Lecture #5

Importance Sampling

Global Illumination
Summer Term 2021

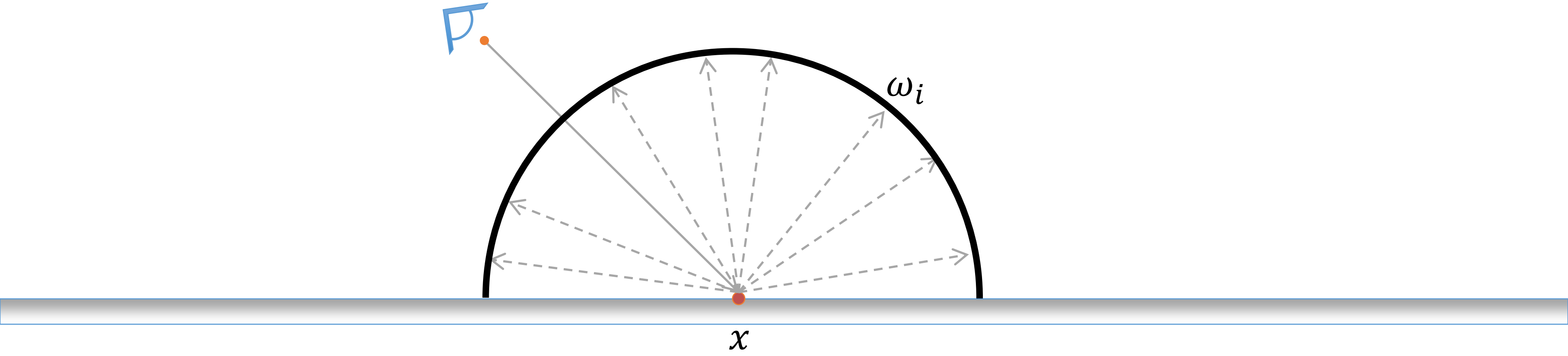
Marc Stamminger / Kai Selgrad

Remember: Rendering Integral – Directional Form

- $L_{out}(x, \omega_{out}) = \int f(x, \omega_{in}, \omega_{out}) < \mathrm{N_x}, \omega_{in} > L_{in}(x, \omega_{in}) d\omega_{in}$

- Compute by numerical integration using random incoming directions ω_i:

$$L_{out}(x, \omega_{out}) \approx \frac{2\pi}{n} \sum_{i=1}^{n} f(x, \omega_i, \omega_{out}) < N_x, \omega_i > L_{in}(x, \omega_i)$$



Remember: Numerical Integration

```
DirectIllumination(x, ω_out)
    n = number of samples, e.g. 100
    Lsum = 0
    for i = 1 to n
        ω_i = choose random direction
        y = intersect_ray(x,ω_i)
        Lsum += f(ω_i,x,ω_out) * dot(N(x), ω_i), Le(y,−ω_i)
    return 2*PI/n * Lsum
```

Remember: Random Directions

- How to compute a random direction ?

- 1st try: use polar coordinates with random angles (θ, ϕ)
- → biased result, more sample towards the poles…

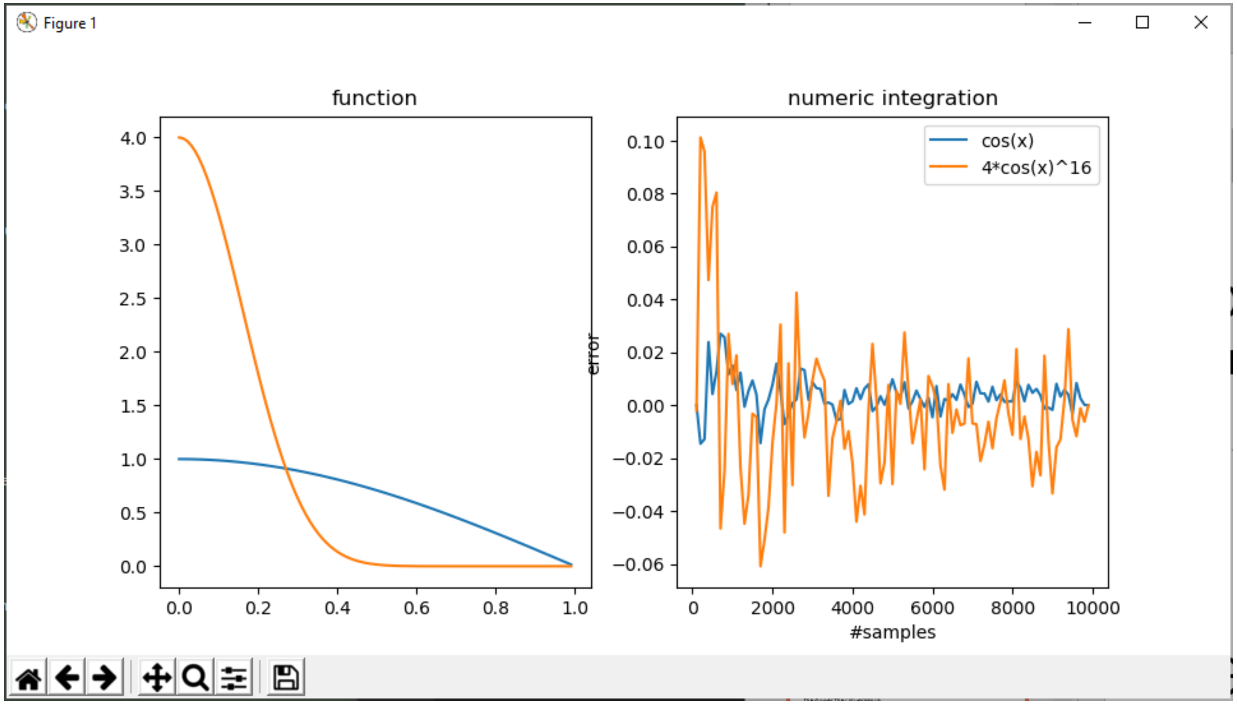
- Better try: use as parameterization $\omega(z, \phi) = \begin{pmatrix} \sqrt{1-z^2} \cdot \cos\phi \\ \sqrt{1-z^2} \cdot \sin\phi \\ z \end{pmatrix}$

  →draw random values for z and ϕ
  → random direction with uniform distribution

- Or: draw a random 3D-point $x \in [-1,1]^3$, discard if $\|x\| > 1$, otherwise normalize

Observation

- Obviously, the precision of numerical integration varies with the smoothness of the integrand
- Example:
  - integrating the functions $f(x) = \cos x$, $g(x) = 4(\cos x)^{16}$
- → the higher the variance, the slower the convergence



Variance in Direct Illumination Integral

- $L_{out}(x, \omega_{out}) = \int f(x, \omega_{in}, \omega_{out}) < \mathrm{N_x}, \omega_{in} > L_{in}(x, \omega_{in}) d\omega_{in}$

- Variance due to BRDF:
  - Glossy surfaces. The glossier, the higher the variance, the worse
- Variance due to scalar product:
  - exists, but not very big. Always the same, independent of scene.
- Variance due to incident light:
  - depends on scene. Bad example: small light sources

General Idea of Importance Sampling

- Up to now:

$$\int_0^1 f(x)\,dx \;\approx\; \frac{1}{n}\sum_{i=1\ldots n} f(x_i)$$

- Assumes that the x_i are uniformly distributed (each sample position has equal probability)

- But we can also use a non-uniform distribution of the samples:

$$\int_0^1 f(x)\,dx \;\approx\; \frac{1}{n}\sum_{i=1\ldots n} \frac{f(x_i)}{p(x_i)}$$

- assumes that the x_i are distributed according to some probability $p(x_i)$

Probability Distribution Functions

- $p(x)$ is a (continuous) probability distribution function (PDF)
- continuous → probability of a particular value x is zero, but probability that a value is from a range $[x_0, x_1]$ is non-zero:

$$P(x \in [x_0, x_1]) = \int_{x_0}^{x_1} p(x)\, dx$$

- Special case: cumulative probability:

