Lecture 7: Monte Carlo Rendering

Chapters 3 and 4 in Advanced GI

Fall 2004

Kavita Bala
Computer Science
Cornell University

Why Monte Carlo?

$$L(x \to \Theta) = L_e(x \to \Theta) +$$
$$\int_{\Omega_x} L_e(y \to -\Psi) f_r(\Psi \leftrightarrow \Theta) \cdot \cos(\Psi, n_x) \cdot d\omega_\Psi +$$
$$\ldots\ldots$$

- Analytical integration is difficult
- Therefore, need numerical techniques

Monte Carlo Integration

- Numerical tool to evaluate integrals
- Use sampling
- Stochastic errors
- Unbiased
  - on average, we get the right answer!

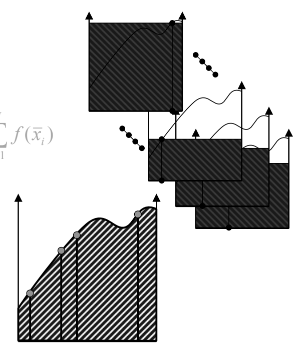
More samples

Secondary estimator

Generate N random samples x_i

Estimator: $\langle I \rangle = I_{sec} = \frac{1}{N}\sum_{i=1}^{N} f(\bar{x}_i)$

Variance $\sigma_{sec}^2 = \sigma_{prim}^2 / N$

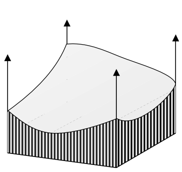


Monte Carlo Integration - 2D

- MC Integration works well for higher dimensions
- Unlike quadrature

$$I = \int_a^b \int_c^d f(x,y)\,dx\,dy$$

$$\langle I \rangle = \frac{1}{N}\sum_{i=1}^{N} \frac{f(x_i, y_i)}{p(x_i, y_i)}$$

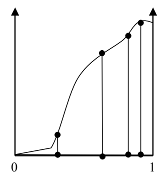


MC Advantages

- Convergence rate of O($\frac{1}{\sqrt{N}}$)
- Simple
  - Sampling
  - Point evaluation
  - Can use black boxes
- General
  - Works for high dimensions
  - Deals with discontinuities, crazy functions,…

Importance Sampling

- Better use of samples by taking more samples in 'important' regions, i.e. where the function is large



Importance Sampling

- Generate samples from density function p(x)

$$\langle I \rangle = \frac{1}{N}\sum_{i=1}^{N}\frac{f(x_i)}{p(x_i)}$$

- Optimal p(x)? $p(x) \approx f(x) / \int f(x)dx$
- General principle:
  - Closer shape of p(x) is to shape of f(x), lower the variance
- Variance can *increase* if p(x) is chosen badly

MC integration - Non-Uniform

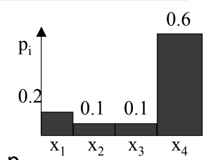
- Some parts of the integration domain have higher importance
- Generate samples according to density function p(x)

$$\langle I \rangle = \frac{1}{N}\sum_{i=1}^{N}\frac{f(x_i)}{p(x_i)}$$

- Estimator?
- What is optimal p(x)? $p(x) \approx f(x) / \int f(x)dx$

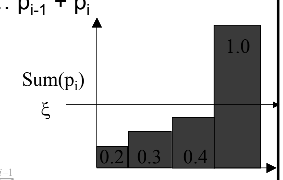
How to sample according to pdf?

- Consider discrete events x_i
  - with probability p_i



- Select x_i if:

  $p_1 + \ldots p_{i-1} < \xi < p_1 + \ldots p_{i-1} + p_i$



$$\sum_{j=1}^{i-1} p_j < \xi < \sum_{j=1}^{i} p_j$$

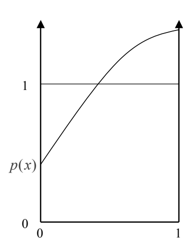
$$P(x_i) = P(\xi \in [\sum_{j=1}^{i-1} p_j, \sum_{j=1}^{i} p_j])$$

$$P(a < \xi < b) = (b - a)$$

$$P(x_i) = P(\xi \in [\sum_{j=1}^{i-1} p_j, \sum_{j=1}^{i} p_j]) = \sum_{j=1}^{i} p_j - \sum_{j=1}^{i-1} p_j = p_i$$

Non-Uniform Samples

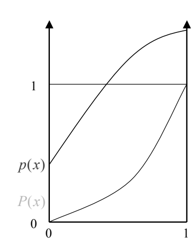
- 1) Choose a normalized probability density function *p(x)*



Non-Uniform Samples

- 1) Choose a normalized probability density function *p(x)*
- 2) Integrate to get a cumulative probability distribution function *P(x)*:

$$P(x) = \int_0^x p(t)dt$$



Note this is similar to computing $\sum_{j=1}^{i} p_j$

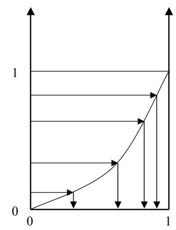
Non-Uniform Samples

- 1) Choose a normalized probability density function *p(x)*
- 2) Integrate to get a probability distribution function *P(x)*:

$$P(x) = \int_0^x p(t)dt$$

- 3) Invert *P*:

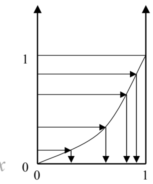
$$x = P^{-1}(\xi)$$



Note this is similar to going from y axis to x in discrete case!

