An Importance Sampling Method for arbitrary BRDFs used in Global Illumination Applications

Rosana Montes

Dpto. Lenguajes y Sistemas Informáticos
E.T.S. Ingeniería Informática y de Telecomunicación
University of Granada

Advisor: Carlos Ureña

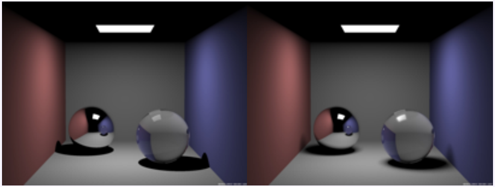
Talk outline

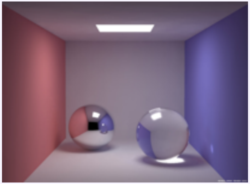
1. Motivation
2. Realistic image synthesis
3. The Monte Carlo method
4. Sampling the BRDF
5. Our adaptive generic sampling method
6. Results for our solution
7. Contributions of this work and future work

Contents

The aim of image synthesis

Produce realistic images from a symbolic description using Local and Global Illumination Methods





Images by H. W. Jensen, 1996

Applications of Global Illumination

- The computer graphics industry demand more realistic computer generated images
- The main goal is to obtain images undistinguishable from the reality

Applications of Global Illumination

- The computer graphics industry demand more realistic computer generated images
- The main goal is to obtain images undistinguishable from the reality

Applications of Global Illumination

- The computer graphics industry demand more realistic computer generated images
- The main goal is to obtain images undistinguishable from the reality



What we see is ... what we describe

- The virtual world is a list description of: lights, objects and the viewer
- The appearance of an object is described by the BRDF
- Some realistic BRDF models are difficult to use in rendering systems

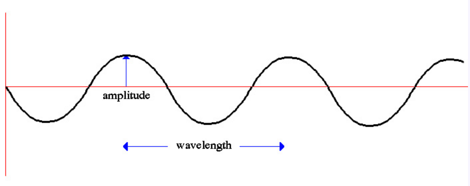
Our main objectives

1. Facilitate the use of 3D geometry models from common graphics formats
2. Edit the object properties and extend them with useful information
3. Allow the use of complex BRDF models in a photorealistic rendering system
    - We have developed tools for the understanding and the use of the BRDF models' parameters:
        - 2D and 3D plot utilities
        - a GPU based BRDF viewer
    - We have developed an efficient sampling algorithm that is independent of how a BRDF is modelled, and is suitable for a Monte Carlo Global Illumination system

Contents

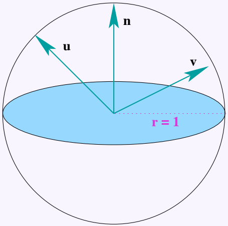
The light model in Global Illumination

The dual nature of light: wave and photon.



A numerical mathematical model represents the transport of the light.

Notation

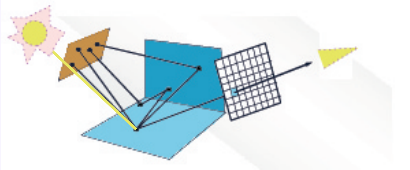


- A vector $\mathbf{w} \in \Omega$ could be expressed as polars $(\theta_\mathbf{w}, \phi_\mathbf{w})$ or cartesian $(x_\mathbf{w}, y_\mathbf{w}, z_\mathbf{w})$.
- In the hemisphere the solid angle measure is noted as σ.
- Another useful measure is the projected solid angle σ_p.

Radiance

L [Watts / ($meter^2$ steradian)]: Power per (projected) surface area per solid angle.

$$dL(x,\mathbf{u}) = \frac{d^2\Phi(x,\mathbf{u})}{dA(x)d\sigma_p(\mathbf{u})}$$



Bidirectional Reflectance Distribution Function

The BRDF function

A function that represents the surface reflectance, given two directions $\mathbf{u}, \mathbf{v} \in \Omega$ and has values between 0 and ∞.

The BRDF

- Describes the relation between the incoming and outgoing radiances at a given point on the surface.
- It is expressed in *steradians*.
- Must be simmetric and energy conservative, to be *physically plausible*

$$f_r(\mathbf{u},\mathbf{v}) = f_r(\mathbf{v},\mathbf{u})$$

$$\int_\Omega f_r(\mathbf{u},\mathbf{v}) cos(\mathbf{v}) d\sigma(\mathbf{v}) \leq 1$$

- There are many reflectance models in Computer Graphics.

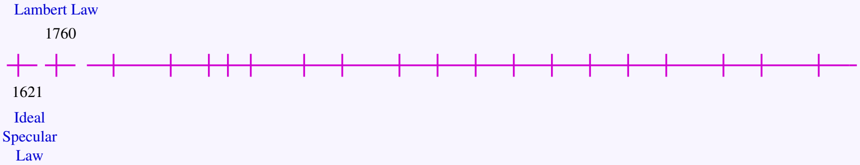
BRDF Classification

There are many reflectance models in Computer Graphics.

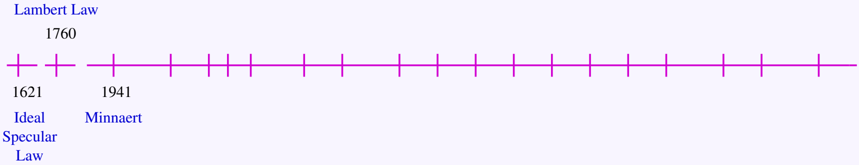
- **Physical** Based on optical properties and the physics of light transport simulation. Parameters are not intuitive or easy to set.
- **Approximated** Simple formulation easy to compute with few parameters. Optimal for computer graphics algorithm. Fast with visually acceptable results.
- **Experimental** Represent acquired data with an analytical formulation.

Time line of some representative *reflectance models*

Time line of some representative *reflectance models*

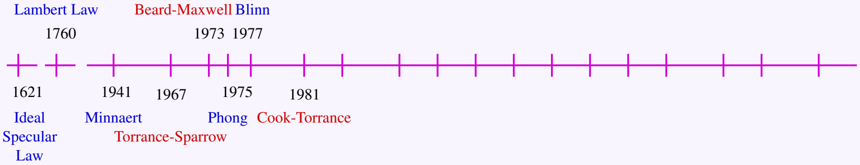


Time line of some representative *reflectance models*

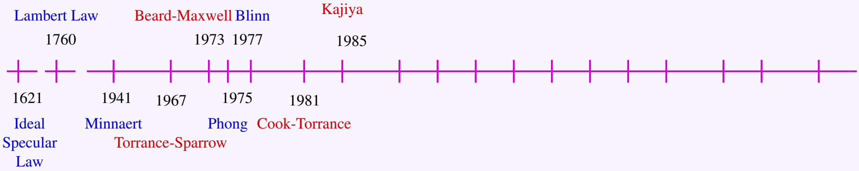


Time line of some representative *reflectance models*

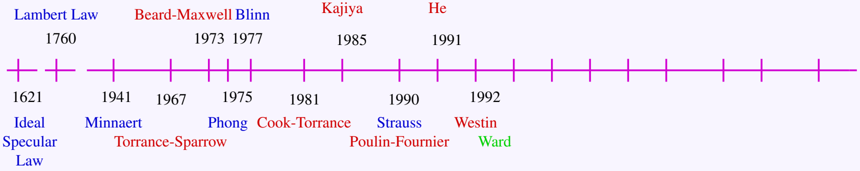
Time line of some representative *reflectance models*



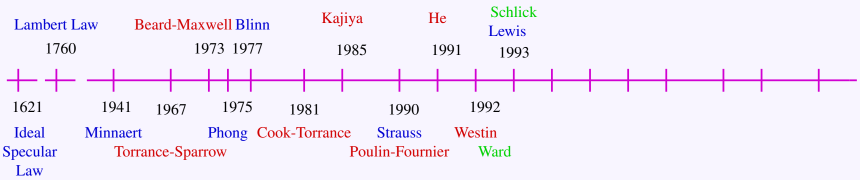
Time line of some representative *reflectance models*



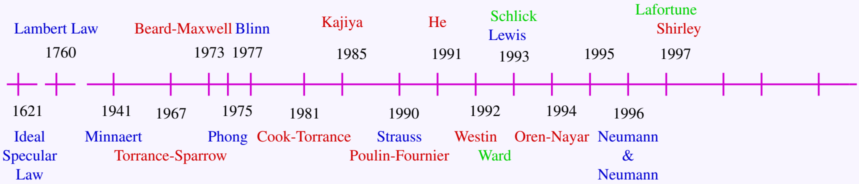
Time line of some representative *reflectance models*



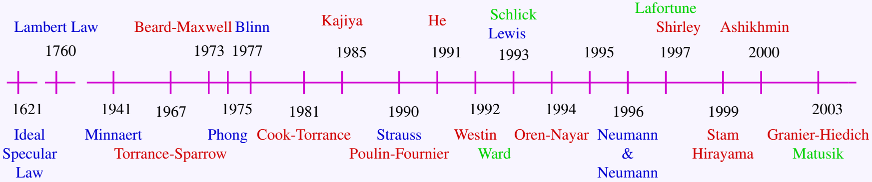
Time line of some representative *reflectance models*



Time line of some representative *reflectance models*



Time line of some representative *reflectance models*



The reflection equation

From the definition of a BRDF and the relationship between irradiance and radiance, we get:

$$f_r(\mathbf{u},\mathbf{v}) = \frac{dL_r(\mathbf{u})}{dE(\mathbf{v})} \qquad dE(\mathbf{v}) = L_i(\mathbf{v})\, cos(\mathbf{v})\, d\sigma\ (\mathbf{v})$$

This means to get the exitant radiance in a direction $\mathbf{u}$, we need to integrate the incident radiance, times the BRDF, times the cosine of the angle with the normal, over all incoming directions in the hemisphere around the surface normal.