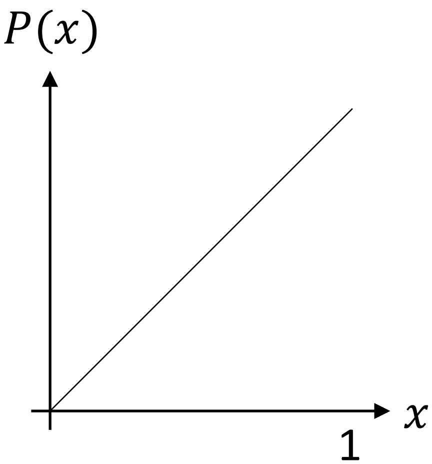
$$P(\bar{x}) = P(x < \bar{x}) = \int_{-\infty}^{\bar{x}} p(x)\, dx$$

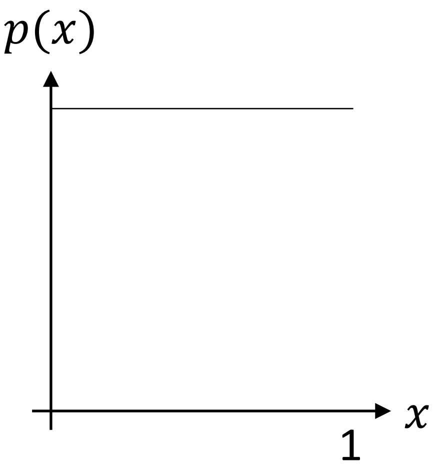
- vice versa:

$$p(x) = \frac{\partial P}{\partial x}(x)$$

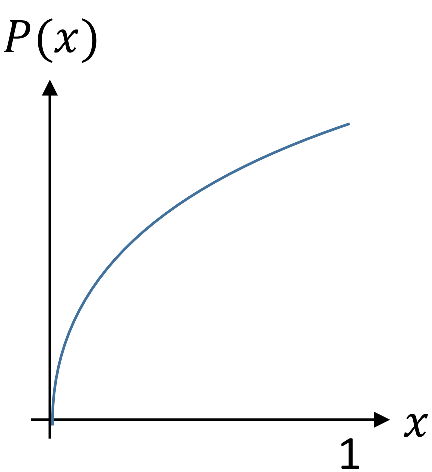
Probability Distribution Functions

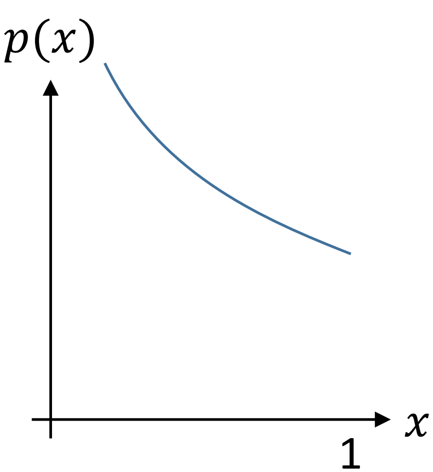
- Example 1: uniform distribution over $[0,1]$
  - $P(X < x) = x$
    $\rightarrow p(x) = 1$





- Example 2: draw uniform random number $\xi \in [0,1]$, then use ξ^2
  - with $\xi \rightarrow \xi^2$ samples are shifted to the left, e.g. 0.1 → 0.01, 0.5 → 0.25
  - $P(\xi^2 < x) = P(\xi < \sqrt{x}) = \sqrt{x}$
    $\rightarrow p(x) = \dfrac{1}{2\sqrt{x}}$
  - p even has a singularity at 0, but is integrable





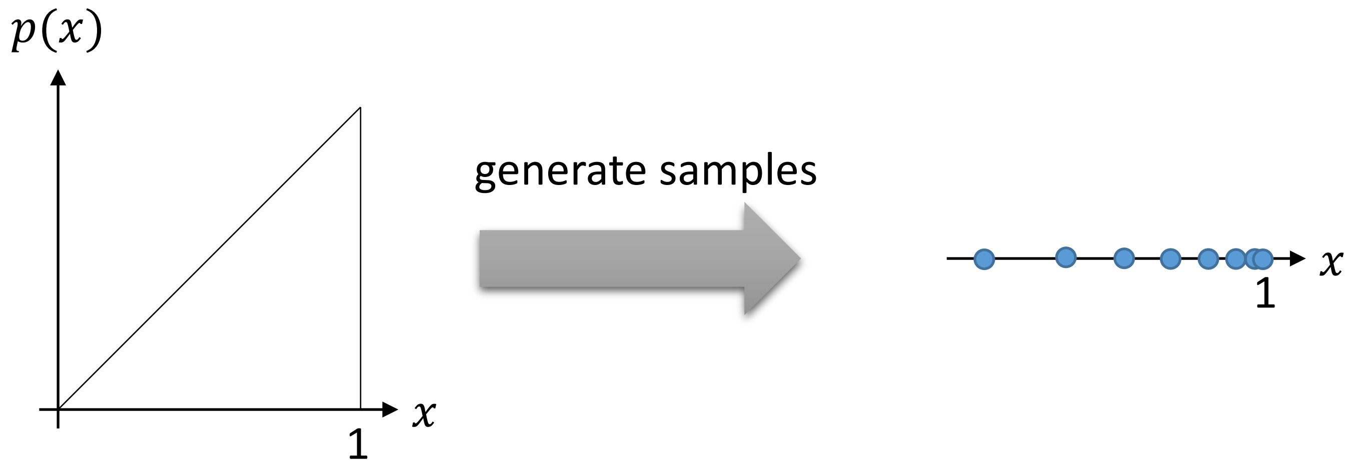
Probability Distribution Functions

- Properties:
  - $p(x) \geq 0$
  - $\int_{-\infty}^{\infty} p(x)\ dx = 1$
  - $p(x)$ can be > 1 for some x !

- Such distributions can also be defined on higher dimensional domains, e.g. also on a triangle, a rectangle, a sphere, or a hemisphere

Probability Distributions for Sampling

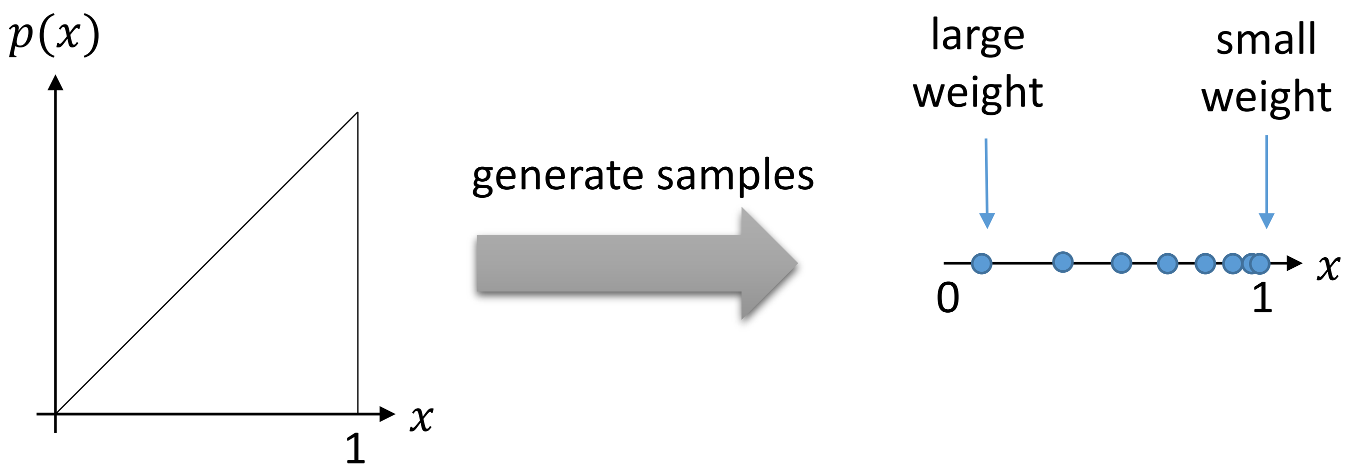
- $p(x)$ is a function on the sampling domain that describes how likely it is that a sample is chosen around x
- Example in 1D:



- Sample density increases to the right
  → values close to $x = 1$ are overemphasized
  → when used for Monte Carlo Integration without correction, a biased result is obtained

Importance Sampling

- weight samples x_i with $\frac{1}{p(x_i)}$
  → there are fewer samples with small x, but these get larger weight
  → and more samples with larger x, but with lower weight
  → correct result, but region close to 1 sampled more densely



- when does this non-uniform sampling make sense ?

