Computer Graphics I

Lecture 11: Numerical integration

Xiaopei LIU

School of Information Science and Technology
ShanghaiTech University

Rendering equation

- **The fundamental rendering equation**
  - Reflection equation
    - Describe how an incident distribution of light at a point is transformed into an outgoing distribution

$$L_o(p, \omega_o) = \int_{\mathcal{S}^2} f(p, \omega_o, \omega_i)\, L_i(p, \omega_i)\, |\cos\theta_i|\, d\omega_i$$

  - Scattering equation
    - More complex integral equation

$$L_o(p_o, \omega_o) = \int_A \int_{\mathcal{H}^2(\mathbf{n})} S(p_o, \omega_o, p_i, \omega_i)\, L_i(p_i, \omega_i)\, |\cos\theta_i|\, d\omega_i\, dA$$

  - **We need to evaluate the integral**
    - Accurately
    - Efficiently

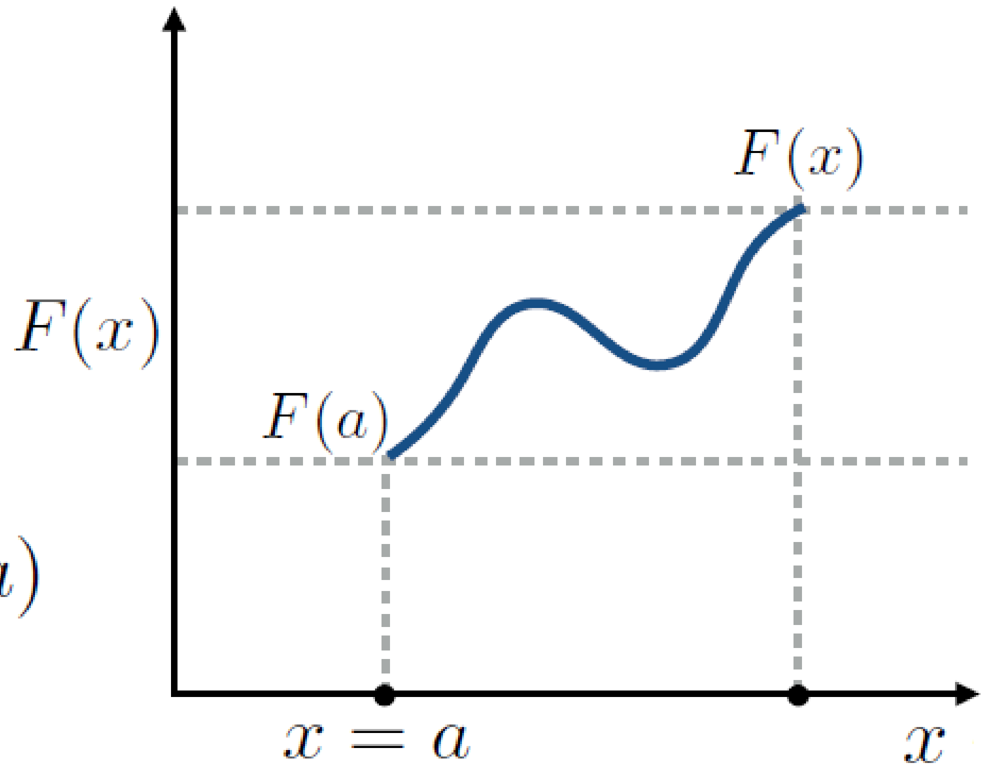
1. Traditional numerical integration

Review: fundamental theorem of calculus

$$\int_a^b f(x)dx = F(b) - F(a)$$

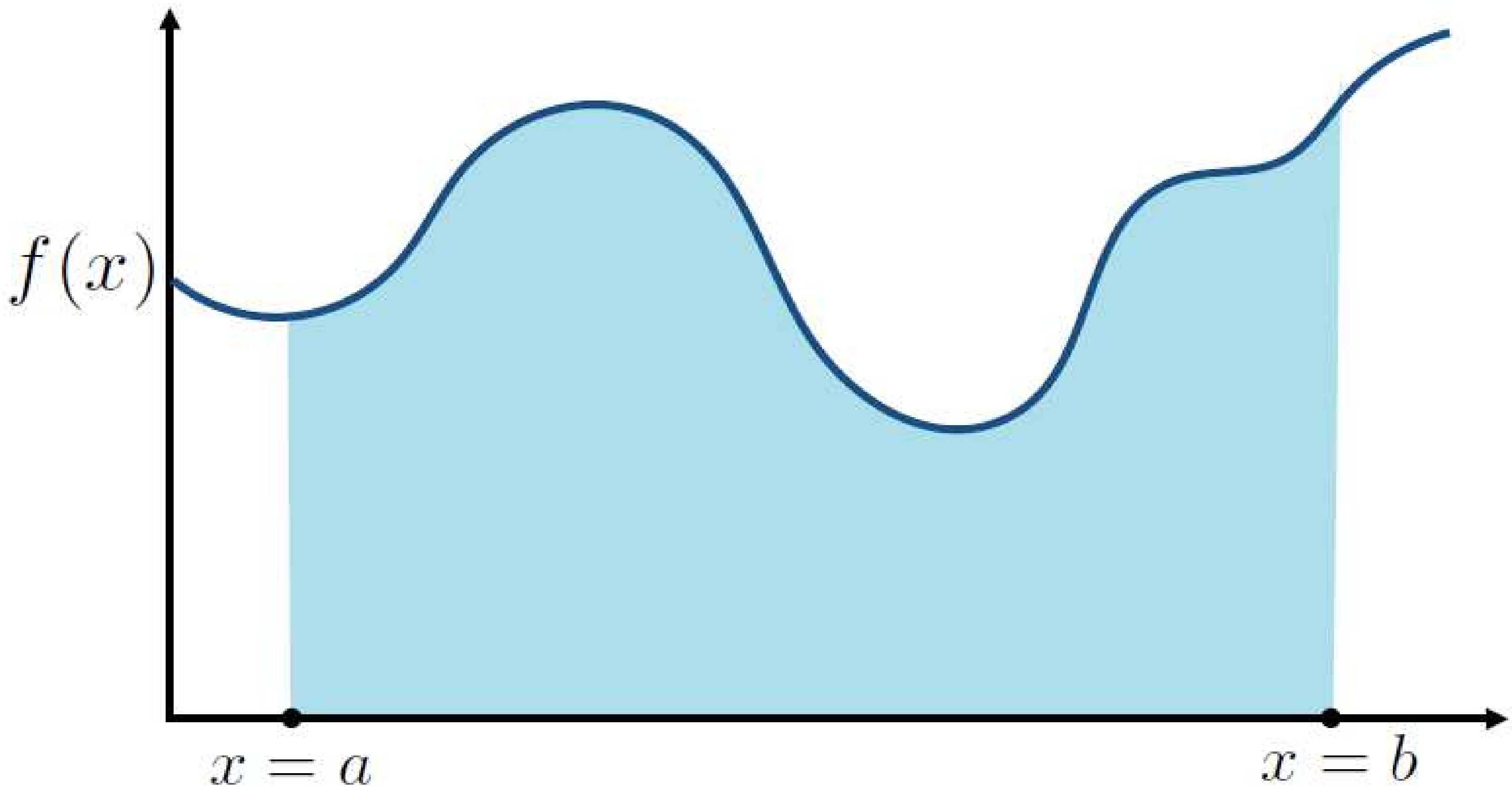
$$f(x) = \frac{d}{dx}F(x)$$

$$\int_a^x f(t)dt = F(x) - F(a)$$



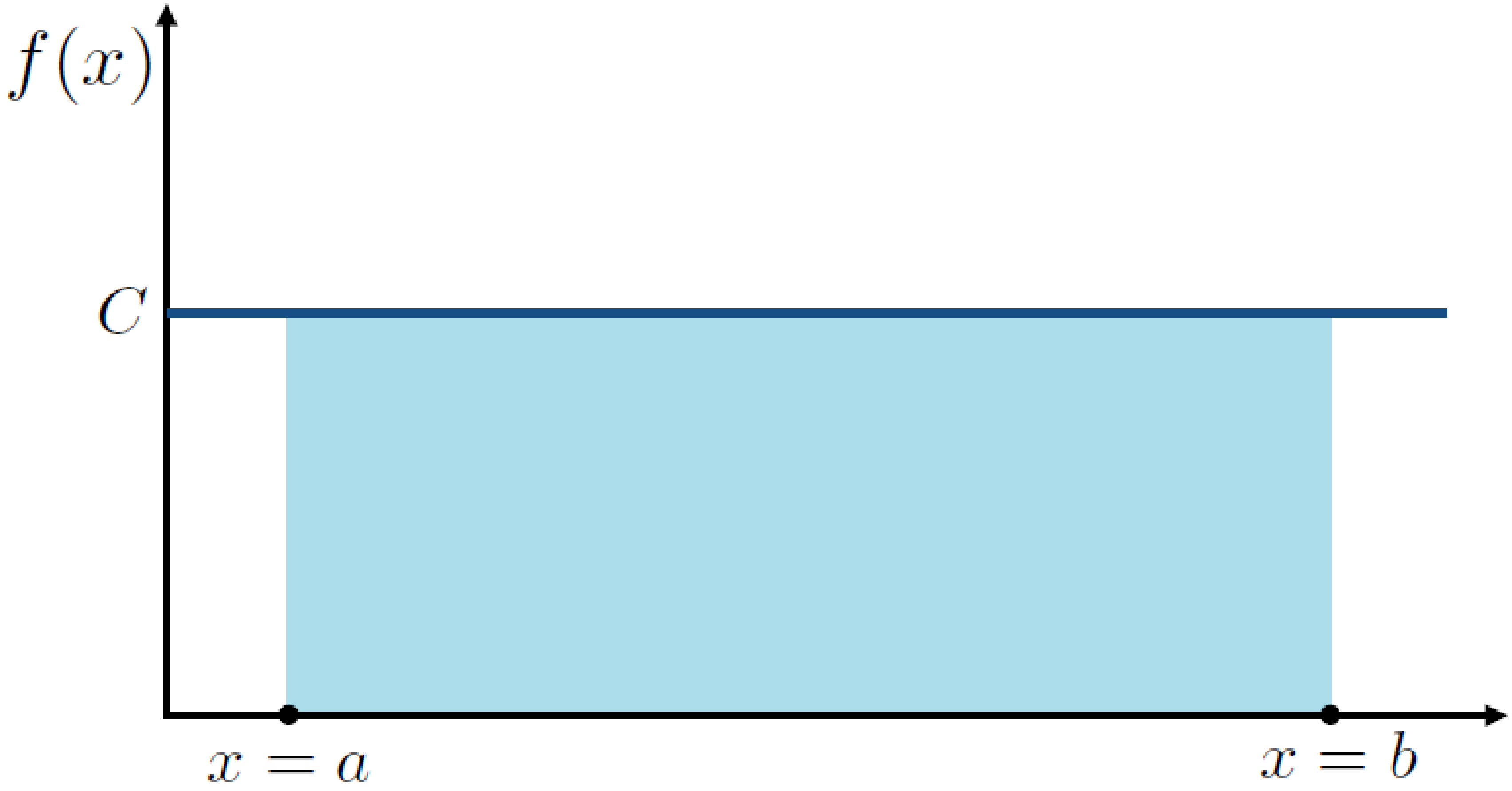
Definite integral as "area under curve"