The reflection equation

From the definition of a BRDF and the relationship between irradiance and radiance, we get:

$$L_r(\mathbf{u}) = \int_\Omega f_r(\mathbf{u}, \mathbf{v})\, L_i(\mathbf{v})\, cos(\mathbf{v})\, d\sigma(\mathbf{v})$$

This means to get the exitant radiance in a direction $\mathbf{u}$, we need to integrate the incident radiance, times the BRDF, times the cosine of the angle with the normal, over all incoming directions in the hemisphere around the surface normal.

Global illumination algorithms

A common classification distinguish *what* is calculated and *how* it is done.

Global illumination algorithms

A common classification distinguish *what* is calculated and *how* it is done.

- Object based vs **Image based** approach

Global illumination algorithms

A common classification distinguish *what* is calculated and *how* it is done.

- Object based vs **Image based** approach
- Deterministic vs **Stochastic**

Global illumination algorithms

A common classification distinguish *what* is calculated and *how* it is done.

- Object based vs **Image based** approach
- Deterministic vs **Stochastic**

Algorithms for radiance computation:
Photon Mapping

Global illumination algorithms

A common classification distinguish *what* is calculated and *how* it is done.

- Object based vs **Image based** approach
- Deterministic vs **Stochastic**

Algorithms for radiance computation:

Recursive Ray Tracing

Global illumination algorithms

A common classification distinguish *what* is calculated and *how* it is done.

- Object based vs **Image based** approach
- Deterministic vs **Stochastic**

Algorithms for radiance computation:

Path Tracing

Contents

Starting with N random samples (namely $X_1, \ldots, X_N$) we can build a random variable X whose mean is the integral we want to compute:

$$\int_S f(x) d\mu(x)$$

Naive Monte-Carlo

$$F_N \equiv F_N(X_1, \ldots, X_N) \;=\; \frac{1}{N}\sum_{i=1}^{N} f(X_i) \;\; with \;\; X_i \sim U$$

Starting with N random samples (namely $X_1, \ldots, X_N$) we can build a random variable X whose mean is the integral we want to compute:

$$\int_S f(x) d\mu(x)$$

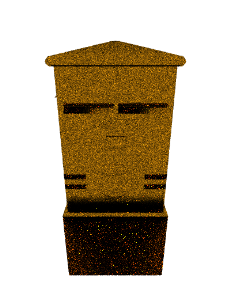
Naive Monte-Carlo

$$F_N \equiv F_N(X_1, \ldots, X_N) = \frac{1}{N} \sum_{i=1}^{N} f(X_i) \;\; with \;\; X_i \sim U$$

Uniform sampling produces high error

Characteristics

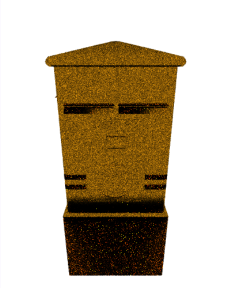
- As $N \to \infty$ the error of this estimation disminish.
- Slow convergence $O(1/\sqrt{N})$.
- Get half of the error implies using 4 times N.



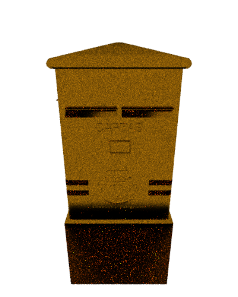
5^2 uniform samples

Characteristics

- As $N \to \infty$ the error of this estimation disminish.
- Slow convergence O$(1/\sqrt{N})$.
- Get half of the error implies using 4 times N.



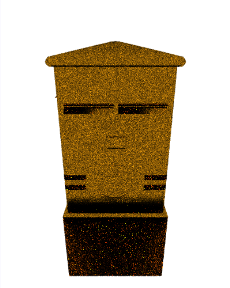
5^2 uniform samples



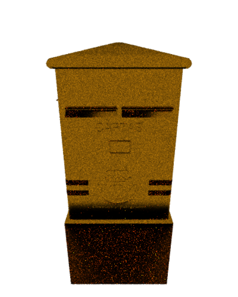
10^2 uniform samples

Characteristics

- As $N \to \infty$ the error of this estimation disminish.
- Slow convergence O$(1/\sqrt{N})$.
- Get half of the error implies using 4 times N.



5^2 uniform samples



10^2 uniform samples

10^2 adaptive samples

Variance

The error produced in the estimation. To diminish it:

- more samples $\Longrightarrow$ more computation
- apply a variance reduction techniques.

Variance

The error produced in the estimation. To diminish it:

- more samples $\Longrightarrow$ more computation
- apply a variance reduction techniques.

- Russian Roulette
- Domain Stratification
- **Importance Sampling**
- Multiple Importance Sampling
- Resampling Importance Sampling
- and many more...

Variance

The error produced in the estimation. To diminish it:

- more samples $\Longrightarrow$ more computation
- apply a variance reduction techniques.

A Monte-Carlo estimator with Importance Sampling

$$F_N \equiv F_N(Y_1, \ldots, Y_N) \; = \; \frac{1}{N} \sum_{i=1}^{N} \frac{f(Y_i)}{p(Y_i)} \;\; \textit{with} \;\; Y_i \sim p$$

The Path Tracing Algorithm We want to approximate the integral of the reflectance equation.

$$L_r(\mathbf{u}) = \int_\Omega f_r(\mathbf{u},\mathbf{v})\,L_i(\mathbf{v})\,cos(\mathbf{v})\,d\sigma(\mathbf{v})$$

On each point we take a chance to stop (absorption) or to reflect based on the reflectance properties of the surface (BRDF).

The Path Tracing Algorithm We want to approximate the integral of the reflectance equation.

$$L_r(\mathbf{u}) = \int_{\Omega} f_r(\mathbf{u},\mathbf{v})\, L_i(\mathbf{v})\, cos(\mathbf{v})\, d\sigma(\mathbf{v})$$

The Path Tracing MC estimator with Importance Sampling

$$L_r(\mathbf{u}) \;=\; \frac{1}{N}\sum_{i=1}^{N} \frac{f_r(\mathbf{u},\mathbf{v})\, L_i(\mathbf{v})\, cos(\mathbf{v})}{p_{\mathbf{u}}(\mathbf{v})} \quad with \;\; \mathbf{v} \sim p_{\mathbf{u}}$$

The Probability Density Function (PDF)

$$L_r(\mathbf{u}) = \int_{\Omega} f_r(\mathbf{u}, \mathbf{v})\, L_i(\mathbf{v})\, cos(\mathbf{v})\, d\sigma(\mathbf{v})$$

Which distribution?

- product of lighting and BRDF
- lighting
- BRDF

The Probability Density Function (PDF)

$$L_r(\mathbf{u}) = \int_\Omega f_r(\mathbf{u},\mathbf{v})\, L_i(\mathbf{v})\, cos(\mathbf{v})\, d\sigma(\mathbf{v})$$

Which distribution?

- **product of lighting and BRDF**
- lighting
- BRDF

The Probability Density Function (PDF)

$$L_r(\mathbf{u}) = \int_\Omega f_r(\mathbf{u},\mathbf{v})\, L_i(\mathbf{v})\, cos(\mathbf{v})\, d\sigma(\mathbf{v})$$

Which distribution?

- **product of lighting and BRDF**
  - Too expensive
- lighting
- BRDF

The Probability Density Function (PDF)

$$L_r(\mathbf{u}) = \int_{\Omega} f_r(\mathbf{u},\mathbf{v})\, L_i(\mathbf{v})\, cos(\mathbf{v})\, d\sigma(\mathbf{v})$$

Which distribution?

- product of lighting and BRDF
  - Too expensive
- **lighting**
- BRDF

The Probability Density Function (PDF)

$$L_r(\mathbf{u}) = \int_\Omega f_r(\mathbf{u},\mathbf{v})\, L_i(\mathbf{v})\, cos(\mathbf{v})\, d\sigma(\mathbf{v})$$

Which distribution?

- product of lighting and BRDF
  - Too expensive
- **lighting**
  - Too many samples for glossy surfaces
- BRDF

The Probability Density Function (PDF)

$$L_r(\mathbf{u}) = \int_\Omega f_r(\mathbf{u}, \mathbf{v})\, L_i(\mathbf{v})\, cos(\mathbf{v})\, d\sigma(\mathbf{v})$$

Which distribution?

- product of lighting and BRDF
  - Too expensive
- lighting
  - Too many samples for glossy surfaces
- **BRDF**

The Probability Density Function (PDF)

$$L_r(\mathbf{u}) = \int_\Omega f_r(\mathbf{u}, \mathbf{v})\, L_i(\mathbf{v})\, cos(\mathbf{v})\, d\sigma(\mathbf{v})$$

Which distribution?

- product of lighting and BRDF
  - Too expensive
- lighting
  - Too many samples for glossy surfaces
- **BRDF**
  - Computationally efficient

The Probability Density Function (PDF)

$$L_r(\mathbf{u}) = \int_\Omega f_r(\mathbf{u}, \mathbf{v})\, L_i(\mathbf{v})\, cos(\mathbf{v})\, d\sigma(\mathbf{v})$$

Which distribution?

