Advanced Computer Graphics
Summary

Matthias Teschner
Outline

- Introduction
- Ray-object intersections
- Light
- Materials
- Radiosity
- Stochastic Raytracing
Ray Tracing

- Tracing rays through a scene to compute the radiance that is perceived by a sensor, i.e. transported along rays
- Tracing a path from a camera through a pixel position of a virtual image plane to compute the color / radiance of an object that is visible along the path
Light

- Is modeled as geometric rays
  - Travels in straight lines (e.g., no diffraction / bending)
  - Travels at infinite speed (steady state of light is computed)
  - Is emitted by light sources
  - Is absorbed / scattered at surfaces
Measuring Light

– Radiance
  – Characterizes strength and direction of radiation / light
  – Is measured by sensors
  – Is computed in computer-generated images
  – Is preserved along lines in space
  – Does not change with distance
Aspects

Light / radiance travels along rays

Light / radiance is emitted at light sources

Incoming light / radiance is absorbed and scattered at surfaces

Cameras capture light / radiance
Ray Tracing - Capabilities

- Reflection
- Refraction
- Soft shadows
- Caustics
- Diffuse interreflections
- Specular interreflections
- Depth of field
- Motion blur
Ray Tracing - Challenges

- Ray shooting (ray-object intersections)
- Number of rays (quality vs. costs)
  - Approximately solving the Rendering equation
- Recursion depth (quality vs. costs)
Ray Tracing vs. Rasterization

- Rasterization
  - Given a set of viewing rays and a primitive, efficiently compute the subset of rays hitting the primitive
  - Loop over all primitives
  - Implicit ray representation
- Ray tracing
  - Given a viewing ray and a set of primitives, efficiently compute the subset of primitives hit by the ray
  - Loop over all viewing rays
  - Explicit ray representation
Ray Tracing vs. Rasterization

– Solve the same problem

Ray Tracers
Compute ray-object intersections to estimate q from p

Rasterizers
Apply modelview, projection and viewport transform to p in order to estimate q

[Ray Tracing Course: SIGGRAPH 2005]
Ray Tracing vs. Rasterization

- Rasterization
  - Well-established, parallelizable algorithms
  - Popular in interactive applications
  - Specialized realizations of global illumination effects
- Ray tracing
  - Natural incorporation of numerous visual effects
  - Unified algorithms for global illumination effects
  - Trade-off between quality and performance
Outline

- Introduction
- Ray-object intersections
- Light
- Materials
- Radiosity
- Stochastic Raytracing
Motivation

- Rays
  - A half-line specified by an origin / position $\mathbf{o}$ and a direction $\mathbf{d}$
  - Parametric form $\mathbf{r}(t) = \mathbf{o} + t\mathbf{d}$ with $0 \leq t \leq \infty$
- Nearest intersection with all objects has to be computed, i.e. intersection with minimal $t \geq 0$
- In implementations, usually $t \geq \varepsilon$ to avoid self-intersections, e.g., if rays start at object surfaces
Implicit Surfaces

– Implicit functions implicitly define a set of surface points
– For a surface point \((x, y, z)\), an implicit function \(f(x, y, z)\) is zero
– An intersection occurs, if a point on a ray satisfies the implicit equation \(f(x, y, z) = f(r(t)) = f(o + td) = 0\)
– E.g., all points \(p\) on a plane with surface normal \(n\) and offset \(r\) satisfy the equation \(n \cdot (p - r) = 0\)
– The intersection with a ray can be computed based on \(t\)

\[
\mathbf{n} \cdot (\mathbf{o} + t\mathbf{d} - \mathbf{r}) = 0 \quad t = \frac{(\mathbf{r} - \mathbf{o}) \cdot \mathbf{n}}{\mathbf{n} \cdot \mathbf{d}} \quad \text{if } \mathbf{d} \text{ is not orthogonal to } \mathbf{n}
\]
Implicit Surfaces - Normal