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



Simple case: constant function

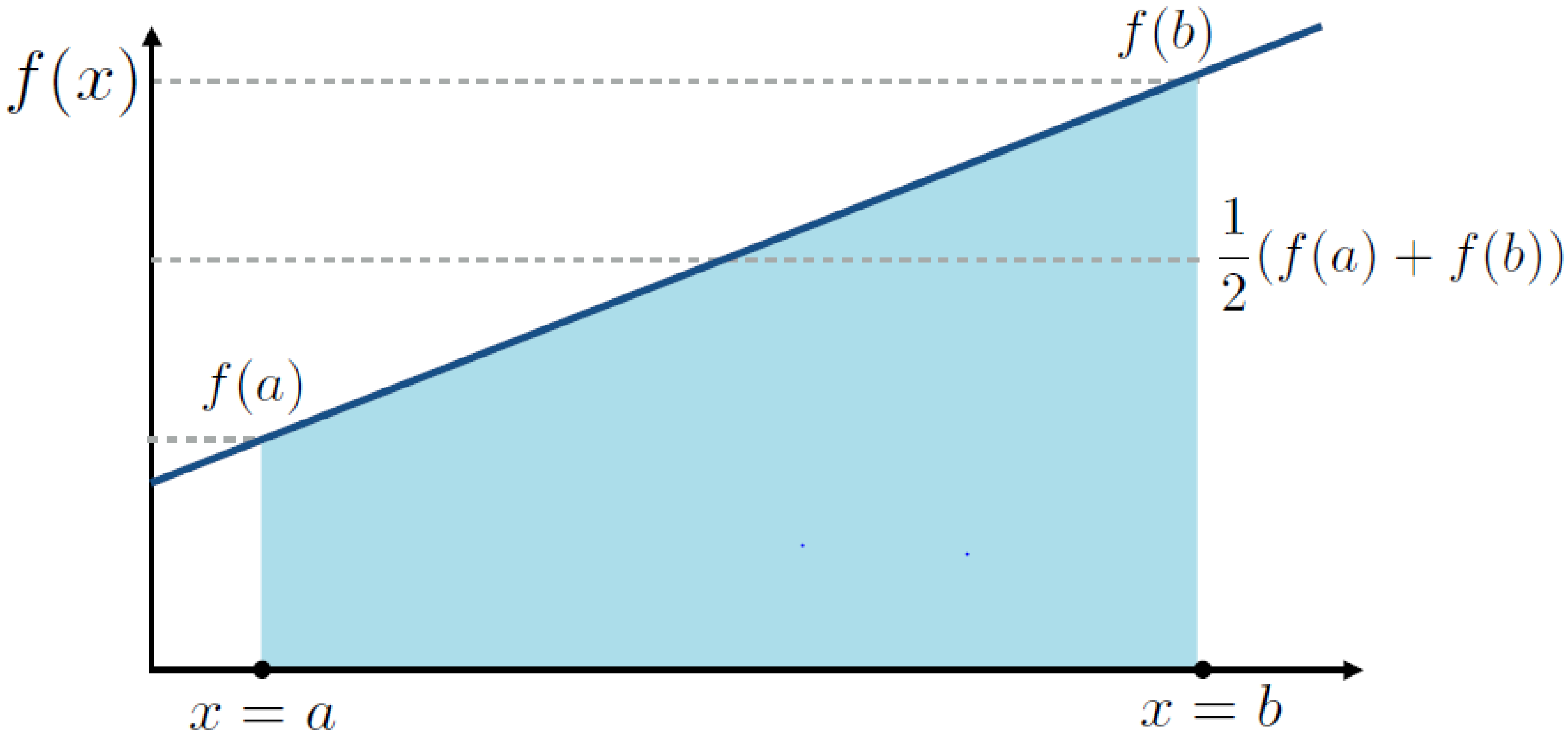
$$\int_a^b C dx = (b-a)C$$



Linear affine function

Affine function: $f(x) = cx + d$

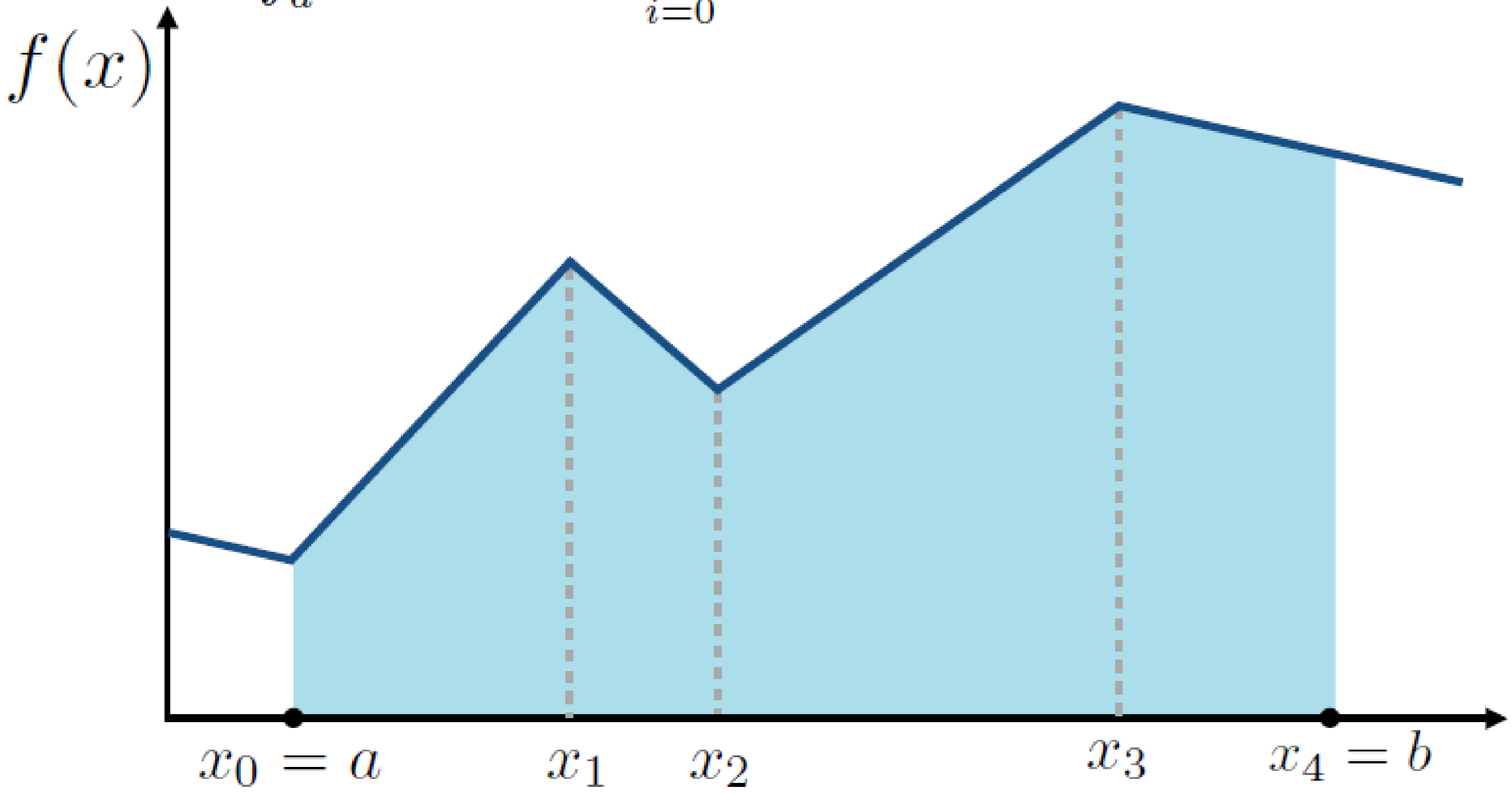
$$\int_a^b f(x)dx = \frac{1}{2}(f(a) + f(b))(b - a)$$



Piecewise affine function

Sum of integrals of individual affine components

$$\int_a^b f(x)dx = \frac{1}{2}\sum_{i=0}^{n-1}(x_{i+1} - x_i)(f(x_i) + f(x_{i+1}))$$