- product of lighting and BRDF
- lighting
- BRDF

Key point

The BRDF function times the cosine term is closer to the integrand

The Probability Density Function (PDF)

Which distribution?

$$L_r(\mathbf{u}) = \int_\Omega f_r(\mathbf{u}, \mathbf{v})\, L_i(\mathbf{v})\, cos(\mathbf{v})\, d\sigma(\mathbf{v})$$

- Now we know that is preferable to sampling proportional to the BRDF function times the cosine term.

The Probability Density Function (PDF)

Which distribution?

$$L_r(\mathbf{u}) = \int_\Omega f_r(\mathbf{u},\mathbf{v})\, L_i(\mathbf{v})\, cos(\mathbf{v})\, d\sigma(\mathbf{v})$$

- L integrated with respect the solid angle measure

The Probability Density Function (PDF)

Which distribution?

$$L_r(\mathbf{u}) = \int_\Omega f_r(\mathbf{u},\mathbf{v})\, L_i(\mathbf{v})\, d\sigma_p(\mathbf{v})$$

- L integrated with respect the projected solid angle measure

$$d\sigma_p(\mathbf{v}) \stackrel{\text{def}}{=} cos(\mathbf{v}) d\sigma(\mathbf{v})$$

The Probability Density Function (PDF)

Which distribution?

$$L_r(\mathbf{u}) = \int_\Omega f_r(\mathbf{u},\mathbf{v})\, L_i(\mathbf{v})\, d\sigma_p(\mathbf{v})$$

Monte-Carlo Estimator with Importance Sampling

$$L_r(\mathbf{u}) \;=\; \frac{1}{N}\sum_{i=1}^{N} \frac{f_r(\mathbf{u},\mathbf{v})\, L_i(\mathbf{v})}{q_{\mathbf{u}}(\mathbf{v})} \;\; with \;\; \mathbf{v} \sim q_{\mathbf{u}}$$

The Probability Density Function (PDF)

Which distribution?

Our contribution

We derive a sampling method based on a PDF of type $q_{\mathbf{u}}$

Monte-Carlo Estimator with Importance Sampling

$$L_r(\mathbf{u}) = \frac{1}{N}\sum_{i=1}^{N}\frac{f_r(\mathbf{u},\mathbf{v})\,L_i(\mathbf{v})}{q_{\mathbf{u}}(\mathbf{v})} \quad \textit{with} \quad \mathbf{v} \sim q_{\mathbf{u}}$$

Contents

Objetive

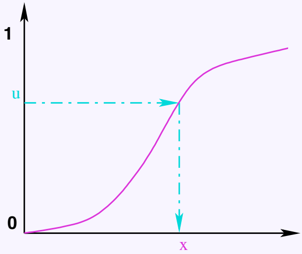
Sampling random values $s \sim p$ using uniform values $\xi \sim U(0,1)$

Objetive

Sampling random values $s \sim p$ using uniform values $\xi \sim U(0,1)$

Inversion of the CDF

1. Compute the *Cumulative Distribution Function*, F and invert it
2. Given $\xi \implies x = F^{-1}(\xi); x \sim p$.

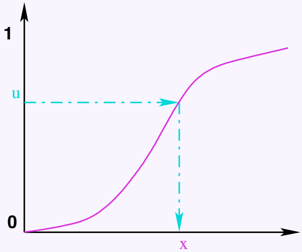


Objetive

Sampling random values $s \sim p$ using uniform values $\xi \sim U(0,1)$

Inversion of the CDF

1. Compute the *Cumulative Distribution Function*, F and invert it
2. Given $\xi \Longrightarrow x = F^{-1}(\xi); x \sim p$.



drawback

Some funtions cannot be analytically inverted.

Objetive

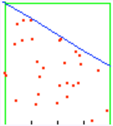
Sampling random values $s \sim p$ using uniform values $\xi \sim U(0,1)$

Objetive

Sampling random values $s \sim p$ using uniform values $\xi \sim U(0,1)$

Rejection Sampling

1. Compute maximum value M of the desired function p
2. Given two random values $x \sim U(0,1)$ and $y \sim U(0,M)$.
3. Reject and repeat until $y \leq f(x)$.

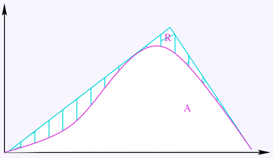


Objetive

Sampling random values $s \sim p$ using uniform values $\xi \sim U(0,1)$

Rejection Sampling

1. Compute maximum value M of the desired function p
2. Given two random values $x \sim U(0,1)$ and $y \sim U(0,M)$.
3. Reject and repeat until $y \leq f(x)$.



disadvantage

- Probability of reject is too high for non-uniform function.
- Unbounded execution times.

Two Approaches

Direct Sampling

- Ideal cases
- Cosine-Lobe based sampling: Lafortune, Phong, Blinn, etc. $\longrightarrow$ not applicable to non-lobe based BRDF models.
- Few BRDF models have an specific PDF

Two Approaches

Direct Sampling

- Ideal cases
- Cosine-Lobe based sampling: Lafortune, Phong, Blinn, etc. $\longrightarrow$ not applicable to non-lobe based BRDF models.
- Few BRDF models have an specific PDF

Generic Sampling

- Independent of the BRDF
- Independent of how the BRDF is represented
  - analytically
  - from measurements

Some Approximate Solutions

Tabular BRDF [Matusik et al. 2003]

- Densely sampled tabular representations of BRDFs (off-line)
- Store the reflectance distribution for dense set of views
- Sampling using numerical inversion of the CDF

Drawback

- Expensive: The total size of this set of CDFs + the BRDF can quickly become prohibitively large
- Aliasing if an insufficient number of slices are stored

Some Approximate Solutions

Factorization of the BRDF [Lawrence et al. 2004]

- Based on numerical approximation of the BRDF times the cosine term
- Compact: BRDF is factorized into 2D and 1D pieces (low dimension)
- Effective sampling of view-independent 1D functions

Key limitations

- The PDF is not exactly proportional to BRDF
- Some samples must be rejected
- Not all functions factorize efficiently
- The user must decide the number of factors

Some Approximate Solutions

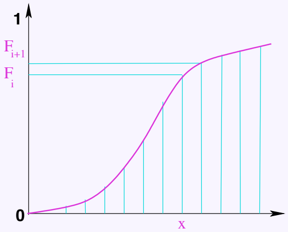
The Cascade CDF [Lawrence et al. 200]

- Store the reflectance distribution for dense set of views

Some Approximate Solutions

The Cascade CDF [Lawrence et al. 200]

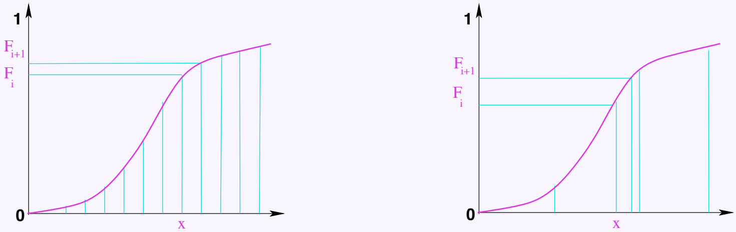
- Store the reflectance distribution for dense set of views
- Sampling using numerical inversion of the CDF: **UniformCDF**



Some Approximate Solutions

The Cascade CDF [Lawrence et al. 200]

- Store the reflectance distribution for dense set of views
- Sampling using numerical inversion of the CDF: **UniformCDF**
- Approximate the CDF curve by segments and reduce storage: **CompressCDF**



Contents

Development of a sampling method

Objectives

- Sample random directions for arbitrary BRDF functions
- Provide a good PDF for *Importance Sampling*
- $p_{\mathbf{u}} \propto f_r(\mathbf{u}, \cdot)$ for arbitrary $\mathbf{u}$ and for a finite set of BRDFs in a scene

Development of a sampling method

Objectives

- Sample random directions for arbitrary BRDF functions
- Provide a good PDF for *Importance Sampling*
- $p_{\mathbf{u}} \propto f_r(\mathbf{u}, \cdot)$ for arbitrary $\mathbf{u}$ and for a finite set of BRDFs in a scene

Our approach

- **A PDF based on the σ_p measure**

Development of a sampling method

Objectives

- Sample random directions for arbitrary BRDF functions
- Provide a good PDF for *Importance Sampling*
- $p_{\mathbf{u}} \propto f_r(\mathbf{u}, \cdot)$ for arbitrary $\mathbf{u}$ and for a finite set of BRDFs in a scene

Our approach

- A PDF based on the σ_p measure
- **A *quadtree* represents the BRDF adaptively**

Development of a sampling method

Objectives

- Sample random directions for arbitrary BRDF functions
- Provide a good PDF for *Importance Sampling*
- $p_{\mathbf{u}} \propto f_r(\mathbf{u}, \cdot)$ for arbitrary $\mathbf{u}$ and for a finite set of BRDFs in a scene

Our approach

- A PDF based on the σ_p measure
- A *quadtree* represents the BRDF adaptively
- **Leaf nodes represent exactly the BRDF function**

Development of a sampling method

Objectives

- Sample random directions for arbitrary BRDF functions
- Provide a good PDF for *Importance Sampling*
- $p_{\mathbf{u}} \propto f_r(\mathbf{u}, \cdot)$ for arbitrary $\mathbf{u}$ and for a finite set of BRDFs in a scene