- Perpendicular to the surface
- Given by the gradient of the implicit function

\[ \mathbf{n} = \nabla f(\mathbf{p}) = \left( \frac{\partial f(\mathbf{p})}{\partial x}, \frac{\partial f(\mathbf{p})}{\partial y}, \frac{\partial f(\mathbf{p})}{\partial z} \right) \]

- E.g., for a point \( \mathbf{p} \) on a plane \( f(\mathbf{p}) = \mathbf{n} \cdot (\mathbf{p} - \mathbf{r}) = 0 \)

\[ \mathbf{n} = \nabla f(\mathbf{p}) = (n_x, n_y, n_z) \]
Triangle

- Parametric representation (barycentric coords)
  \[ p(b_1, b_2) = (1 - b_1 - b_2)p_0 + b_1p_1 + b_2p_2 \]
  \[ b_1 \geq 0 \quad b_2 \geq 0 \quad b_1 + b_2 \leq 1 \]

- Intersection is computed using a linear system
  \[ o + td = (1 - b_1 - b_2)p_0 + b_1p_1 + b_2p_2 \]

- Solution (non-degenerated triangles, not parallel to ray)
  \[
  \begin{pmatrix}
    t \\
    b_1 \\
    b_2
  \end{pmatrix} = \frac{1}{(d \times e_2) \cdot e_1}
  \begin{pmatrix}
    (s \times e_1) \cdot e_2 \\
    (d \times e_2) \cdot s \\
    (s \times e_1) \cdot d
  \end{pmatrix}
  \]
  \[ e_1 = p_1 - p_0 \]
  \[ e_2 = p_2 - p_0 \]
  \[ s = o - p_0 \]
Axis-Aligned (Bounding) Box AABB

- Boxes are represented by slabs
- Intersections of rays with slabs are analyzed to check for ray-box intersection
  - E.g. non-overlapping ray intervals within different slabs indicate that the ray misses the box

\[
\mathbf{n} \cdot (\mathbf{o} + td - \mathbf{r}) = 0
\]
\[
t = \frac{(\mathbf{r} - \mathbf{o}) \cdot \mathbf{n}}{\mathbf{n} \cdot \mathbf{d}}
\]
\[
(1, 0, 0)^T \cdot (\mathbf{o} + td - (x_{0,1}, 0, 0)^T) = 0
\]
\[
t_{x_{\min}, x_{\max}} = \frac{(x_{0,1} - o_x)}{d_x}
\]
Axis-Aligned (Bounding) Box AABB

- Overlapping ray intervals indicate intersections,
  e.g. $t_{x_{\text{min}}} < t_{y_{\text{max}}} \land t_{x_{\text{max}}} > t_{y_{\text{min}}} \Rightarrow \text{intersection}$
  (largest entering value $t$ is smaller than the smallest leaving value $t$, only positive values $t$ are considered)
Outline

– Introduction
– Ray-object intersections
– Light
– Materials
– Radiosity
– Stochastic Raytracing
The Importance of Light Modeling

Light

Color

Material

Rendering equation

Solving the rendering equation (Computing pixel colors)
Light travels along rays

Light is emitted at light sources

Incoming light is absorbed and scattered at surfaces

Cameras capture light

– How to quantify light/color? ⇒ Flux, Irradiance, Radiance
– How to quantify surface illumination? ⇒ Irradiance
– How to quantify pixel colors? ⇒ Radiance
Flux

- Radiant flux $\Phi$
  - Power
  - Radiant energy, i.e. number of photons, per time
  - Brightness, e.g., number of photons emitted by a source per time

Flux is actually radiant energy per time.

$$\Phi = \frac{dQ}{dt}$$

As photons carry varying energy depending on their wavelength, number of photons per time is an approximation that improves the intuition behind flux.
Flux Density

- Irradiance at a position $E(x)$?
  - Issues: position with zero area, no flux per position
  - Solution: infinitesimals, differentials, small quantities

- Consider a small amount of flux $d\Phi(x)$ incident to a small area $dA(x)$ around position $x$

- For $dA(x) \to 0$, we have $d\Phi(x) \to 0$, and the ratio converges to the irradiance at $x$: $E(x) = \frac{d\Phi(x)}{dA(x)}$
Flux Density - Variants

