Advanced Computer Graphics Stochastic Raytracing 1

Matthias Teschner

UNI FREIBURG

From Radiosity to Raytracing

- Radiosity equation governs light transport for diffuse surfaces. ⇨ How to describe light transport for general surfaces?
- How to solve for the light transport?
- How to compute the relevant part of the light transport towards a sensor?

Stochastic Raytracing

- Light transport towards the sensor requires to solve
  $$L(\boldsymbol{p} \rightarrow \boldsymbol{\omega}_o) = L_e(\boldsymbol{p} \rightarrow \boldsymbol{\omega}_o) + \int_S f_r(\boldsymbol{p}, \boldsymbol{\omega}_i \leftrightarrow \boldsymbol{\omega}_o) L(\boldsymbol{p}' \rightarrow -\boldsymbol{\omega}_i) G(\boldsymbol{p}, \boldsymbol{p}') \mathrm{d}A_{p'}$$
- Monte Carlo integration approximates the reflectance integral
  - E.g., $\sum_i f_r(\boldsymbol{p}, \boldsymbol{\omega}_i \leftrightarrow \boldsymbol{\omega}_o) L(\boldsymbol{p}' \rightarrow -\boldsymbol{\omega}_i) G(\boldsymbol{p}, \boldsymbol{p}') A_{p'}$
  - Trace rays into the scene
  - Compute radiance along this ray
  - Associate an area / solid angle with each ray
  - Accumulate all contributions

Outline

- Diffuse vs. general global illumination
- Monte Carlo integration
- Sampling of random variables

Governing Equations

- Rendering equation
  - Governing equation for general global illumination methods

$$L(\boldsymbol{p} \to \boldsymbol{\omega}_o) = L_e(\boldsymbol{p} \to \boldsymbol{\omega}_o) + \int_S f_r(\boldsymbol{p}, \boldsymbol{\omega}_i \leftrightarrow \boldsymbol{\omega}_o) L(\boldsymbol{x} \to -\boldsymbol{\omega}_i) V(\boldsymbol{p}, \boldsymbol{x}) \frac{\cos(\boldsymbol{\omega}_i, \boldsymbol{n}_p) \cos(-\boldsymbol{\omega}_i, \boldsymbol{n}_x)}{r_{px}^2} \mathrm{d}A_x$$

- Radiosity equation
  - Governing equation for diffuse global illumination methods

$$L(\boldsymbol{p} \to \boldsymbol{\omega}_o) = \frac{B(\boldsymbol{p})}{\pi} \qquad f_r(\boldsymbol{p}, \boldsymbol{\omega}_i \leftrightarrow \boldsymbol{\omega}_o) = \frac{\rho(\boldsymbol{p})}{\pi}$$

$$B(\boldsymbol{p}) = B_e(\boldsymbol{p}) + \frac{\rho(\boldsymbol{p})}{\pi} \int_S B(\boldsymbol{x}) V(\boldsymbol{p}, \boldsymbol{x}) \frac{\cos(\boldsymbol{\omega}_i, \boldsymbol{n}_p) \cos(-\boldsymbol{\omega}_i, \boldsymbol{n}_x)}{r_{px}^2} \mathrm{d}A_x$$

A Solution Strategy - Radiosity

- Finite Element Method
- Start with a continuous form / function

$$B(\boldsymbol{p}) = B_e(\boldsymbol{p}) + \frac{\rho(\boldsymbol{p})}{\pi} \int_S B(\boldsymbol{x}) V(\boldsymbol{p}, \boldsymbol{x}) \frac{\cos(\boldsymbol{\omega}_i, \boldsymbol{n}_p) \cos(-\boldsymbol{\omega}_i, \boldsymbol{n}_x)}{r_{px}^2} \mathrm{d}A_x$$

- Discretization

$$\boldsymbol{B} = \boldsymbol{B_e} + \boldsymbol{F}\boldsymbol{B}$$

$$\boldsymbol{B} = (\boldsymbol{I} - \boldsymbol{F})^{-1} \boldsymbol{B_e}$$

- Solving for a vector with unknown radiosities

$$(\boldsymbol{I} - \boldsymbol{F})^{-1} = \sum_{k=0}^{\infty} \boldsymbol{F}^k$$

$$\boldsymbol{B} = \boldsymbol{B_e} + \boldsymbol{F}\boldsymbol{B_e} + \boldsymbol{F}\boldsymbol{F}\boldsymbol{B_e} + \boldsymbol{F}\boldsymbol{F}\boldsymbol{F}\boldsymbol{B_e} + \dots$$

An Alternative Strategy

- Start with the general form of the rendering equation, e.g. in hemispherical form

$$L(\boldsymbol{p} \rightarrow \boldsymbol{\omega}_o) = L_e(\boldsymbol{p} \rightarrow \boldsymbol{\omega}_o) + \int_\Omega f_r(\boldsymbol{p}, \boldsymbol{\omega}_i \leftrightarrow \boldsymbol{\omega}_o) L(\boldsymbol{p} \leftarrow \boldsymbol{\omega}_i) \cos(\boldsymbol{\omega}_i, \boldsymbol{n}_p) \mathrm{d}\omega_i$$

- Solving for a function of unknown radiances $L(\boldsymbol{p} \rightarrow \boldsymbol{\omega}_o)$
  - I.e., radiance at all surface positions into all directions

Operator Form of the Rendering Equation

- Operators transform a function into another one
- Scattering operator

$$(\boldsymbol{K}h)(\boldsymbol{p} \rightarrow \boldsymbol{\omega}_o) = \int_\Omega f_r(\boldsymbol{p}, \boldsymbol{\omega}_i \leftrightarrow \boldsymbol{\omega}_o) h(\boldsymbol{p} \leftarrow \boldsymbol{\omega}_i) \cos(\boldsymbol{\omega}_i, \boldsymbol{n}_p) \mathrm{d}\omega_i$$

  - Applied to an incident radiance function $L(\boldsymbol{p} \leftarrow \boldsymbol{\omega}_i)$, exitant radiance after one bounce / scattering step is returned

$$L(\boldsymbol{p} \rightarrow \boldsymbol{\omega}_o) = (\boldsymbol{K}L)(\boldsymbol{p} \leftarrow \boldsymbol{\omega}_i)$$

  - $\boldsymbol{K}$ operates on an entire function, i.e. on all incident radiances for all positions $\boldsymbol{p}$ and direction $\boldsymbol{\omega}_i$

See, e.g.: Eric Veach: Robust Monte Carlo Methods for Light Transport Simulation, Ph.D. thesis, Stanford University, 1997.

Operator Form of the Rendering Equation

– Propagation operator

$$(\boldsymbol{G}h)(\boldsymbol{p} \leftarrow \boldsymbol{\omega}_i) = h(\boldsymbol{p}' \rightarrow -\boldsymbol{\omega}_i)$$

p′ indicates the raycast operator applied to *p*

– Applied to an exitant radiance function $L(\boldsymbol{p}' \rightarrow -\boldsymbol{\omega}_i)$, incident radiance at $\boldsymbol{p}$ from direction $\boldsymbol{\omega}_i$ is returned

$$L(\boldsymbol{p} \leftarrow \boldsymbol{\omega}_i) = (\boldsymbol{G}L)(\boldsymbol{p}' \rightarrow -\boldsymbol{\omega}_i)$$

– Radiance is preserved / propagated along the line between $\boldsymbol{p}$ and $\boldsymbol{p}'$

– $\boldsymbol{p}$ and $\boldsymbol{p}'$ can be reversed, i.e. $L(\boldsymbol{p}' \leftarrow -\boldsymbol{\omega}_o) = (\boldsymbol{G}L)(\boldsymbol{p} \rightarrow \boldsymbol{\omega}_o)$

See, e.g.: Eric Veach: Robust Monte Carlo Methods for Light Transport Simulation, Ph.D. thesis, Stanford University, 1997.

Operator Form of the Rendering Equation

- Light transport operator

  $$\boldsymbol{T} = \boldsymbol{K}\boldsymbol{G}$$

  - Composition of scattering and propagation
  - Maps an exitant radiance function to the exitant radiance function after one scattering step
  - Remember: $\boldsymbol{G}$ maps exitant radiance to incident radiance propagated along a direction. Then, $\boldsymbol{K}$ maps incident radiance to exitant radiance after scattering

See, e.g.: Eric Veach: Robust Monte Carlo Methods for Light Transport Simulation, Ph.D. dissertation, Stanford University, 1997.

UNI FREIBURG

Operator Form of the Rendering Equation

$$L(\boldsymbol{p} \to \boldsymbol{\omega}_o) = L_e(\boldsymbol{p} \to \boldsymbol{\omega}_o) + \int_\Omega f_r(\boldsymbol{p}, \boldsymbol{\omega}_i \leftrightarrow \boldsymbol{\omega}_o) L(\boldsymbol{p} \leftarrow \boldsymbol{\omega}_i) \cos(\boldsymbol{\omega}_i, \boldsymbol{n}_p) \mathrm{d}\omega_i$$

– Can be written as

$$L(\boldsymbol{p} \to \boldsymbol{\omega}_o) = L_e(\boldsymbol{p} \to \boldsymbol{\omega}_o) + (\boldsymbol{K}\boldsymbol{G}L)(\boldsymbol{p} \to \boldsymbol{\omega}_o)$$

– Light transport equation

$$L = L_e + \boldsymbol{T}L$$ Infinite number of equations with an infinite number of unknown exitant radiances

- $\boldsymbol{T}$ relates exitant radiance functions
- Represents the light propagation equilibrium

Light Transport Equation

$$L = L_e + \mathbf{T}L$$

– Solving for the unknown radiance function

$$(\mathbf{I} - \mathbf{T})L = L_e$$

$$L = (\mathbf{I} - \mathbf{T})^{-1} L_e$$

  – Neumann series

$$L = \sum_{k=0}^{\infty} (\mathbf{T}^k L_e)$$

$$\approx L_e + \mathbf{T}L_e + \mathbf{T}\mathbf{T}L_e + \mathbf{T}\mathbf{T}\mathbf{T}L_e + \ldots$$

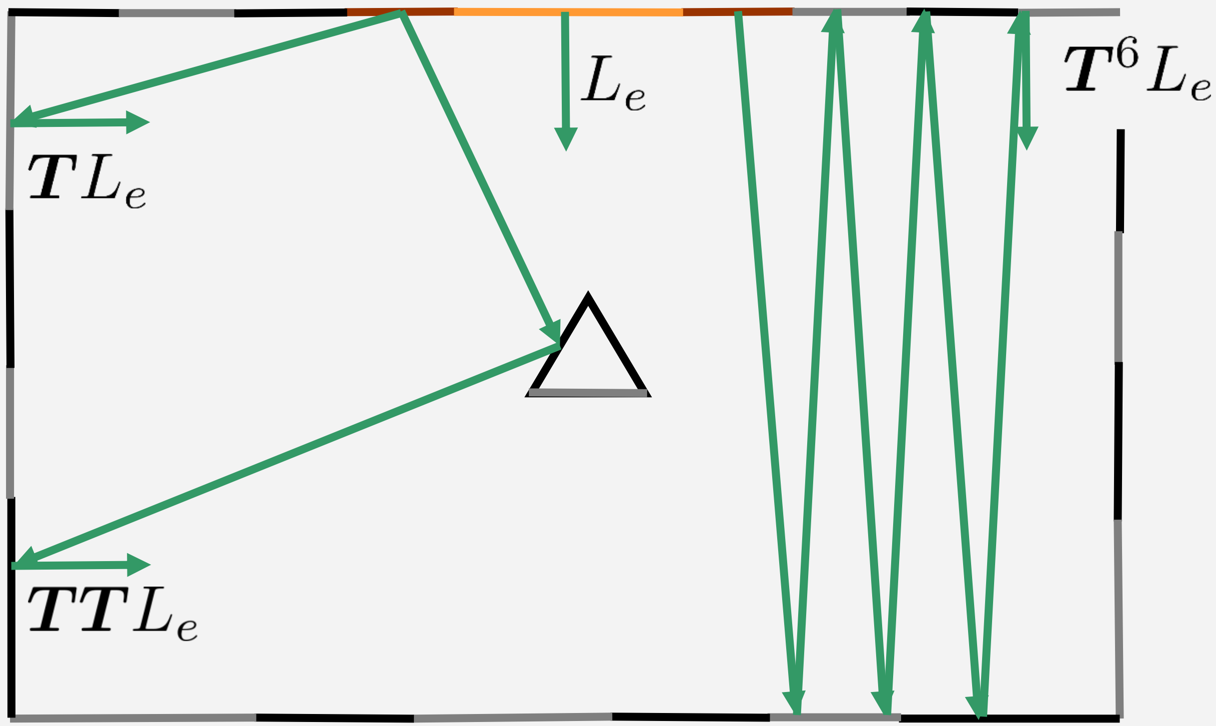
Light Transport Equation

- Discussion
  - Radiance function is a sum of
    - Emitted radiance L_e
    - Emitted radiance after one scattering $\boldsymbol{T}L_e$
    - Emitted radiance after two scatterings $\boldsymbol{TT}L_e$
    - …

$$L \approx L_e + \boldsymbol{T}L_e + \boldsymbol{TT}L_e + \boldsymbol{TTT}L_e + \dots$$

