Lecture 9: Monte Carlo Rendering
Chapters 4 and 5 in Advanced GI

Fall 2004
Kavita Bala
Computer Science
Cornell University

Homework

• HW 1 out, due Oct 5

• Assignments done separately
  – Might revisit this policy for later assignments
Stochastic Ray Tracing

• Parameters?
  – # starting rays per pixel
  – # random rays for each surface point
    (branching factor)

• Path Tracing
  – Branching factor == 1

Algorithm so far ...

• Shoot # viewing rays through each pixel

• Shoot # indirect rays, sampled over hemisphere
  – Path tracing shoots only 1 indirect ray

• Terminate recursion using Russian Roulette
Performance/Error

• Want better quality with smaller number of samples
  – Fewer samples/better performance
  – Stratified sampling
  – Quasi Monte Carlo: well-distributed samples

• Faster convergence
  – Importance sampling: next-event estimation

Stratified Sampling

• Samples could be arbitrarily close

• Split integral in subparts

\[ I = \int_{x_1} f(x)dx + \ldots + \int_{x_N} f(x)dx \]

• Estimator

\[ I_{strat} = \frac{1}{N} \sum_{i=1}^{N} \frac{f(x_i)}{p(x_i)} \]

• Variance:

\[ \sigma_{strat} \leq \sigma_{sec} \]
Higher Dimensions

- Stratified grid sampling:
  \[ N^d \] samples

- N-rooks sampling:
  \[ N \] samples

Quasi Monte Carlo

- Eliminates randomness to find well-distributed samples
- Samples are deterministic but “appear” random
Performance/Error

- Want better quality with smaller number of samples
  - Fewer samples/better performance
  - Stratified sampling
  - Quasi Monte Carlo: well-distributed samples

- Faster convergence
  - Importance sampling: next-event estimation

Next Event Estimation

\[ L(x \rightarrow \Theta) = L_e + L_{\text{direct}} + L_{\text{indirect}} \]

\[ = L_e + \int_{\Omega_d} f_r \cdot \cos + \int_{\Omega_i} f_r \cdot \cos \]

- So … sample direct and indirect with separate MC integration
How to sample direct illumination

• Sampling a single light source

• Sampling for many lights

Direct Illumination

\[ L(x \rightarrow \Theta) = \int_{A_{\text{source}}} f_r(x_r, -\Psi \leftrightarrow \Theta) \cdot L(y \rightarrow \Psi) \cdot G(x, y) \cdot dA_y \]

\[ G(x, y) = \frac{\cos(n_{x, \Theta}) \cos(n_{y, \Psi}) \text{Vis}(x, y)}{r_{xy}^2} \]

hemisphere integration  area integration
Estimator for direct lighting

- Pick a point on the light’s surface with pdf $p(y)$

- For N samples, direct light at point $x$ is:

$$E(x) = \frac{1}{N} \sum_{i=1}^{N} f_{r} L_{source} \frac{\cos \theta_{x} \cos \theta_{y_{i}}}{r_{x_{i}y_{i}}^{2}} \frac{Vis(x, y_{i})}{p(y_{i})}$$

PDF for sampling light

- Uniform $p(y) = \frac{1}{\text{Area}_{source}}$

- Pick a point uniformly over light’s area
  - Can stratify samples

- Estimator:

$$E(x) = \frac{\text{Area}_{source}}{N} \sum_{i=1}^{N} f_{r} L_{source} \frac{\cos \theta_{x} \cos \theta_{y_{i}}}{r_{x_{i}y_{i}}^{2}} \frac{Vis(x, y_{i})}{p(y_{i})}$$
Different pdfs

• Uniform

\[ p(y) = \frac{1}{\text{Area}_{\text{source}}} \]

• Solid angle sampling
  – Removes cosine and distance from integrand
  – Better when significant foreshortening

\[ E(x) = \frac{1}{N} \sum_{i=1}^{N} f_r L_{\text{source}} \frac{\cos \theta_x \cos \theta_{r_i}}{r_{xy_i}^2} \frac{\text{Vis}(x, \bar{y}_i)}{p(\bar{y}_i)} \]

Parameters

• How to distribute paths within light source?
  – Uniform
  – Solid angle
  – What about light distribution?

