdot r^2 &\geq 0 \\ \iff (k+1) \cdot c^2 \cdot r^2 &\leq 1 \end{aligned} \tag{56}$$

For $k \leq -1$, the validity condition is met for $r \in \mathbb{R}$. For $k > -1$, the condition is met when,

$$\begin{aligned} r^2 &\leq \frac{1}{c^2 \cdot (k+1)} \\ \iff r &\leq \frac{|R|}{\sqrt{k+1}} \end{aligned} \tag{57}$$

[1] Note the use of this kind of surface in optics dates back to at least Kepler in this present form and all the way back to the Greeks for the general idea.

4.4.2 Link with quadrics

We exhibit the link between the `standard` surface and the more general quadrics expressions.

General quadrics form An expression for general quadrics can be written as [14].

$$Ax^2 + By^2 + Cz^2 + Dxy + Eyz + Fxz + Gx + Hy + Iz + J = 0 \tag{58}$$

With,

- $A, B, C, D, E, F, G, H, I, J$ all in $\mathbb{R}$,
- At least one of A, B or C is non-zero.

We can easily see to which family of quadrics the points defined by the `standard` altitude formula *belong to*. We start with the altitude expression and re-express it in the general quadrics form.

$$\begin{aligned}
z &= \frac{c \cdot r^2}{1 + \sqrt{1 - (k+1) \cdot c^2 \cdot r^2}} \\
\implies & z + z \cdot \sqrt{1 - (k+1) \cdot c^2 \cdot r^2} = c \cdot r^2 \\
\implies & z - c \cdot r^2 = -z \cdot \sqrt{1 - (k+1) \cdot c^2 \cdot r^2} \\
\implies & \frac{c \cdot r^2}{z} - 1 = \sqrt{1 - (k+1) \cdot c^2 \cdot r^2} \\
\implies & \frac{c^2 \cdot r^4}{z^2} + 1 - \frac{2 \cdot c \cdot r^2}{z} = 1 - (k+1) \cdot c^2 \cdot r^2 \\
\implies & \frac{c \cdot r^2}{z^2} - \frac{2}{z} = -(k+1) \cdot c \\
\implies & (k+1) \cdot z^2 - \frac{2}{c} \cdot z + r^2 = 0 \\
\implies & x^2 + y^2 + (k+1) \cdot z^2 - 2R \cdot z = 0
\end{aligned} \tag{59}$$

We readily see this quadric can be defined from the general quadric expression by setting:

$$\begin{cases} A = B = 1 \\ C = k + 1 \\ I = -2R \\ D = E = F = G = H = J = 0 \end{cases} \tag{60}$$

Normal form We operate the suitable change of coordinates in order to reduce the quadric expression to a so-called *normal form* ([14], [7] p.384) and better identify the types of quadrics we are working with.

First, we treat the case $k = -1$. The quadric takes the following form.

$$\begin{aligned} & x^2 + y^2 - 2R \cdot z = 0 \\ \iff & z = \frac{c}{2} \cdot x^2 + \frac{c}{2} \cdot y^2 \end{aligned} \tag{61}$$

This form is a *circular paraboloid.*

Then, we treat the case $k \neq -1$. Let $\varepsilon = k + 1$.

$$\begin{aligned} & x^2 + y^2 + \varepsilon \cdot z^2 - 2R \cdot z = 0 \\ \iff & x^2 + y^2 + \varepsilon \cdot \left(z^2 - \frac{2R}{\varepsilon} \cdot z \right) = 0 \\ \iff & x^2 + y^2 + \varepsilon \cdot \left(z - \frac{R}{\varepsilon} \right)^2 - \frac{R^2}{\varepsilon} = 0 \end{aligned} \tag{62}$$

We operate the change of coordinate $z' = z - \frac{R}{\varepsilon}$.