Terms in the Neumann Series

– Example contributions to terms

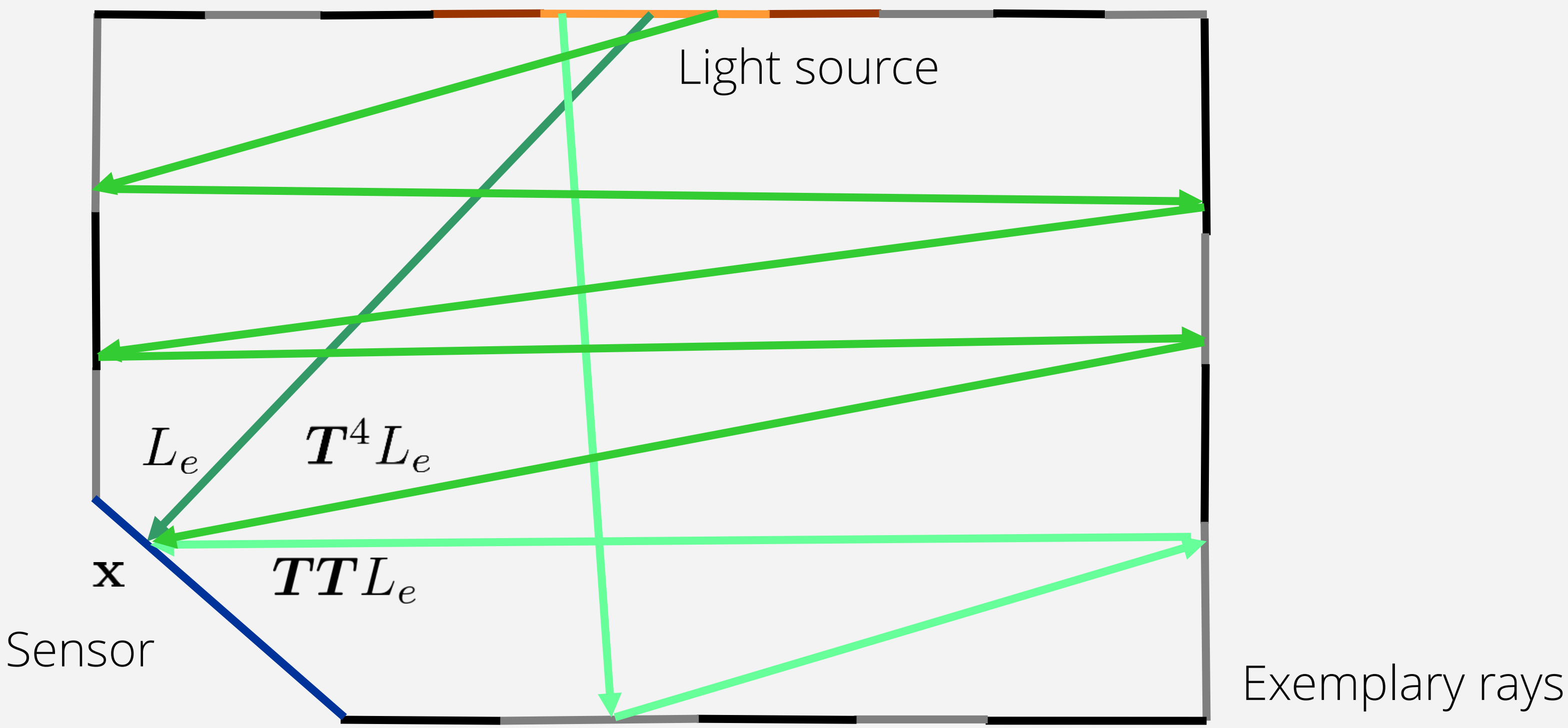


Forward Raytracing

- Send rays / propagate radiance from all light source positions into all directions ⇨ L_e
- At all intersection points $\boldsymbol{p}$, solve the integral
  $L_1(\boldsymbol{p} \to \boldsymbol{\omega}_o) = \int_\Omega f_r(\boldsymbol{p}, \boldsymbol{\omega}_i \leftrightarrow \boldsymbol{\omega}_o) \boldsymbol{G} L_e \cos(\boldsymbol{\omega}_i, \boldsymbol{n}_p) \mathrm{d}\omega_i$
  for all direction $\boldsymbol{\omega}_o$ ⇨ $\boldsymbol{T} L_e$
- Trace rays to propagate $\boldsymbol{T} L_e$
- At all intersection points $\boldsymbol{p}$, solve the integral
  $L_2(\boldsymbol{p} \to \boldsymbol{\omega}_o) = \int_\Omega f_r(\boldsymbol{p}, \boldsymbol{\omega}_i \leftrightarrow \boldsymbol{\omega}_o) \boldsymbol{G} L_1 \cos(\boldsymbol{\omega}_i, \boldsymbol{n}_p) \mathrm{d}\omega_i$
  for all direction $\boldsymbol{\omega}_o$ ⇨ $\boldsymbol{T}\boldsymbol{T} L_e$

Forward Raytracing

– At a sensor: Accumulate radiance contributions of rays after *n* scattering steps, i.e. compute $L_e + \boldsymbol{T} L_e + \boldsymbol{T}\boldsymbol{T} L_e + \ldots$



Operator Form of the Rendering Equation

$$L(\boldsymbol{p} \to \boldsymbol{\omega}_o) = L_e(\boldsymbol{p} \to \boldsymbol{\omega}_o) + \int_\Omega f_r(\boldsymbol{p}, \boldsymbol{\omega}_i \leftrightarrow \boldsymbol{\omega}_o) L(\boldsymbol{p} \leftarrow \boldsymbol{\omega}_i) \cos(\boldsymbol{\omega}_i, \boldsymbol{n}_p) \mathrm{d}\omega_i$$

– Can be written as

$$L(\boldsymbol{p} \to \boldsymbol{\omega}_o) = L_e(\boldsymbol{p} \to \boldsymbol{\omega}_o) + (\boldsymbol{KG}L)(\boldsymbol{p} \to \boldsymbol{\omega}_o)$$

– Light transport equation

$$L = L_e + \boldsymbol{T}L$$ Infinite number of equations with an infinite number of unknown exitant radiances

 – $\boldsymbol{T}$ relates exitant radiance functions
 – Represents the light propagation equilibrium

UNI FREIBURG

Light Transport Equation

$$L = L_e + \mathbf{T}L$$

– Solving for the unknown radiance function

$$(\mathbf{I} - \mathbf{T})L = L_e$$

$$L = (\mathbf{I} - \mathbf{T})^{-1}L_e$$

– Neumann series

$$L = \sum_{k=0}^{\infty}(\mathbf{T}^k L_e)$$

$$\approx L_e + \mathbf{T}L_e + \mathbf{T}\mathbf{T}L_e + \mathbf{T}\mathbf{T}\mathbf{T}L_e + \ldots$$

Light Transport Equation

- Discussion
  - Radiance function is a sum of
    - Emitted radiance L_e
    - Emitted radiance after one scattering $\boldsymbol{T} L_e$
    - Emitted radiance after two scatterings $\boldsymbol{T}\boldsymbol{T} L_e$
    - …

$$L \approx L_e + \boldsymbol{T} L_e + \boldsymbol{T}\boldsymbol{T} L_e + \boldsymbol{T}\boldsymbol{T}\boldsymbol{T} L_e + \ldots$$