• How many paths (“shadow-rays”)?
  – Total?
  – Per light source? (~intensity, importance, …)
Formulation over all lights

- When $M$ is large, each direct lighting sample is very expensive
- We would like to importance sample the lights
- Instead of $M$ integrals

\[
L(x \rightarrow \Theta) = \frac{1}{M} \sum_{i=1}^{M} \int_{A_{source}} f_r(x,-\Psi \leftrightarrow \Theta) \cdot L_{source}(y \rightarrow -\Psi') \cdot G(x,y) \cdot dA_y
\]

- Formulation over 1 integration domain

\[
L(x \rightarrow \Theta) = \int_{A_{all \ lights}} f_r(x,-\Psi \leftrightarrow \Theta) \cdot L_{source}(y \rightarrow -\Psi') \cdot G(x,y) \cdot dA_y
\]

Why?

- Do not need a minimum of $M$ rays/sample
- Can use only one ray/sample

- Still need N samples, but 1 ray/sample

- Ray is distributed over the whole integration domain
  - Can importance sample the lights
Anti-aliasing

• Can piggy-back on the anti-aliasing of pixel

How to sample the lights?

• A discrete pdf $p_L(k_i)$ picks the light $k_i$

• A surface point is then picked with pdf $p(y_i|k_i)$

• Estimator with $N$ samples:

$$E(x) = \frac{1}{N} \sum_{i=1}^{N} \frac{f_r L_{source} G(x, \bar{y}_i)}{p_L(k_i) p(y_i | k_i)}$$
Strategies for picking light

- Uniform \[ p_L(k) = \frac{1}{M} \]

- Area \[ p_L(k) = \frac{A_k}{\sum A_k} \]

- Power \[ p_L(k) = \frac{P_k}{\sum P_k} \]

Don’t take visibility into account

Research on many lights

- Ward ‘91

- Shirley, Wang, Zimmerman ‘94

- Fernandez, Bala, Greenberg ‘02

- Wald and Slusallek ‘03
Alternative direct paths

- Shoot paths at random over hemisphere; check if they hit light source

  - paths not used efficiently
  - noise in image
  - might work if light source occupies large portion on hemisphere

Alternative direct paths

1 path / point  16 paths / point  256 paths / point
Direct paths

• Different path generators produce different estimators and different error characteristics
• Direct illumination general algorithm:

```cpp
compute_radiance (point, direction)
    est_rad = 0;
    for (i=0; i<n; i++)
        p = generate_path;
        est_rad += energy_transfer(p) / probability(p);
    est_rad = est_rad / n;
    return(est_rad);
```

Stochastic Ray Tracing

• Sample area of light source for direct term
• Sample hemisphere with random rays for indirect term

• Optimizations:
  – Stratified sampling
  – Importance sampling
  – Combine multiple probability density functions into a single PDF
Indirect Illumination

• Paths of length > 1

• Many different path generators possible

• Efficiency depends on:
  – BRDFs along the path
  – Visibility function
  – ...

Indirect paths - surface sampling

• Simple generator (path length = 2):
  – select point on light source
  – select random point on surfaces

  – per path:
    ▪ 2 visibility checks
Indirect paths - surface sampling

- Indirect illumination (path length 2):

\[
L(x \rightarrow \Theta) = \int \int_l L(y \rightarrow \psi_1 f_j(z, z, -\psi_1) G(z, y) f_j(x, -\psi_2) G(z, x) dA_1 dA_2
\]

\[
\langle L(x \rightarrow \Theta) \rangle = \frac{1}{N} \sum_{i=1}^N L(y_i \rightarrow \psi_0 f_j(z, z, -\psi_1) G(z, y_i) f_j(x, -\psi_2) G(z, x) / p_j(y_i) p_j(z, x)
\]

- 2 visibility values cause noise
  - which might be 0

Indirect paths - source shooting

- Shoot ray from light source, find hit location
- Connect hit point to receiver

- per path:
  - 1 ray intersection
  - 1 visibility check
Indirect paths - receiver gathering

- Shoot ray from receiver point, find hit location
- Connect hit point to random point on light source

- per path:
  - 1 ray intersection
  - 1 visibility check

Indirect paths

Surface sampling
- 2 visibility terms; can be 0

Source shooting
- 1 visibility term
  - 1 ray intersection

Receiver gathering
- 1 visibility term
  - 1 ray intersection
More variants ...