- Irradiance $E$ - incident / incoming flux per surface
- Radiosity $B$ - outgoing flux (reflected plus emitted) per surface

Irradiance – Incident flux per area
Radiosity – Outgoing flux per area
Solid Angle

- Area of a sphere surface divided by the squared sphere radius
  \[ \Omega = \frac{A}{r^2} \]
- E.g., solid angle of the entire sphere surface \( \Omega = \frac{4\pi r^2}{r^2} = 4\pi \)
  - Independent from the radius
- E.g., solid angle of a hemisphere
  \[ \Omega = \frac{1}{2} \frac{4\pi r^2}{r^2} = 2\pi \]
Infinitesimal Solid Angle and Surface Area

- \( \Omega \approx \frac{A \cos \theta}{r^2} \) is an approximation

- If an infinitesimally small area \( dA(x) \) at position \( x \) converges to zero, then the solid angle \( d\omega \) also converges to zero and the relation \( d\omega = \frac{dA(x) \cos \theta_x}{r_x^2} \) is correct in the limit
Visibility Function

- Position $\mathbf{x}$ only contributes to $\int_{\text{Surface}} \frac{\cos \theta_x}{r_x^2} dA(\mathbf{x})$ if it is visible from $\mathbf{c}$.
- Therefore,
  \[ \Omega = \int_{\text{Surface}} V(\mathbf{c}, \mathbf{x}) \frac{\cos \theta_x}{r_x^2} dA(\mathbf{x}) \]
  with $V(\mathbf{c}, \mathbf{x}) = 1$, if $\mathbf{x}$ is visible from $\mathbf{c}$ and $V(\mathbf{c}, \mathbf{x}) = 0$, if $\mathbf{x}$ is not visible from $\mathbf{c}$.

\[
\begin{align*}
V(\mathbf{c}, \mathbf{x}_1) &= 1 \\
V(\mathbf{c}, \mathbf{x}_2) &= 0 \\
V(\mathbf{c}, \mathbf{x}_3) &= 0
\end{align*}
\]
Radiance at a Position in a Direction

- Actual setting
\[ L(x, \omega) = \frac{d^2 \Phi}{dA(x) \cdot \cos \theta \cdot d\omega} \]
  - Flux that is transported through an infinitesimally small cone

- Simplified notion
\[ L(x, \omega) \]
  - Radiance \( L \) at position \( x \) in direction \( \omega \)
  - Flux that is transported along a ray
Radiance and Oriented Surfaces

- Two areas $dA_1, dA_2$ around positions $x_1, x_2$ with $x_1 = x_2$
- Angles between surface normal and flux direction $\omega$:
  $\theta_1 = 0, \theta_2 \neq 0$
- Radiance at $x_1$:
  \[ L(x_1, \omega) = \frac{d^2\Phi}{dA_1 \cdot \cos \theta_1 \cdot d\omega} = \frac{d^2\Phi}{dA^\perp \cdot d\omega} \]
- Radiance at $x_2$:
  \[ L(x_2, \omega) = \frac{d^2\Phi}{dA_2 \cdot \cos \theta_2 \cdot d\omega} = \frac{d^2\Phi}{dA^\perp \cdot d\omega} \]

Radiance describes the flux within the grey area independent from the plane (sensor) orientation.
Irradiance and Oriented Surfaces

- Irradiance at \( \mathbf{x}_1 : E(\mathbf{x}_1) = \frac{d\Phi}{dA_1} \)
- Irradiance at \( \mathbf{x}_2 : E(\mathbf{x}_2) = \frac{d\Phi}{dA_2} \)
- \( dA_2 = \frac{dA_1}{\cos \theta_2} \Rightarrow E(\mathbf{x}_2) = \cos \theta_2 \cdot E(\mathbf{x}_1) \)

\[
dA_\perp = dA_1 \cos \theta_1 = dA_2 \cos \theta_2 = dA_i \cos \theta_i
\]

\[
\Rightarrow E_\perp = \frac{E(\mathbf{x}_1)}{\cos \theta_1} = \frac{E(\mathbf{x}_2)}{\cos \theta_2} = \frac{E(\mathbf{x}_i)}{\cos \theta_i}
\]

- Lambert’s Cosine Law
  - Irradiance on a surface is proportional to the cosine of the angle between surface normal and flux direction.