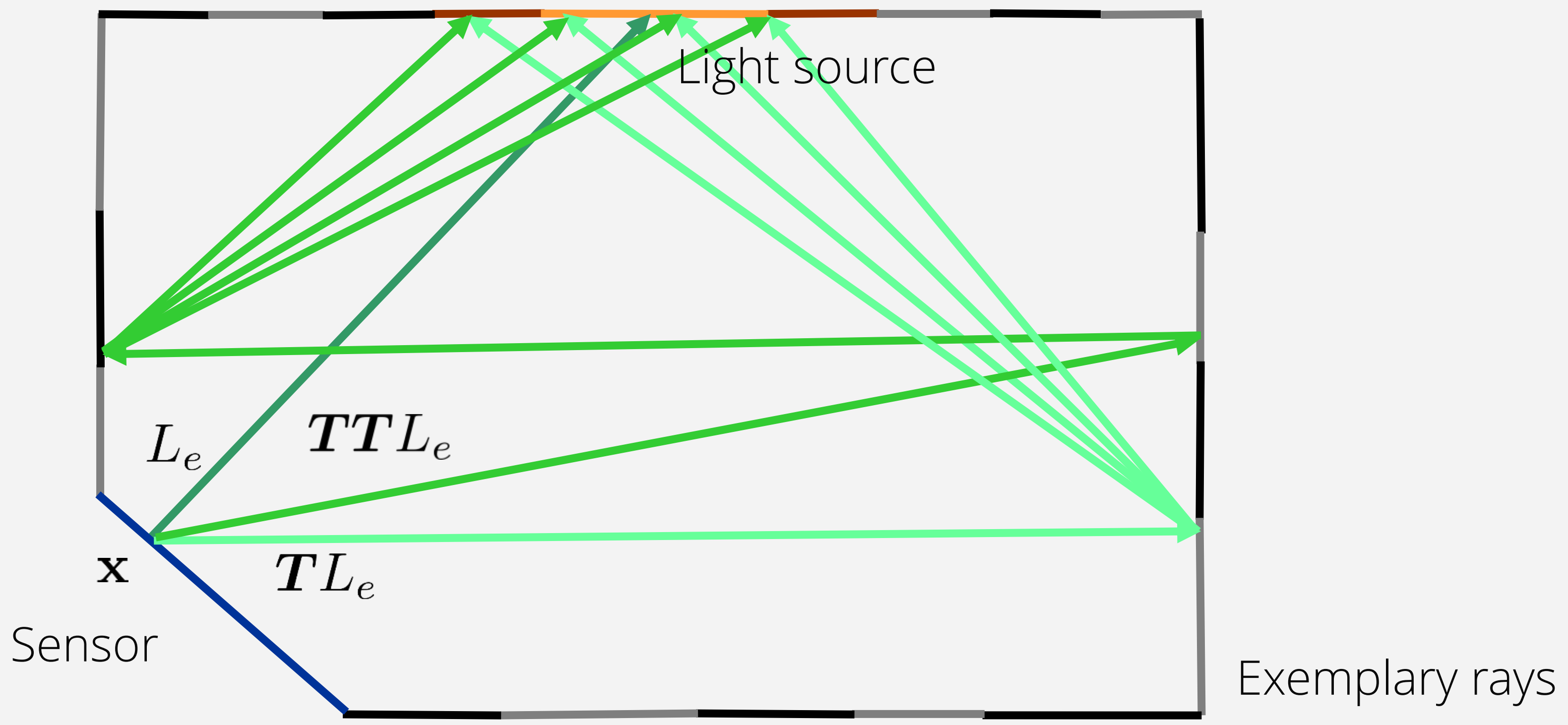
UNI FREIBURG

Backward Raytracing

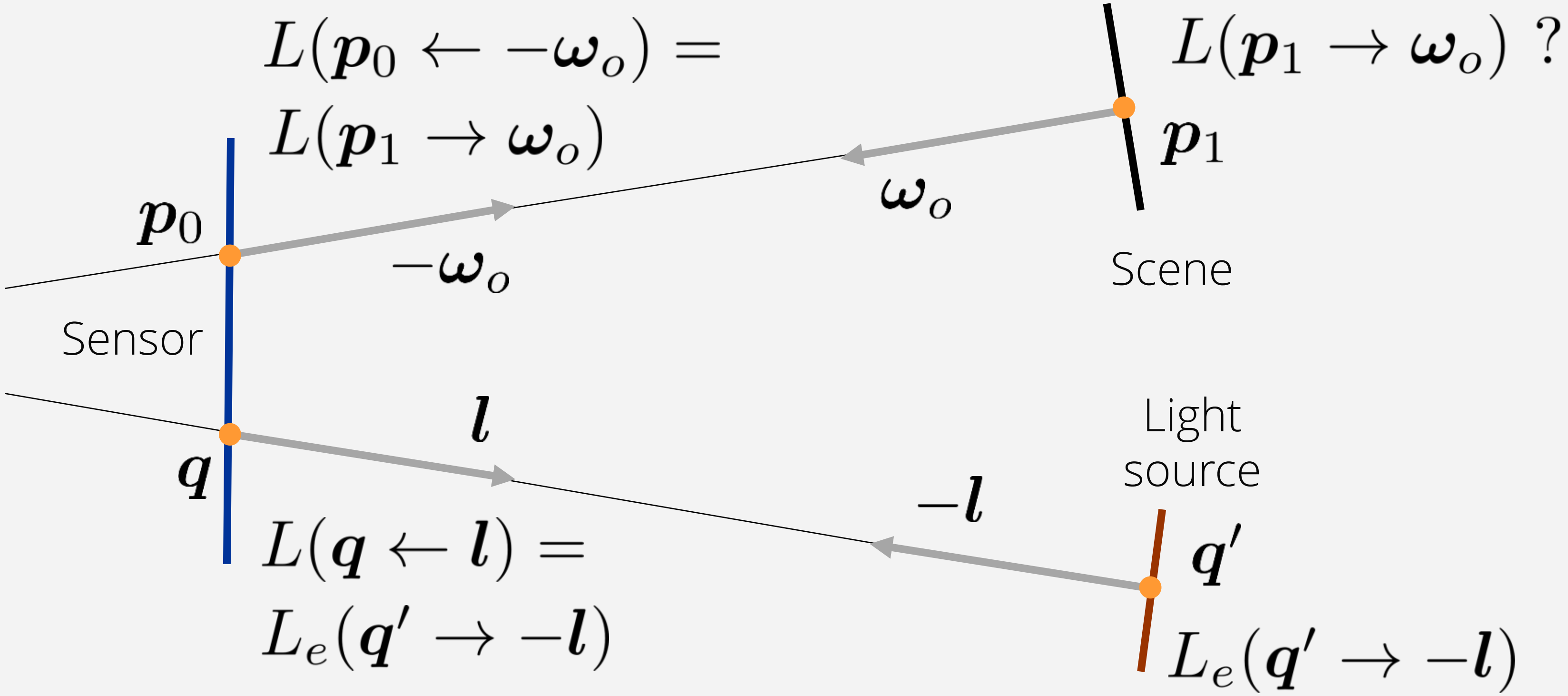
- Consider rays from the sensor into the scene
- Propagate radiance from visible light sources
- ⇨ part of L_e visible to the sensor
- At intersection points $\boldsymbol{p}$ with the scene, compute radiance $L(\boldsymbol{p} \rightarrow \boldsymbol{\omega}_o) = \int_\Omega f_r(\boldsymbol{p}, \boldsymbol{\omega}_i \leftrightarrow \boldsymbol{\omega}_o) L_e(\boldsymbol{p} \leftarrow \boldsymbol{\omega}_i) \cos(\boldsymbol{\omega}_i, \boldsymbol{n}_p) \mathrm{d}\omega_i$ that is propagated in direction $\boldsymbol{\omega}_o$ towards the sensor
- ⇨ part of $\boldsymbol{T} L_e$ visible to the sensor
- …

Backward Raytracing

– Trace rays from the sensor into the scene



Setting at Sensor

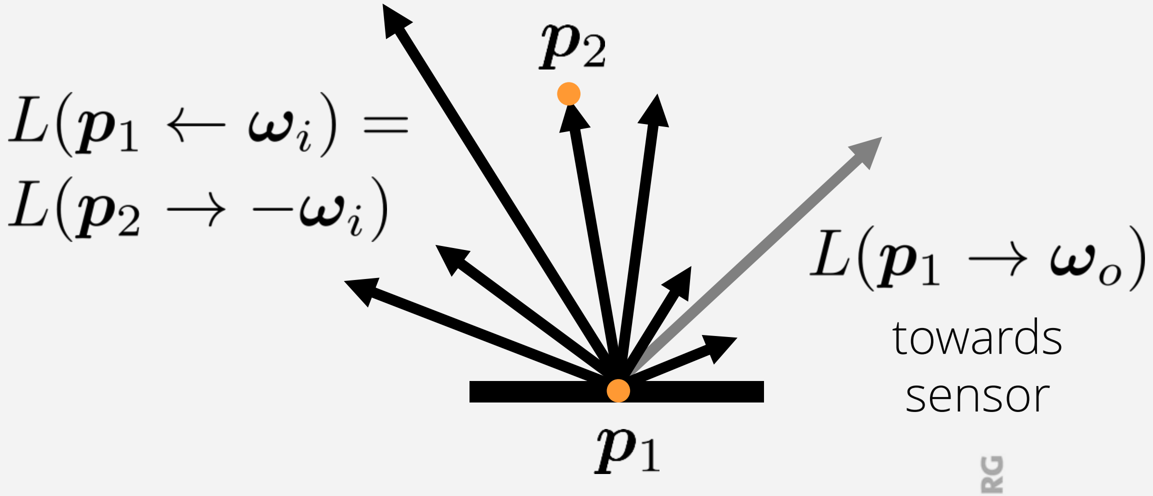


– How to compute $L(\boldsymbol{p}_1 \rightarrow \boldsymbol{\omega}_o)$ and what is its the relation to $L_e + \boldsymbol{T}L_e + \boldsymbol{TT}L_e + \ldots$?

Setting at First-Level Intersections

$$L(\boldsymbol{p}_1 \rightarrow \boldsymbol{\omega}_o) = \int_S f_r(\boldsymbol{p}_1, \boldsymbol{\omega}_i \leftrightarrow \boldsymbol{\omega}_o) L(\boldsymbol{p}_2 \rightarrow -\boldsymbol{\omega}_i) G(\boldsymbol{p}_1, \boldsymbol{p}_2) \mathrm{d}A_{p_2}$$
$$= \int_{\text{Light Sources}} f_r(\boldsymbol{p}_1, \boldsymbol{\omega}_i \leftrightarrow \boldsymbol{\omega}_o) L_e(\boldsymbol{p}_2 \rightarrow -\boldsymbol{\omega}_i) G(\boldsymbol{p}_1, \boldsymbol{p}_2) \mathrm{d}A_{p_2}$$
$$+ \int_{\text{Scene}} f_r(\boldsymbol{p}_1, \omega_i \leftrightarrow \boldsymbol{\omega}_o) L(\boldsymbol{p}_2 \rightarrow -\boldsymbol{\omega}_i) G(\boldsymbol{p}_1, \boldsymbol{p}_2) \mathrm{d}A_{p_2}$$

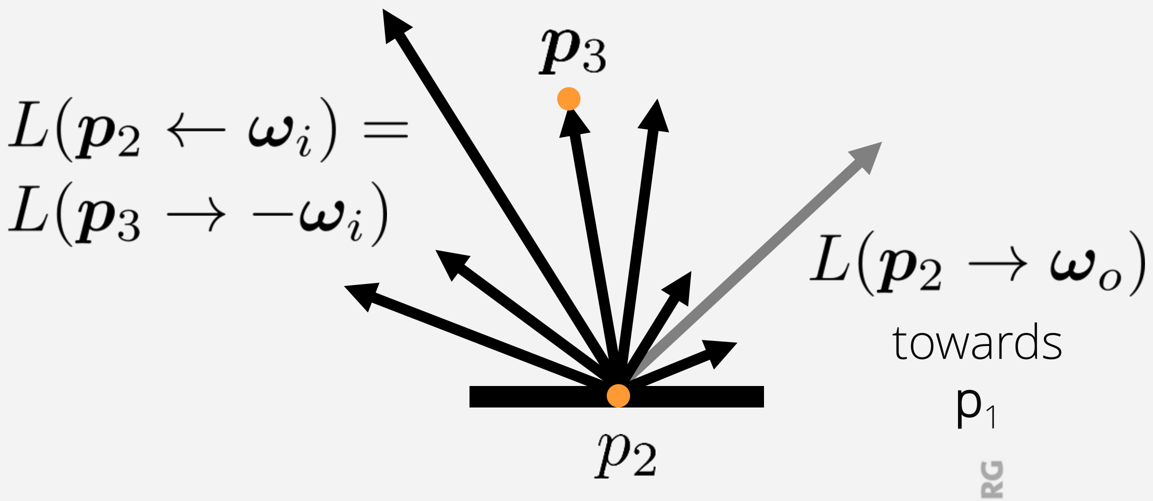
- $\int_{\text{Light Sources}} \cdots$ is the part of $\boldsymbol{T} L_e$ visible to the sensor
- Computation of $\int_{\text{Scene}} \cdots$ requires $L(\boldsymbol{p}_2 \rightarrow -\boldsymbol{\omega}_i)$



Setting at Second-Level Intersections

$$L(\boldsymbol{p}_2 \rightarrow \boldsymbol{\omega}_o) = \int_S f_r(\boldsymbol{p}_2, \boldsymbol{\omega}_i \leftrightarrow \boldsymbol{\omega}_o) L(\boldsymbol{p}_3 \rightarrow -\boldsymbol{\omega}_i) G(\boldsymbol{p}_2, \boldsymbol{p}_3) \mathrm{d}A_{p_3}$$
$$= \int_{\text{Light Sources}} f_r(\boldsymbol{p}_2, \boldsymbol{\omega}_i \leftrightarrow \boldsymbol{\omega}_o) L_e(\boldsymbol{p}_3 \rightarrow -\boldsymbol{\omega}_i) G(\boldsymbol{p}_2, \boldsymbol{p}_3) \mathrm{d}A_{p_3}$$
$$+ \int_{\text{Scene}} f_r(\boldsymbol{p}_2, \boldsymbol{\omega}_i \leftrightarrow \boldsymbol{\omega}_o) L(\boldsymbol{p}_3 \rightarrow -\boldsymbol{\omega}_i) G(\boldsymbol{p}_2, \boldsymbol{p}_3) \mathrm{d}A_{p_3}$$

- $\int_{\text{Light Sources}} \cdots$ is the part of $\boldsymbol{TT}L_e$ visible to the sensor
- Computation of $\int_{\text{Scene}} \cdots$ requires $L(\boldsymbol{p}_3 \rightarrow -\boldsymbol{\omega}_i)$



Summary

– Recursive evaluation of

$$L(\boldsymbol{p} \to \boldsymbol{\omega}_o) = \int_S f_r(\boldsymbol{p}, \boldsymbol{\omega}_i \leftrightarrow \boldsymbol{\omega}_o) L(\boldsymbol{p}' \to -\boldsymbol{\omega}_i) G(\boldsymbol{p}, \boldsymbol{p}') \mathrm{d}A_{p'}$$
$$= \int_{\text{Light Sources}} f_r(\boldsymbol{p}, \boldsymbol{\omega}_i \leftrightarrow \boldsymbol{\omega}_o) L_e(\boldsymbol{p}' \to -\boldsymbol{\omega}_i) G(\boldsymbol{p}, \boldsymbol{p}') \mathrm{d}A_{p'}$$
$$+ \int_{\text{Scene}} f_r(\boldsymbol{p}, \boldsymbol{\omega}_i \leftrightarrow \boldsymbol{\omega}_o) L(\boldsymbol{p}' \to -\boldsymbol{\omega}_i) G(\boldsymbol{p}, \boldsymbol{p}') \mathrm{d}A_{p'}$$