Piecewise affine function

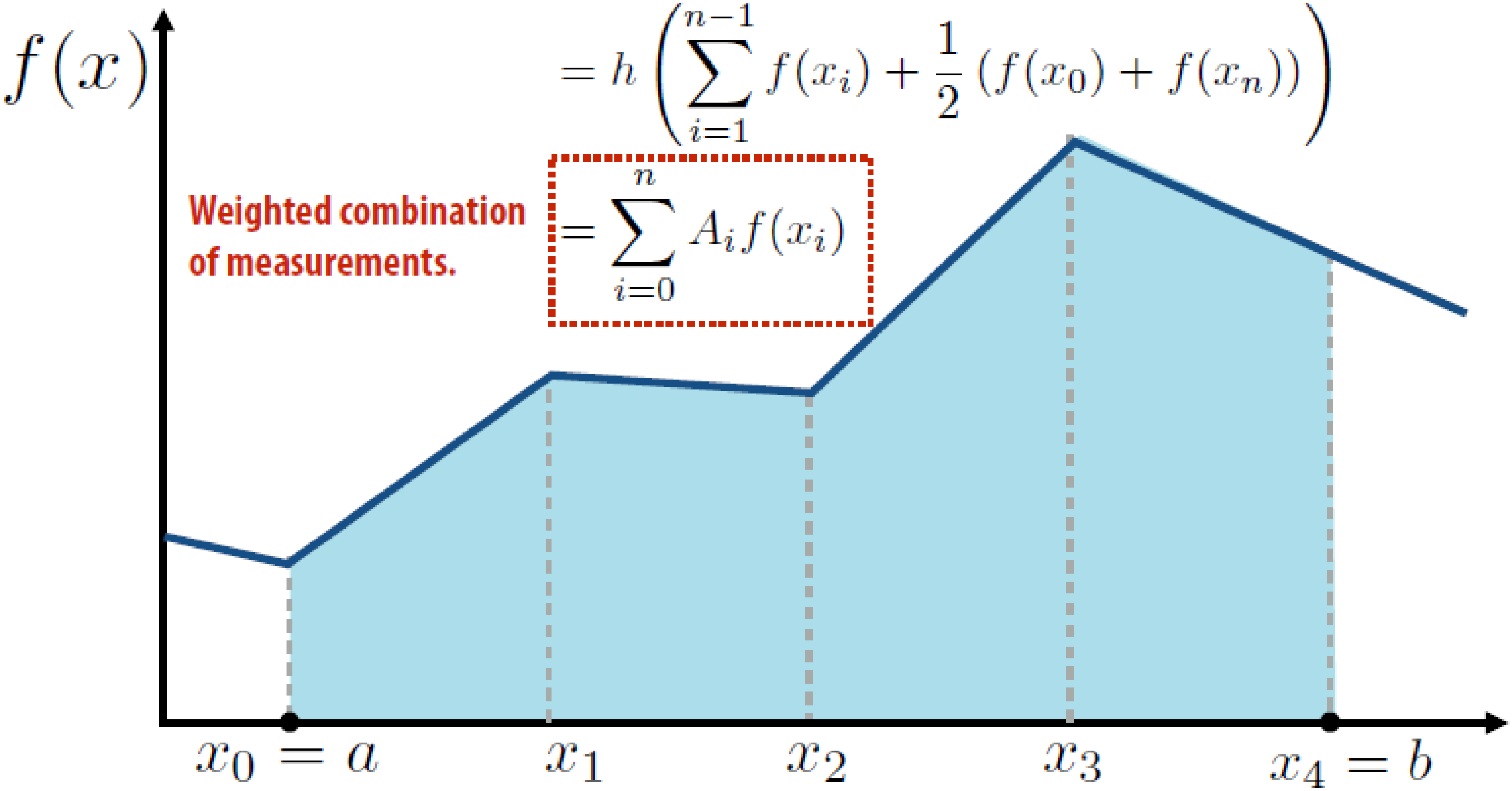
If N-1 segments are of equal length: $h = \dfrac{b-a}{n-1}$

$$\int_a^b f(x)dx = \frac{h}{2}\sum_{i=0}^{n-1}(f(x_i) + f(x_{i+1}))$$

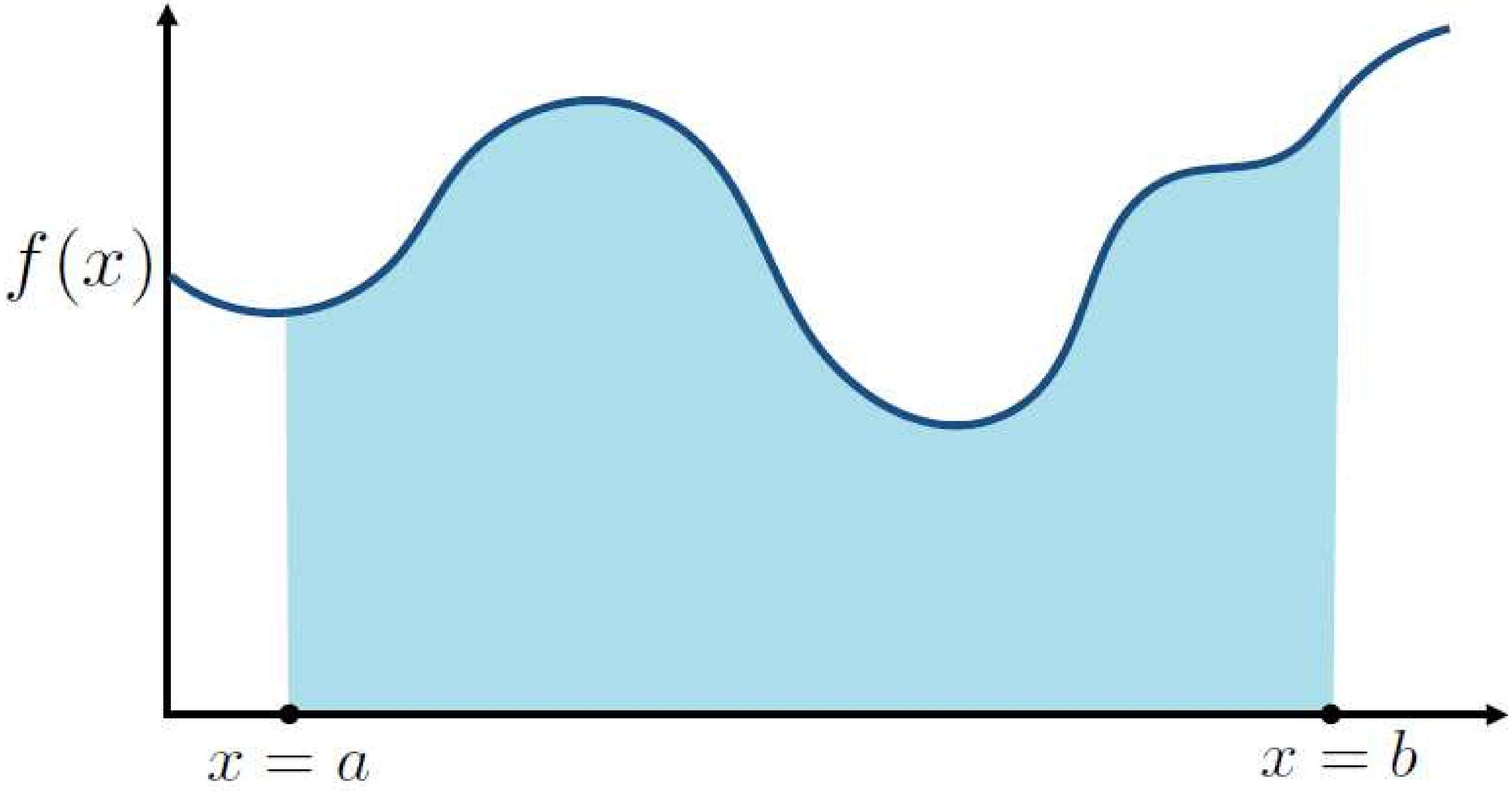
$$= h\left(\sum_{i=1}^{n-1} f(x_i) + \frac{1}{2}(f(x_0) + f(x_n))\right)$$

Weighted combination of measurements.

$$\boxed{= \sum_{i=0}^{n} A_i f(x_i)}$$



Polynomials?



Aside: interpolating polynomials

Consider n+1 measurements of a function $f(x)$

$$f(x_0), f(x_1), f(x_2), \cdots, f(x_n)$$

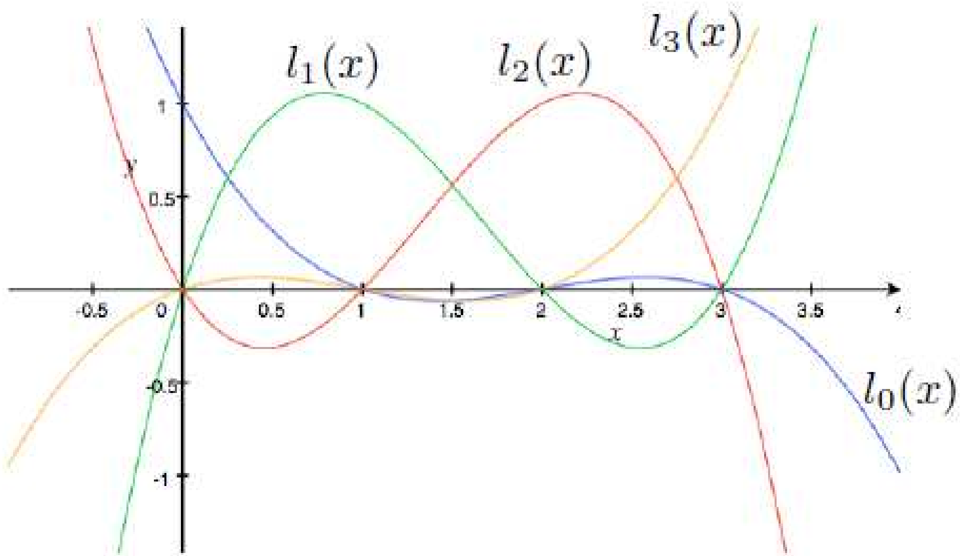
There is a unique degree ≤ n polynomial that interpolates the points:

$$p(x) = \sum_{i=0}^{n} f(x_i) \prod_{j \neq i, j=0}^{n} \left(\frac{x - x_j}{x_i - x_j} \right)$$

$$= \sum_{i=0}^{n} f(x_i) l_i(x)$$

Note: $l_i(x)$ **is 1 at** x_i **and 0 at all other measurement points**

Lagrange polynomial



Gaussian quadrature theorem

If $f(x)$ is a polynomial of degree of up to 2n+1, then its integral over [a,b] is computed exactly by a weighted combination of n+1 measurements in this range.