$$\begin{aligned} & x^2 + y^2 + \varepsilon \cdot z'^2 - \frac{R^2}{\varepsilon} = 0 \\ \iff & x^2 \cdot \frac{\varepsilon}{R^2} + y^2 \cdot \frac{\varepsilon}{R^2} + z'^2 \cdot \frac{\varepsilon^2}{R^2} - 1 = 0 \\ \iff & x^2 \cdot \frac{\text{sign}(\varepsilon) \cdot |\varepsilon|}{R^2} + y^2 \cdot \frac{\text{sign}(\varepsilon) \cdot |\varepsilon|}{R^2} + z'^2 \cdot \frac{\varepsilon^2}{R^2} - 1 = 0 \\ \iff & x^2 \cdot \frac{|\varepsilon|}{R^2} + y^2 \cdot \frac{|\varepsilon|}{R^2} + z'^2 \cdot \text{sign}(\varepsilon) \cdot \frac{\varepsilon^2}{R^2} - \text{sign}(\varepsilon) = 0 \end{aligned} \tag{63}$$

Let,

$$\begin{cases} a^2 = \frac{R^2}{|\varepsilon|} \\ b^2 = \frac{R^2}{\varepsilon^2} \\ \epsilon = \text{sign}(\varepsilon) \end{cases} \tag{64}$$

A readable *normal form* is:

$$\frac{x^2}{a^2} + \frac{y^2}{a^2} + \epsilon \cdot \frac{z'^2}{b^2} - \epsilon = 0 \tag{65}$$

The case $k < -1$ gives the form,

$$\frac{x^2}{a^2} + \frac{y^2}{a^2} - \frac{z'^2}{b^2} = -1 \tag{66}$$

which is a *hyperboloid of revolution of two sheets.*

The case $k > -1$ gives the form,

$$\frac{x^2}{a^2} + \frac{y^2}{a^2} + \frac{z'^2}{b^2} = 1 \tag{67}$$

which is a *spheroid*. We note that for $k = 0$, we have $\varepsilon = 1$ and thus $a = b$, which describes a *sphere*.

4.4.3 Rationale for defining the shape from quadrics

We explore the rationale which leads from the general quadric (with potentially two sheets) to the `standard` altitude definition.

The general quadric is defined as,

$$x^2 + y^2 + (k+1) \cdot z^2 - 2R \cdot z = 0 \tag{68}$$

We solve for z.

In the $k = -1$ case, the quadric is a circular paraboloid and has a single sheet, thus the solution is unambiguous.

$$z = \frac{c \cdot r^2}{2} \tag{69}$$

In the $k \neq -1$ cases, the solution is given by solving the quadratic equation.

$$\begin{cases} \Delta = 4R^2 - 4(k+1) \cdot r^2 \\ z = \frac{2R \pm \sqrt{\Delta}}{2(k+1)} \end{cases} \tag{70}$$

We have two solutions given by,

$$z = \frac{R \pm \sqrt{R^2 - (k+1) \cdot r^2}}{k+1} \tag{71}$$

These define the two sheets of the spheroid or the hyperboloid of revolution of two sheets. An explicit altitude formula may only contain a single sheet. We choose to have $z(r = 0) = 0$.

$$\begin{aligned} & z(r=0) = 0 \\ \iff & \frac{R \pm \sqrt{R^2}}{k+1} = 0 \\ \iff & R \pm |R| = 0 \end{aligned} \tag{72}$$

We have to distinguish cases based on the sign of R.

- $R \geq 0$: we choose the solution with minus sign in order to constrain $z(r = 0) = 0$,
- $R < 0$: we choose the solution with plus sign.

Therefore, we build our solution as:

$$
\begin{aligned}
& z = \frac{R - \operatorname{sign}(R) \cdot \sqrt{R^2 - (k+1) \cdot r^2}}{k+1} \\
\iff & (k+1) \cdot z = R - \operatorname{sign}(R) \cdot \sqrt{R^2 - (k+1) \cdot r^2} \\
\iff & (k+1) \cdot z \cdot \left(R + \operatorname{sign}(R) \cdot \sqrt{R^2 - (k+1) \cdot r^2}\right) = R^2 - (R^2 - (k+1) \cdot r^2) \\
\iff & z \cdot \left(R + \operatorname{sign}(R) \cdot \sqrt{R^2 - (k+1) \cdot r^2}\right) = r^2 \\
\iff & z = \frac{r^2}{R + \operatorname{sign}(R) \cdot \sqrt{R^2 - (k+1) \cdot r^2}} \\
\iff & z = \frac{r^2}{R + \operatorname{sign}(R) \cdot |R| \cdot \sqrt{1 - (k+1) \cdot r^2 \cdot c^2}} \\
\iff & z = \frac{c \cdot r^2}{1 + \sqrt{1 - (k+1) \cdot r^2 \cdot c^2}}
\end{aligned}
\tag{73}
$$

Which is the `standard` altitude formula in use. We see the $k = -1$ case is still compatible with this formula.