Irradiance describes the effect of the flux within the grey area onto a surface. I.e., the orientation of the surface with respect to the flux direction matters.
Conservation of Radiance

- Radiosity at \( \mathbf{x}_1 \): \( B(\mathbf{x}_1) = \frac{d\Phi}{dA_1} \)
- Irradiance at \( \mathbf{x}_2 \): \( E(\mathbf{x}_2) = \frac{d\Phi}{dA_2} \neq B(\mathbf{x}_1) \)
- Radiance at \( \mathbf{x}_1 \):
  \[
  L(\mathbf{x}_1, \omega_1) = \frac{d^2\Phi}{dA_1 \cdot \cos \theta_1 \cdot d\omega_1} \quad d\omega_1 = \frac{dA_2 \cdot \cos \theta_2}{r^2}
  \]
  \[
  L(\mathbf{x}_1, \omega_1) = \frac{r^2 \cdot d^2\Phi}{dA_1 \cdot \cos \theta_1 \cdot dA_2 \cdot \cos \theta_2}
  \]
- Radiance at \( \mathbf{x}_2 \):
  \[
  L(\mathbf{x}_2, \omega_2) = \frac{d^2\Phi}{dA_2 \cdot \cos \theta_2 \cdot d\omega_2} \quad d\omega_2 = \frac{dA_1 \cdot \cos \theta_1}{r^2}
  \]
  \[
  L(\mathbf{x}_2, \omega_2) = \frac{r^2 \cdot d^2\Phi}{dA_1 \cdot \cos \theta_1 \cdot dA_2 \cdot \cos \theta_2} = L(\mathbf{x}_1, \omega_1)
  \]

Conservation of radiance. Radiance describes flux transported along a ray.
**Discussion – Inverse Square Law**

- Irradiance at an illuminated surface decreases quadratically with the distance from a light source
  - Surfaces appear darker with growing distance from light
  - Flux generated at A, arriving at $A_1$ and $A_2$: $L \cdot A \cdot \Omega$

- Areas
  
  $A_1 \sim \Omega \cdot r_1^2 \quad A_2 \sim \Omega \cdot r_2^2$

- Irradiances
  
  $E_1 \sim \frac{\Phi}{A_1} = \frac{L \cdot A \cdot \Omega}{\Omega \cdot r_1^2} \quad E_2 \sim \frac{\Phi}{A_2} = \frac{L \cdot A \cdot \Omega}{\Omega \cdot r_2^2} \quad E \sim \frac{1}{r^2}$

All planes are orthogonal to $\omega$. Thus, $\cos \theta = 1$ for all planes.
Irradiance and Radiance

- **Illumination strength at a surface** can be characterized by **irradiance** (flux per area)
  - Depends quadratically on the distance between surface and light source
- **Illumination strength at a sensor element** can be characterized by **radiance** (flux per area per solid angle)
  - Does not depend on the distance between surface and sensor
Irradiance and Radiance

Object surface receives less flux from light 2 than from light 1 (inverse square law). Both lights contribute to the illumination of the same surface element.

Camera captures the same radiance for both lights. $d\omega_1$ and $d\omega_2$ are of the same size into different directions. Light 1 and 2 contribute to different sensor elements.

Object surface

Brightness depends on irradiance.

Camera

Sensor response depends on Radiance.