– Each recursion level computes parts of the functions $L_e, \boldsymbol{T}L_e, \boldsymbol{TT}L_e, \ldots$ that are visible to the sensor

Numerical Integration

- The integral $\int_S \dots$ is approximately computed with a sum of samples $\sum_i \dots$
- For each sample *i*,
  - A ray is cast into the scene
  - Intersection with the scene is computed
  - Radiance along the ray is computed

Numerical Integration

- Typically, $\int_S \dots = \int_{\text{Scene}} \dots + \int_{\text{Light Sources}} \dots \approx \sum_{\text{Scene}_i} \dots + \sum_{\text{Light Source}_i} \dots$ is considered
- For $\sum_{\text{Light Source}_i} \cdots$, light source areas are sampled and rays towards those positions are processed
- For $\sum_{\text{Scene}_i} \cdots$, the respective solid angle is sampled and rays towards those directions are processed

Numerical Integration

- Due to the recursive nature, the number of processed rays grows exponentially with the recursion level
- ⇨ Monte Carlo integration
  - Efficient for multidimensional integral
  - Adaptive sample distribution
  - Very flexible in terms of the number of used samples
  - Even one sample can be used to approximate an integral

  ⇨ e.g., Path tracing
    - At each recursion level, trace a fixed number of rays to light sources and one ray into the scene (which generates a ray path)

Advanced Computer Graphics Stochastic Raytracing 2

Matthias Teschner

UNI FREIBURG

Outline

- Diffuse vs. general global illumination
- Monte Carlo integration
- Sampling of random variables

Goal

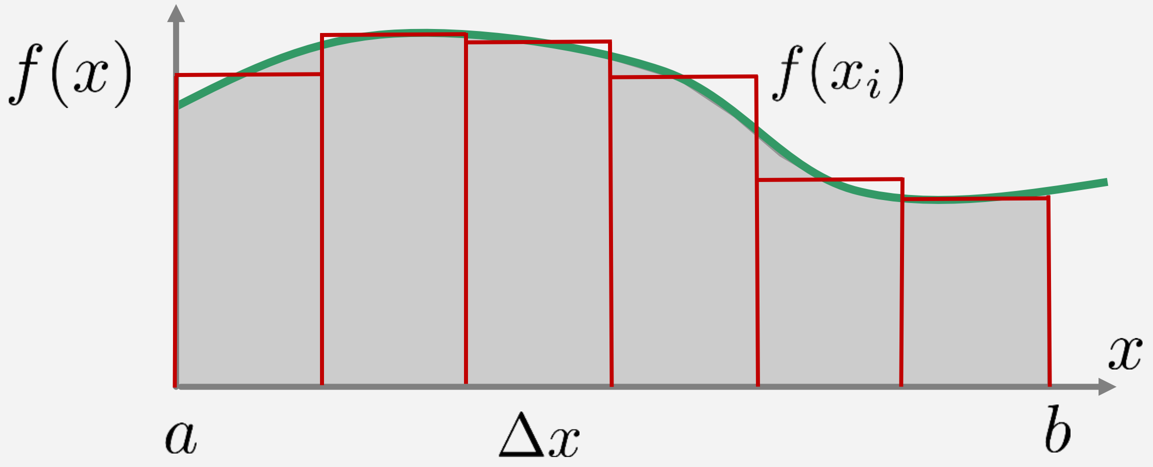
- Approximating the solution of the light transport equation $L = \sum_{k=0}^{\infty}(\boldsymbol{T}^k L_e)$
- Recursive evaluation of

$$L(\boldsymbol{p} \rightarrow \boldsymbol{\omega}_o) = L_e(\boldsymbol{p} \rightarrow \boldsymbol{\omega}_o) + \int_S f_r(\boldsymbol{p}, \boldsymbol{\omega}_i \leftrightarrow \boldsymbol{\omega}_o) L(\mathbf{p}' \rightarrow -\boldsymbol{\omega}_i) G(\boldsymbol{p}, \mathbf{p}') \mathrm{d}A_{p'}$$

- Each recursion level computes parts of the functions $L_e, \boldsymbol{T}L_e, \boldsymbol{T}\boldsymbol{T}L_e, \ldots$ that are visible to the sensor

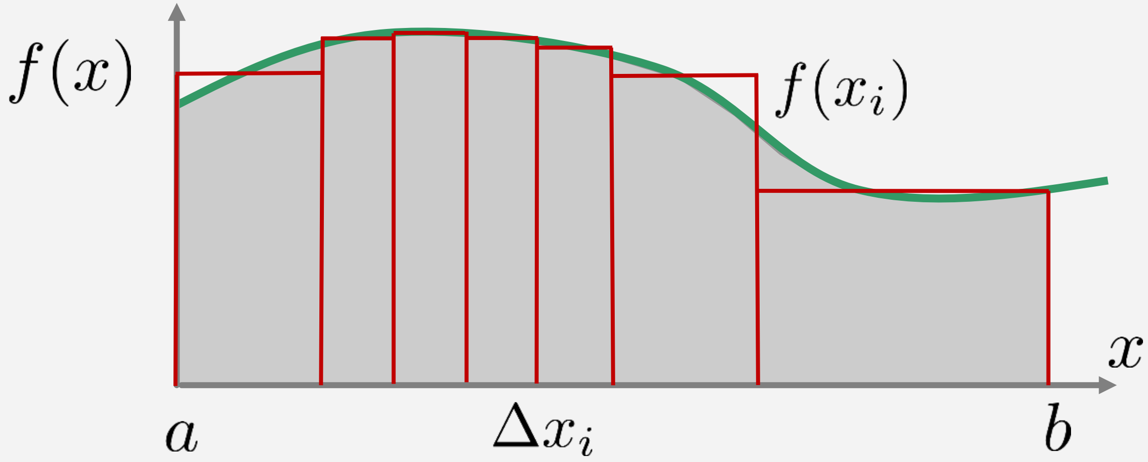
Numerical Integration – Fixed Sample Size

- E.g. Riemann sum
  - $\int_a^b f(x)\mathrm{d}x \approx \sum_i f(x_i)\Delta x \quad \Delta x = \frac{b-a}{N}$
  - More / smaller samples ⇨ better accuracy
  - d dimensional integrals require N^d samples



Numerical Integration – Adaptive Sample Size

- E.g., Monte Carlo integration
  - $\int_a^b f(x)\mathrm{d}x \approx \sum_i f(x_i)\Delta x_i$, adaptive sample size Δx_i
  - More / smaller samples ⇨ better accuracy
  - d dimensional integrals work with arbitrary sample numbers
  - Sample size is only approximated ⇨ noise



Stochastic Raytracing - Concept

- Approximately evaluate the reflectance integral $\int_\Omega f_r(\boldsymbol{p}, \boldsymbol{\omega}_i \leftrightarrow \boldsymbol{\omega}_o) L(\boldsymbol{p} \leftarrow \boldsymbol{\omega}_i) \cos(\boldsymbol{\omega}_i, \boldsymbol{n}_p) \mathrm{d}\omega_i$ by
  - Tracing rays into randomly sampled 2D directions
  - Computing the incoming radiances
- Integral is approximated with $\sum_i f_r(\boldsymbol{p}, \boldsymbol{\omega}_i \leftrightarrow \boldsymbol{\omega}_o) L(\boldsymbol{p} \leftarrow \boldsymbol{\omega}_i) \cos(\boldsymbol{\omega}_i, \boldsymbol{n}_p) \Delta\Omega_i$
  - 2 dimensional sample directions $\boldsymbol{\omega}_i = (\theta_i, \phi_i)$
  - $\Delta\Omega_i$ is an approximation of the solid angle of sample direction $\boldsymbol{\omega}_i = (\theta_i, \phi_i)$

Introduction

- Challenges
  - Approximate the integral as exact as possible
  - Trace as few rays as possible / use as few samples as possible
  - Trace relevant rays / use relevant samples
    - Rays / samples to light sources are very relevant
      (Rays / samples to occluded light sources are irrelevant)
    - For diffuse surfaces, rays / samples in normal direction are more relevant than rays / samples perpendicular to the normal
    - For specular surfaces, rays / samples in reflection direction are relevant

Properties

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

Properties

- Drawbacks
  - Using *n* samples, the scheme converges to the correct result with $O(n^{½})$
  - 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 and for a direction is expensive (ray intersection tests)

Continuous Random Variables

- Motivation: random sampling of directions
- Continuous random variables X
  - In contrast to discrete random variables, infinite number of possible values
- Canonical uniform random variable $0 \leq \xi < 1$
  - Sample sets with arbitrary distributions can be computed from ξ

Probability Density Function PDF p(x)

- Motivation: PDF governs the size / solid angle of a sample / sample direction
- Probability of a random variable taking certain value ranges
- $p(x) \geq 0 \quad \forall x \in [a, b]$
- $Pr(x_0 \leq X \leq x_1) = \int_{x_0}^{x_1} p(x) \mathrm{d}x$ The probability, that the random variable has a certain exact value x_0=x_1, is 0.
- $\int_a^b p(x) \mathrm{d}x = 1$ The probability, that the random variable is in the specified domain, is 1.
- Example
  - Uniform PDF for $0 \leq X \leq 5$
  - $1 = \int_0^5 p(x) \mathrm{d}x = p(x) \int_0^5 \mathrm{d}x = 5\ p(x) \qquad p(x) = \frac{1}{5}$

UNI FREIBURG

Cumulative Distribution Function CDF P(x)

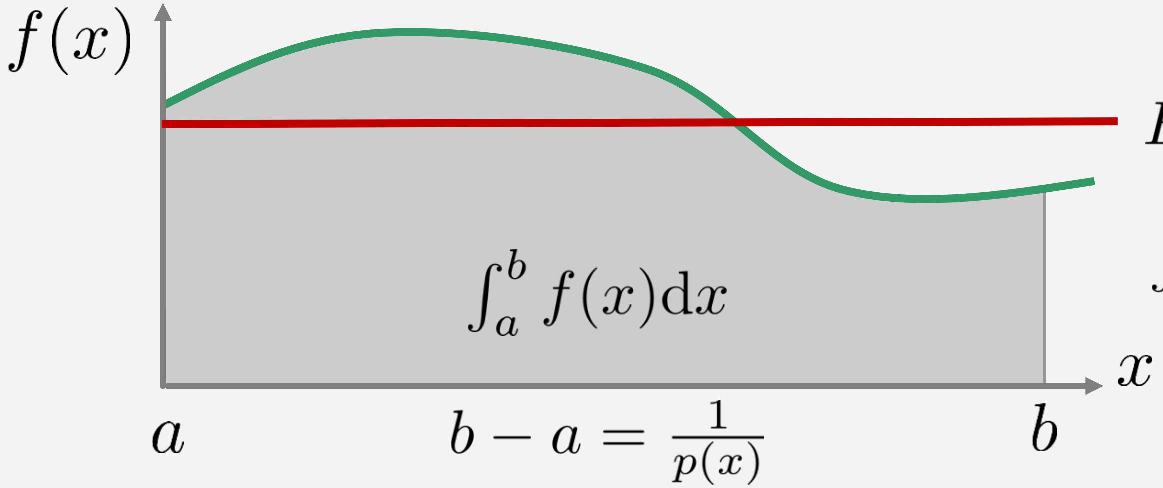
- Motivation: CDFs are required to generate sample sets for arbitrary PDFs from uniform sample sets
- Probability of a random variable to be less or equal to x
- $P(x) = Pr(X \leq x) = \int_a^x p(x)\mathrm{d}x$
- $P(a) = 0 \leq P(x) \leq 1 = P(b)$
- $Pr(x_0 \leq X \leq x_1) = P(x_1) - P(x_0)$