4.4.4 Intersection formula

The intersection of a ray with a `standard` shape may be expressed in closed-form. Finding it involves substituting the usual ray equation in the `standard` surface expression.

The ray equation being, as usual,

$$
\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} x_P \\ y_P \\ 0 \end{bmatrix} + t \cdot \begin{bmatrix} l \\ m \\ n \end{bmatrix} \tag{74}
$$

and the quadric implicit equation,

$$
x^2 + y^2 + (k+1) \cdot z^2 - 2R \cdot z = 0 \tag{75}
$$

Finding the expression for solutions is easy, but finding an expression which is numerically well behaved is more involved. Thankfully, Welford found such an expression [11], which we use with a slight modification.

$$
\begin{cases}
f = c \cdot ({x_P}^2 + {y_P}^2) \\
g = n - c \cdot (l \cdot x_P + m \cdot y_P) \\
t = \frac{f}{g + \operatorname{sign}(n) \cdot \sqrt{g^2 - c \cdot f \cdot (1 + k \cdot n^2)}}
\end{cases}
\tag{76}
$$

Our minor modification is the addition of $\operatorname{sign}(n)$ which allows choosing the right root regardless of ray orientation.

The case where $g^2 - c \cdot f \cdot (1 + k \cdot n^2) \leq 0$ corresponds to an error case of absence of intersection or ray tangency.

In non-error cases, the intersection point is given as usual with:

$$I = P + t \cdot V \tag{77}$$

The hand validation being laborious, we validated the intersection formula with a symbolic math tool.

4.4.5 Normal vector

The normal vector at some point on the surface is found by computing the gradient of the implicit quadric expression, this is the method followed by Welford [11].

Let $\varepsilon = k + 1$, the implicit quadric expression is:

$$F(x, y, z) = \frac{c}{2} \cdot (x^2 + y^2 + \varepsilon \cdot z^2) - z = 0 \tag{78}$$

We compute the gradient:

$$\begin{cases} \frac{\partial F}{\partial x} = c \cdot x \\ \frac{\partial F}{\partial y} = c \cdot y \\ \frac{\partial F}{\partial z} = c \cdot \varepsilon \cdot z - 1 \end{cases} \tag{79}$$

The norm of the unit vector is then,

$$\left|\overrightarrow{N}\right| = \sqrt{c^2 \cdot (x^2 + y^2) + c^2 \cdot \varepsilon^2 \cdot z^2 + 1 - 2c \cdot \varepsilon \cdot z} \tag{80}$$

We know from the quadric expression that,

$$x^2 + y^2 = \frac{2}{c} \cdot z - \varepsilon \cdot z^2 \tag{81}$$

Hence,

$$\begin{aligned} \left|\overrightarrow{N}\right| &= \sqrt{c^2 \cdot \left(\frac{2}{c} \cdot z - \varepsilon \cdot z^2\right) + c^2 \cdot \varepsilon^2 \cdot z^2 + 1 - 2c \cdot \varepsilon \cdot z} \\ &= \sqrt{1 - 2c \cdot (\varepsilon - 1) \cdot z + c^2 \cdot \varepsilon \cdot (\varepsilon - 1) \cdot z^2} \end{aligned} \tag{82}$$

The unit vector with correct orientation is then:

$$\overrightarrow{N} = -\begin{bmatrix} c \cdot x \\ c \cdot y \\ c \cdot \varepsilon \cdot z - 1 \end{bmatrix} \cdot \frac{\operatorname{sign}(n) \cdot \operatorname{sign}(c \cdot \varepsilon \cdot z - 1)}{\left|\overrightarrow{N}\right|} \tag{83}$$

Let $D = \varepsilon \cdot c \cdot z - 1$, we can simplify the expression of $\overrightarrow{N}$ if we know this sign. Let's determine it.