Direction: $\omega_1 \neq \omega_2$

Solid angle: $d\omega_1 = d\omega_2$
Outline

- Introduction
- Ray-object intersections
- Light
- Materials
- Radiosity
- Stochastic Raytracing
Materials

– Can be described by **relating incident and exitant flux**
Light Interaction at a Surface

- Incident radiance \( L_i(p, \omega_i) \) at position \( p \) from direction \( \omega_i \) induces irradiance at \( p \):
  \[
dE_i(p, \omega_i) = L_i(p, \omega_i) \cos \theta_i d\omega_i
\]
- Flux is partially absorbed:
  \[
  dB_i(p, \omega_i) = \rho(p) dE_i(p, \omega_i)
  \]
- Reflected flux into direction \( \omega_o \)
  \[
  dL_o(p, \omega_o) \sim dB_i(p, \omega_i) \sim dE_i(p, \omega_i)
  \]
  \( 0 \leq \rho \leq 1 \) is a reflectance coefficient.

\( \omega_i \) represents the direction of the incident radiance. Per definition, all directions point away from the surface. I.e., incident radiance travels along \(-\omega_i\).
BRDF Definition

– For all pairs of directions $\omega_i$ and $\omega_o$, the ratio of outgoing radiance towards $\omega_o$ and irradiance due to incoming radiance from $\omega_i$ is referred to as BRDF: $f_r(p, \omega_i, \omega_o) = \frac{dL_o(p, \omega_o)}{dE_i(p, \omega_i)}$

– BRDF typically depends on a position and two directions.
  – Directions form a solid angle of $2\pi$ for opaque surfaces and $4\pi$ for transparent surfaces
  – Various variants. E.g., BRDF can depend on two positions for subsurface scattering $f_r(p_i, p_o, \omega_i, \omega_o) = \frac{dL_o(p_o, \omega_o)}{dE_i(p_i, \omega_i)}$
BRDF Application

- Relation between irradiance and exitant radiance
  \[ dL_o(p, \omega_o) = f_r(p, \omega_i, \omega_o) dE_i(p, \omega_i) \]

- Irradiance is induced by radiance
  \[ dL_o(p, \omega_o) = f_r(p, \omega_i, \omega_o) L_i(p, \omega_i) \cos \theta_i d\omega_i \]

- Integration over the hemisphere \( \Rightarrow \) reflectance equation
  \[ L_o(p, \omega_o) = \int_{2\pi} f_r(p, \omega_i, \omega_o) L_i(p, \omega_i) \cos \theta_i d\omega_i \]

- Reflectance equation establishes a relation between incident and exitant radiance

The portion of light from an incoming direction that is scattered in an outgoing direction
## BRDF Properties

- **Definition:** \( f_r(\omega_i, \omega_o) = \frac{dL_o(\omega_o)}{dE_i(\omega_i)} = \frac{dL_o(\omega_o)}{L_i(\omega_i) \cdot \cos \theta_i \cdot d\omega_i} \)

- **Positive:** \( f_r(\omega_i, \omega_o) \geq 0 \)

- **Helmholtz reciprocity:** \( f_r(\omega_o, \omega_i) = f_r(\omega_i, \omega_o) \)
  - Incident and exitant radiance can be reversed

- **Energy conservation:** \( \forall \omega_i : \int_{2\pi}^{+} f_r(\omega_i, \omega_o) \cos \theta_o \, d\omega_o \leq 1 \)

- **Linearity**
  
  - If a material is defined as a sum of BRDFs, the contributions of the BRDFs are added for the total outgoing radiance

  \[ \int (f_{r,1} + f_{r,2}) L_i \cos \theta_i \, d\omega_i = \int f_{r,1} L_i \cos \theta_i \, d\omega_i + \int f_{r,2} L_i \cos \theta_i \, d\omega_i \]
BRDF Materials

- Diffuse
  \[ f_{r,d}(\omega_i, \omega_o) = \frac{\rho}{\pi} \]
- Mirror
  \[ f_{r,m}(\omega_i, \omega_o) = \rho \frac{1}{\cos \theta_i \sin \theta_i} \delta(\theta_o - \theta_i) \delta(\phi_o \pm \pi - \phi_i) \]
- Specular
  \[ f_{r,s}(\omega_i, \omega_o) = \rho ((2(n \cdot \omega_i) \cdot n - \omega_i) \cdot \omega_o)^e \]

\( n, \omega_i, \omega_o \) are represented with 3D normalized vectors.
BRDF for Diffuse Reflecting Material