UNI FREIBURG

Expected Value

- Motivation: expected value of an estimator function is equal to the reflectance integral
- Expected value $E[f(x)]$ of a function $f(x)$ is defined as the weighted average value of the function over a domain D
- $E[f(x)] = \int_D f(x)\ p(x)\ \mathrm{d}x$ with $\int_D p(x)\ \mathrm{d}x = 1$ Processes an infinite number of samples x according to a PDF p(x)
- Properties
  - $E[af(x)] = aE[f(x)]$
  - $E\left[\sum_i f(X_i)\right] = \sum_i E[f(X_i)]$ For independent random variables X_i

Expected Value

- Examples for uniform PDF $p(x)$
  - $f(x) = \cos(x) \quad D = [0, \pi] \quad p(x) = \frac{1}{\pi}$
    $E[\cos(x)] = \int_0^\pi \cos(x)\ \frac{1}{\pi}\ \mathrm{d}x = \frac{1}{\pi}(-\sin\pi + \sin 0) = 0$
  - $f(x) = x \quad D = [0, 6] \quad p(x) = \frac{1}{6}$
    $E[x] = \int_0^6 x\ \frac{1}{6}\ \mathrm{d}x = \frac{1}{6}(\frac{6^2}{2} - 0) = 3$
  - 

$E[f(x)] = \frac{1}{b-a} \int_a^b f(x)\mathrm{d}x$

$\int_a^b f(x)\mathrm{d}x = E[f(x)](b-a)$

Monte Carlo Estimator - Uniform Random Variables

- Motivation: approximation of the reflectance integral
- Goal: computation of $\int_a^b f(x)\mathrm{d}x$
- Uniformly distributed random variables $X_i \in [a, b]$
- Probability density function $p(x) = \frac{1}{b-a}$ Constant and integration to one
- Monte Carlo estimator $F_N = \frac{b-a}{N} \sum_{i=1}^{N} f(X_i)$
- Expected value of F_N is equal to the integral $\int_a^b f(x)\mathrm{d}x$
  - $E[F_N] = \int_a^b f(x)\mathrm{d}x$

Monte Carlo Estimator - Uniform Random Variables

$$
\begin{aligned}
E[F_N] &= E\left[\frac{b-a}{N}\sum_{i=1}^{N} f(X_i)\right] \\
&= \frac{b-a}{N}\sum_{i=1}^{N} E[f(X_i)] \\
&= \frac{b-a}{N}\sum_{i=1}^{N} \int_a^b f(x)p(x)\mathrm{d}x \\
&= \frac{b-a}{N}\sum_{i=1}^{N} \int_a^b f(x)\frac{1}{b-a}\mathrm{d}x \\
&= \frac{1}{N}\sum_{i=1}^{N} \int_a^b f(x)\mathrm{d}x \\
&= \int_a^b f(x)\mathrm{d}x
\end{aligned}
$$

Monte Carlo Estimator - Uniform Random Variables

- PDF $p(x) = \frac{1}{b-a}$
- Estimator $F_N = \frac{b-a}{N} \sum_{i=1}^{N} f(X_i)$
- Integral
  - $\int_a^b f(x)\mathrm{d}x \approx \frac{b-a}{N} \sum_{i=1}^{N} f(X_i) = \sum_{i=1}^{N} f(X_i) \frac{b-a}{N} = \sum_{i=1}^{N} f(X_i) \frac{1}{N\ p(X_i)}$
  - Function value $f(X_i)$
  - Approximate sample size $\frac{1}{N\ p(X_i)}$

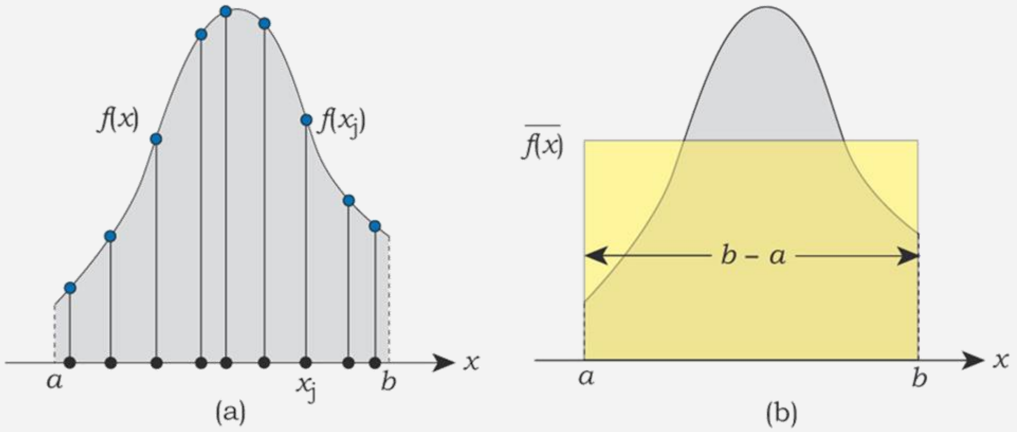
Examples - Uniform Random Variables

- Integral $\int_0^1 5x^4 \mathrm{d}x = 1$
- Estimator $F_N = \frac{1-0}{N} \sum_{i=1}^{N} 5X_i^4$ Sample size approx. 1/N
- For an increasing number of uniformly distributed random variables X_i, the estimator converges to one
- $F_N = \sum_{i=1}^{N} f(X_i) \frac{b-a}{N}$

$$F_N = (b-a) \frac{1}{N} \sum_{i=1}^{N} f(X_i)$$

$$= (b-a) \overline{f(x)}$$

$$E[F_N] = \int_a^b f(x) \mathrm{d}x$$



Uniformly distributed random samples

[Suffern]

Monte Carlo Estimator - Non-uniform Random Variables

- Monte Carlo estimator $F_N = \frac{1}{N} \sum_{i=1}^{N} \frac{f(X_i)}{p(X_i)}$ $p(X_i) \neq 0$
- $E[F_N] = E\left[\frac{1}{N} \sum_{i=1}^{N} \frac{f(X_i)}{p(X_i)}\right]$

$$= \frac{1}{N} \sum_{i=1}^{N} \int_a^b \frac{f(x)}{p(x)} p(x) \mathrm{d}x$$

$$= \frac{1}{N} \sum_{i=1}^{N} \int_a^b f(x) \mathrm{d}x$$

$$= \int_a^b f(x) \mathrm{d}x$$

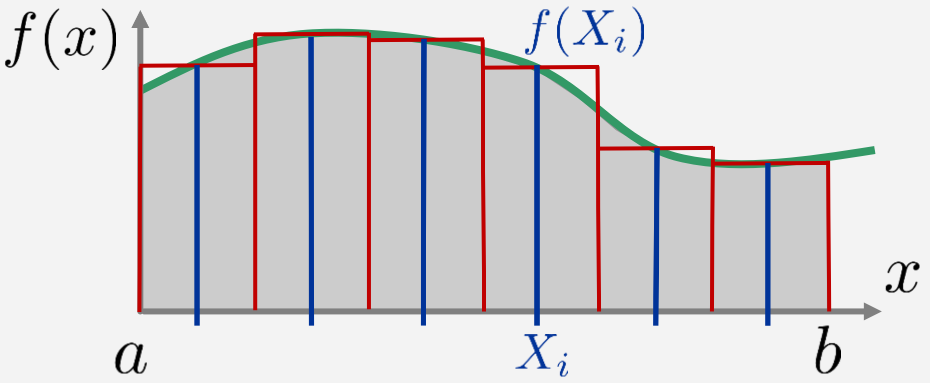
UNI FREIBURG

Monte Carlo Estimator - Non-uniform Random Variables

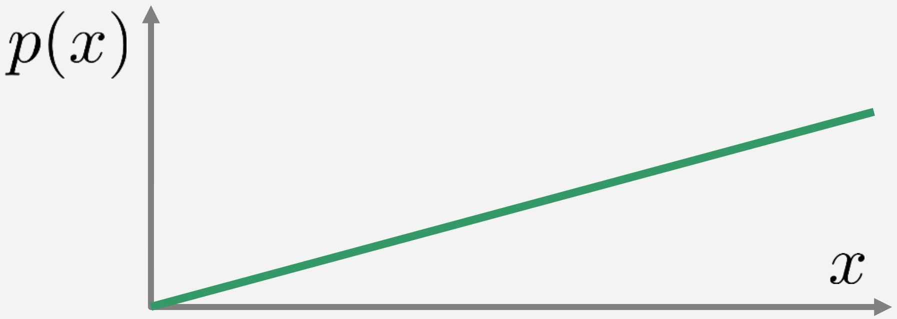
- PDF $p(x)$
- Estimator $F_N = \frac{1}{N} \sum_{i=1}^{N} \frac{f(X_i)}{p(X_i)}$
- Integral
  - $\int_a^b f(x)\mathrm{d}x \approx \frac{1}{N} \sum_{i=1}^{N} \frac{f(X_i)}{p(X_i)} = \sum_{i=1}^{N} f(X_i) \frac{1}{N\ p(X_i)}$
  - Function value $f(X_i)$
  - Approximate sample size $\frac{1}{N\ p(X_i)}$

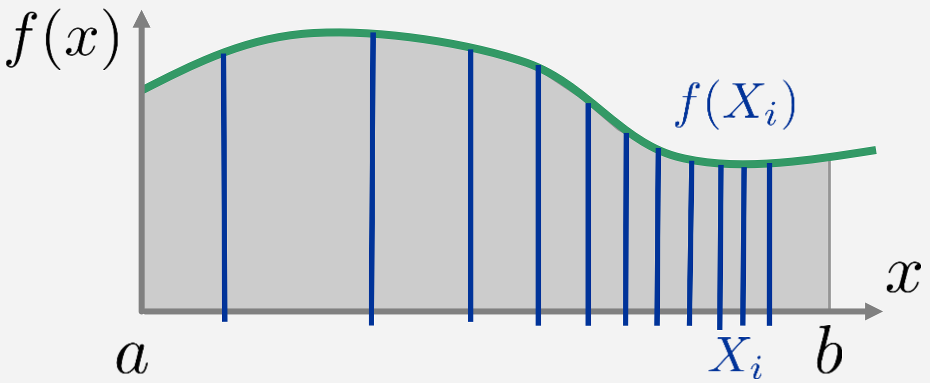
Approximate Sample Size

– Sample size / distance for uniform PDF: $\approx \frac{b-a}{N} = \frac{1}{Np(X_i)}$



– Sample size for non-uniform PDF: $\approx \frac{1}{Np(X_i)}$





Monte Carlo Estimator - Multiple Dimensions

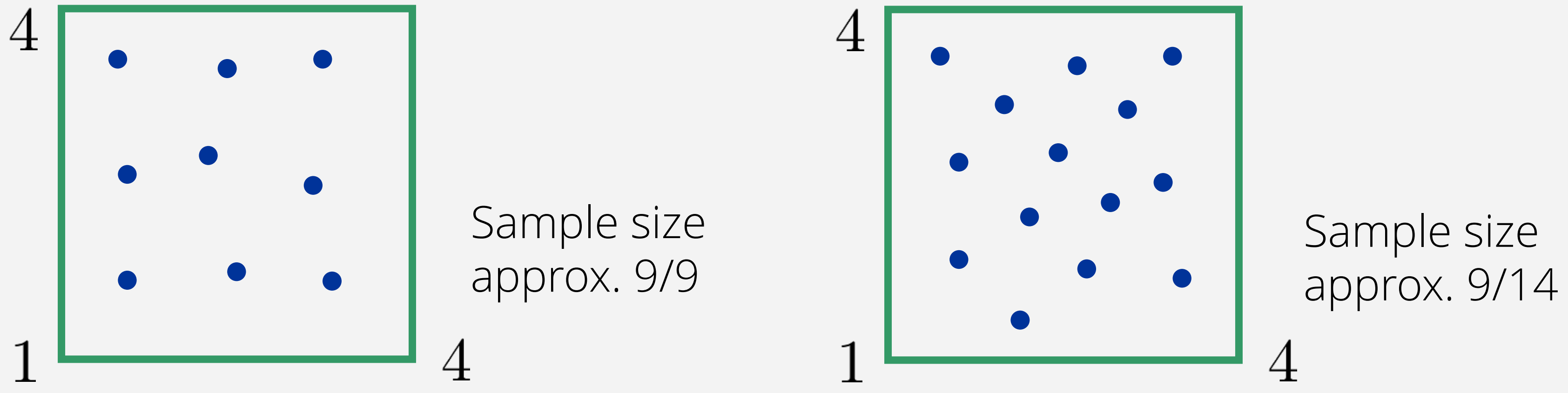
- E.g., $\int_{x_0}^{x_1} \int_{y_0}^{y_1} f(x,y)\mathrm{d}x\mathrm{d}y$
- Samples $\boldsymbol{X}_i$ are two-dimensional
- Uniformly distributed random samples