Our approach

- A PDF based on the σ_p measure
- A *quadtree* represents the BRDF adaptively
- Leaf nodes represent **exactly** the BRDF function
- **Optimal Rejection sampling on leaf nodes**

Development of a sampling method

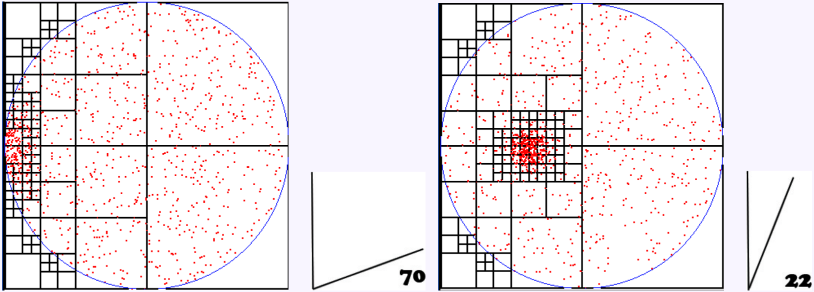
Objectives

- Sample random directions for arbitrary BRDF functions
- Provide a good PDF for *Importance Sampling*
- $p_{\mathbf{u}} \propto f_r(\mathbf{u}, \cdot)$ for arbitrary $\mathbf{u}$ and for a finite set of BRDFs in a scene

Our approach

- A PDF based on the σ_p measure
- A *quadtree* represents the BRDF adaptively
- Leaf nodes represent **exactly** the BRDF function
- Optimal Rejection sampling on leaf nodes
- **Few and simple parameters**

Adaptive Sampling of the BRDF



Benefit

Density of the samples exactly proportional to the BRDF

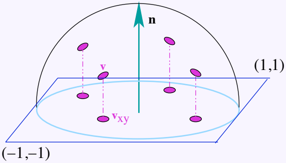
Optimal *Rejection Sampling*

- While loop executed following a *geometric distribution*
- The **average number of trials** is a parameter of our algorithm
- We guarantee a probability for accepting a sample $\geq \frac{1}{n_{max}}$

A change in the domain of integration

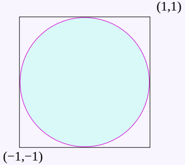
key idea

The PDF has implicit a cosine term



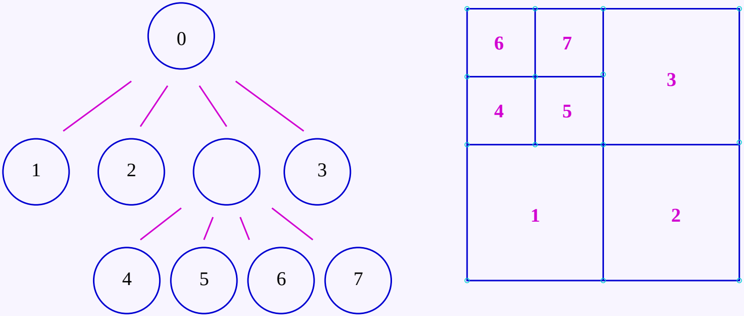
The Unit Disk Domain

- The 3D direction is projected into a 2D point.
- The solid projected angle σ_p is the measure in $\mathcal{D}^2$.



$$d\sigma_p(\mathbf{v}) \stackrel{\text{def}}{=} (\mathbf{v}\cdot\mathbf{n})\, d\sigma(\mathbf{v}) \Longleftrightarrow dA(\mathbf{v}_{xy})$$

The data structure



The data structure

For each Node i

$$I_i = \int_{x,y \in R_i} f_r(\mathbf{u}, \mathbf{v}_{xy}) \, dx \, dy$$

$$V_i = \int_{x,y \in R_i} \bar{f}_r(\mathbf{u}, \mathbf{v}_{xy}) \, dx \, dy = size_i^2 \times M_i$$

$$M_i \stackrel{\text{def}}{=} max_{(x,y) \in R_i} \{ f_r(\mathbf{u}, \mathbf{v}_{xy}) \}$$

The data structure

For each Node i

$$I_i = \int_{x,y \in R_i} f_r(\mathbf{u}, \mathbf{v}_{xy})\, dx\, dy$$

$$V_i = \int_{x,y \in R_i} \bar{f}_r(\mathbf{u}, \mathbf{v}_{xy})\, dx\, dy = size_i^2 \times M_i$$

$$M_i \stackrel{\text{def}}{=} max_{(x,y) \in R_i} \{ f_r(\mathbf{u}, \mathbf{v}_{xy}) \}$$

$$P_{\mathbf{u}}(R_i) \stackrel{\text{def}}{=} \frac{I_i}{I_0}$$

The data structure

For each Node i

$$I_i = \int_{x,y \in R_i} f_r(\mathbf{u}, \mathbf{v}_{xy})\, dx\, dy$$

$$V_i = \int_{x,y \in R_i} \bar{f}_r(\mathbf{u}, \mathbf{v}_{xy})\, dx\, dy = size_i^2 \times M_i$$

$$M_i \stackrel{\text{def}}{=} max_{(x,y)\in R_i} \left\{ f_r(\mathbf{u}, \mathbf{v}_{xy}) \right\}$$

The criteria ensures:

$$n_{max} \frac{I_i}{V_i} \geq 1$$

The Adaptive PDF

The Adaptive PDF $q_{\mathbf{u}}$ is represented as a *quadtree* for each BRDF and for any given direction $\mathbf{u}$

Our precomputation

We store n quadtrees for n incident angles in a preprocess step

The sampling procedure

Generation of a single direction

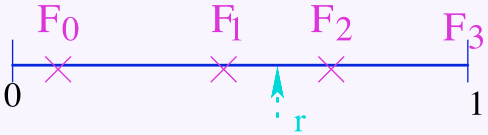
1. Traverse the *quadtree* and select a leaf node i
2. Adjust the domain of the region if necessary
3. Perform rejection sampling on R_i
4. Do inverse projection $\mathbf{s}_{xy}$ and get $\mathbf{s} \in \Omega$
5. $q_{\mathbf{u}}(\mathbf{s}^i_{xy})$ is $P_{\mathbf{u}}(\mathbf{s}^i_{xy})$ times $P_{\mathbf{u}}(R_i)$

Step 1

How to traverse the *quadtree*

From root node till a leaf is found. On each non-leaf node do:

1. Compute de Cumulative Distribution Function $F(R_j)$ for children regions
2. Get an uniform random value r
3. If $F(R_{j-1}) < r \leq F(R_j)$ then descend to j-child

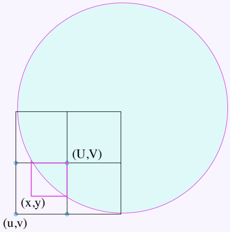


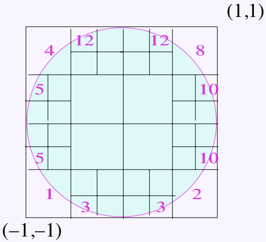
Step 2

The improved area of a region

Given $R_i = [u_i, v_i] \times [u_i + size_i, v_i + size_i]$:

- Four distances are computed
- Four boolean tests leads to a binary value $b_0 b_1 b_2 b_3$
- Update region bounds





Step 3 and so on

How to perfom *Rejection Sampling*

Inside the optimal limit $(x, y) \times (U, V)$ (as an example):

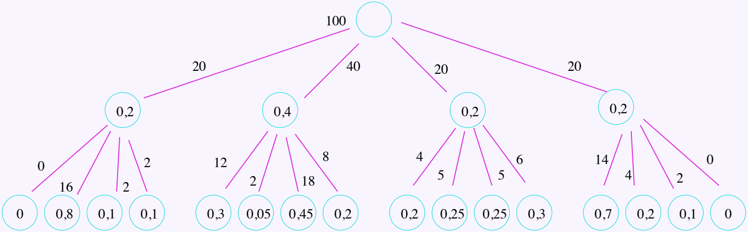
1. Get an uniform point in ($x_{\mathbf{s}}$,$y_{\mathbf{s}}$) in $[x, U] \times (U, V)$
2. Get an uniform value z in $[0, M_i]$
3. Compute the BRDF value of $f_r(\mathbf{u}, \mathbf{s}_{xy})$
4. Reject and repeat while $f_r \leq z$
5. Do inverse projection:

$$(x_{\mathbf{s}}, y_{\mathbf{s}}) \in \mathcal{D}^2 \Longleftrightarrow \left(x_{\mathbf{s}}, y_{\mathbf{s}}, \sqrt{x_{\mathbf{s}}^2 + y_{\mathbf{s}}^2} \right) \in \Omega$$

An optimization

Sampling N-directions

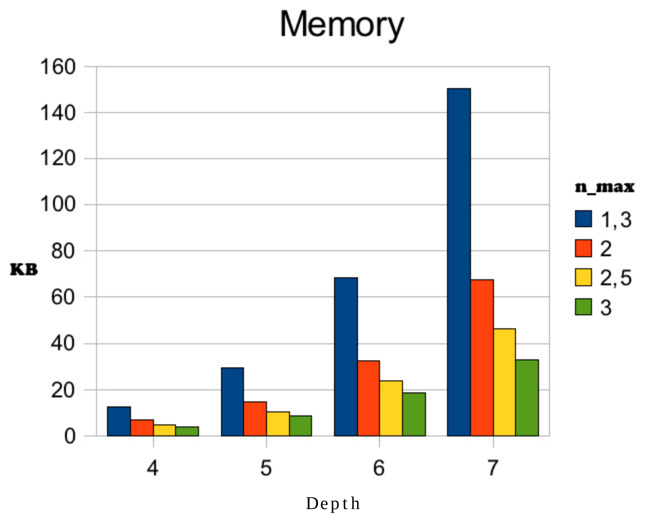
1. Get N samples using a single call
2. Split N using the CDF of children nodes