- Illumination $L_i(\omega_i)$
- Induced surface irradiance $dE_i(\omega_i) = L_i(\omega_i) \cdot \cos \theta_i \cdot d\omega_i$
- Overall irradiance $E = \int_{2\pi} L_i(\omega_i) \cdot \cos \theta_i \cdot d\omega_i$
- Partially absorbed. Resulting radiosity

$$B = \rho \cdot E = \int_{2\pi} \rho \cdot L_i(\omega_i) \cdot \cos \theta_i \cdot d\omega_i \quad 0 \leq \rho \leq 1$$

$$B = \int_{2\pi} L_o(\omega_o) \cdot \cos \theta_o \cdot d\omega_o = L_o \cdot \int_{2\pi} \cos \theta_o d\omega_o = L_o \cdot \pi$$

$$\rho \cdot E = \pi \cdot L_o \quad L_o = \frac{\rho}{\pi} E = \int_{2\pi} \frac{\rho}{\pi} \cdot L_i(\omega_i) \cdot \cos \theta_i \cdot d\omega_i$$

$$f_{r,d}(\omega_i, \omega_o) = \frac{\rho}{\pi} \quad \text{BRDF is constant for diffuse reflecting material}$$
Outline

– Introduction
– Ray-object intersections
– Light
– Materials
– Radiosity
– Stochastic Raytracing
Solution of the Rendering Equation

- Exitant radiances from all scene points into all directions

$$L_e(p \rightarrow \omega_o)$$

$$L(p \rightarrow \omega_o)$$

Cornell box
Rendering of the Solution

- At an arbitrarily placed and oriented sensor
  - Cast a ray through position \( p \) in an image plane into direction \( \omega_i \)
  - Lookup \( L(p \leftarrow \omega_i) = L(r_c(p, \omega_i) \rightarrow -\omega_i) = L(p' \rightarrow -\omega_i) \)
Simplified Setting

- Lambertian material
  - Exitant radiance independent from direction
  - Radiance into arbitrary direction can be computed from radiosity
    \[ L(p \rightarrow \omega_o) = \frac{B(p)}{\pi} \]
- Discretized scene representation with faces, e.g., triangles
  - Assume constant radiosity per face
  \[ \Rightarrow \text{Problem is simplified to n radiosity values for n faces} \]
  \[ \Rightarrow n \text{ instances of the rendering equation govern the solution} \]
Radiosity Integral Equation

- Rendering equation

\[ L(p \rightarrow \omega_o) = L_e(p \rightarrow \omega_o) + \]
\[ \int_S f_r(p, \omega_i \leftrightarrow \omega_o) L(x \rightarrow -\omega_i) V(p, x) \frac{\cos(\omega_i, n_p) \cos(-\omega_i, n_x)}{r_{px}^2} dA_x \]

- Radiance can be computed from radiosity for Lambertian surfaces: \( L(p \rightarrow \omega_o) = \frac{B(p)}{\pi} \)

- Radiosity equation

\[ B(p) = B_e(p) + \int_S f_r(p, \omega_i \leftrightarrow \omega_o) B(x) V(p, x) \frac{\cos(\omega_i, n_p) \cos(-\omega_i, n_x)}{r_{px}^2} dA_x \]
\[ B(p) = B_e(p) + \frac{\rho(p)}{\pi} \int_S B(x) V(p, x) \frac{\cos(\omega_i, n_p) \cos(-\omega_i, n_x)}{r_{px}^2} dA_x \]

Constant BRDF for Lambertian surfaces
Discretization of the Radiosity Equation

– Continuous form, per surface position
\[ B(p) = B_e(p) + \frac{\rho(p)}{\pi} \int_S B(x) V(p, x) \cos(\omega_i, n_p) \cos(-\omega_i, n_x) \, dA_x \]

– Discretized form, per face / triangle
\[ B_i = B_{ei} + \sum_j \rho_i F_{ij} B_j \]
\[ B_i - \sum_j \rho_i F_{ij} B_j = B_{ei} \]