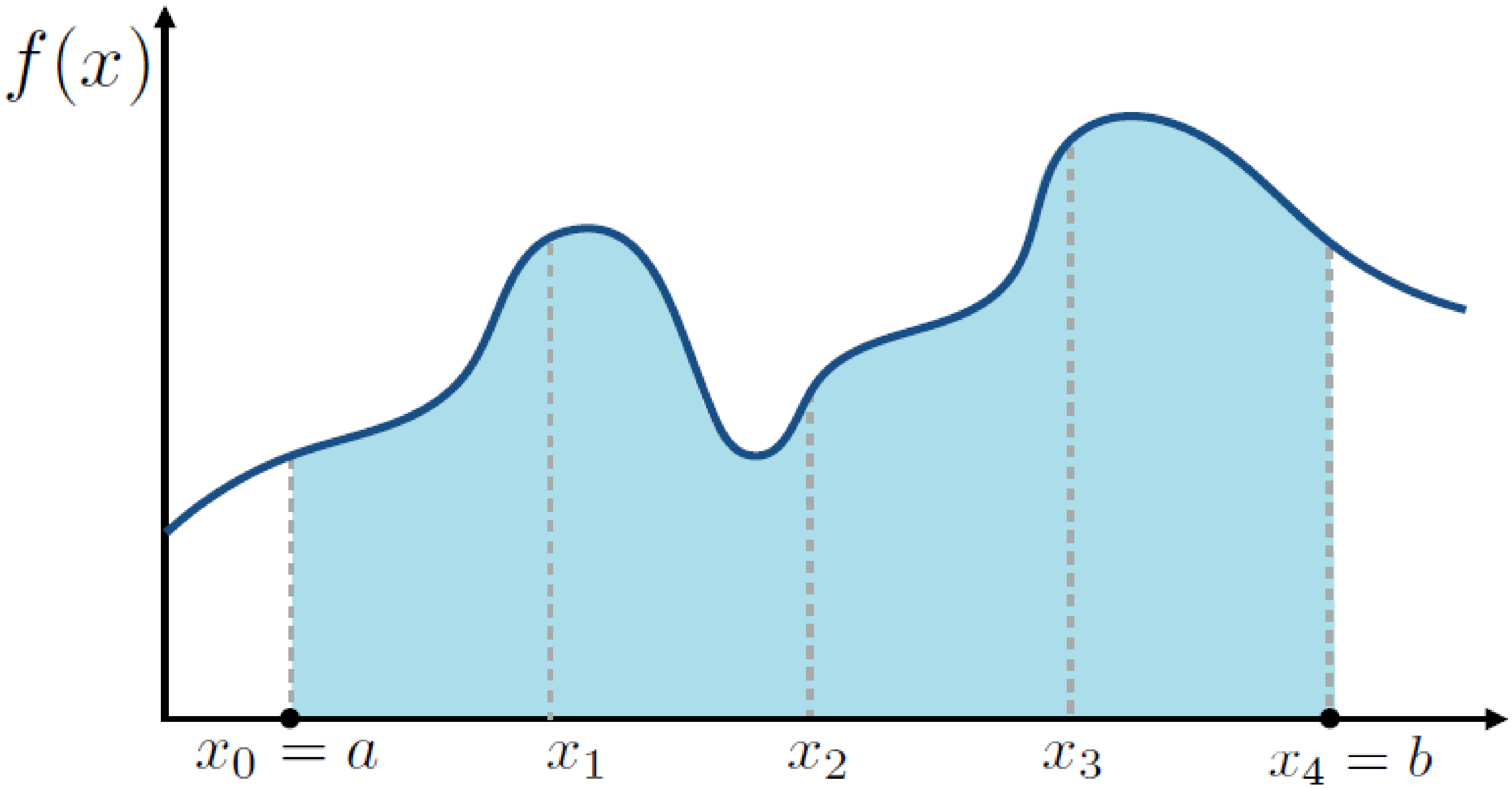
$$\int_a^b f(x)dx = \sum_{i=0}^{n} A_i f(x_i) \qquad A_i = \int_a^b l_i(x)dx$$

Where are these points?

Roots of degree n+1 polynomial $q(x)$ where:

$$\int_a^b x^k q(x)dx = 0 \qquad 0 \leq k \leq n$$

Arbitrary function f(x)?

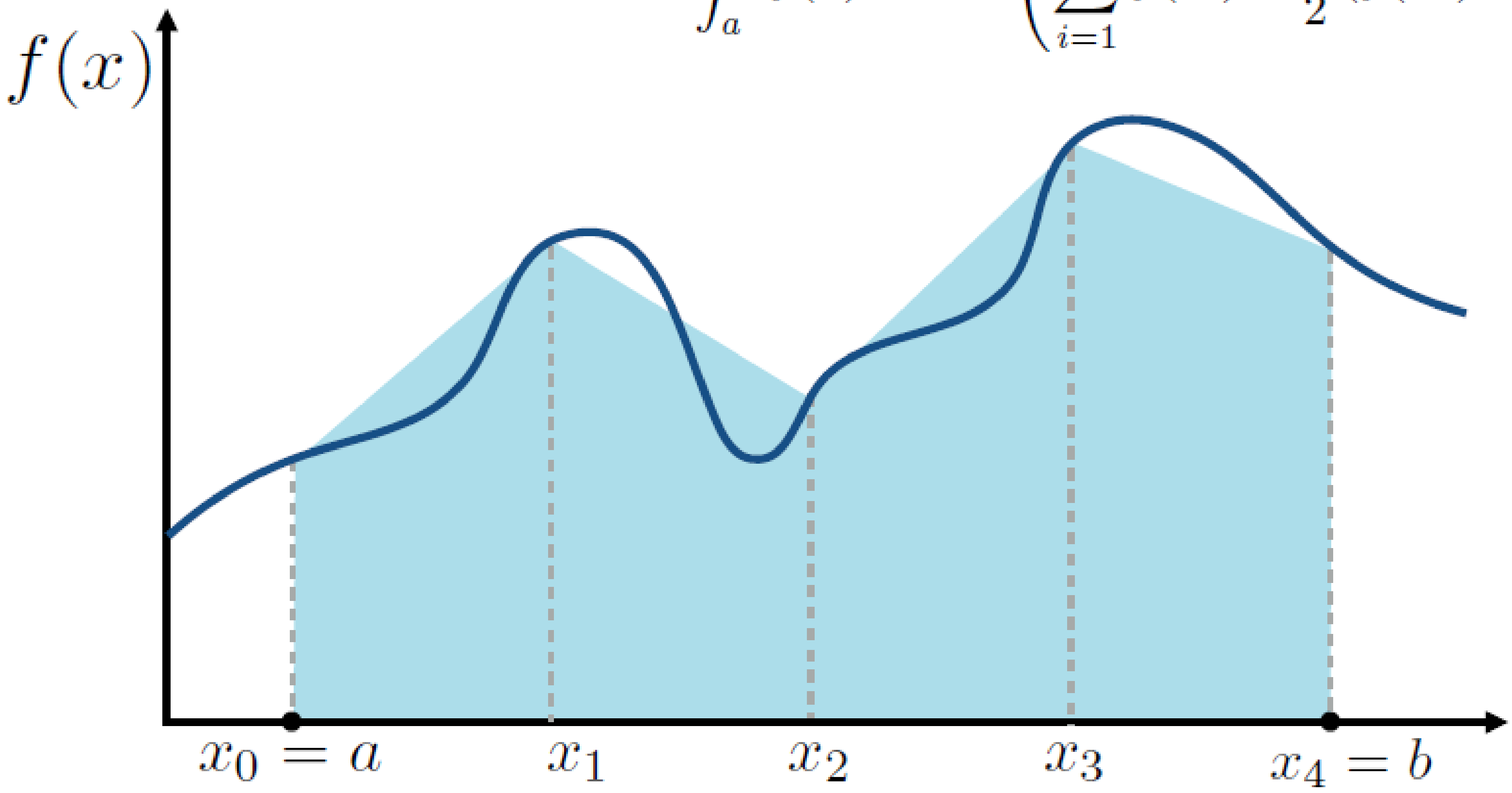


Trapezoidal rule

<u>Approximate</u> integral of f(x) by assuming function is piecewise linear

For equal length segments: $h = \frac{b-a}{n-1}$

$$\int_a^b f(x)dx = h\left(\sum_{i=1}^{n-1} f(x_i) + \frac{1}{2}\left(f(x_0) + f(x_n)\right)\right)$$