Importance Sampling

- We know:

$$\int_0^1 f(x)\,dx \;\approx\; \frac{1}{n} \sum_{i=1\ldots n} \frac{f(x_i)}{p(x_i)}$$

- Effectively, we integrate f/p
- If we roughly know f, we can choose p such that f/p has low variance
  → faster convergence!
- p should have a "similar shape" as f
- Ideally, $p = c \cdot f$, then we effectively integrate $\frac{f}{p} = \frac{1}{c} = const$, and we get a proper result with one single sample !
- ???
- We have to choose c such that p becomes a probability distribution:

$$\int p(x)dx = 1 \;\rightarrow c = 1/\int f(x)dx$$

- → we must know the integral

Importance Sampling

- How can we apply all this for our rendering integral ?

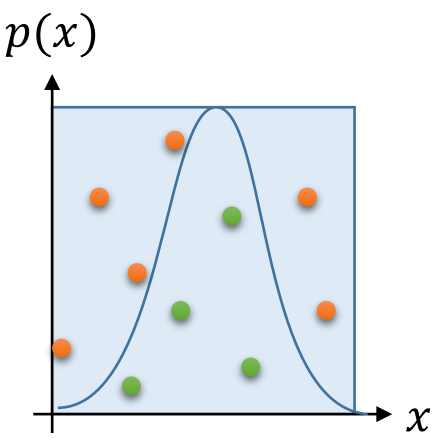
$$L_{out}(x, \omega_{out}) = \int f(x, \omega_{in}, \omega_{out}) < \mathrm{N_x}, \omega_{in} > L_{in}(x, \omega_{in})\, d\omega_{in}$$

- We can compute a PDF which is proportional to f and generate sample directions ω_{in} according to this PDF
  → depends on the BRDF, solutions are known e.g. for Phong
  → variance due to f is canceled out
  → faster convergence
- Or we generate a PDF proportional to the scalar product and sample with respect to this
- Or we generate a PDF according to L_{in}
  → see Lecture **Environment Lighting**
- Or we generate PDFs with respect to all factors and combine these
  → see Lecture **Multiple Importance Sampling**

PDF Sampling

- How can we generate samples with respect to a given PDF ?

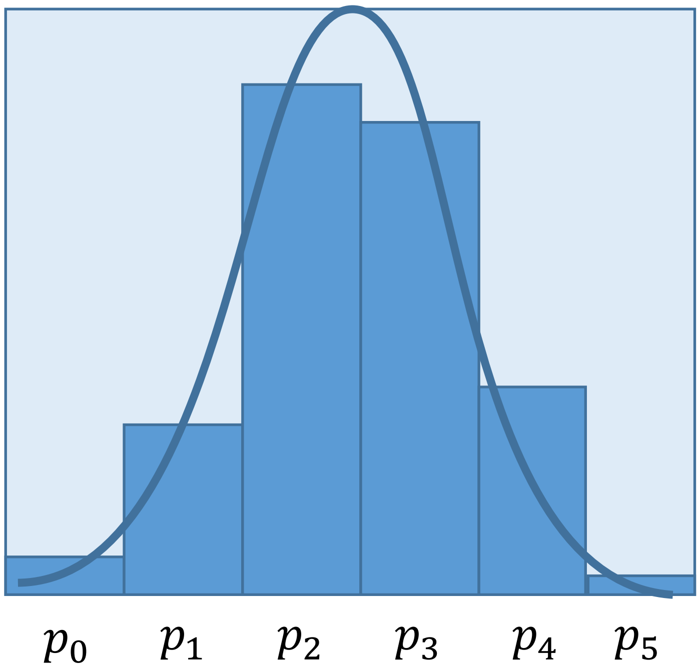
- Rejection Sampling:
  - generate a box around the graph of p
  - draw random samples within this box
  - only use samples under the curve
    → probability of a sample to survive
    is proportional to p

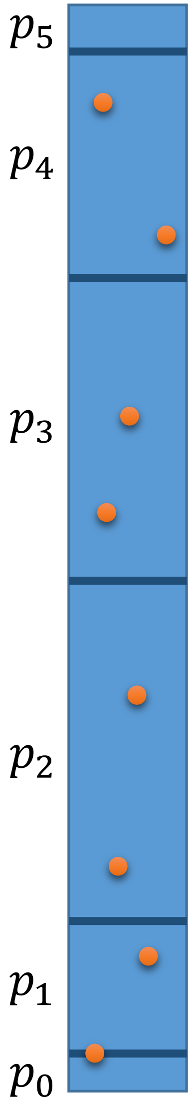


- Issues:
  - We need to know the maximum of p
  - Can become very slow if maximum is large
    → most samples get rejected

PDF Sampling – Discretized Version

- By discretization: Discretize PDF and stack rectangles
- draw a random point in the resulting rectangle (→ 2D-sample)
- requires binary search to find rectangle for a given height value