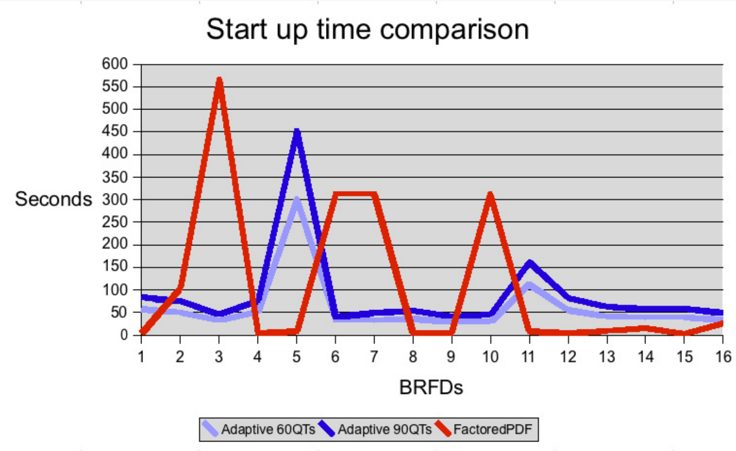
Editing and sampling the BRDF: a demo

Contents

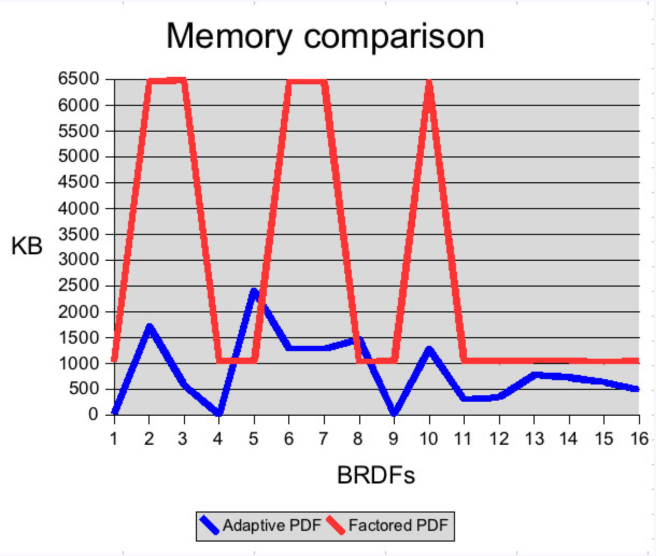
With varying parameters



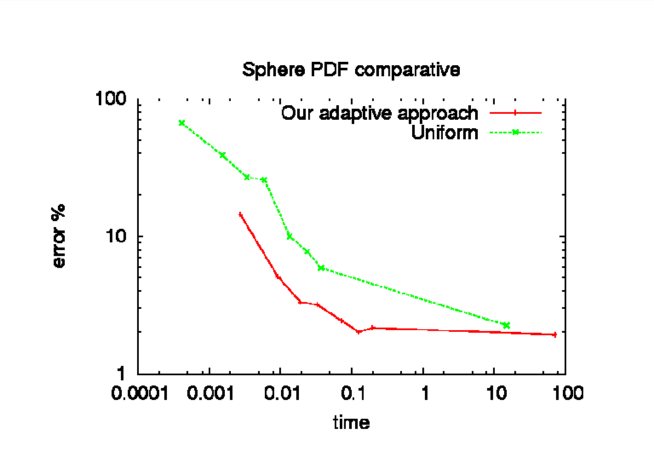
The cost of the precomputation



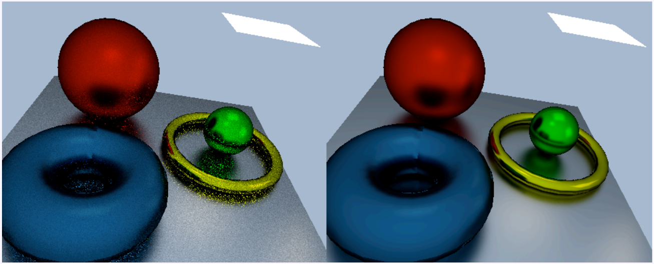
The cost of the precomputation



Uniform vs Adaptive Sampling

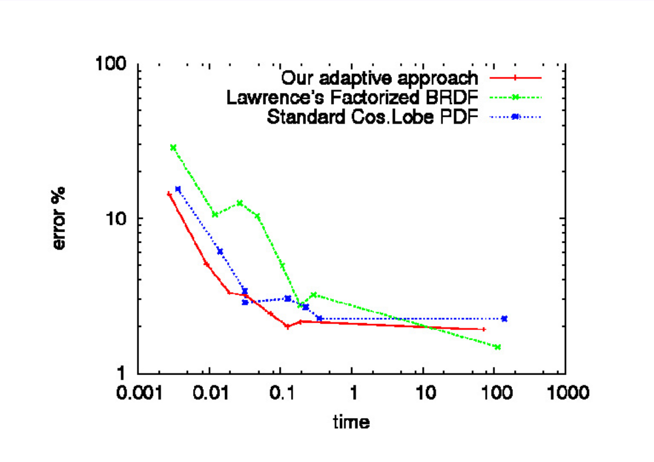


Uniform vs Adaptive Sampling

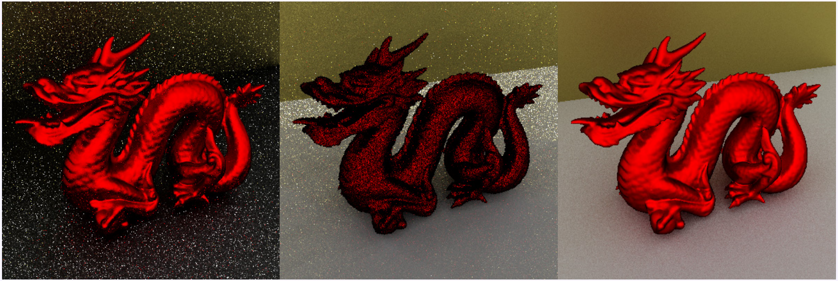


Uniform: 2500 samples in 150.54 secs.
Adaptive: 100 samples in 43 secs.

Direct and General vs Adaptive Sampling

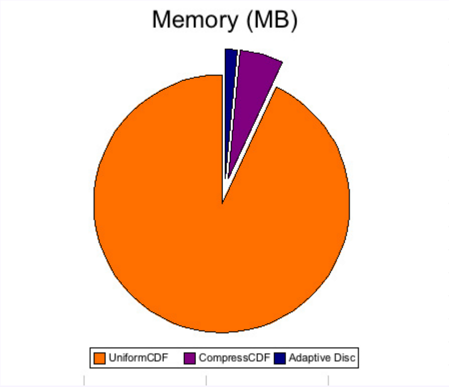


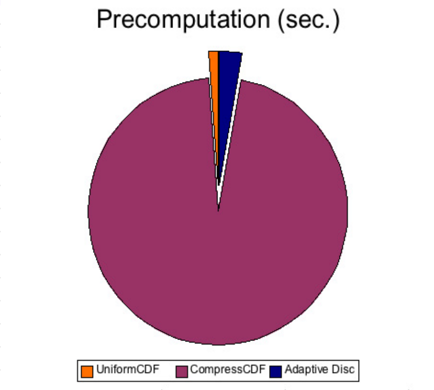
Direct an General vs Adaptive Sampling



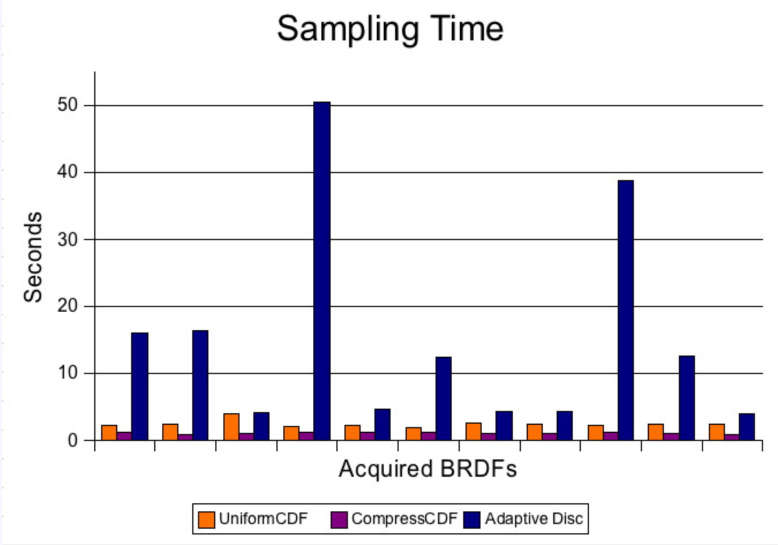
Cosine Lobe PDF	–	–	29.06 sec
Factored PDF	120.19 sec	10.25 MB	24.539 sec
Adaptive PDF	60.48 sec	1.19 MB	42.632 sec