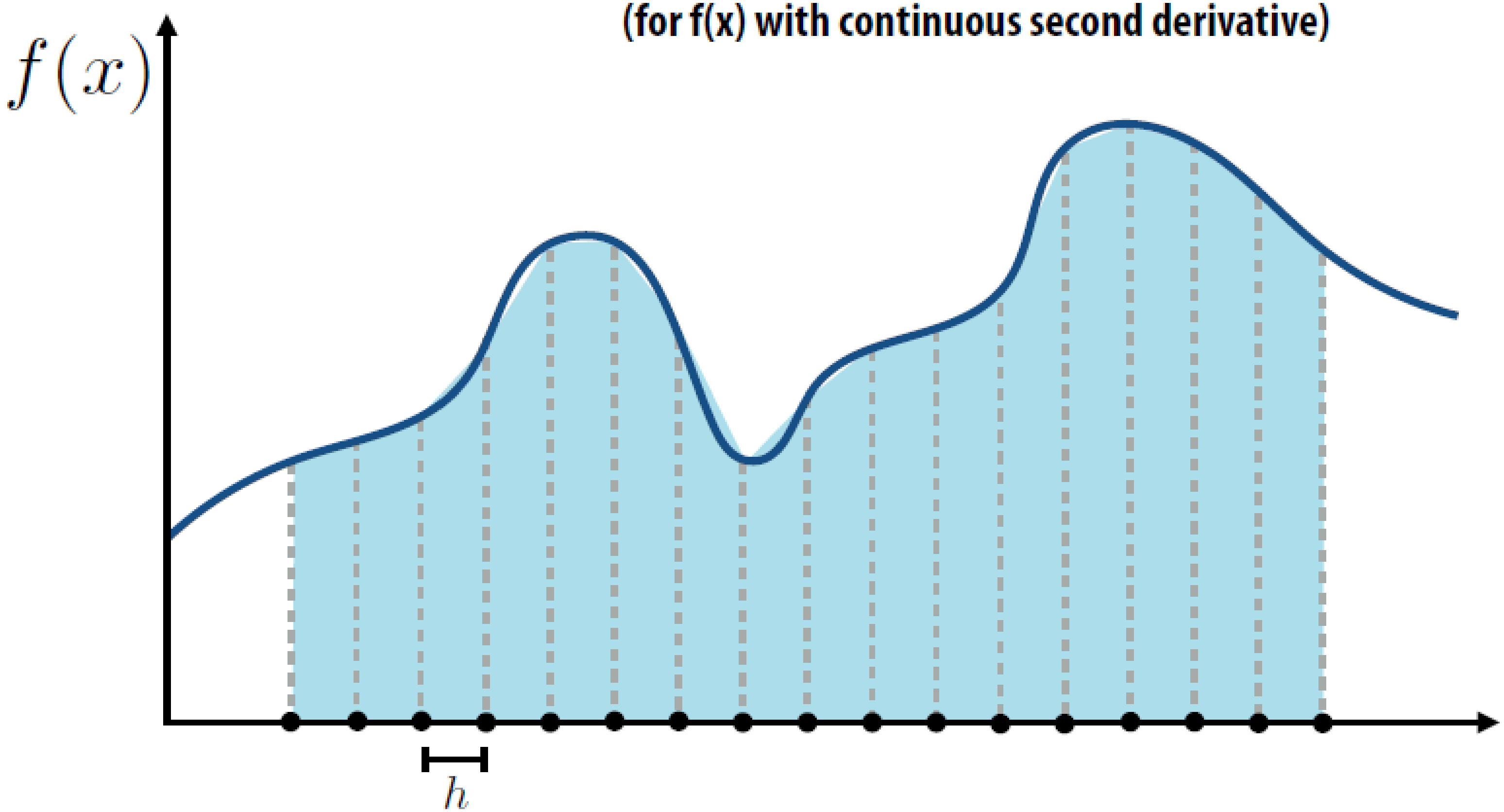
Trapezoidal rule

Consider cost and accuracy of estimate as $n \to \infty$ **(or** $h \to 0$**)**

Work: $O(n)$

Error can be shown to be: $O(h^2) = O(\frac{1}{n^2})$

(for f(x) with continuous second derivative)



Integration in 2D

Consider integrating $f(x,y)$ using the trapezoidal rule (apply rule twice: when integrating in *x* and in *y*)

$$\int_{a_y}^{b_y}\int_{a_x}^{b_x} f(x,y)dxdy = \int_{a_y}^{b_y}\left(\boxed{O(h^2) + \sum_{i=0}^{n} A_i f(x_i,y)}\right)dy$$

First application of rule

$$= O(h^2) + \sum_{i=0}^{n} A_i \int_{a_y}^{b_y} f(x_i,y)dy$$

$$= O(h^2) + \sum_{i=0}^{n} A_i \left(\boxed{O(h^2) + \sum_{j=0}^{n} A_j f(x_i,y_j)}\right)$$

Second application

$$= O(h^2) + \sum_{i=0}^{n}\sum_{j=0}^{n} A_i A_j f(x_i,y_j)$$

Errors add, so error still: $O(h^2)$

But work is now: $O(n^2)$

(*n* x *n* set of measurements)

Must perform much more work in 2D to get same error bound on integral!

In K-D, let $N = n^k$

Error goes as: $O\left(\frac{1}{N^{2/k}}\right)$

Look at the rendering equation again

- **The reflection and scattering equations**

$$L_o(p, \omega_o) = \int_{\mathbb{S}^2} f(p, \omega_o, \omega_i)\, L_i(p, \omega_i)\, |\cos\theta_i|\, d\omega_i$$

$$L_o(p_o, \omega_o) = \int_A \int_{\mathcal{H}^2(\mathbf{n})} S(p_o, \omega_o, p_i, \omega_i)\, L_i(p_i, \omega_i)\, |\cos\theta_i|\, d\omega_i\, dA$$

  - **Very high dimensional**
    - Consider the tracing process: infinite dimensional
    - Conventional numerical integration becomes prohibitive in computation

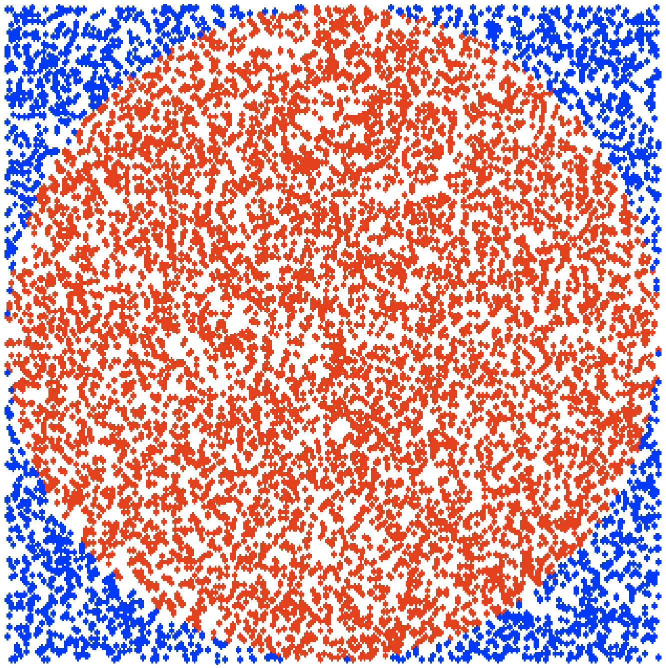
How to do realistic rendering?

- **How to evaluate the integral efficiently?**
  - Rendering equations are usually high dimensional, hard to directly evaluate
  - Sampling? How many samples needed?
  - Convergence?

2. Monte-Carlo integration

Monte-Carlo integration

- **A technique for numerical integration**
  - Using random numbers (probabilistic rather than deterministic)
  - Algorithm gives the correct value of integral “on average”
  - Particularly useful for higher-dimensional integrals
  - Statistically very similar to the true answer



Review of probability

- **Random variable X**
  - A variable whose value is chosen by a random process
  - Applying function *f* to a random variable X results in a new random variable Y = *f*(X)

- **Probability Pr**
  - The measure of the likelihood that an event will occur

- **Cumulative distribution function (CDF)**

$$P(x) = Pr\{X \leq x\}$$

Review of probability

- **Continuous random variables x**
  - A random variable taking values over ranges of continuous domains

- **Probability density function (PDF)**
  - The relative likelihood for the random variable to take on a given value (non-negative)

$$p(x) = \frac{\mathrm{d}P(x)}{\mathrm{d}x}$$

  - For uniform random variables

$$p(x) = \begin{cases} 1 & x \in [0, 1) \\ 0 & \text{otherwise} \end{cases}$$

Review of probability

- **Computing probability from PDF**
  – The probability that a random variable lies inside the interval

$$P(x \in [a, b]) = \int_a^b p(x)\, \mathrm{d}x$$

- **Expected value**
  – Average value of a function over some distribution of values p(x) over its domain

$$E_p[f(x)] = \int_D f(x)\, p(x)\, \mathrm{d}x$$

Review of probability

- **Variance**
  - The expected deviation of the function from its expected value

$$V[f(x)] = E\left[\left(f(x) - E[f(x)]\right)^2\right]$$

  - Properties

$$E[af(x)] = aE[f(x)]$$

$$E\left[\sum_i f(X_i)\right] = \sum_i E[f(X_i)]$$

$$V[af(x)] = a^2 V[f(x)]$$

$$V[f(x)] = E\left[(f(x))^2\right] - E[f(x)]^2$$

$$\sum_i V[f(X_i)] = V\left[\sum_i f(X_i)\right]$$

Review of probability

- **Joint distribution function**
  - Give the probability that each of *X*, *Y*, … falls in any particular range of values specified for that variable
- **Marginal density function**
  - The probabilities of various values of the variables in the subset without reference to the values of the other variables

$$p(x) = \int p(x, y)\, \mathrm{d}y$$

- **Conditional probability**
  - A measure of the probability of an event given that another event has occurred

$$p(y|x) = \frac{p(x, y)}{p(x)}$$

Monte-Carlo estimator

- **Approximate the value of an arbitrary integral**
  - The foundation of the light transport algorithms

- **1D evaluation**
  - A one-dimensional integral $\int_a^b f(x)\,\mathrm{d}x$
  - Given a supply of uniform random variables $X_i \in [a, b]$, the expected value of the integral estimator

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

Monte-Carlo estimator

- **Expected value**
  - Equal to the integral

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

Monte-Carlo estimator

- **More general**
  - An arbitrary non-zero distribution function $f(x)$
  - Random variables X_i drawn from arbitrary PDF $p(x)$

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

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

Monte-Carlo estimator

- **Multi-dimensional function estimation**
  - N samples X_i are taken from a multidimensional PDF
  - The estimator is applied as in 1D

- **Consider 3D integral**

$$\int_{x_0}^{x_1} \int_{y_0}^{y_1} \int_{z_0}^{z_1} f(x, y, z)\, \mathrm{d}x\, \mathrm{d}y\, \mathrm{d}z$$