Looking at the implicit altitude formula (Eq. 9), we see z and c are of the same sign, so we can write,

$$D = \varepsilon \cdot |c| \cdot |z| - 1 \tag{84}$$

We can express the range of validity of ε in the definition of the `standard` shape.

$$\begin{aligned} & 1 - (k+1) \cdot c^2 \cdot r^2 \geq 0 \\ \iff & 1 - \varepsilon \cdot c^2 \cdot r^2 \geq 0 \\ \iff & \varepsilon \leq \frac{1}{c^2 \cdot r^2} \end{aligned} \tag{85}$$

Plugging this limitation into the expression of D, we get,

$$\begin{aligned} D &\leq \frac{1}{c^2 \cdot r^2} \cdot |c| \cdot |z| - 1 \\ &\leq \frac{1}{|c|} \cdot \frac{|z|}{r^2} - 1 \end{aligned} \tag{86}$$

By the surface definition (Eq. 9), we have,

$$|z| \leq |c| \cdot r^2 \tag{87}$$

Thus,

$$\begin{aligned} D &\leq \frac{|c| \cdot r^2}{|c| \cdot r^2} - 1 \\ &\leq 0 \end{aligned} \tag{88}$$

The simplified expression for the unit normal vector opposite to the incoming ray is then,

$$\overrightarrow{N} = \begin{bmatrix} c \cdot x \\ c \cdot y \\ c \cdot \varepsilon \cdot z - 1 \end{bmatrix} \cdot \frac{\operatorname{sign}(n)}{\sqrt{1 - 2c \cdot (\varepsilon - 1) \cdot z + c^2 \cdot \varepsilon \cdot (\varepsilon - 1) \cdot z^2}} \tag{89}$$

5 Tests and benchmarks

We document the rationale for tests performed on the components of the software. We also detail a representative performance report.

5.1 Tests

Our rationale for testing is the following.

- Every function must be called at least once.
- Every eventual error case and condition must be reached at least once.
- The correctness is assessed on a few samples, through the means of either:
  - Pinning the function under test to a reference function which is very clearly expressed with respect to the documentation.
  - Pinning the function result to an externally computed result.

As the development proceeds, we add cases arising from fixed bugs in order to prevent regression.

5.2 Performance report

We provide benchmark results for the operations we wrote. The goal is to indicate what order of magnitude of performance can be expected.

5.2.1 Hypotheses and limitations

We run each function on sizeable arrays of rays, one ray at a time. The functions are evaluated independently. This may be unrepresentative of a real raytracing computation flow where complex operations may be chained on only a few rays. The operations are run on CPU on a single thread.

5.2.2 Hardware

We ran the benchmark on a Void Linux desktop computer, with kernel version 6.0.11_1. We took no particular steps to configure the OS. The computer has a AMD Athlon 3000G CPU and DDR4 RAM clocked at 2667 MT/s. We compiled the benchmark with gcc 10.2.1 with the flags `-march=native -O2`.

5.2.3 Results

The benchmark results are listed in Tab. 2.

6 Notations

6.1 Multiplication and dot product

The symbol $\cdot$ may refer to:

- Scalar multiplication
- Scalar/Vector or Scalar/Matrix multiplication
- Vector dot product

Operation	Mean timing per op (ns)	Standard deviation (ns)
reflect	2.2	0.3
refract	5.6	0.6
transfer	5.6	0.9
standard.intersect	5.6	0.6
standard.normal	5.9	0.9

Table 2: Benchmark results

6.2 Absolute value and 2-norm

The unary operator notation $|\cdot|$ may refer to:

- Absolute value of a scalar
- 2-norm (or *Euclidian* norm) of a vector

6.3 Sign function

We use a sign function, defined as in (Eq. 90).

$$\text{sign}(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ -1 & \text{otherwise} \end{cases} \tag{90}$$

7 Glossary

Some technical terms are recurring. They are specific either to this document or to optical design and thus must be defined.

7.1 Local plane

The *local plane* is the $z = 0$ plane in a LCS (Section 2.2).

7.2 Concave or Convex

We must disambiguate what we mean by *convex* and *concave*, as other conventions are used in other fields. These adjectives apply to surface shapes, often in the context of raytracing as seen by incoming rays. It refers to the sign of the curvature of the shape. The two terms are illustrated on Fig. 11. A surface shape may be described as convex/concave either globally or locally at the ray intersection point.