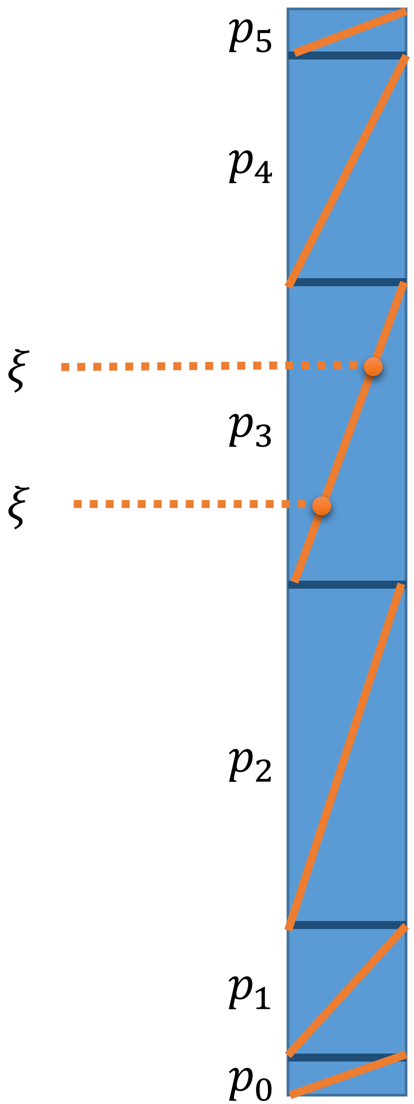




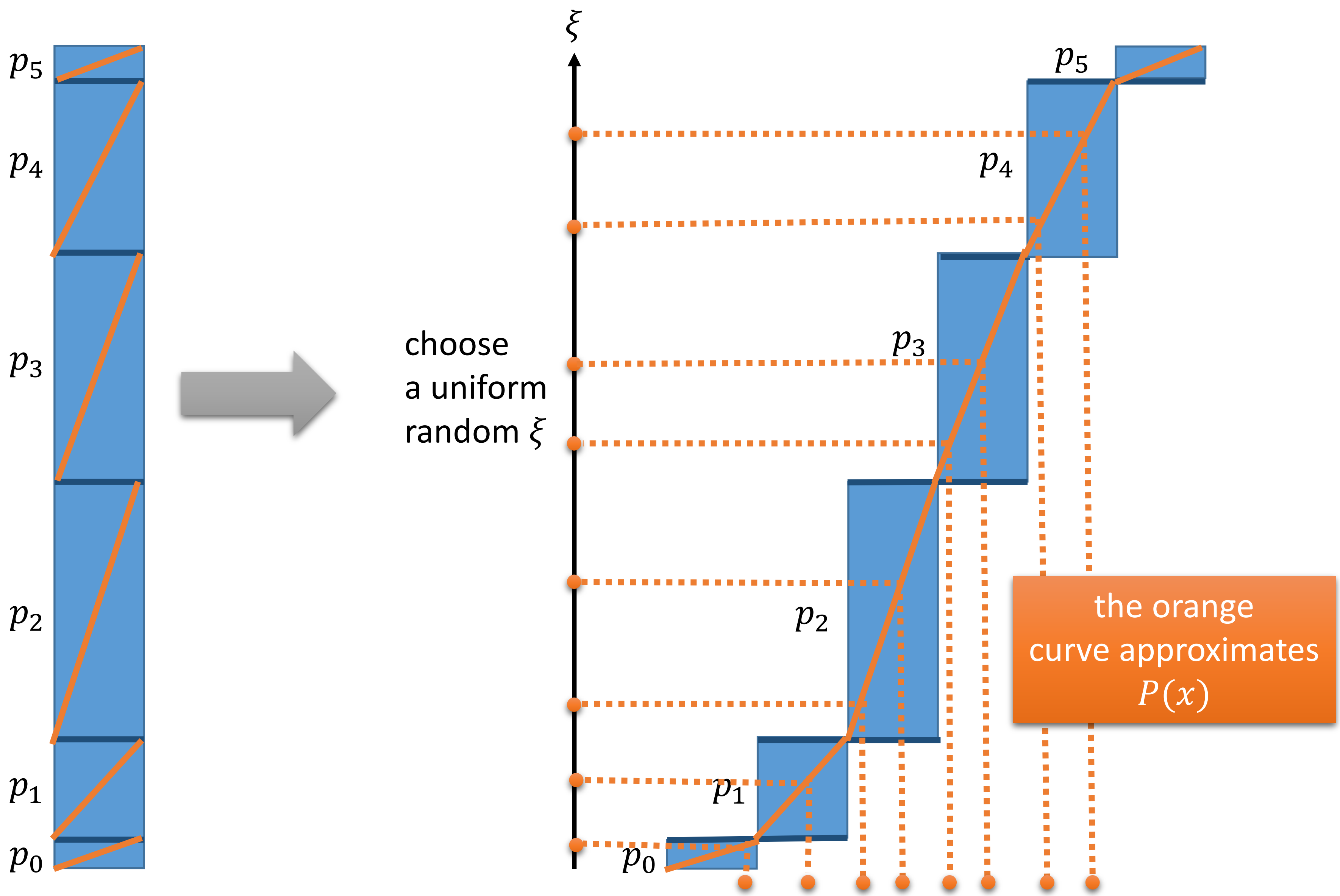
- approximate only
- more difficult for multidimensional PDFs

PDF Sampling – Discretized Version

- Variant: we can cope with only one random sample
- Say, we use random sample ξ to select rectangle (y-coordinate)
- Use position of ξ within this rectangle to determine x-position



PDF Sampling – Discretized Version



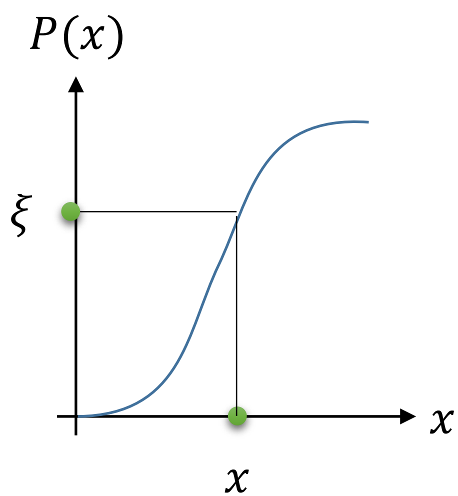
PDF Sampling

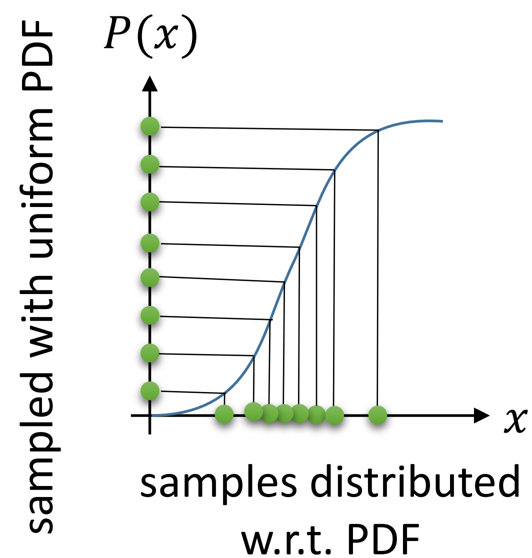
- Continuous version of previous approach: Inversion method
- Compute cumulative probability:

$$P(x) = \int_{-\infty}^{x} p(x')\,dx'$$

corresponds to stacking on previous slides

- Invert P
- draw a random number $\xi \in [0,1]$
- use as sample $x = P^{-1}(\xi)$





PDF Sampling

- For the inversion method, we must be able to normalize f, to compute P, and to invert it
- Typical programming interface for a BRDF thus looks like this:

```
BRDF::sample(
      const vec3 &normal, const vec3 &w_out,
      vec3 &w_in, float &pdf)
```

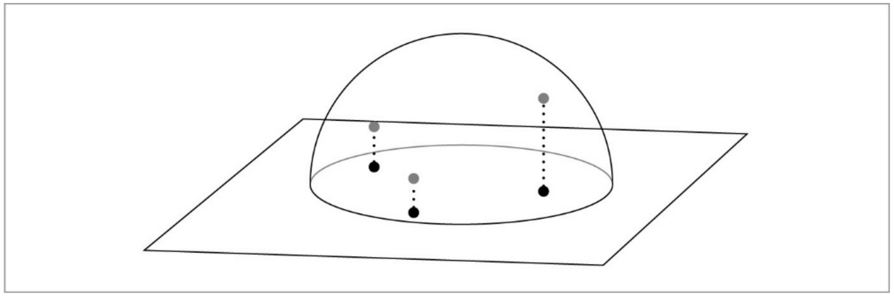
→ for a given normal and outgoing direction, generate a sample direction **w_out** and its corresponding PDF-value
→ simply weight sample by 1/***pdf**
→ default implementation returns uniform direction and pdf
→ if implementation supports importance sampling, you will automatically get better results

PDF Sampling

- Example 1: Sampling with respect to the cosine (scalar product):
  → **Malley's method:**

  - Sample unit disk → (x, y)
    → sample point on unit square, reject all outside the unit disk

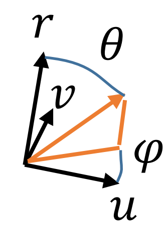
  - Lift to hemisphere → $\left(x, y, \sqrt{1 - x^2 - y^2}\right)$



From: Physically Based Rendering, Section 13

PDF Sampling

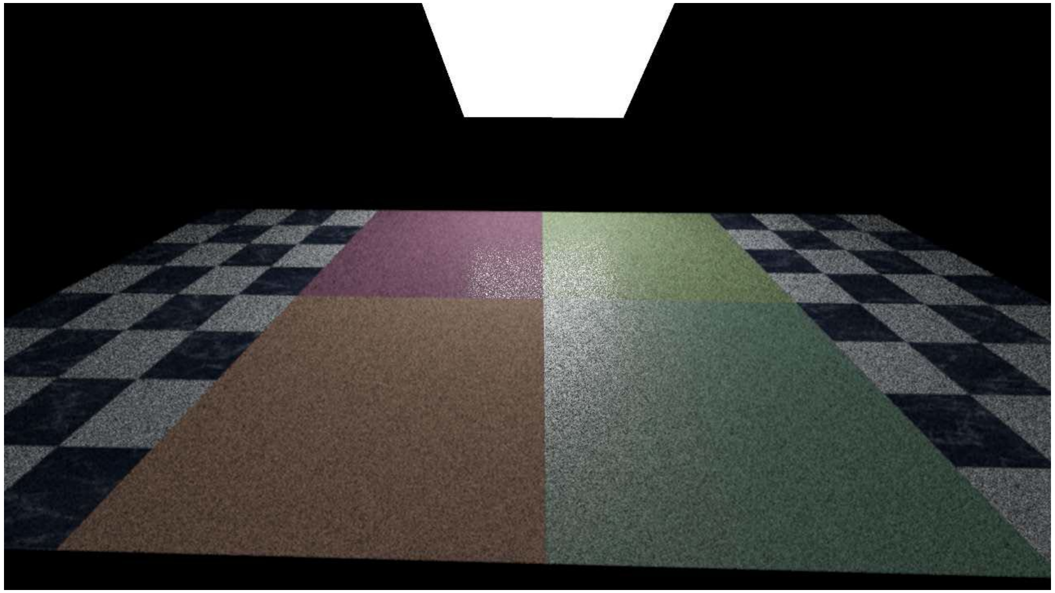
- Example 2: importance sampling of Phong BRDF:
  - Compute reflection direction r
  - Generate basis vectors (u, v, r)
  - draw random variables ξ_1, ξ_2
  - $\theta = arc\cos \sqrt[n+1]{\xi_1}, \varphi = \xi_2 \cdot 2\pi$
  - generate direction $\omega = \cos\theta \cdot r + \sin\theta \cdot (\cos\varphi \cdot u + \sin\varphi \cdot v)$