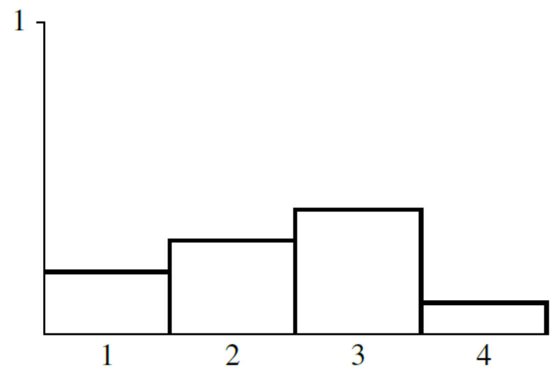
  - Assuming separable joint distribution
    - The estimator

$$\frac{(x_1 - x_0)(y_1 - y_0)(z_1 - z_0)}{N} \sum_i f(X_i)$$

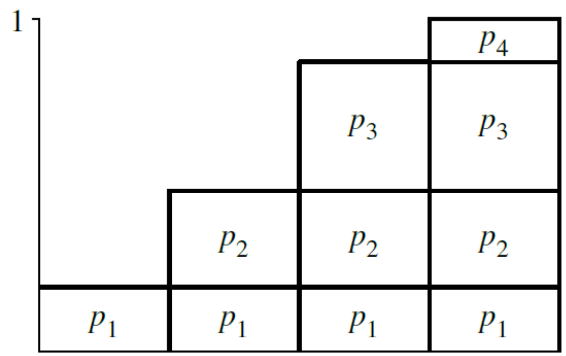
3. Sampling of random variables

Basic sampling of random variables

- **Inversion method**
  - Discrete case
    - Probability function

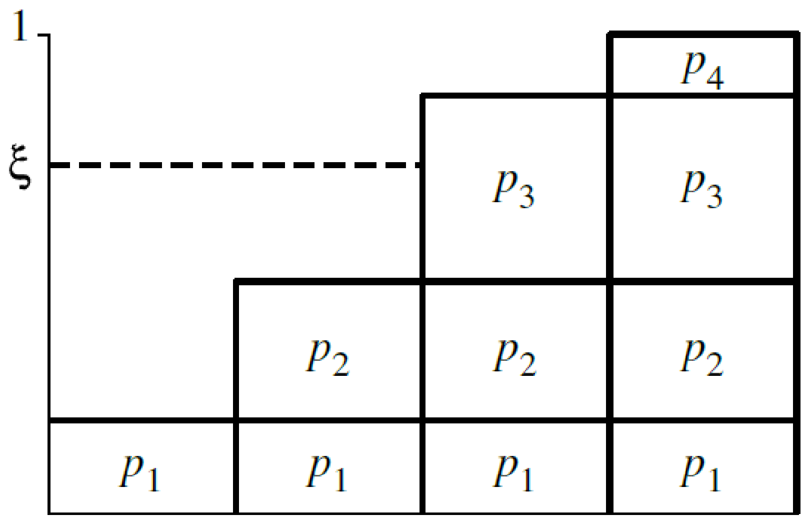


    - Cumulative distribution function



Basic sampling of random variables

- **Inversion method**
  - A canonical uniform random variable on vertical axis



  - The inverse based on ξ value conforms to the desired distribution

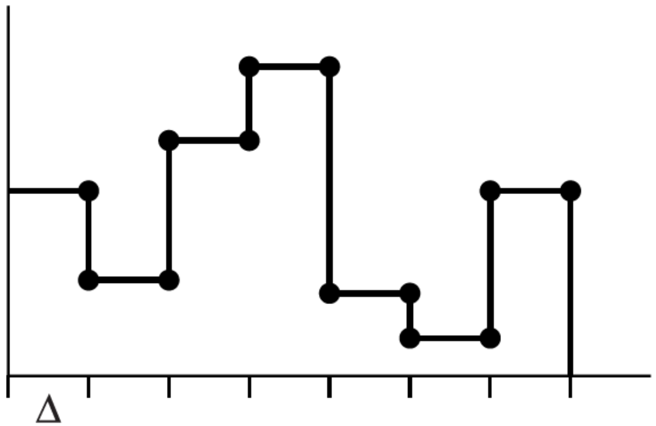
Basic sampling of random variables

- **Inversion method**
  - Application to continuous random variables
    - 1. Compute the CDF $P(x) = \int_0^x p(x')\,\mathrm{d}x'$
    - 2. Compute the inverse $P^{-1}(x)$
    - 3. Obtain a uniformly distributed random number ξ
    - 4. Compute $X_i = P^{-1}(\xi)$

Basic sampling of random variables

- **Inversion method**
  - Piecewise-constant 1D functions over [0,1]

$$f(x) = \begin{cases} v_0 & x_0 \le x < x_1 \\ v_1 & x_1 \le x < x_2 \\ \vdots & \end{cases}$$



  - The integral $\int f(x)\,\mathrm{d}x$

$$c = \int_0^1 f(x)\,\mathrm{d}x = \sum_{i=0}^{N-1} \Delta v_i = \sum_{i=0}^{N-1} \frac{v_i}{N} \quad \Rightarrow \quad p(x) = f(x)/c$$

Basic sampling of random variables

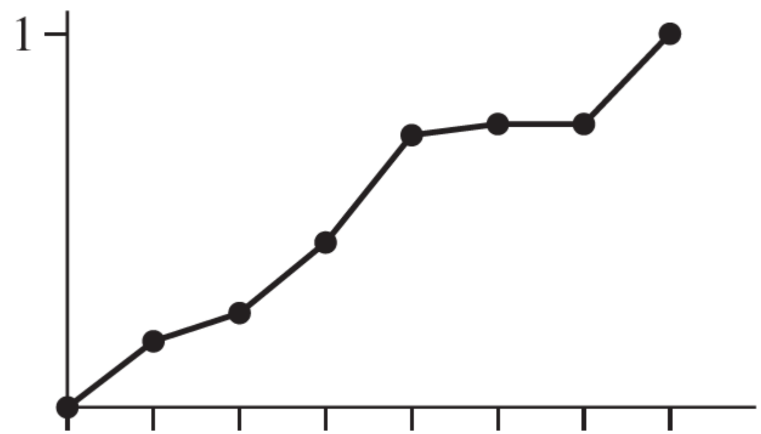
- **Inversion method**
  - Computing cumulative distribution function

$$P(x_0) = 0$$

$$P(x_1) = \int_{x_0}^{x_1} p(x)\,\mathrm{d}x = \frac{v_0}{Nc} = P(x_0) + \frac{v_0}{Nc}$$

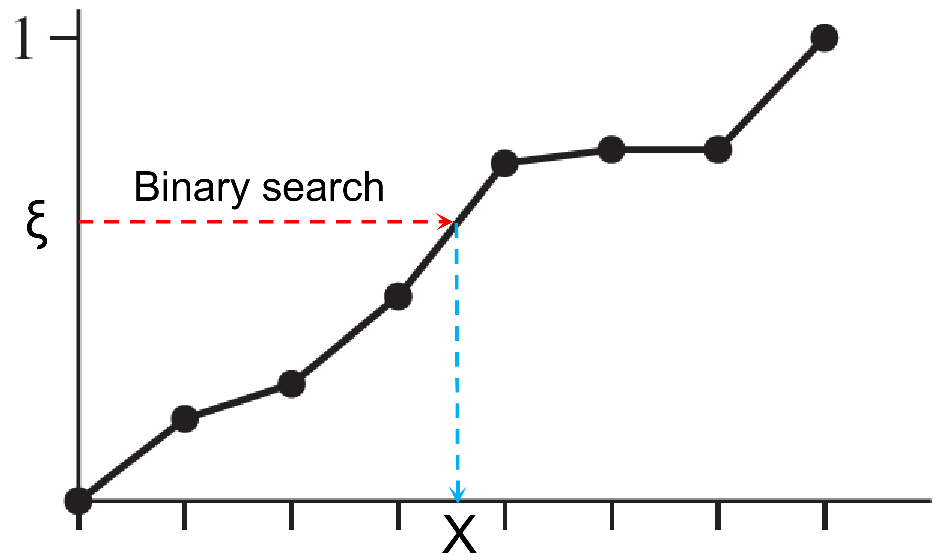
$$P(x_2) = \int_{x_0}^{x_2} p(x)\,\mathrm{d}x = \int_{x_0}^{x_1} p(x)\,\mathrm{d}x + \int_{x_1}^{x_2} p(x)\,\mathrm{d}x = P(x_1) + \frac{v_1}{Nc}$$

$$P(x_i) = P(x_{i-1}) + \frac{v_{i-1}}{Nc}$$



Basic sampling of random variables

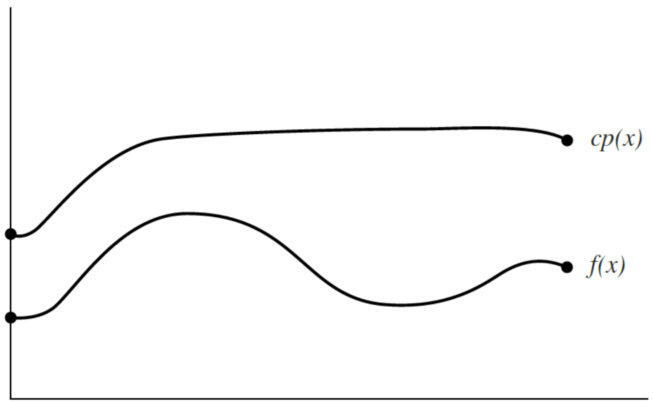
- **Inversion method**
  - Compute the inverse



Rejection method

- **Problem with inversion method**
  - Sometimes difficult to compute the CDF integral
  - Sometimes unable to obtain function inverse
- **Rejection method**
  - A dart-throwing approach
  - Find a PDF p(x) from which we know how to sample
  - $p(x)$ must satisfy