Non-Uniform Samples

- This transforms uniform samples into non-uniform samples!
- Why? $\Pr(x \le y) = CDF(y) = \int_{-\infty}^{y} p(x)dx$



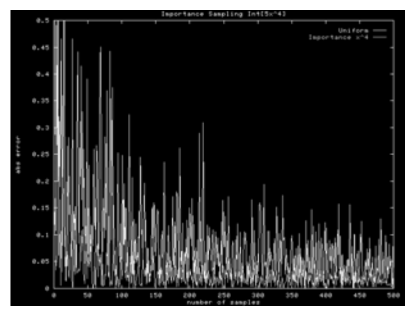
- Need:
  - CDF P(x)
  - Inverse CDP P^{-1}

Importance Sampling

$$p(x) = \frac{f(x)}{\int_D f(x)}$$

- General principle:
  The closer the shape of p(x) is to the shape of f(x), the lower the variance
- Variance can also increase if p(x) is chosen badly

Numerical Example



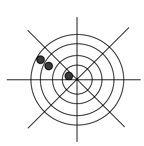
Example: Sampling according to p(x)

- Area of a circle: $A = \frac{1}{\pi}\int_0^{2\pi}\int_0^1 r dr d\theta = \frac{1}{\pi}\left[\frac{r^2}{2}\theta\right] = 1$

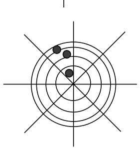
$$A = \int_0^{2\pi}\int_0^1 f(r,\theta) dr d\theta$$

$$f(r,\theta) = \frac{r}{\pi}$$

- Uniform sampling of r and θ



- Equal area sampling of r and θ

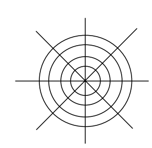


Example: Sampling according to p(x)

$$A = \int_0^{2\pi}\int_0^1 f(r,\theta) dr d\theta$$

$$p(r,\theta) = f(r,\theta) = \frac{r}{\pi}$$

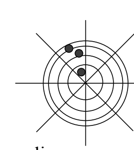
$$CDF(r,\theta) = P(r,\theta) = \int_0^{\theta}\int_0^{r} \frac{r}{\pi} dr d\theta = r^2 \frac{\theta}{2\pi}$$



$$y = x^2 \to x = \sqrt{y} \Leftrightarrow P_r(r) = r^2, P^{-1}{}_r(r) = \sqrt{r}$$

$$x_\theta = P^{-1}{}_\theta(\xi_1) = 2\pi\xi_1$$

$$x_r = P^{-1}{}_r(\xi_2) = \sqrt{\xi_2}$$



Equal area sampling

Cosine distribution

$$f = \frac{1}{\pi}\int_0^{2\pi}\int_0^{1}\cos\theta\sin\theta\, d\theta\, d\phi$$

$$p(\theta,\phi) = \frac{\cos\theta\sin\theta}{\pi}$$

$$CDF(\theta,\phi) = \int_0^{\theta}\int_0^{\phi}\frac{\cos\theta\sin\theta}{\pi}\, d\theta\, d\phi = (1-\cos^2\theta)\frac{\phi}{2\pi}$$

$$F(\theta) = 1-\cos^2\theta$$

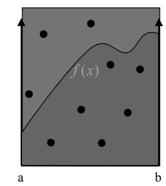
$$F(\phi) = \frac{\phi}{2\pi}$$

$$\phi_i = 2\pi\xi_1 \qquad \theta_i = \cos^{-1}\sqrt{\xi_2}$$

Rejection Methods

- Pick ξ_1, ξ_2

$$I = \int_a^b f(x)dx$$



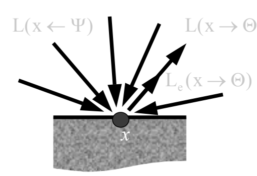
- If $\xi_2 < f(\xi_1)$, select ξ_2
- Is this efficient? What determines efficiency? A(f)/A(rectangle)

Summary

- What is Monte Carlo Integration?
- Estimators
- Sampling non-uniform distributions
  – Importance Sampling
  – Rejection Sampling
- Next time: How does MC apply to RE