Sampling the Merl BRDF Database





Sampling the Merl BRDF Database



Sampling the Merl BRDF Database

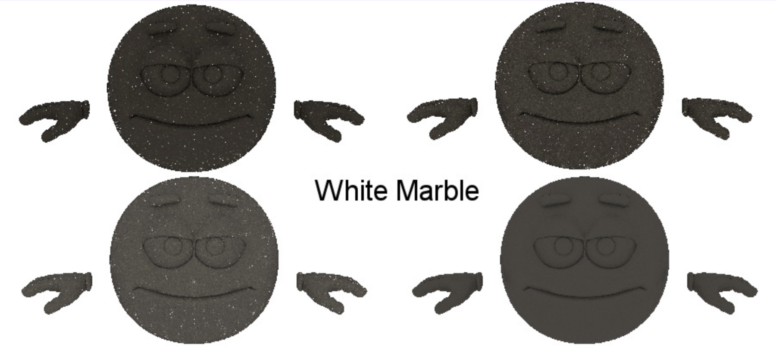
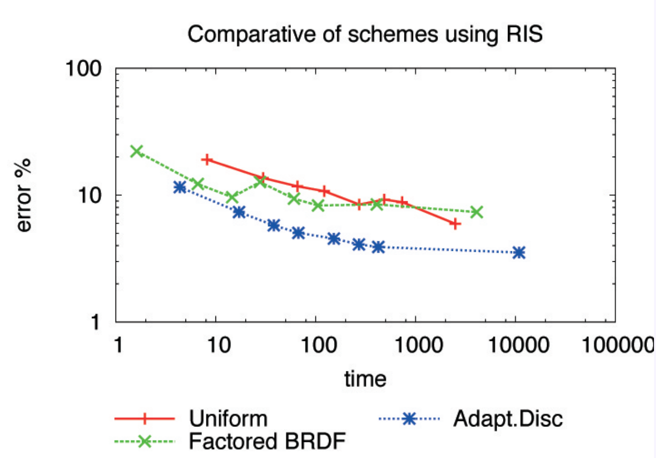


Image: 450x200 pixels and 25 samples

$$L_r(\mathbf{u}) = \int_{\Omega} f_r(\mathbf{u}, \mathbf{v})\, L_i(\mathbf{v})\, cos(\mathbf{v})\, d\sigma(\mathbf{v})$$

with **Resampling Importance Sampling**



Sampling the product of the BRDF and lighting

Importance Sampling

Resampling Importance Sampling

Contents

Contributions of this work

1. Survey of the reflectance models in Computer Graphics
2. A generic sampling method
   - We have an exact a sampling scheme suitable for MCGI.
   - Generic: sample any analytical or acquired BRDF model.
   - With the same number of samples, it achieve less error than others.
   - No need of user guidance.
   - Combines with others variance reduction techniques like RIS.
3. Tools for designers for rendering and modeling GI scenes

Future work

1. Compare ours with more generic sampling schemes
2. Use it with anisotropic measured reflectance data
3. Optimize the sampling scheme
   - reduce the sampling times with the GPU
   - use an area-preserving mapping
   - use QMC sequences
4. Apply this sampling to other MC applications

Thank you for your attention

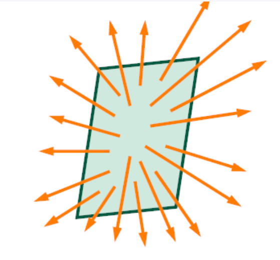
Ph.D. Presentation Rosana Montes

An Importance Sampling Method for arbitrary BRDFs used in Global Illumination Applications

Granada, June 27 2008

Radian Flux

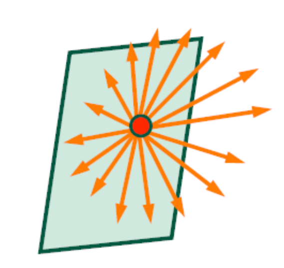
Total radiant power [Watts]: ϕ



In this example, we are looking at a window. The power of all the light pouring through all parts of the window, in all directions, is radiant flux.

Radiant Exitance

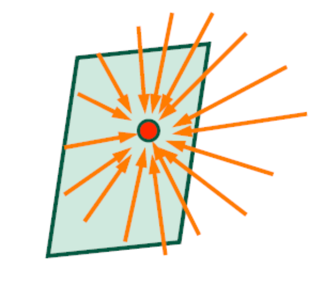
Power per surface area [Watts / $meter^2$]: $M = d\Phi_r/dA$.



The area density of the power exitant (coming out of) a single point on the window is radiant exitance, also called **radiosity** (B).

Irradiance

Power per surface area [Watts / $meter^2$]: $E = d\Phi_r/dA$.

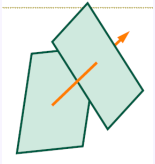


Irradiance is very similar to radiosity, but it measures incident (incoming) light rather than exitant light.

Radiance

Power per (projected) surface area per solid angle [Watts / ($meter^2$ steradian)]: L

$$dL(x, \mathbf{u}) = \frac{d^2\Phi(x, \mathbf{u})}{dA(x)d\sigma_p(\mathbf{u})}$$



Materials

Can be classified into two categories.