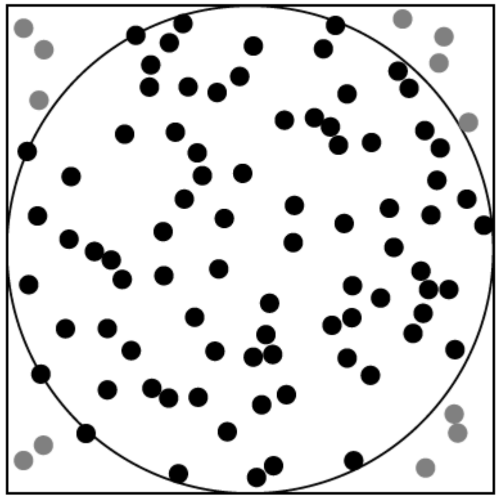
$$f(x) < c\ p(x)$$



Rejection method

- **Random sample generation**
  - Start loop
    - Sample x from p's distribution
    - Choose a random variable ξ
    - If $\xi < f(x)/(cp(x))$ then

      Return x



- Efficiency
  - Depends on how close cp(x) bounds f(x)

- Rejection method isn't used in Monte-Carlo method for rendering

Metropolis sampling

- **A sampling technique with remarkable property**
  - Generate samples from any non-negative function f
  - Distributed proportional to f's value
  - Only require the ability to evaluate f
  - Can efficiently generate samples from a wider variety of functions

Metropolis sampling

- **Basic algorithm**
  - Generate a set of samples X_i from a function f defined over an arbitrary dimensional space Ω
    - Select the first sample X_0
    - Each sample Xi is generated using a random mutation to Xi-1 to compute a proposed sample X'
    - In order to compute X', we must compute a tentative transition function T(X → X') : the transition probability
    - Compute the acceptance probability a(X → X')

$$a(X \to X') = \min\left(1, \frac{f(X')\,T(X' \to X)}{f(X)\,T(X \to X')}\right) \quad \Longrightarrow \quad a(X \to X') = \min\left(1, \frac{f(X')}{f(X)}\right)$$

Symmetric proposal

Metropolis sampling

- **Basic sampling pseudo-code**

```
X = X0
for i = 1 to n
    X' = mutate(X)
    a = accept(X, X')
    if (random() < a)
        X = X'
    record(X)
```

  – The recorded *X* sequence will be used for integration

Metropolis sampling

- **Choosing mutation strategies**
  - More freedom
  - Subject to being able to compute the tentative transition density $T(X \to X')$

Several choices

- Local perturbation

$$x_i' = x_i \pm s\,\xi \qquad\qquad x_i' = x_i \pm b\,\mathrm{e}^{-\log(b/a)\,\xi}$$

- Global uniform random

$$x_i = \xi$$

- Match some part of the function being sampled

$$T(X \to X') = p(X')$$

Estimating integrals with Metropolis sampling

- **We can apply Metropolis algorithm**
  - Evaluate integrals

$$\int f(x)g(x)\,\mathrm{d}\Omega$$

  - Standard Monte-Carlo estimator

$$\int_\Omega f(x)g(x)\,\mathrm{d}\Omega \approx \frac{1}{N}\sum_{i=1}^{N}\frac{f(X_i)g(X_i)}{p(X_i)}$$

  - Apply Metropolis sampling to generate samples from a density function that is proportional to f(x)

$$\int_\Omega f(x)g(x)\,\mathrm{d}\Omega \approx \left[\frac{1}{N}\sum_{i=1}^{N} g(X_i)\right]\cdot\int_\Omega f(x)\,\mathrm{d}\Omega$$

Transforming between distributions

- **Function of a random variable**
  - Suppose we are given random variable X_i with PDF $p_x(x)$
  - Given $Y_i=y(X_i)$, the following equality satisfies

$$Pr\{Y \leq y(x)\} = Pr\{X \leq x\} \quad \Rightarrow \quad P_y(y) = P_y(y(x)) = P_x(x)$$

  - Differentiating

$$p_y(y)\frac{\mathrm{d}y}{\mathrm{d}x} = p_x(x) \quad \Rightarrow \quad p_y(y) = \left(\frac{\mathrm{d}y}{\mathrm{d}x}\right)^{-1} p_x(x)$$

  - Usually we know $p_y(y)$ (and P(y)), how to sample y?

$$y(x) = P_y^{-1}\left(P_x(x)\right)$$

Transforming between distributions

- **Transformation in multiple dimensions**
  - Let n-dimensional random variable X with density function $p_x(x)$
  - Let Y=T(X), T is a bijective mapping

$$p_y(y) = p_y(T(x)) = \frac{p_x(x)}{|J_T(x)|}$$

- Jacobian:

$$\begin{pmatrix} \partial T_1/\partial x_1 & \cdots & \partial T_1/\partial x_n \\ \vdots & \ddots & \vdots \\ \partial T_n/\partial x_1 & \cdots & \partial T_n/\partial x_n \end{pmatrix}$$

Transforming between distributions

- **Example**
  - Polar coordinates

$$x = r \cos\theta$$
$$y = r \sin\theta$$

- Suppose we draw samples from some density p(r,θ)
- Computing the Jacobian

$$J_T = \begin{pmatrix} \frac{\partial x}{\partial r} & \frac{\partial x}{\partial \theta} \\ \frac{\partial y}{\partial r} & \frac{\partial y}{\partial \theta} \end{pmatrix} = \begin{pmatrix} \cos\theta & -r\sin\theta \\ \sin\theta & r\cos\theta \end{pmatrix}$$

- The determinant: $r\left(\cos^2\theta + \sin^2\theta\right) = r$
- So

$$p(x, y) = p(r, \theta)/r \quad \Longrightarrow \quad p(r, \theta) = r\, p(x, y)$$

Transforming between distributions

- **Example**
  - Spherical coordinates

$$x = r \sin\theta \cos\phi$$
$$y = r \sin\theta \sin\phi$$
$$z = r \cos\theta$$

- Computing the Jacobian determinant: $|J_T| = r^2 \sin\theta$
- The corresponding density function

$$p(r, \theta, \phi) = r^2 \sin\theta \; p(x, y, z)$$

- Solid angle defined with spherical coordinates $d\omega = \sin\theta \, \mathrm{d}\theta \, \mathrm{d}\phi$
- If we have a density function defined over a solid angle

$$p(\theta, \phi) \, \mathrm{d}\theta \, \mathrm{d}\phi = p(\omega) \, \mathrm{d}\omega \Rightarrow p(\theta, \phi) = \sin\theta \; p(\omega)$$

2D sampling with multi-dimensional transformations

- **2D joint density function p(x,y)**
  - We wish to draw samples (X,Y) from
- **Sometimes separable**

$$p(x, y) = p_x(x)\, p_y(y)$$

  - Random variable (X,Y) can be found independently
  - Many useful densities aren't separable
- **Basic idea**
  - Compute the marginal density to isolate one particular variable, and draw sample with 1D technique
  - Compute the conditional probability and draw a sample from that distribution

2D sampling with multi-dimensional transformations

- **Example**
  - Sampling a unit disk uniformly
    - Wrong approach: $r = \xi_1, \theta = 2\pi\xi_2$
    - PDF p(x,y) by normalization is: $p(x, y) = 1/\pi$
    - Transform into polar coordinate: $p(r, \theta) = r/\pi$ $p(r, \theta) = r\ p(x, y)$
    - Compute the marginal and conditional densities

$$p(r) = \int_0^{2\pi} p(r, \theta)\, \mathrm{d}\theta = 2r$$

$$p(\theta | r) = \frac{p(r, \theta)}{p(r)} = \frac{1}{2\pi}$$

- Integrating and inverting to find $P(r)$, $P^{-1}(r)$, $P(\theta)$, and $P^{-1}(\theta)$

$$r = \sqrt{\xi_1}$$
$$\theta = 2\pi\xi_2$$

3D sampling with multi-dimensional transformations

- **Example**
  - Uniformly sampling a hemisphere
    - Uniform sampling means $p(\omega) = c$
    - Normalization:

$$p(\theta, \phi) = \sin\theta \; p(\omega)$$

$$\int_{\mathcal{H}^2} p(\omega)\, \mathrm{d}\omega = 1 \Rightarrow c \int_{\mathcal{H}^2} \mathrm{d}\omega = 1 \Rightarrow c = \frac{1}{2\pi} \longrightarrow p(\omega) = 1/(2\pi) \longrightarrow p(\theta, \phi) = \sin\theta/(2\pi)$$

    - Consider sampling θ:

$$p(\theta) = \int_0^{2\pi} p(\theta, \phi)\, \mathrm{d}\phi = \int_0^{2\pi} \frac{\sin\theta}{2\pi}\, \mathrm{d}\phi = \sin\theta$$

    - Compute the conditional density for φ:

$$p(\phi|\theta) = \frac{p(\theta, \phi)}{p(\theta)} = \frac{1}{2\pi}$$

3D sampling with multi-dimensional transformations

- **Example**
  - Uniformly sampling a hemisphere
    - Use 1D inversion technique to sample:

$$P(\theta) = \int_0^\theta \sin\theta' \, \mathrm{d}\theta' = 1 - \cos\theta$$

$$P(\phi|\theta) = \int_0^\phi \frac{1}{2\pi} \, \mathrm{d}\phi' = \frac{\phi}{2\pi}$$

    - Inversion is straightforward

$$\theta = \cos^{-1}\xi_1$$
$$\phi = 2\pi\xi_2.$$

$$\Rightarrow$$

$$x = \sin\theta\cos\phi = \cos(2\pi\xi_2)\sqrt{1-\xi_1^2}$$
$$y = \sin\theta\sin\phi = \sin(2\pi\xi_2)\sqrt{1-\xi_1^2}$$
$$z = \cos\theta = \xi_1$$

3D sampling with multi-dimensional transformations

- **Cosine-weighted hemisphere sampling**
  - It is useful to have a cosine distribution over the hemisphere (the incident cosine term)
  - We require: $p(\omega) \propto \cos\theta$
  - Derive as before:

$$\int_{\mathcal{H}^2} c\ p(\omega)\ \mathrm{d}\omega = 1 \qquad d\omega = \sin\theta\ \mathrm{d}\theta\ \mathrm{d}\phi \qquad p(\theta,\phi) = \sin\theta\ p(\omega)$$