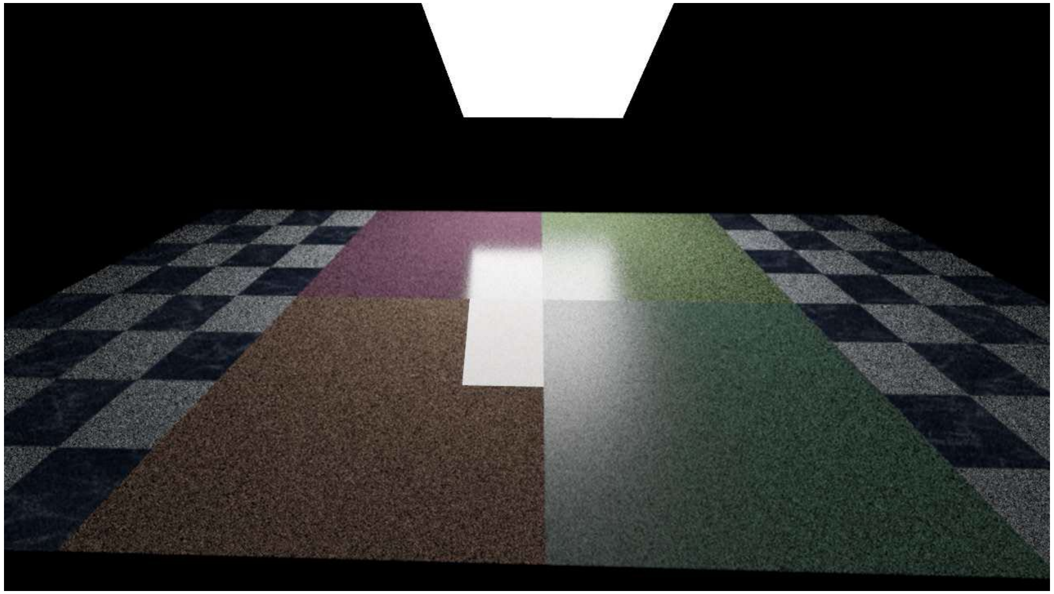
- ω distributed with respect to Phong BRDF with exponent n

Current Assignment

- Uniform sampling of hemisphere

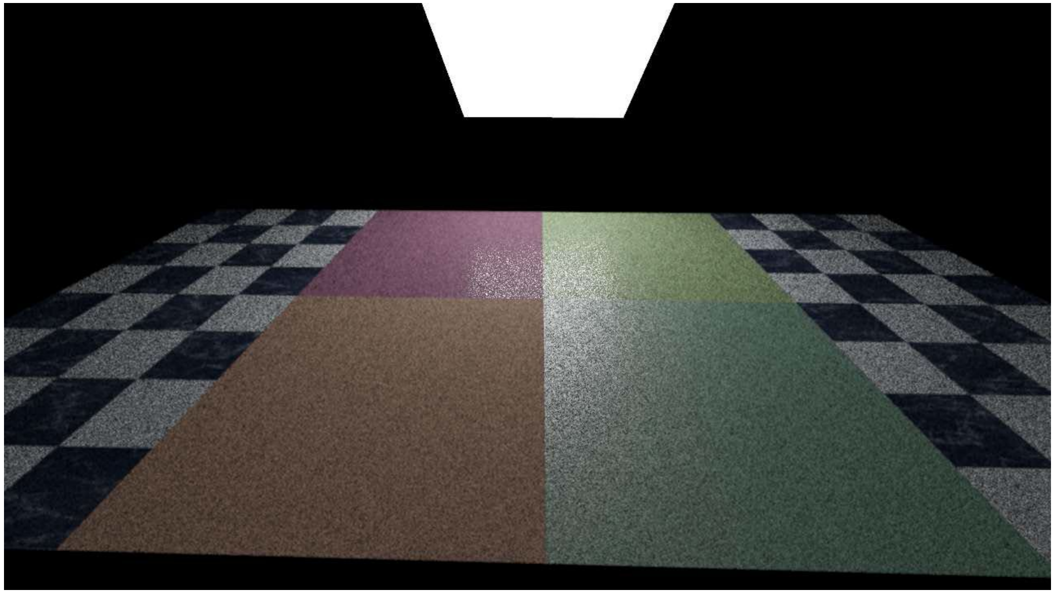


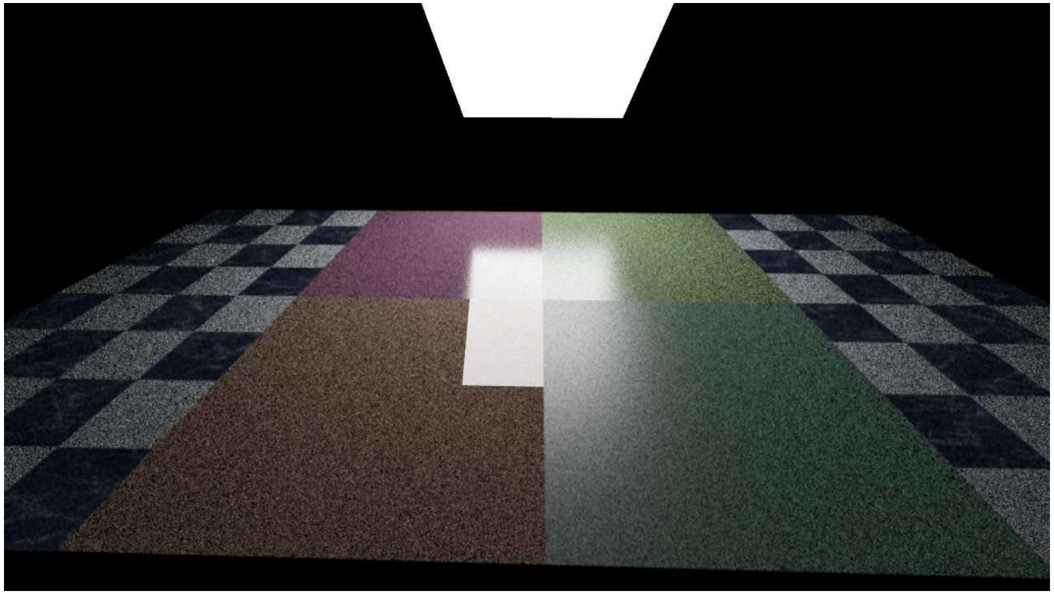
- Sampling w.r.t. BRDF



Current Assignment

- in this example:
  light comes only from a single light source
- → we only consider rays that hit the light source
- → wasteful, if light source is small





Alternative Formulation

- $L_{out}(x, \omega_{out}) = \int f\left(x, \frac{x-y}{\|x-y\|}, \omega_{out}\right)\ G(x,y)\ V(x,y)\ L\left(y, \frac{y-x}{\|y-x\|}\right) dA_y$

- We integrate over light emitting surfaces
  → makes particular sense for light sources

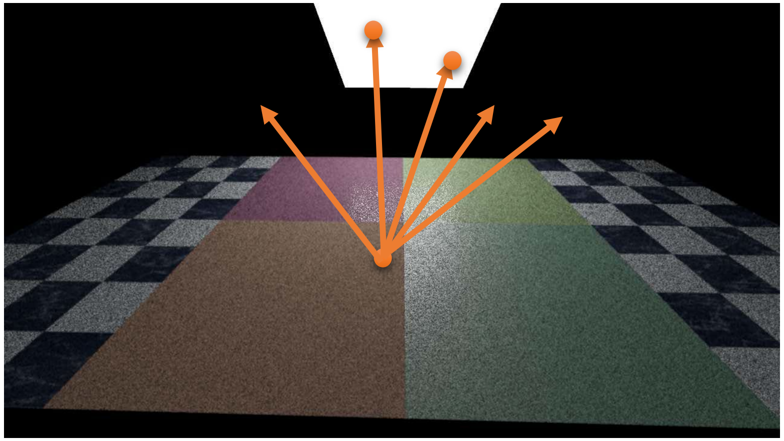
direct lighting
→ we only use L_e here !