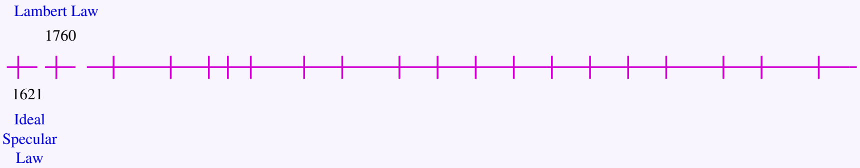
- **Isotropic**
- **Anisotropic**

- **Metals**
- **Dielectrics**

Time line of some representative *reflectance models*

- mirror surfaces
- plausible
- direct sampling
- isotropic

Time line of some representative *reflectance models*

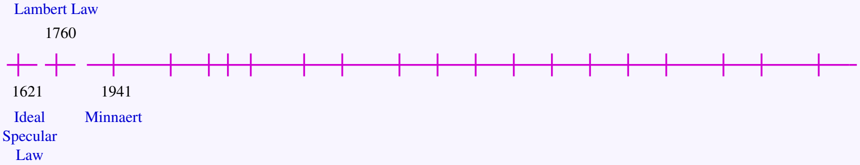


- diffuse surfaces

- plausible

- direct sampling
- isotropic

Time line of some representative *reflectance models*

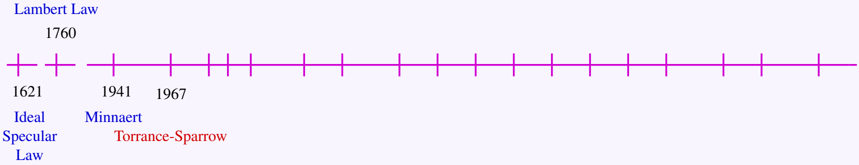


- diffuse surfaces

- moon reflectance
- not energy conservative

- isotropic

Time line of some representative *reflectance models*

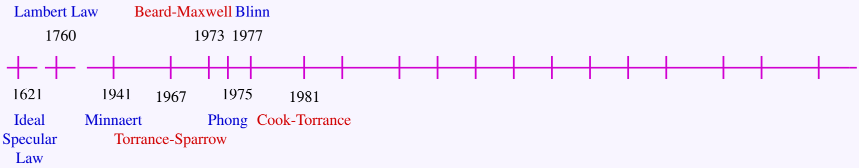


- diffuse surfaces

- glossy surfaces; metals
- not energy conservative

- isotropic

Time line of some representative *reflectance models*

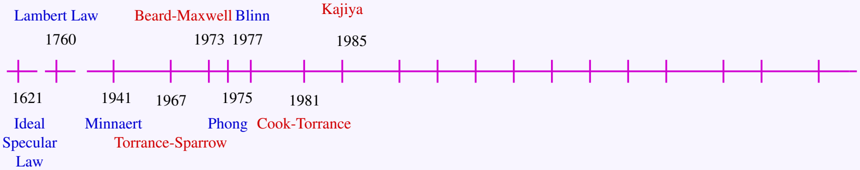


- diffuse surfaces

- paint / poolished surfaces
- not energy conservative

- isotropic

Time line of some representative *reflectance models*

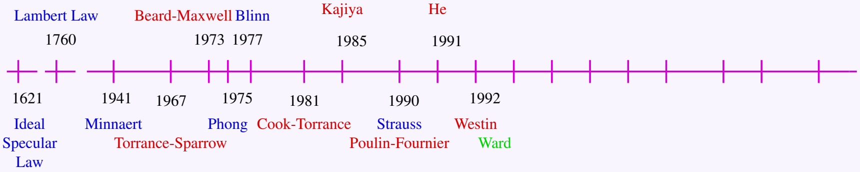


- matte / plastics
- diffuse surfaces

- not energy conservative

- isotropic

Time line of some representative *reflectance models*

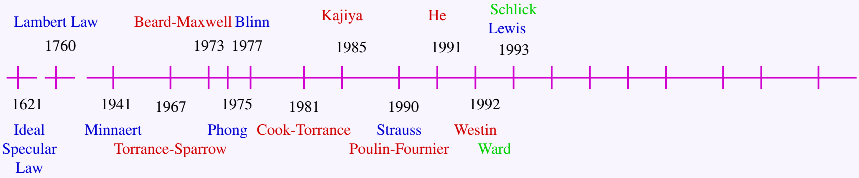


- matte / plastics
- diffuse surfaces

- not energy conservative

- isotropic

Time line of some representative *reflectance models*

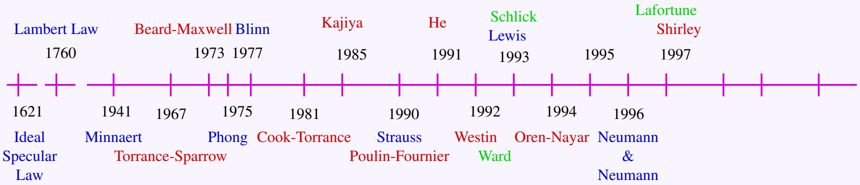


- diffuse surfaces
- metal or dielectrics

- glossy surfaces; metals
- not energy conservative

- isotropic

Time line of some representative *reflectance models*

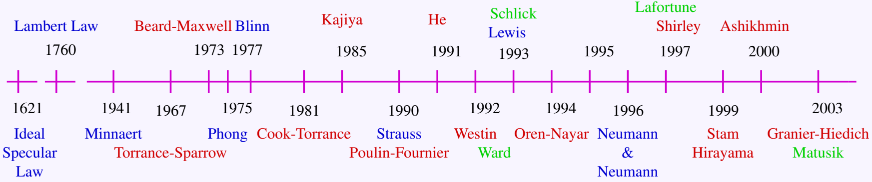


- diffuse surfaces

- glossy surfaces; metals
- not energy conservative

- anisotropic

Time line of some representative *reflectance models*

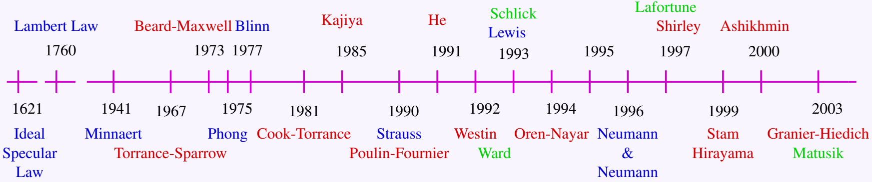


- diffuse surfaces
- metal or dielectrics

- not energy conservative

- anisotropic

Time line of some representative *reflectance models*

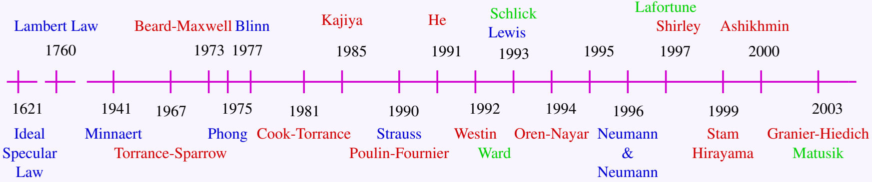


- diffuse surfaces

- glossy surfaces; metals
- plausible

- anisotropic

Time line of some representative *reflectance models*

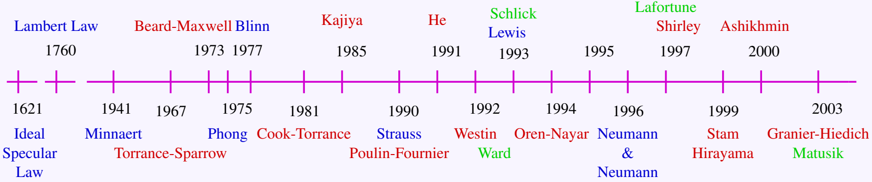


- diffuse surfaces

- glossy surfaces; metals
- plausible

- direct sampling
- anisotropic

Time line of some representative *reflectance models*

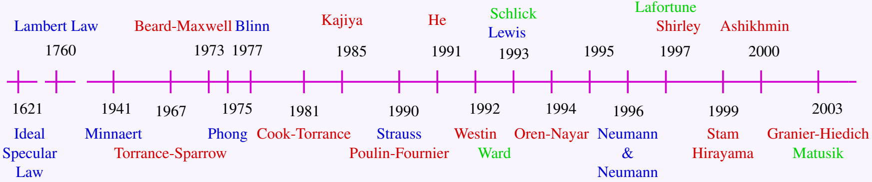


- matte / plastics
- diffuse surfaces

- plausible

- direct sampling
- anisotropic

Time line of some representative *reflectance models*

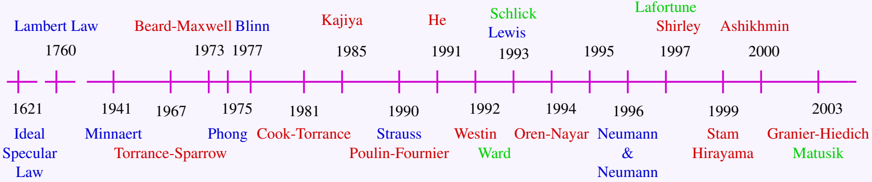


- diffuse surfaces
- cloth, sand and some plastics

- plausible

- isotropic

Time line of some representative *reflectance models*

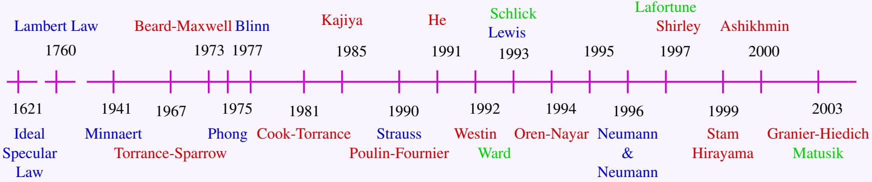


- diffuse surfaces
- cloth, sand and some plastics

- glossy surfaces; metals
- plausible

- direct sampling
- anisotropic

Time line of some representative *reflectance models*

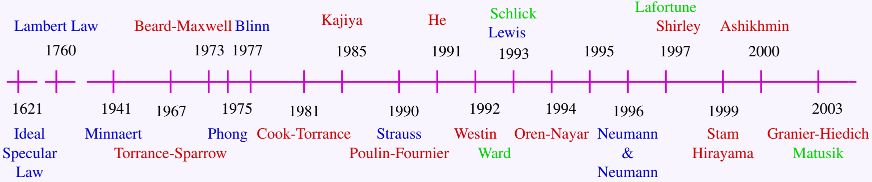


- matte / plastics
- diffuse surfaces

- plausible

- direct sampling
- isotropic

Time line of some representative *reflectance models*

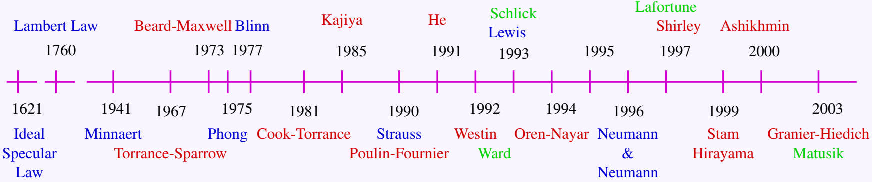


- diffuse surfaces

- paint / poolished surfaces
- plausible

- direct sampling
- anisotropic

Time line of some representative *reflectance models*



- diffuse surfaces

- acquired real surfaces

- direct sampling
- isotropic

Some Terminology for Monte-Carlo

- Probability measure $P_{\mathbf{u}}(\Omega) = 1$
- Cumulative Distribution Function
  - $F_X(x) = P_r(X \leq x)$
  - $F_X = \int_{\infty}^{x} p(y)dy$
- Probability Distribution Function
  - $p(x) = \frac{dF_X(x)}{dx}$
  - $P_r(X \in [a, b]) = \int_a^b p(x)dx$
- Expected value $E[f] = \int f(x)p(x)dx$
- Error or Bias $\beta[f] = E[f - Q]$
- Variance $V[f] = E[f^2] - E[f]^2$
- Efficiency $\epsilon[f] \ = \ \frac{1}{V[f]\,T[f]}$.

Variance

The error produced in the estimation. To diminish it:

- more samples $\Longrightarrow$ more computation
- apply a variance reduction techniques.

Variance

The error produced in the estimation. To diminish it:

- more samples $\Longrightarrow$ more computation
- apply a variance reduction techniques.

- Russian Roulette

- With probability q stop the *path-walk*
- With probability $1 - q$ integrand is evaluated
- Unbiased estimate by weighting

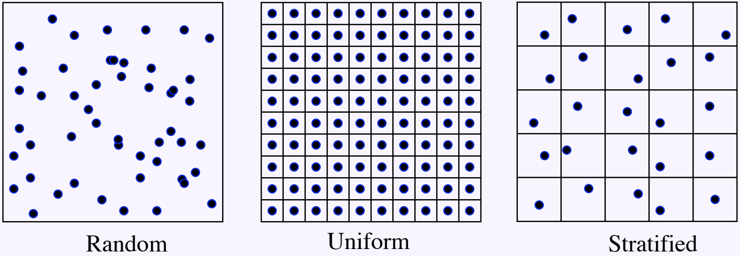
Variance

The error produced in the estimation. To diminish it:

- more samples $\Longrightarrow$ more computation
- apply a variance reduction techniques.

- Domain Stratification

- Divided domain into N disjoints subregions
- Instead of N-sampling the entire domain, 1 sample on each region



- Domain Stratification

Variance

The error produced in the estimation. To diminish it:

- more samples $\Longrightarrow$ more computation
- apply a variance reduction techniques.

- Importance Sampling

- Use an easier to sample function p, instead of the interest f
- $p \propto f$
- not uniform

Variance

The error produced in the estimation. To diminish it:

- more samples $\Longrightarrow$ more computation
- apply a variance reduction techniques.