  $(x_0, y_0) \leq \boldsymbol{X}_i = (x_i, y_i) \leq (x_1, y_1)$
- Probability density function $p(\boldsymbol{X}_i) = \frac{1}{x_1 - x_0} \frac{1}{y_1 - y_0}$
- Monte Carlo estimator

  $F_N = \frac{(x_1 - x_0)(y_1 - y_0)}{N} \sum_{i=1}^{N} f(\boldsymbol{X}_i)$
- Approximate sample size is $\frac{(x_1 - x_0)(y_1 - y_0)}{N}$

Monte Carlo Estimator - Multiple Dimensions

- E.g., $\int_1^4 \int_1^4 f(x, y)\mathrm{d}x\mathrm{d}y$
- Uniformly distributed random samples
- Probability density function $p(\boldsymbol{X}_i) = \frac{1}{4-1}\frac{1}{4-1} = \frac{1}{9}$
- Monte Carlo estimator $F_N = \frac{9}{N}\sum_{i=1}^{N} f(\boldsymbol{X}_i)$
- Approximate sample size $\frac{9}{N}$



UNI FREIBURG

Monte Carlo Estimator - Integration over a Hemisphere

- Approximate computation of the irradiance at a point

$$E_i(\boldsymbol{p}) = \int_{2\pi^+} L_i(\boldsymbol{p}, \boldsymbol{\omega}) \cos\theta \mathrm{d}\omega$$
$$= \int_0^{2\pi} \int_0^{\frac{\pi}{2}} L_i(\boldsymbol{p}, \theta, \phi) \cos\theta \sin\theta \mathrm{d}\theta \mathrm{d}\phi$$

- Estimator $F_N = \frac{1}{N} \sum_{i=1}^{N} \frac{f(\boldsymbol{X}_i)}{p(\boldsymbol{X}_i)} = \frac{1}{N} \sum_{i=1}^{N} \frac{L_i(\boldsymbol{p},\theta_i,\phi_i) \cos\theta_i \sin\theta_i}{p(\theta_i,\phi_i)}$
- Choosing a PDF This flexibility is an important aspect of Monte Carlo integration.
  - 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
  - $p(\theta, \phi) \propto \cos\theta$

Monte Carlo Estimator - Integration over a Hemisphere

– Probability distribution

$$\int_{2\pi^+} c\ \tilde{p}(\boldsymbol{\omega})\mathrm{d}\omega = 1 \qquad \tilde{p}(\theta,\phi) = \cos\theta$$

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

$$c\ \frac{2\pi}{1+1} = 1$$

$$c = \frac{1}{\pi}$$

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

– Estimator

$$F_N = \frac{1}{N}\sum_{i=1}^{N} \frac{L_i(\boldsymbol{p},\theta_i,\phi_i)\cos\theta_i\sin\theta_i}{p(\theta_i,\phi_i)}$$

$$= \frac{\pi}{N}\sum_{i=1}^{N} L_i(\boldsymbol{p},\theta_i,\phi_i) \qquad \approx \int_0^{2\pi}\int_0^{\frac{\pi}{2}} L_i(\boldsymbol{p},\theta,\phi)\cos\theta\sin\theta\mathrm{d}\theta\mathrm{d}\phi$$

If θ and ϕ are sampled according to PDF $p(\theta, \phi)$

UNI FREIBURG

Monte Carlo Estimator - Integration over a Hemisphere

- Integral $\int_0^{2\pi} \int_0^{\frac{\pi}{2}} L_i(\boldsymbol{p}, \theta, \phi) \cos\theta \sin\theta \mathrm{d}\theta \mathrm{d}\phi$
- PDF $p(\theta, \phi) = \frac{\cos\theta \sin\theta}{\pi}$
- Estimator $\frac{\pi}{N} \sum_{i=1}^{N} L_i(\boldsymbol{p}, \theta_i, \phi_i)$

  $= \sum_{i=1}^{N} L_i(\boldsymbol{p}, \theta_i, \phi_i) \cos\theta_i \frac{\pi}{N \cos\theta_i}$
- Function value $L_i(\boldsymbol{p}, \theta_i, \phi_i) \cos\theta_i$ for direction (θ_i, ϕ_i)
- Approximate sample size / solid angle $\frac{\pi}{N \cos\theta_i}$
  - For large N
  - The PDF in terms of the solid angle is $p(\boldsymbol{\omega}_i) = \frac{\cos\theta_i}{\pi}$

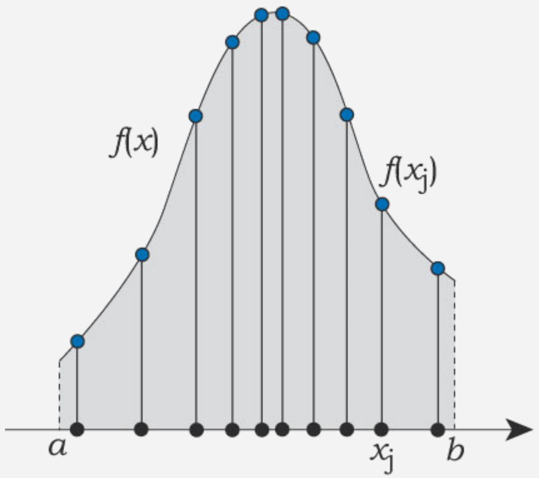
UNI FREIBURG

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

Monte Carlo Estimator - Error Reduction

- Importance sampling
  - Motivation: contributions of larger sample values are more important
  - PDF should be similar to the shape of the function
  - Optimal PDF $p(x) = \frac{f(x)}{\int f(x)\mathrm{d}x}$
  - E.g., if incident radiance is weighted with $\cos\theta$, the PDF should choose more samples close to the normal direction



[Suffern]

Monte Carlo Estimator - Error Reduction

- Stratified sampling
  - Domain subdivision into strata
  - E.g., handling direct and indirect illumination differently
  - $$\begin{aligned} L(\boldsymbol{p} \rightarrow \boldsymbol{\omega}_o) &= L_e(\boldsymbol{p} \rightarrow \boldsymbol{\omega}_o) + \int_S f_r(\boldsymbol{p}, \boldsymbol{\omega}_i \leftrightarrow \boldsymbol{\omega}_o) L(\boldsymbol{p}' \rightarrow -\boldsymbol{\omega}_i) G(\boldsymbol{p}, \boldsymbol{p}') \mathrm{d}A_{p'} \\ &= L_e(\boldsymbol{p} \rightarrow \boldsymbol{\omega}_o) \\ &\quad + \int_{\text{Light Sources}} f_r(\boldsymbol{p}, \boldsymbol{\omega}_i \leftrightarrow \boldsymbol{\omega}_o) L_e(\boldsymbol{p}' \rightarrow -\boldsymbol{\omega}_i) G(\boldsymbol{p}, \boldsymbol{p}') \mathrm{d}A_{p'} \\ &\quad + \int_{\text{Scene}} f_r(\boldsymbol{p}, \boldsymbol{\omega}_i \leftrightarrow \boldsymbol{\omega}_o) L(\boldsymbol{p}' \rightarrow -\boldsymbol{\omega}_i) G(\boldsymbol{p}, \boldsymbol{p}') \mathrm{d}A_{p'} \end{aligned}$$

Advanced Computer Graphics
Stochastic Raytracing 3

Matthias Teschner

UNI FREIBURG

Monte Carlo Estimator - Integration over a Hemisphere

- Approximate computation of the irradiance at a point

$$E_i(\boldsymbol{p}) = \int_{2\pi^+} L_i(\boldsymbol{p}, \boldsymbol{\omega}) \cos\theta \mathrm{d}\omega$$
$$= \int_0^{2\pi} \int_0^{\frac{\pi}{2}} L_i(\boldsymbol{p}, \theta, \phi) \cos\theta \sin\theta \mathrm{d}\theta \mathrm{d}\phi$$

- Estimator $F_N = \frac{1}{N} \sum_{i=1}^{N} \frac{f(\boldsymbol{X}_i)}{p(\boldsymbol{X}_i)} = \frac{1}{N} \sum_{i=1}^{N} \frac{L_i(\boldsymbol{p},\theta_i,\phi_i) \cos\theta_i \sin\theta_i}{p(\theta_i,\phi_i)}$
- Choosing a PDF This flexibility is an important aspect of Monte Carlo integration.
  - 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
  - $p(\theta, \phi) \propto \cos\theta$

Outline

- Diffuse vs. general global illumination
- Monte Carlo integration
- Sampling of random variables
  - Inversion method
  - Rejection method
  - Transforming between distributions
  - 2D sampling
  - Examples

Motivation - Rendering Equation

- Hemispherical form

$$L(\boldsymbol{p} \rightarrow \boldsymbol{\omega}_o) = L_e(\boldsymbol{p} \rightarrow \boldsymbol{\omega}_o) + \int_\Omega f_r(\boldsymbol{p}, \boldsymbol{\omega}_i \leftrightarrow \boldsymbol{\omega}_o) L(\boldsymbol{p}' \rightarrow -\boldsymbol{\omega}_i) \cos(\boldsymbol{\omega}_i, \boldsymbol{n}_p) \mathrm{d}\omega_i$$

- Area form

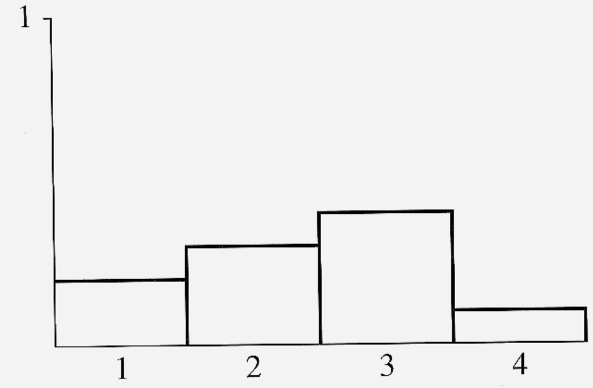
$$L(\boldsymbol{p} \rightarrow \boldsymbol{\omega}_o) = L_e(\boldsymbol{p} \rightarrow \boldsymbol{\omega}_o) + \int_S f_r(\boldsymbol{p}, \boldsymbol{\omega}_i \leftrightarrow \boldsymbol{\omega}_o) L(\boldsymbol{x} \rightarrow -\boldsymbol{\omega}_i) V(\boldsymbol{p}, \boldsymbol{x}) \frac{\cos(\boldsymbol{\omega}_i, \boldsymbol{n}_p) \cos(-\boldsymbol{\omega}_i, \boldsymbol{n}_x)}{r_{px}^2} \mathrm{d}A_x$$

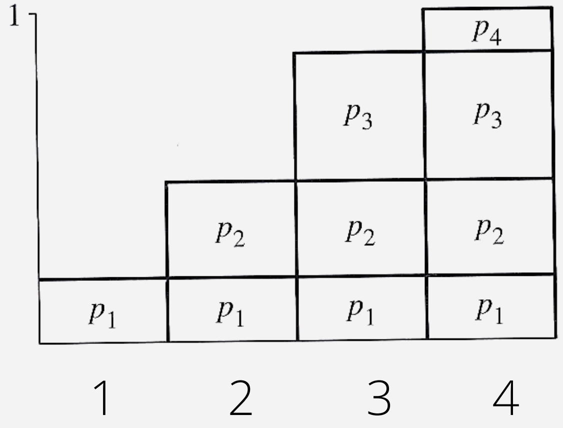
Motivation - Monte Carlo Integration

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

Inversion Method

- Mapping of a uniform random variable to a goal distribution
- Discrete example
  - Four outcomes with probabilities p_1, p_2, p_3, p_4 and $\sum_i p_i = 1$
  - Computation of the cumulative distribution function $P(i) = \sum_{j=1}^{i} p_j$