- Direct Lighting:

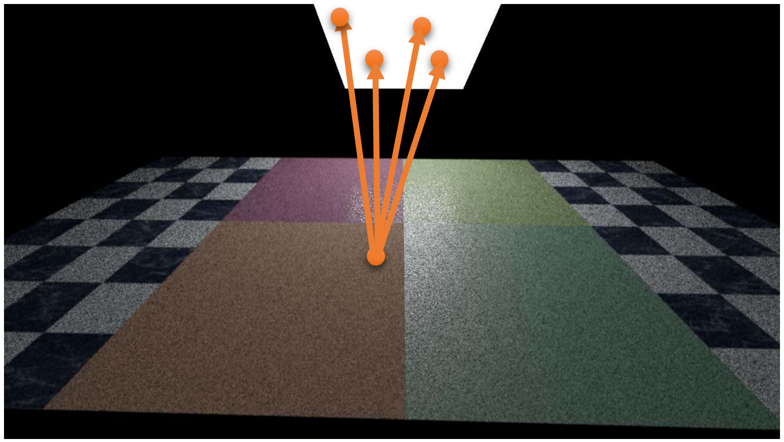
$$L_{out}(x, \omega_{out}) = \int f\left(x, \frac{x-y}{\|x-y\|}, \omega_{out}\right)\ G(x,y)\ V(x,y)\ L_e\left(y, \frac{y-x}{\|y-x\|}\right) dA_y$$

Direct Lighting Computation

- Approach #1:
  Cast rays over hemisphere
  → only rays hitting light contribute
  → BRDF importance sampling possible



- Approach #2:
  Sample light source
  → we need to check visibility
  → most rays contribute
  → BRDF importance sampling
  not directly possible…



Direct Lighting Computation

- Generate a random sample…
- … on a rectangle a, b, c, d:
  - parameterize rectangle with two parameters $\alpha, \beta \in [0,1]^2$:
    $$x(\alpha, \beta) = a + \alpha(b - a) + \beta(d - a)$$
  - draw random values for α, β
- … on a triangle a, b, c:
  - parameterize with two parameters $\alpha, \beta \in [0,1]^2$ and $\alpha + \beta < 1$:
    $$x(\alpha, \beta) = a + \alpha(b - a) + \beta(c - a)$$
  - draw random values for α, β until $\alpha + \beta < 1$
  - or: draw random α, β, if $\alpha + \beta > 1$ then $\alpha \leftarrow 1 - \alpha, \beta \leftarrow 1 - \beta$
- … on a polygon:
  - compute plane and a bounding rectangle
  - draw random points on rectangle, until one inside polygon is found

Direct Lighting Computation

- Generate a random sample…
- … on a sphere:
  - parameterize sphere with two parameters $\alpha \in [0,1], \beta \in [-1,1]$:

$$x(\alpha,\beta) = \begin{pmatrix} \sqrt{1-\beta^2}\cos 2\pi\alpha \\ \sqrt{1-\beta^2}\sin 2\pi\alpha \\ \beta \end{pmatrix}$$

  - draw random (α, β) → random point on sphere with uniform distribution
  - or draw a random point in $[-1,1]^3$ until it lies in unit sphere

- To draw the (random) samples α, β, we can use the sampling techniques shown in Lecture #2

Direct Lighting – Importance Sampling

- $\int f\left(x, \frac{x-y}{\|x-y\|}, \omega_{out}\right) \; G(x,y) \; V(x,y) \; L_e\left(y, \frac{x-y}{\|x-y\|}\right) dA_y$

high variance for glossy BRDF

high variance for close light sources

high variance along shadow boundary

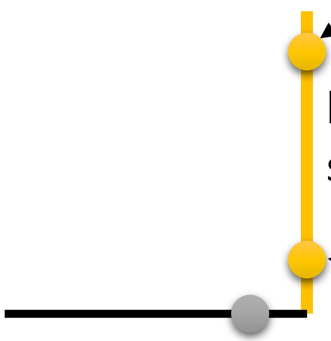
usually no variance

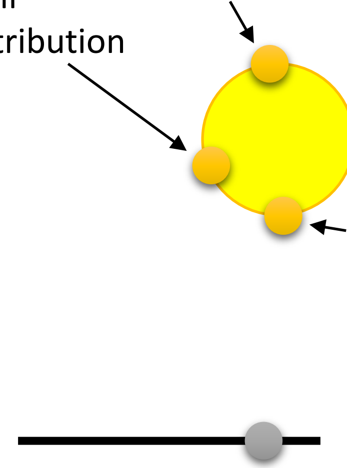
Direct Lighting – Importance Sampling

- Importance sampling with respect to BRDF
  → just done
- Importance sampling w.r.t. geometric term G ?
  → next
- Importance sampling w.r.t. visibility V ?
  → difficult, we do not know V and sampling it is expensive
- Importance sampling w.r.t. light emission ?
  → next lecture, needed e.g. for **environment lighting**
- Importance sampling w.r.t. multiple terms
  → later lecture: **multiple importance sampling**

Importance Sampling of Geometric Term

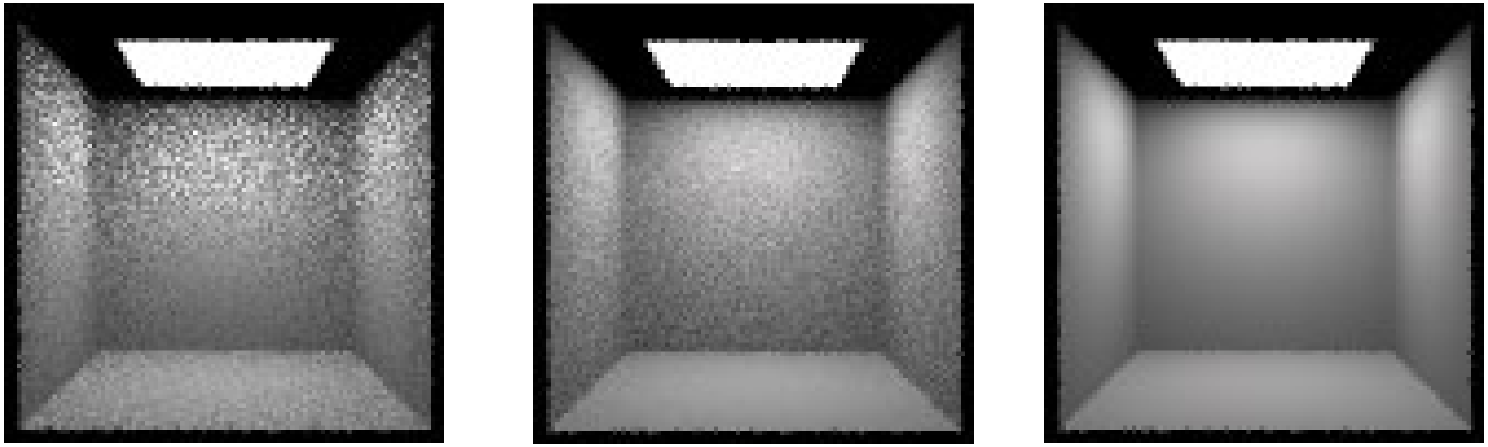
- Shirley et al.: "Monte Carlo Techniques for Direct Lighting Computations"
- For now, let's assume we have diffuse surfaces
  → BRDF=const → variation mostly due to G and V
- We know nothing about V, and evaluation is expensive
  → importance sampling w.r.t. G
- Examples:





Importance Sampling of Geometric Term

- Applying the inversion method for typical light source geometries (rectangle, sphere, triangle, …) is difficult, but possible
- E.g., for rectangles, taken from "The Direct Lighting Computation in Global Illumination Methods":