- Multiple Importance Sampling

- More than one density in a single estimator
- Easiest to sample each one that combined

Variance

The error produced in the estimation. To diminish it:

- more samples $\Longrightarrow$ more computation
- apply a variance reduction techniques.

- Resampling Importance Sampling

- Use an easier to sample function p, instead of the interest g
- Get M samples $\sim p$ and its weights $w(x_i) = g(x_i)/p(x_i)$
- Get y proportional to the M weights $\Longrightarrow y \sim g$

The Radiance Eq.

Previous $\Longrightarrow$

$$L_r(\mathbf{u}) \;=\; L_e(\mathbf{u}) + \int_{\Omega} f_r(\mathbf{u},\mathbf{v})\, L_i(\mathbf{v})\; (\mathbf{v}\cdot\mathbf{n})\; d\sigma\;(\mathbf{v}).$$

With a change in the domain of integration $\Longrightarrow$

$$L_r(\mathbf{u}) \;=\; L_e(\mathbf{u}) + \int_{\mathcal{D}^2} f_r(\mathbf{u},\mathbf{v}_{xy})\, L_i(\mathbf{v}_{xy})\, dA(\mathbf{v}_{xy})$$

The Monte-Carlo Estimator

$$X_n(\mathbf{s}_1,\ldots,\mathbf{s}_n) \;\stackrel{\text{def}}{=}\; \frac{1}{n}\sum_{j=1}^{n}\frac{f_r(\mathbf{u},\mathbf{s}^j_{xy})L_i(\mathbf{s}^j_{xy})}{q(\mathbf{s}^j_{xy})}$$

with random directions $\mathbf{s}_{xy} \sim q_{\mathbf{u}}$.

Two Approaches

Direct Sampling

- Ideal cases

Ideal Specular

- Light is reflected in a single direction.
- The PDF is constant.

Two Approaches

Direct Sampling

- Ideal cases

Ideal Diffuse Sample a random direction uniformly in the hemi-sphere.

$$p(\mathbf{v}) = \frac{1}{\pi} cos(\mathbf{v})$$

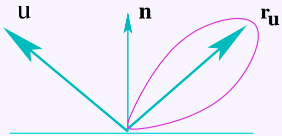
Two Approaches

Direct Sampling

- Ideal cases
- Cosine-Lobe based sampling: Lafortune, Phong, Blinn, etc. $\longrightarrow$ not applicable to non-lobe based BRDF models.

Non-Ideal distributions Sample acoording to a lobe around the axe vector $\mathbf{r_u}$.
Shape is controled with the exponent parameter.

$$p_{\mathbf{u}}(\mathbf{v}) = \frac{n+1}{2\pi} cos^n(\mathbf{r_u} \cdot \mathbf{v})$$



Two Approaches

Direct Sampling

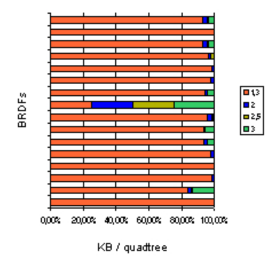
- Ideal cases
- Cosine-Lobe based sampling: Lafortune, Phong, Blinn, etc. $\longrightarrow$ not applicable to non-lobe based BRDF models.
- Few BRDF models have an especific PDF

Non-Ideal distributions

- Ward BRDF.
- Neumann-Neumman BRDF.
- Ashikhmin BRDF.

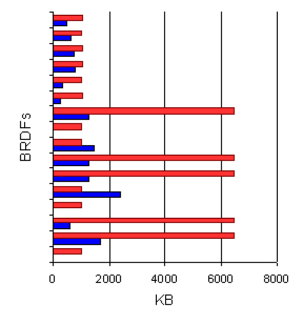
Memory

Single QT	nmax - Adaptive Disc PDF			
BRDF	1.3	2.0	2.5	3.0
Ashikhmin	1188.3	0.103	0.103	0.103
Beardmax	1412.8	28.710	12.360	230.332
Blinn	522.410	6.914	0.103	0.103
Coupled	322.969	0.103	0.103	0.103
He	1346.3	28.711	10.184	0.103
Lafortune	2344.6	30.891	11.273	111.539
Lewis	4701.9	40.699	12.363	304.441
Minnaert	974.695	28.711	9.094	8.549
Oren	0.103	0.103	0.103	0.103
Phong	2455.7	21.082	6.914	115.898
Poulin	308.801	6.914	0.103	0.103
Schlick (D)	522.410	9.094	0.103	0.103
Schlick (S)	522.410	9.094	0.103	0.103
Strauss	380.730	13.453	6.914	15.633
Torrance	728.391	0.103	0.103	0.103
Ward	340.406	8.004	4.734	13.453
Avg. (KB)	1129.5	14.54	4.67	50.05



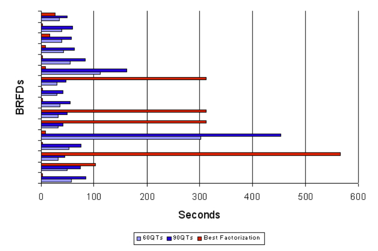
Memory

BRDF	60 QT	Factored PDF
Ashikhmin	6.25	1031
BeardMax.	1713.25	6454
Blinn	582.25	6481
Coupled	6.25	1033
He	2407.25	1034
Lafortune	1275.25	6445
Lewis	1279.25	6445
Minnaert	1461.25	1031
Oren	6.25	1033
Phong	1279.25	6445
Poulin	297.25	1038
Schlick (D)	342.25	1033
Schlick (S)	780.25	1043
Strauss	727.25	1052
Torrance	631.25	1029
Ward	483.25	1038
Average (KB)	**829.88**	2729.06



Computation Time

	Adaptive Disc		Factored PDF
BRDF	**60QTs**	**90QTs**	Best params
Ashikhmin	57.61	83.46	3.12
BeardMaxwell	49.08	74.75	102.9
Blinn	32.12	45.08	565.7
Coupled	51.25	76.17	3.21
He	300.8	452.7	8.00
Lafortune	32.33	40.34	312.5
Lewis	32.49	48.86	312.4
Minnaert	35.33	54.28	3.17
Oren-Nayar	28.99	40.99	3.08
Phong	29.99	45.69	312.3
Poulin-Fournier	112.7	160.9	8.09
Schlick (D)	55.17	82.49	3.24
Schlick (S)	41.65	63.03	8.30
Strauss	38.82	56.86	15.36
Cook-Torrance	38.84	58.43	2.97
Ward	33.57	48.99	26.62
Avg. (sec)	**60.67**	**89.56**	105.69



The BRDF parameters.

List of the reflectance models, with the parametrization used through the evaluation of our model.

BRDF	Parameters
ASHIKHMIN	nu =25 nv =25 ks =0.39 kd =0.6
BEARDMAXWELL	Ω =1 τ =0 n_{real} =0.37 n_{imag} =2.82 rs =0.4 rd =0.75 rv =0.5
BLINN	n =100 ks =1 kd =0.67
COUPLED	R_m =1 R_o =0.05
HE	σ =0.5 τ =3 λ =800
LAFORTUNE	cx =-1 cy =-1 cz =1 n =100 kp =0.96
LEWIS	n =100 ks =1 kd =0
MINNAERT	k =0.8 kd =1
OREN	s =0 r =1
PHONG	n =100 ks =0.4 kd =0.6
POULIN	d =2 h =0.01 n =100 ks =0.8 kd =0.2
SCHLICK (D)	sc =0.7 sr =0.31 sp =1 ly =true dc =1 dr =0 dp =1
SCHLICK (S)	sc =0.7 sr =0.31 sp =1 ly =false
STRAUSS	s =0.76 m =0.91 ks =0.64 kd =0.5
TORRANCE	m =0.35 n_{real} =0.617 n_{imag} =2.63 ks =0.7 kd =0.3
WARD	sx =0.2 sy =0.2 ks =0.5 kd =0.5

We adjust manually the seven parameters for each BRDF factorization. They minimize the average original matrix value and its approximation.

BRDF	$N_{\theta_\mathbf{u}} \times N_{\phi_\mathbf{u}}$	$N_{\theta_\mathbf{p}} \times N_{\phi_\mathbf{p}}$	$J \times K$	**Reparam.**
ASHIKHMIN	16×16	32×16	1×2	false
BEARDMAXWELL	16×16	100×32	2×3	false
BLINN	16×16	100×32	3×3	true
COUPLED	16×16	32×16	1×3	false
HE	16×16	32×16	2×1	false
LAFORTUNE	16×16	100×32	2×2	true
LEWIS	16×16	100×32	2×2	true
MINNAERT	16×16	32×16	1×2	false
OREN	16×16	32×16	1×3	false
PHONG	16×16	100×32	2×2	true
POULIN	16×16	32×16	2×2	false
SCHLICK (D)	16×16	32×16	1×3	true
SCHLICK (S)	16×16	32×16	2×3	false
STRAUSS	16×16	32×16	3×3	false
TORRANCE	16×16	32×16	1×1	false
WARD	16×16	32×16	2×2	true

The right word

For smooth, shining or bright surfaces that reflects light:

- **Polish**: if this surface is produced by rubbing or friction (the car's mirrorlike polish).
- **Gloss**: hard smoothness associated with lacquered or varnished surfaces (a high-gloss paint).
- **Luster**: light reflected from the surfaces o certain materials, such as silk or pearl.
- **Sheen**: A glistering or radiant brightness associated with specific materials (his hair had a velvety sheen).