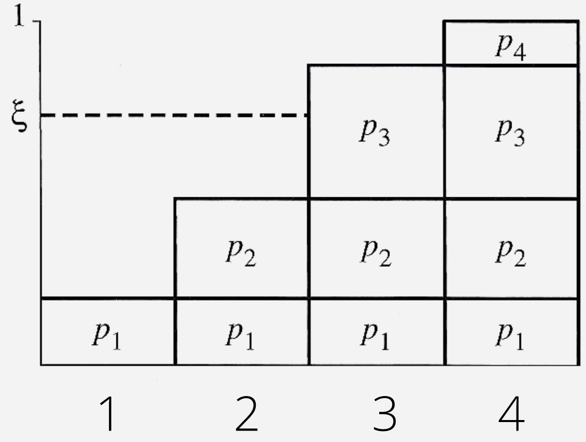




[Pharr, Humphreys]

Inversion Method

- Discrete example cont.
  - Take a uniform random variable ξ
  - $P^{-1}(\xi)$ has the desired distribution
- Continuous case
  - P and P^{-1} are continuous functions
  - Start with the desired PDF $p(x)$
  - Derive $P(x) = \int_0^x p(x')\mathrm{d}x'$
  - Compute the inverse $P^{-1}(x)$
  - Obtain a uniformly distributed variable
  - Compute $X_i = P^{-1}(\xi)$ which adheres to $p(x)$

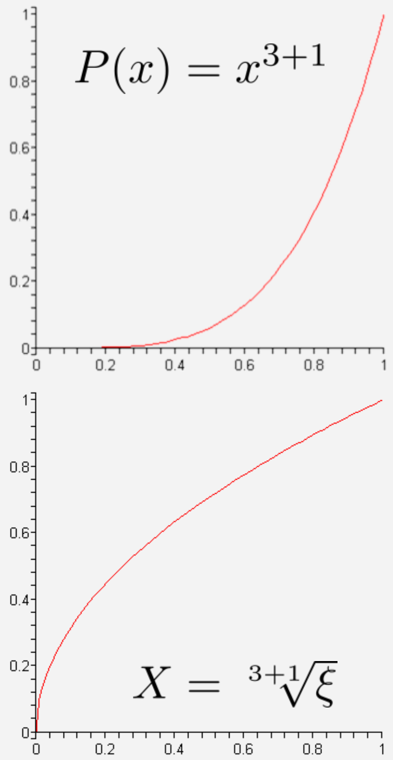


[Pharr, Humphreys]

Inversion Method - Example 1

- Power distribution $p(x) \propto x^n$
  - E.g., for sampling the Blinn microfacet model
- Computation of the PDF
  - $\int_0^1 c\, x^n \mathrm{d}x = 1 \Rightarrow c\, \frac{x^{n+1}}{n+1}\Big|_0^1 = 1 \Rightarrow c = n+1$
- PDF $p(x) = (n+1)x^n$
- CDF $P(x) = \int_0^x p(x')\mathrm{d}x' = x^{n+1}$
- Inverse of the CDF $P^{-1}(x) = \sqrt[n+1]{x}$
- Sample generation
  - Generate uniform random samples $0 \leq \xi \leq 1$
  - $X = \sqrt[n+1]{\xi}$ are samples from the distribution $p(x) = (n+1)x^n$

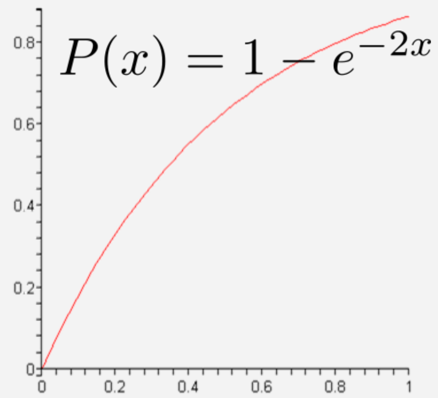
$p(x) = 4x^3$

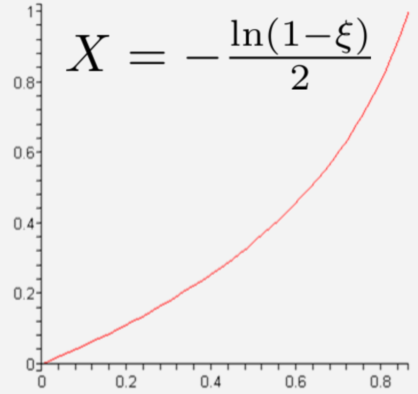


Inversion Method - Example 2

- Exponential distribution $p(x) \propto e^{-ax}$
  - E.g., for considering participating media
- Computation of the PDF
  - $\int_0^\infty c\ e^{-ax} \mathrm{d}x = -\frac{c}{a}\ e^{-ax}\big|_0^\infty = \frac{c}{a} = 1$
- PDF $p(x) = a\ e^{-ax}$
- CDF $P(x) = \int_0^x p(x')\mathrm{d}x' = 1 - e^{-ax}$
- Inverse of the CDF $P^{-1}(x) = -\frac{\ln(1-x)}{a}$
- Sample generation
  - Generate uniform random samples $0 \leq \xi \leq 1$
  - $X = -\frac{\ln(1-\xi)}{a}$ are samples from the distribution $p(x) = a\ e^{-ax}$

$p(x) = 2e^{-2x}$





Outline

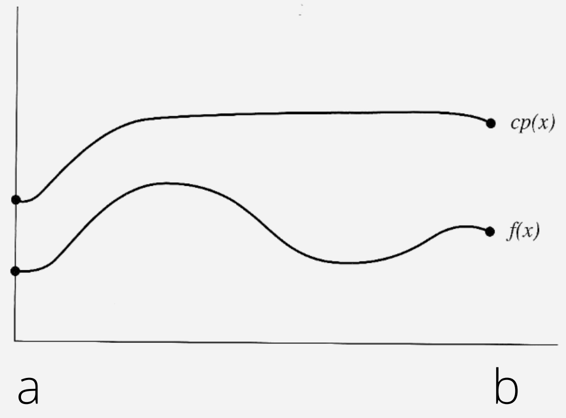
- Context
- Diffuse vs. general global illumination
- Monte Carlo integration
- Sampling of random variables
  - Inversion method
  - Rejection method
  - Transforming between distributions
  - 2D sampling
  - Examples

Rejection Method

- Draws samples according to a function $f(x)$
  - Dart-throwing approach
  - Works with a PDF $p(x)$ and a scalar c with $f(x) < c \cdot p(x)$
- Properties
  - $f(x)$ is not necessarily a PDF
  - PDF, CDF and inverse CDF do not have to be computed
  - Simple to implement
  - Useful for debugging purposes

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]

Outline

- Context
- Diffuse vs. general global illumination
- Monte Carlo integration
- Sampling of random variables
  - Inversion method
  - Rejection method
  - Transforming between distributions
  - 2D sampling
  - Examples

UNI FREIBURG

Transforming Between Distributions

- Computation of a resulting PDF, when a function is applied to samples from an arbitrary distribution
  - Random variables X_i are drawn from $p_x(x)$
  - Bijective transformation (one-to-one mapping) $Y_i = y(X_i)$
  - How does the distribution $p_y(y)$ look like?

UNI FREIBURG

Transforming Between Distributions

- $Pr\{Y \leq y(x)\} = Pr\{X \leq x\}$

  $P_y(y) = P_y(y(x)) = P_x(x)$

  $p_y(y) = \frac{p_x(x)}{|y'(x)|}$
- Example $p_x(x) = 2x \quad 0 \leq x \leq 1$
  - $y(x) = \sin x \quad x(y) = \arcsin y$
  - $y'(x) = \cos x$
  - $p_y(y) = \frac{p_x(x)}{|\cos x|} = \frac{2x}{|\cos x|} = \frac{2 \arcsin y}{|\cos \arcsin(y)|} = \frac{2 \arcsin y}{\sqrt{1-y^2}}$

Transforming Between Distributions

- Multiple dimensions
  - $\boldsymbol{X}_i$ is an n-dimensional random variable
  - $\boldsymbol{Y}_i = \boldsymbol{T}(\boldsymbol{X}_i)$ is a bijective transformation
- Transformation of the PDF
  - $p_y(y) = \frac{p_x(x)}{|\boldsymbol{J}_T(x)|}$ $\boldsymbol{J}_T(x) = \begin{pmatrix} \frac{\partial T_1}{\partial x_1} & \cdots & \frac{\partial T_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial T_n}{\partial x_1} & \cdots & \frac{\partial T_n}{\partial x_n} \end{pmatrix}$

Transforming Between Distributions

- Example (polar coordinates)
  - Samples (r, θ) with density $p(r, \theta)$
  - Corresponding density $p(x, y)$ with $x = r\cos\theta$ and $y = r\sin\theta$
  - $\boldsymbol{J}_T(x) = \begin{pmatrix} \cos\theta & -r\sin\theta \\ \sin\theta & r\cos\theta \end{pmatrix} \quad |\boldsymbol{J}_T(x)| = r(\cos^2\theta + \sin^2\theta) = r$
  - $p(x, y) = \frac{1}{r} p(r, \theta) \quad p(r, \theta) = r \cdot p(x, y)$

UNI FREIBURG

Transforming Between Distributions

- Example (spherical coordinates)
  - $x = r \sin\theta \cos\phi$
  - $y = r \sin\theta \sin\phi$
  - $z = r \cos\theta$
  - $p(r, \theta, \phi) = r^2 \sin\theta \cdot p(x, y, z)$
- Example (solid angle)
  - $Pr\{\boldsymbol{\omega} \in \Omega\} = \int_\Omega p(\boldsymbol{\omega}) \mathrm{d}\omega$
  - $\mathrm{d}\omega = \sin\theta \ \mathrm{d}\theta \ \mathrm{d}\phi$
  - $p(\theta, \phi) = \sin\theta \cdot p(\boldsymbol{\omega})$

Outline

- Context
- Diffuse vs. general global illumination
- Monte Carlo integration
- Sampling of random variables
  - Inversion method
  - Rejection method
  - Transforming between distributions
  - 2D sampling
  - Examples

Concept

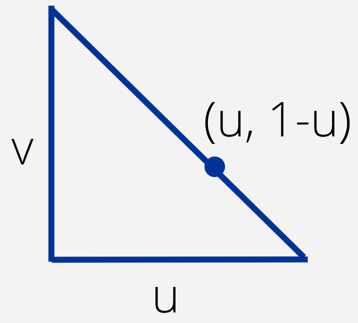
- Samples from a 2D joint density function $p(x,y)$
- Procedure
  - Compute the marginal density function $p_x(x) = \int p(x,y)\mathrm{d}y$
  - Compute the conditional density function $p_y(y|x) = \frac{p(x,y)}{p_x(x)}$
  - Generate a sample X according to $p_x(x)$
  - Generate a sample Y according to $p_y(y|X) = \frac{p(x,y)}{p_x(X)}$
- Marginal density function
  - Integral of $p(x,y)$ for a particular x over all y-values
- Conditional density function
  - Density function for y given a particular x

Outline

- Context
- Diffuse vs. general global illumination
- Monte Carlo integration
- Sampling of random variables
  - Inversion method
  - Rejection method
  - Transforming between distributions
  - 2D sampling
  - Examples

Uniform Sampling of a Triangle

- Sampling an isosceles right triangle of area 0.5
  - u, v can be interpreted as Barycentric coordinates
  - Can be used to generate samples for arbitrary triangles
- $p(u,v) = 2$
- Marginal density function
  - $p_u(u) = \int_0^{1-u} p(u,v)\ \mathrm{d}v = 2\int_0^{1-u} \mathrm{d}v = 2(1-u)$
- Conditional density $p_v(v|u) = \frac{p(u,v)}{p_u(u)} = \frac{1}{1-u}$
- Inversion method
  - $P_u(u) = \int_0^u 2 - 2u' \mathrm{d}u' = 2u - u^2$
  - $P_v(v|u) = \int_0^v \frac{1}{1-u} \mathrm{d}v' = \frac{v}{1-u}$