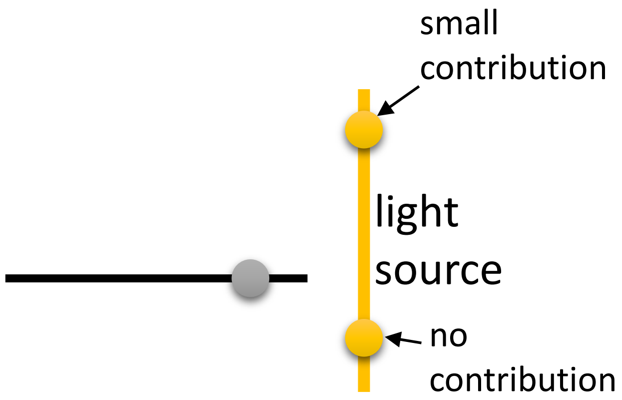


$$p_1 = \frac{1}{A} \qquad p_2 \sim \frac{\cos\theta_2}{d^2} \qquad p_3 \sim \frac{\cos\theta_1\ \cos\theta_2}{d^2}$$

- Inversion formulas are very complicated, and require numerical inversion

Importance Sampling of Geometric Term

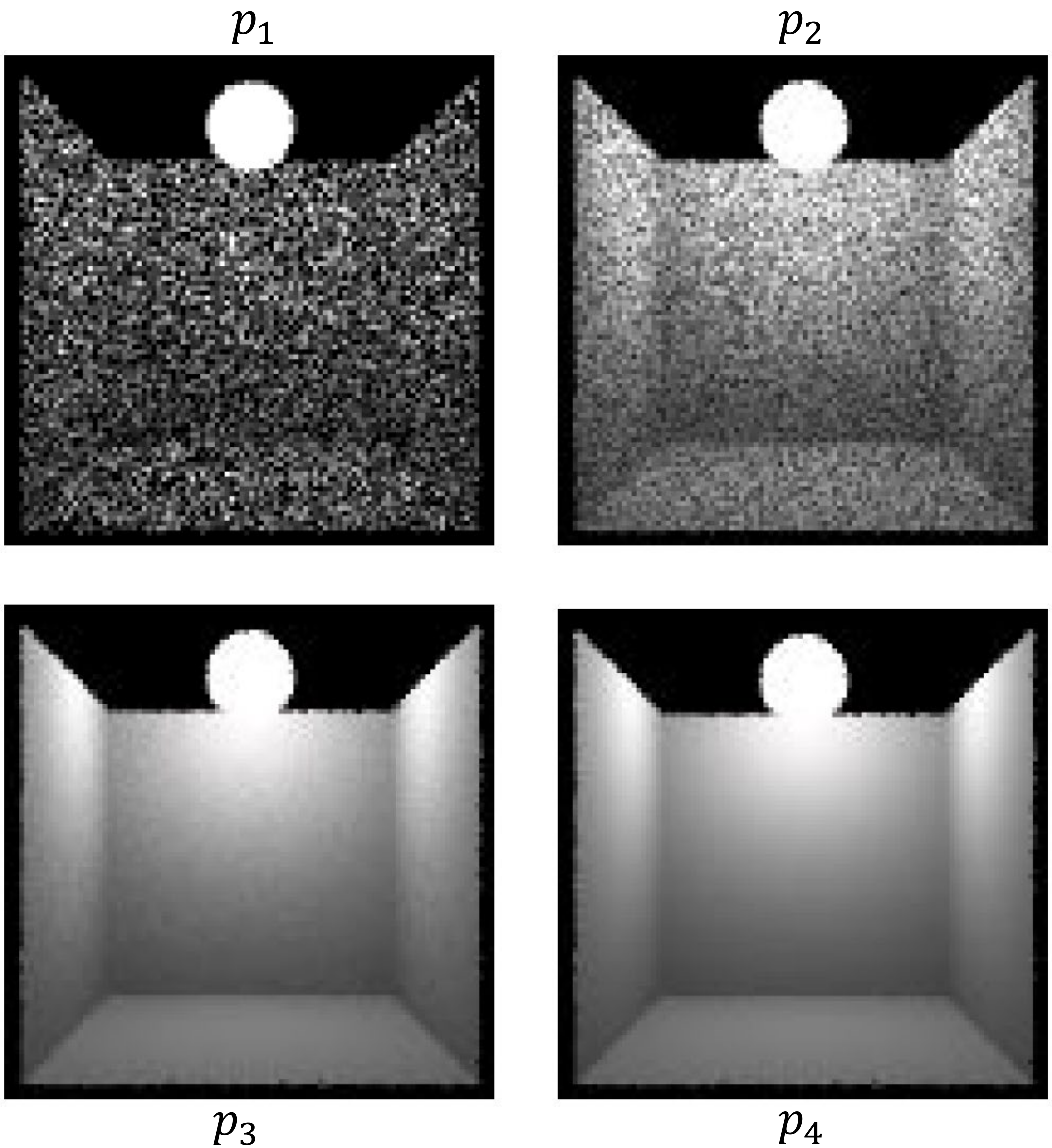
- Practical problem for rectangles:
  lights "below the horizon"



- More practical approach for rectangular lights:
  - use uniform sampling on rectangle
  - check the sample for being "under the horizon" ($\cos\theta_1 < 0$) before evaluating visibility
  - can be seen as "rejection sampling"

Importance Sampling of Geometric Term

- For spherical luminaires:
  - sample entire sphere:
  - $p_1 = \frac{1}{A}$
  - sample visible part of sphere:
  $$p_2 = \frac{1}{A_{vis}}$$
  - sample solid angle:
  $$p_3 \sim \frac{\cos\theta_2}{d^2}$$
  → closed form for inversion available !
  - sample w.r.t. G:
  $$p_4 \sim G \sim \frac{\cos\theta_1 \, cos\theta_2}{d^2}$$
  → numerical inversion needed
  → sampling is expensive



Multiple Light Sources

- Up to now we considered a single light source
- How about multiple, maybe many (1000s) of light sources L_i ?

- $\sum_i \int_{L_i} f(\ldots)G(\ldots)V(\ldots)dA_y$

- Do we have to sample each light source ?

- No, we can unite all light sources to a single large one !
  → no one claimed that a light source must be connected
  → $L = \cup L_i$

- Then we can generate an arbitrary number of samples on this light
  → correct expected value with less samples than light sources !

Multiple Light Sources

- How to do this ?

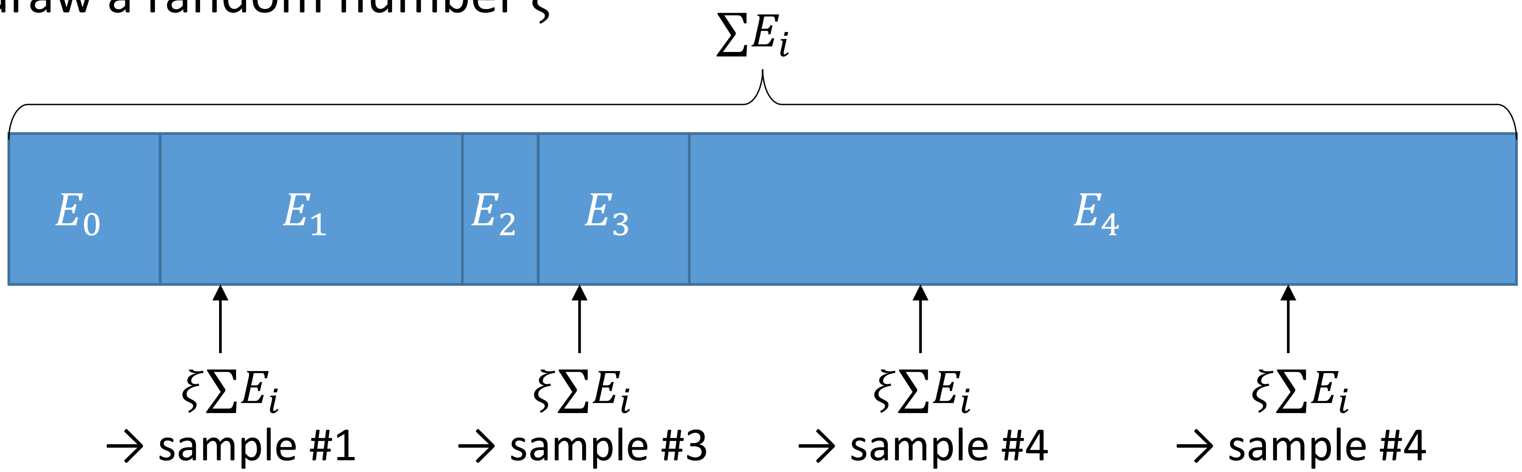
- Solution #1:
  - for each sample, draw a random number $\xi \in [0,1]$
  - select light source $i = \lfloor \xi n \rfloor$ (n: number of light sources)
  - chose a random sample on light source i

  - → gives a proper result with correct expected value

  - all light sources get same sample number, maybe we can distribute samples better (more samples to brighter lights) ?