Acronyms

LCS Local Coordinate System
TIR Total Internal Reflection

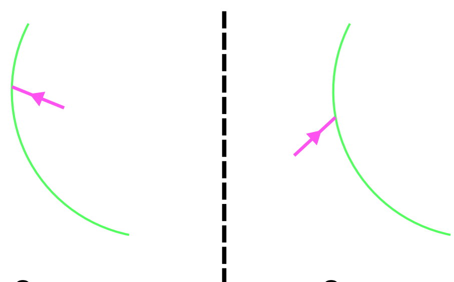

Figure 11: Illustration of a ray incoming onto a concave surface (left), and on a convex surface (right).

References

[1] Xavier Bec. *Faster Refraction Formula, and Transmission Color Filtering.* Online – Ray Tracing News Vol.10 number 1. 1997. URL: https://www.realtimerendering.com/resources/RTNews/html//rtnv10n1.html.

[2] Peter Comninos. *Mathematical and computer programming techniques for computer graphics.* Springer Science & Business Media, 2010.

[3] Andrew S. Glassner. *An Introduction to Ray Tracing.* Academic Press Ltd., 1989. URL: https://www.realtimerendering.com/blog/an-introduction-to-ray-tracing-is-now-free-for-download/.

[4] Alan W. Greynolds. "Superconic and subconic surfaces in optical design". In: *International Optical Design Conference Technical Digest.* 2002.

[5] Paul S. Heckbert and Pat Hanrahan. "Beam tracing polygonal objects". In: *Proceedings of the 11th annual conference on Computer graphics and interactive techniques - SIGGRAPH '84.* ACM Press, 1984. DOI: 10.1145/800031.808588.

[6] Adam Marrs, Peter Shirley, and Ingo Wald, eds. *Ray Tracing Gems II.* Apress, 2021. DOI: 10.1007/978-1-4842-7185-8.

[7] Venit Stewart, Wayne Bishop, and Brown Jason. *Elementary linear algebra.* First Canadian Edition. Nelson College Indigenous, 2008.

[8] Houllier Thomas. "Freeform imaging optical systems". PhD thesis. Univ Jean Monnet – Saint-Etienne, March 2021. DOI: http://dx.doi.org/10.13140/RG.2.2.12722.27841.

[9] Eric W. Weisstein. *Direction cosine.* From MathWorld–A Wolfram Web Resource. [Online; accessed 22-Sep-2022]. URL: https://mathworld.wolfram.com/DirectionCosine.html.

[10] Eric W. Weisstein. *Normal vector.* From MathWorld–A Wolfram Web Resource. [Online; accessed 15-Oct-2022]. URL: https://mathworld.wolfram.com/NormalVector.html.

[11] Walter Thompson Welford. *Aberrations of optical systems.* Ed. by E. R. Pike, B. E. A. Saleh, and W. T. Welford. Adam Hilger, 1986.

[12] Turner Whitted. "An improved illumination model for shaded display". In: *ACM SIGGRAPH 2005 Courses on - SIGGRAPH '05.* ACM Press, 2005. DOI: 10.1145/1198555.1198743.

[13] Wikipedia contributors. *Normal (geometry) — Wikipedia, The Free Encyclopedia.* [Online; accessed 4-December-2022]. 2022. URL: https://en.wikipedia.org/w/index.php?title=Normal_(geometry)&oldid=1115796469.

[14] Wikipedia contributors. *Quadric — Wikipedia, The Free Encyclopedia.* [Online; accessed 1-December-2022]. 2022. URL: https://en.wikipedia.org/w/index.php?title=Quadric&oldid=1124841009.

[15] Wikipedia contributors. *Rotation matrix — Wikipedia, The Free Encyclopedia.* [Online; accessed 26-November-2022]. 2022. URL: https://en.wikipedia.org/w/index.php?title=Rotation_matrix&oldid=1122062121.

[16] Wikipedia contributors. *Snell's law — Wikipedia, The Free Encyclopedia.* https://en.wikipedia.org/w/index.php?title=Snell%27s_law&oldid=1117944194. [Online; accessed 3-November-2022]. 2022.

[17] Wikipedia contributors. *Specular reflection — Wikipedia, The Free Encyclopedia.* https://en.wikipedia.org/w/index.php?title=Specular_reflection&oldid=1116077720. [Online; accessed 17-October-2022]. 2022.