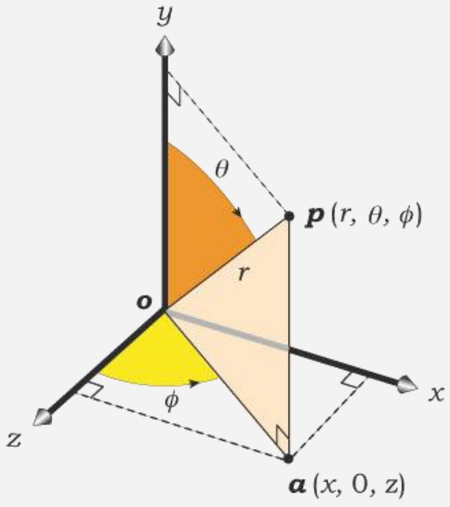
[Pharr, Humphreys]

Uniform Sampling of a Triangle

- Inversion method cont.
  - Inverse functions of the cumulative distribution functions
  - $u = 1 - \sqrt{1 - \xi_1}$ u is generated between 0 and 1
  - $v = \xi_2 \sqrt{1 - \xi_1}$ v is generated between 0 and 1-u=(1-ξ)½
  - Generating uniformly sampled random values ξ_1 and ξ_2
  - Applying the inverse CDFs to obtain u and v

Uniform Sampling of a Hemisphere

- PDF is constant with respect to a solid angle $p(\boldsymbol{\omega}) = c$
- $\int_{2\pi^+} p(\boldsymbol{\omega}) \mathrm{d}\omega = 1 \;\Rightarrow\; c\int_{2\pi^+} \mathrm{d}\omega = 1 \;\Rightarrow\; c = \frac{1}{2\pi}$
- $p(\boldsymbol{\omega}) = \frac{1}{2\pi} \;\Rightarrow\; p(\theta, \phi) = \frac{\sin\theta}{2\pi}$
- Marginal density function
  - $p_\theta(\theta) = \int_0^{2\pi} p(\theta, \phi)\mathrm{d}\phi = \int_0^{2\pi} \frac{\sin\theta}{2\pi}\mathrm{d}\phi = \sin\theta$
- Conditional density for ϕ
  - $p_\phi(\phi|\theta) = \frac{p(\theta,\phi)}{p_\theta(\theta)} = \frac{1}{2\pi}$
- Inversion method
  - $P_\theta(\theta) = \int_0^\theta \sin\theta' \mathrm{d}\theta' = -\cos\theta + 1$
  - $P_\phi(\phi|\theta) = \int_0^\phi \frac{1}{2\pi}\mathrm{d}\phi' = \frac{\phi}{2\pi}$



[Suffern]

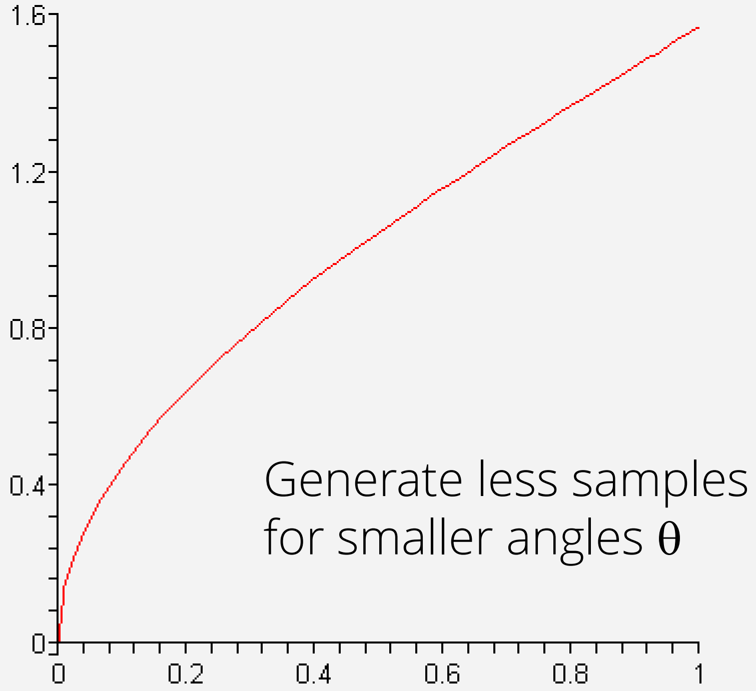
Uniform Sampling of a Hemisphere

- Inversion method cont.
  - Inverse functions of the cumulative distribution functions
  - $\theta = \arccos(1 - \xi_1)$
  - $\phi = 2\pi\xi_2$
  - Generating uniformly sampled random values ξ_1 and ξ_2
  - Applying the inverse CDFs to obtain θ and ϕ
- Conversion to Cartesian space
  - $x = \sin\theta\cos\phi = \cos(2\pi\xi_2)\sqrt{1 - (1 - \xi_1)^2}$
  - $y = \sin\theta\sin\phi = \sin(2\pi\xi_2)\sqrt{1 - (1 - \xi_1)^2}$
  - $z = \cos\theta = 1 - \xi_1$
- $(x, y, z)^\mathsf{T}$ is a normalized direction

UNI FREIBURG

Uniform Sampling of a Hemisphere

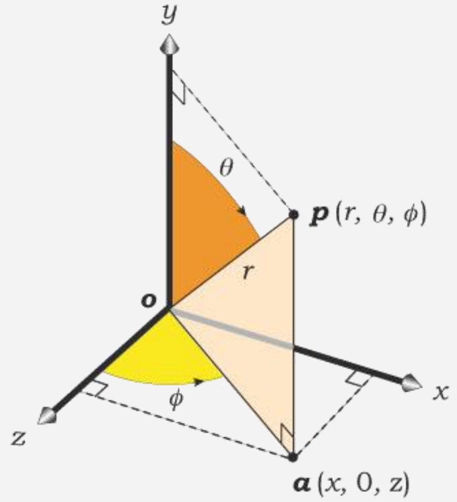
– Illustration for θ



$$\theta = \arccos(1 - \xi_1)$$

Cosine-Weighted Sampling of a Hemisphere

- PDF is proportional to $\cos\theta$: $p(\boldsymbol{\omega}) \propto \cos\theta$
  - $\int_{2\pi^+} c\, p(\boldsymbol{\omega})\ \mathrm{d}\omega = 1 = \int_0^{2\pi} \int_0^{\frac{\pi}{2}} c\ \cos\theta \sin\theta\ \mathrm{d}\theta\ \mathrm{d}\phi = c\ 2\pi \int_0^{\frac{\pi}{2}} \cos\theta \sin\theta\ \mathrm{d}\theta = c\ 2\pi \frac{1}{2} = 1$
  - $p(\theta, \phi) = \frac{1}{\pi} \cos\theta \sin\theta$
- Marginal density function
  - $p_\theta(\theta) = \int_0^{2\pi} p(\theta, \phi) \mathrm{d}\phi = \int_0^{2\pi} \frac{1}{\pi} \cos\theta \sin\theta \mathrm{d}\phi = 2 \cos\theta \sin\theta$
- Conditional density for ϕ
  - $p_\phi(\phi|\theta) = \frac{p(\theta,\phi)}{p_\theta(\theta)} = \frac{1}{2\pi}$
- Inversion method
  - $P_\theta(\theta) = \int_0^\theta 2 \cos\theta' \sin\theta' \mathrm{d}\theta' = 2 \left[-\frac{\cos^2\theta'}{2} \right]_0^\theta$
    $= 2 \left(-\frac{\cos^2\theta}{2} + \frac{1}{2} \right) = \sin^2\theta$
  - $P_\phi(\phi|\theta) = \int_0^\phi \frac{1}{2\pi} \mathrm{d}\phi' = \frac{\phi}{2\pi}$



[Suffern]

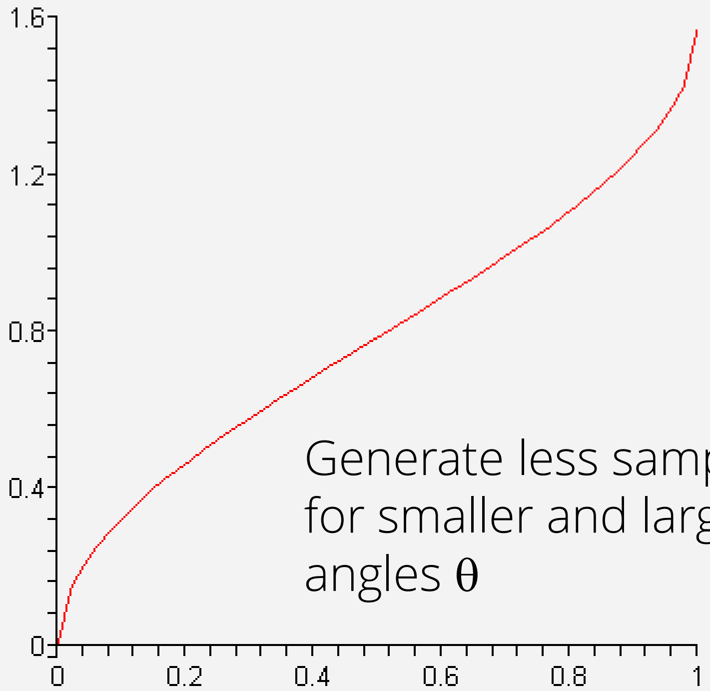
Cosine-Weighted Sampling of a Hemisphere

- Inversion method cont.
  - Inverse functions of the cumulative distribution functions
  - $\theta = \arcsin(\sqrt{\xi_1})$
  - $\phi = 2\pi\xi_2$
  - Generating uniformly sampled random values ξ_1 and ξ_2
  - Applying the inverse CDFs to obtain θ and ϕ
- Conversion to Cartesian space
  - $x = \sin\theta\cos\phi = \cos(2\pi\xi_2)\sqrt{\xi_1}$
  - $y = \sin\theta\sin\phi = \sin(2\pi\xi_2)\sqrt{\xi_1}$
  - $z = \cos\theta = \sqrt{1-\xi_1}$
- $(x, y, z)^\mathsf{T}$ is a normalized direction

x- y- values uniformly sample a unit disk, i.e., cosine-weighted samples of the hemisphere can also be obtained by uniformly sampling a unit sphere and projecting the samples onto the hemisphere

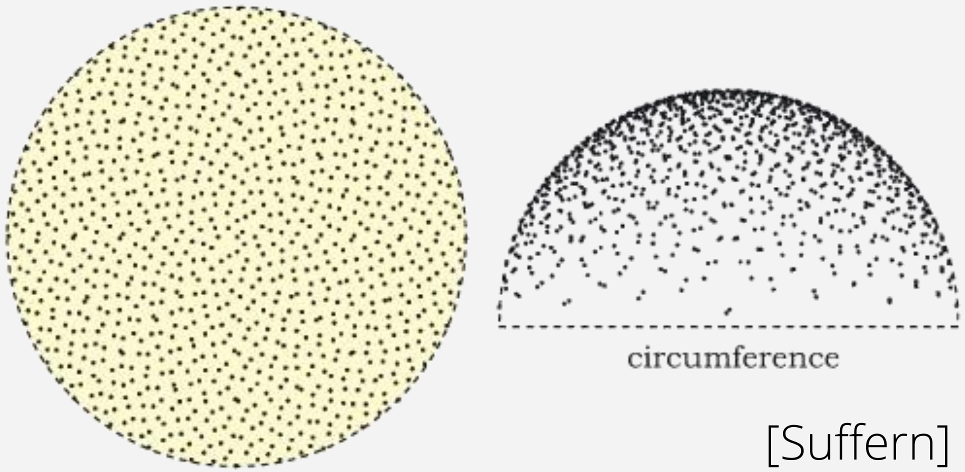
Cosine-Weighted Sampling of a Hemisphere

– Illustration for θ



Generate less samples for smaller and larger angles θ

$$\theta = \arcsin(\xi_1)$$



[Suffern]

Cosine-weighted hemisphere (top view, side view)