- Shoot ray from receiver point, find hit location
- Shoot ray from hit point, check if on light source

\[ \text{per path:} \]
- 2 ray intersections
- \( L_0 \) might be zero

Indirect paths

- Same principles apply to paths of length > 2
  - generate multiple surface points
  - generate multiple bounces from light sources and connect to receiver
  - generate multiple bounces from receiver and connect to light sources
  - ...

- Estimator and noise characteristics change with path generator
Indirect paths

\[
\text{compute_radiance (point, direction)}
\]
\[\text{est_rad} = 0;\]
\[\text{for } (i=0; i<n; i++)\]
\[\quad q = \text{generate_indirect_path};\]
\[\quad \text{est_rad} += \text{energy_transfer}(q) / p(q);\]
\[\quad \text{est_rad} = \text{est_rad} / n;\]
\[\text{return(\text{est_rad});}\]

Stochastic Ray Tracing

- Sample area of light source for direct term
- Sample hemisphere with random rays for indirect term
- Optimizations:
  - Stratified sampling
  - Importance sampling
  - Combine multiple probability density functions into a single PDF
Sampling strategies

• Uniform sampling over the hemisphere

\[ L(x \rightarrow \Theta) = \int_{\Omega_x} L(x \leftarrow \Psi) \cdot f_r(\Psi \leftrightarrow \Theta) \cdot \cos(\Psi, n_x) \cdot d\omega_\Psi \]

\[ p(\Theta) = \frac{1}{2\pi} \]

Sampling strategies

• Sampling according to the cosine factor

\[ L(x \rightarrow \Theta) = \int_{\Omega_x} L(x \leftarrow \Psi) \cdot f_r(\Psi \leftrightarrow \Theta) \cdot \cos(\Psi, n_x) \cdot d\omega_\Psi \]

\[ p(\Theta) = \frac{\cos \theta}{\pi} \]
**Sampling strategies**

- Sampling according to the BRDF

\[
L(x \rightarrow \Theta) = \int_{\Omega_x} L(x \leftarrow \Psi) f_r(\Psi \leftarrow \Theta) \cos(\Psi, n_x) \cdot d\omega_y
\]

\[
p(\Theta) \sim f_r(\Theta \leftrightarrow \Psi)
\]

**Example: sample according to BRDF**

- Discrete pdf \(q_1, q_2, q_3\)

\[
q_1 + q_2 + q_3 = 1
\]

\[
L_{\text{indirect}} = L_{\text{diffuse}} + L_{\text{specular}}
\]

\[
\langle L_{\text{indirect}} \rangle = \begin{cases} 
\frac{L(x \leftarrow \Psi_i)k_i \cos(N, \Psi_i)}{q_1 p_1(\Psi_i)} | \xi < q_1 \\
\frac{L(x \leftarrow \Psi_i)k_i \cos^a(R, \Psi_i) \cos(N, \Psi_i)}{q_2 p_2(\Psi_i)} | q_1 \leq \xi < q_2 \\
0 | \text{otherwise}
\end{cases}
\]
Sampling strategies

- Sampling according to the BRDF times the cosine

\[ L(x \to \Theta) = \int_{\Omega_x} L(x \leftarrow \Psi) \cdot f_r(\Psi \leftrightarrow \Theta) \cdot \cos(\Psi, n_x) \cdot d\omega \]

\[ p(\Theta) \sim f_r(\Theta \leftrightarrow \Psi) \cos \theta \]

Comparison

With importance sampling (brdf on sphere)  Without importance sampling

© Kavita Bala, Computer Science, Cornell University