– \( B_{ei} \) is a source, i.e. the emitted radiosity at face \( i \)
– \( B_i, B_j \) are unknown radiosities at faces \( i \) and \( j \)
– \( \rho_i, F_{ij} \) are known coefficients

Finite Element Method
System of Linear Equations

\[
\begin{pmatrix}
1 - \rho_1 F_{11} & -\rho_1 F_{12} & \ldots & -\rho_1 F_{1n} \\
-\rho_2 F_{21} & 1 - \rho_2 F_{22} & \ldots & -\rho_2 F_{2n} \\
\vdots & \vdots & \ddots & \vdots \\
-\rho_n F_{n1} & \ldots & -\rho_n F_{nn-1} & 1 - \rho_n F_{nn}
\end{pmatrix}
\begin{pmatrix}
B_1 \\
B_2 \\
\vdots \\
B_n
\end{pmatrix}
=
\begin{pmatrix}
B_{e1} \\
B_{e2} \\
\vdots \\
B_{en}
\end{pmatrix}
\]

Solving the Linear System

- Typically with iterative schemes, e.g. relaxed Jacobi
  - Initialize, e.g., $B^0_i = 0$
  - Iteratively update $B^{l+1}_i = B^l_i + \frac{\lambda_i}{1-\rho_i F_{ii}} (B_{ei} - (B^l_i - \sum_j \rho_i F_{ij} B^l_j))$

- Intuition
  - Changes from $B^l_i$ to $B^{l+1}_i$ are proportional to $B_{ei} - (B^l_i - \sum_j \rho_i F_{ij} B^l_j)$
  - If $B_{ei} - (B^l_i - \sum_j \rho_i F_{ij} B^l_j) = 0$, i.e. $B^l_i - \sum_j \rho_i F_{ij} B^l_j = B_{ei}$, the solver has converged and $B^{l+1}_i = B^l_i$

Superscript indicates solver iteration
$\lambda_i$ is a user-defined parameter that governs the solver convergence
Outline

- Introduction
- Ray-object intersections
- Light
- Materials
- Radiosity
- Stochastic Raytracing
Concept

- Approximately evaluate the integral
  \[ \int_{\Omega} f_r(p, \omega_i \leftrightarrow \omega_o) L(p \leftarrow \omega_i) \cos(\omega_i, n_p) d\omega_i \]
  by
  - Tracing rays into randomly sampled 2D directions
  - Computing the incoming radiances

- Integral is approximated with
  \[ \sum_i f_r(p, \omega_i \leftrightarrow \omega_o) L(p \leftarrow \omega_i) \cos(\omega_i, n_p) \Delta\Omega_i \]
  - 2 dimensional sample directions \( \omega_i = (\theta_i, \phi_i) \)
  - \( \Delta\Omega_i \) is an approximation of the solid angle of sample direction \( \omega_i = (\theta_i, \phi_i) \)
Properties

- Benefits
  - Processes only evaluations of the integrand at arbitrary surface points in the domain
  - Appropriate for integrals of arbitrary dimensions
  - Allows for non-uniform sample patterns / adaptive sample sizes
  - Works for a large variety of integrands, e.g., it handles discontinuities
Properties

- Drawbacks
  - Using $n$ samples, the scheme converges to the correct result with $O(n^{1/2})$
  - I.e., to half the error, $4n$ samples are required
  - Errors are perceived as noise, i.e. pixels are arbitrarily too bright or dark (due to the erroneous approximation of the sample size)
  - Evaluation of the integrand at a point is expensive (ray intersections tests)
Monte Carlo Estimator - Non-uniform Random Variables

- PDF $p(x)$
- Estimator $F_N = \frac{1}{N} \sum_{i=1}^{N} \frac{f(X_i)}{p(X_i)}$
- Integral
  - $\int_a^b f(x)dx \approx \frac{1}{N} \sum_{i=1}^{N} \frac{f(X_i)}{p(X_i)} = \sum_{i=1}^{N} f(X_i) \frac{1}{N p(X_i)}$
- Function value $f(X_i)$
- Approximate sample size $\frac{1}{N p(X_i)}$
Monte Carlo Estimator - Integration over a Hemisphere