MC applied to RE

$$\underline{L(x\to\Theta)} = \underline{L_e(x\to\Theta)} + \int_{\Omega_x} f_r(\Psi\leftrightarrow\Theta)\cdot \underline{L(x\leftarrow\Psi)}\cdot\cos(\Psi,n_x)\cdot d\omega_\Psi$$

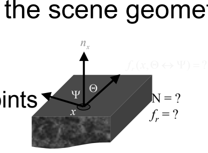


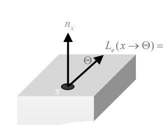
Radiance Evaluation

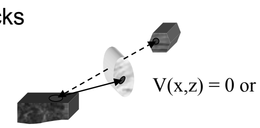
- Many different light paths contribute to single radiance value
  – many paths are unimportant
- Tools we need:
  – generate the light paths
  – sum all contributions of all light paths
  – clever techniques to select important paths

Assumptions: black boxes

- Can query the scene geometry and materials
  – surface points
  – light sources
  – visibility checks
  – tracing rays







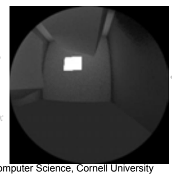
Rendering Equation

$$L(x \to \Theta) = L_e(x \to \Theta) + \int_{\Omega_x} f_r(\Psi \leftrightarrow \Theta) \cdot L(x \leftarrow \Psi) \cdot \cos(\Psi, n_x) \cdot d\omega_\Psi$$

function to integrate over all incoming directions over the hemisphere around x

Value we want

$= L_e + \int_{\Omega_x}$



$\cdot f_r \cdot \cos$

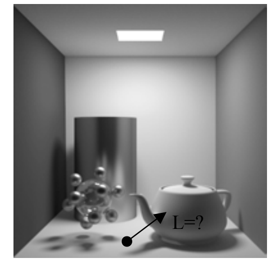
How to compute?

L(x→Θ) = ?

Check for $L_e(x \to \Theta)$

Now add $L_r(x \to \Theta)$ =

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



How to compute?

- Monte Carlo!
- Generate random directions on hemisphere Ω_x, using pdf p(Ψ)

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

$$\langle L(x \to \Theta) \rangle = \frac{1}{N} \sum_{i=1}^{N} \frac{f_r(\Psi_i \leftrightarrow \Theta) \cdot L(x \leftarrow \Psi_i) \cdot \cos(\Psi_i, n_x)}{p(\Psi_i)}$$

How to compute?

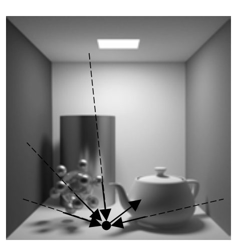
Generate random directions Ψ_i

$$\langle L \rangle = \frac{1}{N} \sum_{i=1}^{N} \frac{f_r(\ldots) \cdot L(x \leftarrow \Psi_i) \cdot \cos(\ldots)}{p(\Psi_i)}$$

- evaluate brdf
- evaluate cosine term
- evaluate $L(x \leftarrow \Psi_i)$

How to compute?

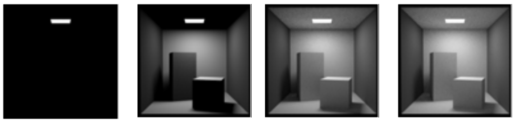
- evaluate $L(x \leftarrow \Psi_i)$?
- Radiance is invariant along straight paths
- vp(x, Ψ_i) = first visible point
- $L(x \leftarrow \Psi_i) = L(vp(x, \Psi_i) \to \Psi_i)$



How to compute? Recursion ...

- Recursion ….
- Each additional bounce adds one more level of indirect light
- Handles ALL light transport
- “Stochastic Ray Tracing”

When to end recursion?



- Contributions of further light bounces become less significant
- If we just ignore them, estimators will be biased!

Russian Roulette

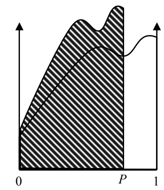
Integral

$$I = \int_0^1 f(x)dx = \int_0^1 \frac{f(x)}{P} P dx = \int_0^P \frac{f(y/P)}{P} dy$$

Estimator

$$\langle I_{roulette} \rangle = \begin{cases} \frac{f(x_i)}{P} & \textbf{if } x_i \le P, \\ 0 & \textbf{if } x_i > P. \end{cases}$$

Variance $\sigma_{roulette} > \sigma$



Russian Roulette

- Pick some ‘absorption probability’ α
  - probability 1-α that ray will bounce
  - estimated radiance becomes L/ (1-α)
- E.g. α = 0.9
  - only 1 chance in 10 that ray is reflected
  - estimated radiance of that ray is multiplied by 10
  - instead of shooting 10 rays, we shoot only 1, but count the contribution of this one 10 times