Multiple Light Sources

- Solution #2:
  Compute emitted power of each light source: $E_i = L_i A_i$
- Choose a sample with a probability proportional to E_i
  → draw a random number ξ

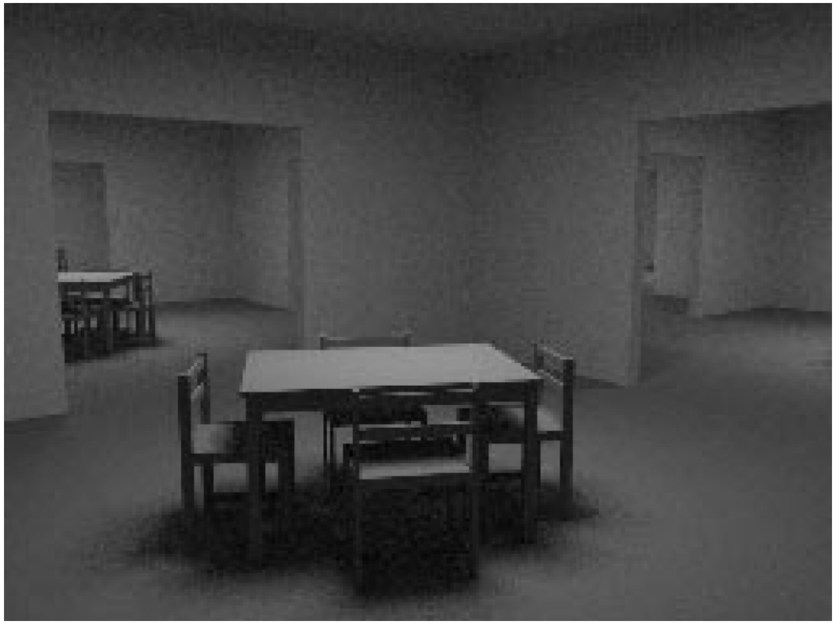


- This is importance sampling of the light source w.r.t. its power !

Multiple Light Sources

- Even better solution #3:
  We can adapt the importance of a light source for each surface point

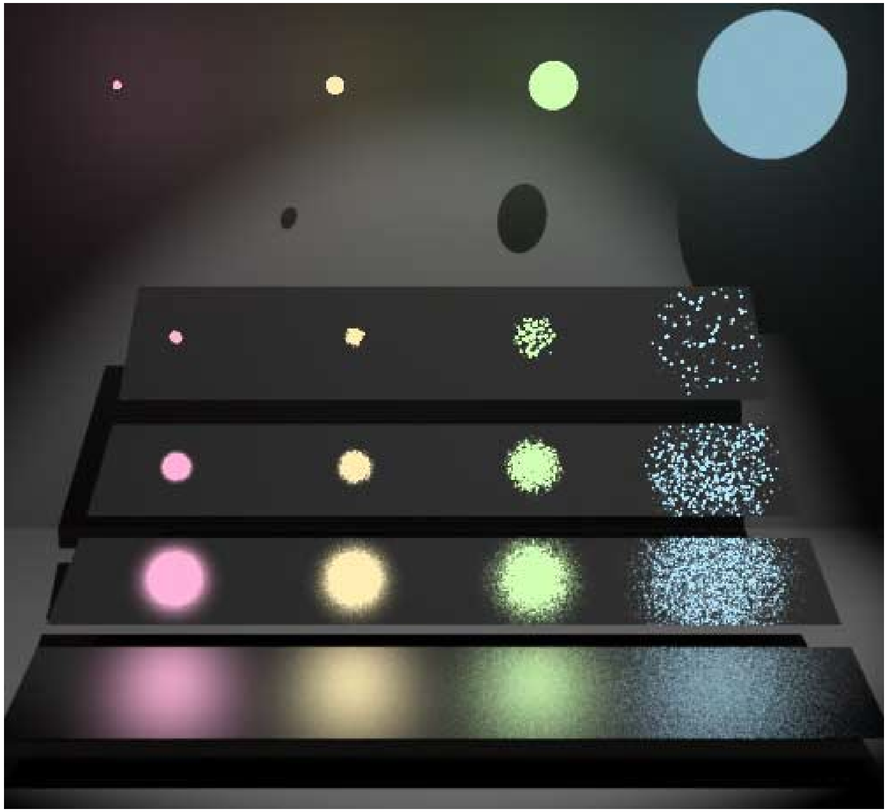
- E.g. estimate the contribution of each light source for the current point, ignoring visibility
  - e.g. using one sample of G
  - or using an analytic formula,
    e.g. for spheres
  - be careful not to underestimate the contribution as zero
    → light source will get no sample,
    even if it contributes…
- recompute E_i for each sample



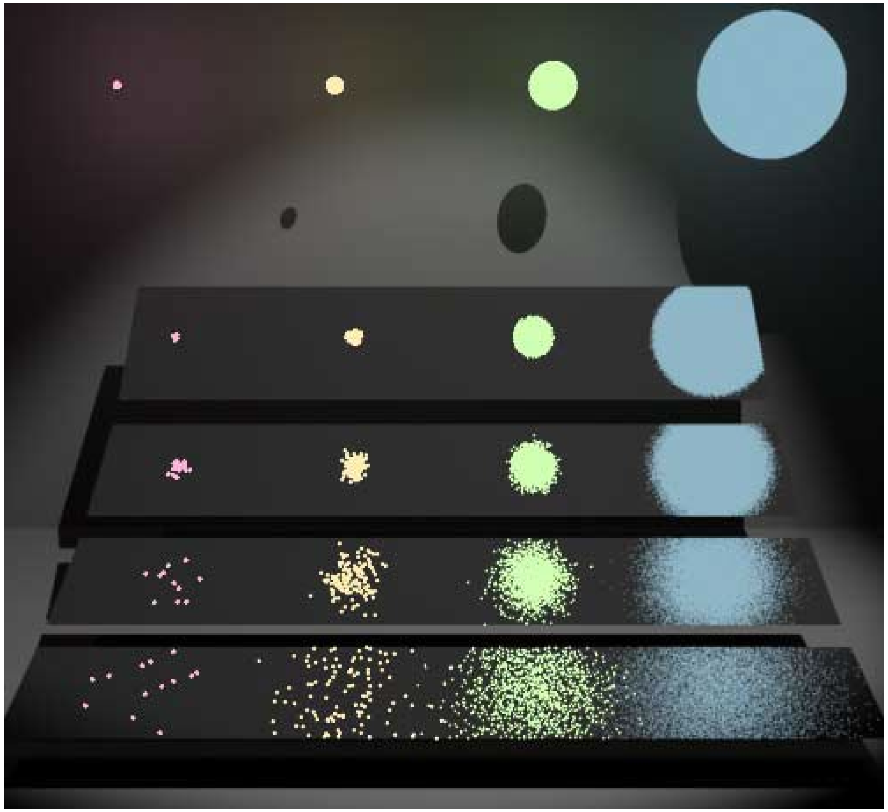
Shirley at al.:
A scene with 100 luminaires,
sampled by 49 samples per pixel

Sampling Strategies

- Sampling the light source



- Sampling the BRDF



Veach et al.: "[Optimally Combining Sampling Techniques for Monte Carlo Rendering]"

Next Week: Combining Sampling Methods