- Approximate computation of the irradiance at a point

\[ E_i(p) = \int_{2\pi} L_i(p, \omega) \cos \theta d\omega \]

\[ = \int_{\theta=0}^{\frac{\pi}{2}} \int_{\phi=0}^{2\pi} L_i(p, \theta, \phi) \cos \theta \sin \theta d\theta d\phi \]

- Estimator

\[ F_N = \frac{1}{N} \sum_{i=1}^{N} \frac{f(X_i)}{p(X_i)} = \frac{1}{N} \sum_{i=1}^{N} \frac{L_i(p, \theta_i, \phi_i) \cos \theta_i \sin \theta_i}{p(\theta_i, \phi_i)} \]

- Choosing a PDF

  - Should be similar to the shape of the integrand
  - As incident radiance is weighted with \( \cos \theta \), it is appropriate to generate more samples close to the top of the hemisphere
  - \( p(\theta, \phi) \propto \cos \theta \)

This flexibility is an important aspect of Monte Carlo integration.
Monte Carlo Estimator - Integration over a Hemisphere

- Probability distribution

\[
\int_{2\pi}^{\pi} c \ p(\omega) d\omega = 1 \\
\int_0^{2\pi} \int_0^{\pi/2} c \ \cos \theta \sin \theta \ d\theta \ d\phi = 1 \\
c \ \frac{2\pi}{1+1} = 1 \\
c = \frac{1}{\pi} \\
p(\theta, \phi) = \frac{\cos \theta \sin \theta}{\pi}
\]

- Estimator

\[
F_N = \frac{1}{N} \sum_{i=1}^{N} \frac{L_i(p, \theta_i, \phi_i) \cos \theta_i \sin \theta_i}{p(\theta_i, \phi_i)} \\
= \frac{\pi}{N} \sum_{i=1}^{N} L_i(p, \theta_i, \phi_i) \approx \int_0^{2\pi} \int_0^{\pi/2} L_i(p, \theta, \phi) \cos \theta \sin \theta d\theta d\phi
\]

If \( \theta \) and \( \phi \) are sampled according to PDF \( p(\theta, \phi) \)
Monte Carlo Estimator - Integration over a Hemisphere

- Integral \( \int_{0}^{2\pi} \int_{0}^{\frac{\pi}{2}} L_i(p, \theta, \phi) \cos \theta \sin \theta \, d\theta \, d\phi \)
- PDF \( p(\theta, \phi) = \frac{\cos \theta \sin \theta}{\pi} \)
- Estimator \( \frac{\pi}{N} \sum_{i=1}^{N} L_i(p, \theta_i, \phi_i) \)
  \( = \sum_{i=1}^{N} L_i(p, \theta_i, \phi_i) \cos \theta_i \sin \theta_i \frac{\frac{\pi}{N} \cos \theta_i \sin \theta_i}{N \cos \theta_i \sin \theta_i} \)
- Function value \( L_i(p, \theta_i, \phi_i) \cos \theta_i \sin \theta_i \) for direction \((\theta_i, \phi_i)\)
- Approximate sample size / solid angle \( \frac{\pi}{N \cos \theta_i \sin \theta_i} \)
Monte Carlo Integration - Steps

– Choose an appropriate probability density function
– Generate random samples according to the PDF
– Evaluate the function for all samples
– Accumulate sample values weighted with their approximate sample size
Inversion Method

- $P$ and $P^{-1}$ are continuous functions
- Start with the desired PDF $p(x)$
- Compute $P(x) = \int_0^x p(x')dx'$
- Compute the inverse $P^{-1}(x)$
- Obtain a uniformly distributed variable $\xi$
- Compute $X_i = P^{-1}(\xi)$ which adheres to $p(x)$
Rejection Method

- Sample generation
  - Generate a uniform random sample $0 \leq \xi < 1$
  - Generate a sample $X$ according to $p(x)$
  - Accept $X$ if $\xi \cdot c \cdot p(X) \leq f(X)$

[Pharr, Humphreys]