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

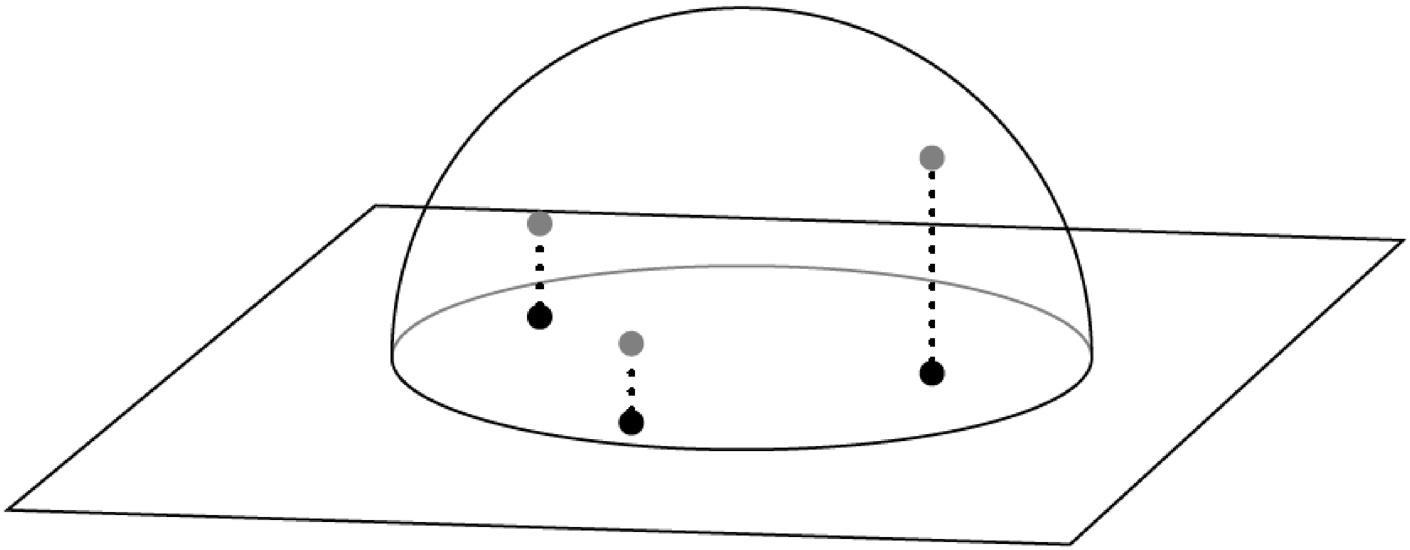
$$c\ 2\pi\int_0^{\pi/2}\cos\theta\sin\theta\ \mathrm{d}\theta = 1$$

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

$$\Rightarrow \boxed{p(\theta,\phi) = \frac{1}{\pi}\cos\theta\ \sin\theta}$$

3D sampling with multi-dimensional transformations

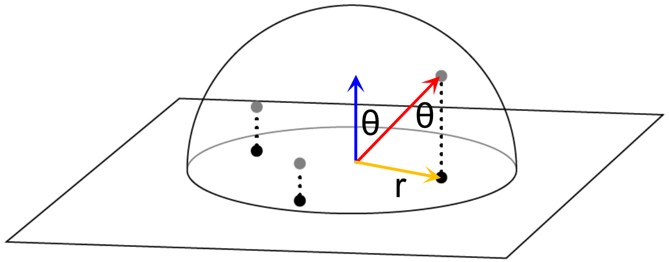
- **Cosine-weighted hemisphere sampling**
  - Malley's method
    - Sampling a unit disk and project onto the sphere



3D sampling with multi-dimensional transformations

- **Why Malley's method works?**
  - Let (r,ϕ) be polar coordinates on disk, we know

$$p(r,\phi) = r/\pi$$



  - Vertical projection gives: $\sin\theta = r$
  - To complete the (r , ϕ)⟶(sin θ , ϕ) transformation, we need the determinant of the Jacobian

$$|J_T| = \begin{vmatrix} \cos\theta & 0 \\ 0 & 1 \end{vmatrix} = \cos\theta$$

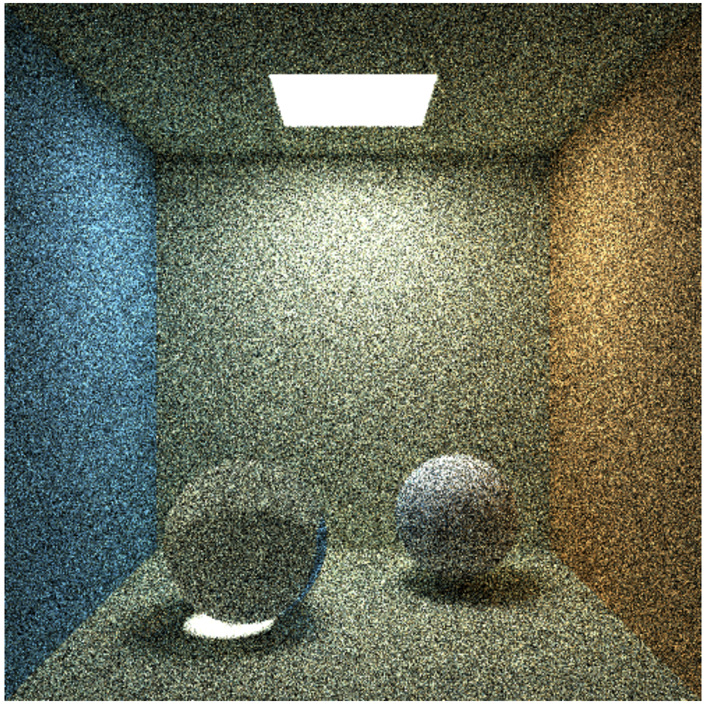
  - Therefore:

$$p(\theta,\phi) = |J_T|\,p(r,\phi) = \cos\theta\, r/\pi = \boxed{(\cos\theta\,\sin\theta)/\pi}$$

4. Sampling efficiency

Estimating efficiency

- **Variance in Monte-Carlo ray tracing**
  - Manifest as image noise



Different sampling rate

- **Efficiency of an estimator**

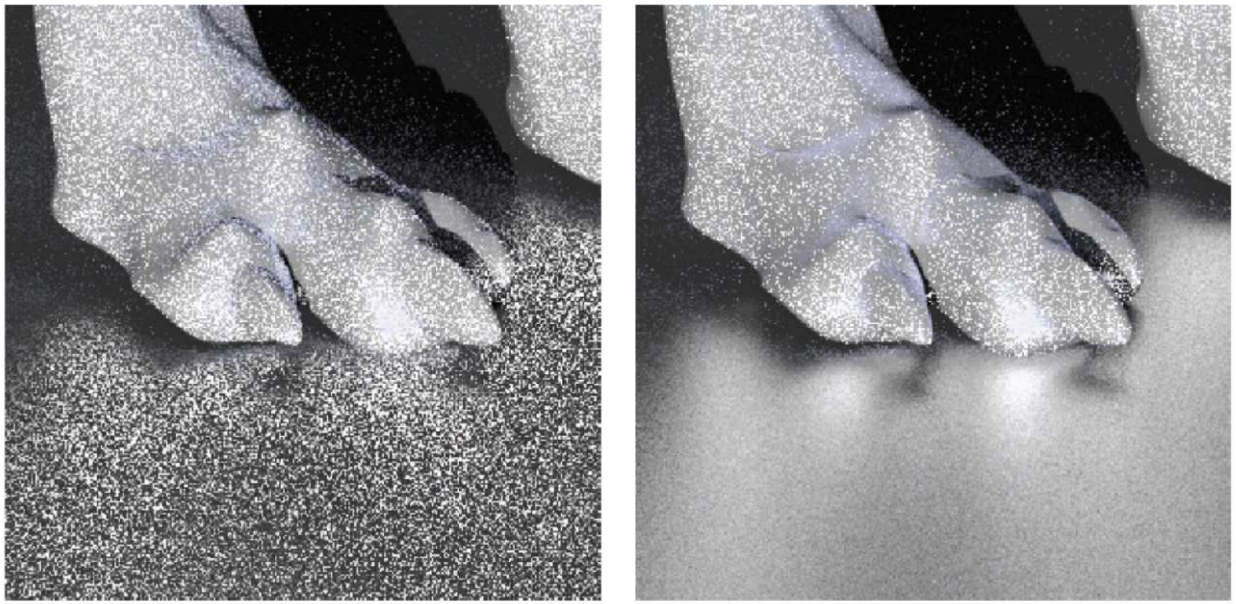
$$\epsilon[F] = \frac{1}{V[F]T[F]}$$

  - V[F]: sampling variance
  - T[F]: running time to compute

- **Improve efficiency**
  - Importance sampling

Stratified sampling

- **Stratified sampling**
  - Subdivide the integration domain into n non-overlapping regions Λ_i
  - We draw n_i samples from each region Λ_i according to density p_i inside each region
  - Suffer from "curse of dimensionality"



Quasi Monte-Carlo sampling

- **Low-discrepancy sampling**
  - Poisson disk / best-candidate sampling
  - Foundation of a branch of Monte-Carlo sampling
- **Advantage**
  - Quasi Monte-Carlo converges asymptotically faster
  - Generally better for smooth integrand
- **Disadvantage**
  - Asymptotic convergence rate is not applicable to discontinuous integrand

Sampling bias

- **Another approach of variance reduction**
  - Introduce bias into the computation
  - Sacrifice for larger error in expected value for variance reduction
- **Unbiased estimator**
  - The expected value is equal to the correct answer
- **Bias estimation**

$$\beta = E[F] - \int f(x)\,\mathrm{d}x$$

Sampling bias

- **Why bias is sometimes desirable?**
  - Consider computing estimation of the mean value of a distribution $X_i \sim p$ over [0,1]
  - Two estimators
    - 1. $$\frac{1}{N}\sum_{i=1}^{N} X_i$$
    - 2. $$\frac{1}{2}\max(X_1, X_2, \ldots, X_N)$$
  - The first estimator has variance $O(N^{-1})$

Sampling bias

- **Why bias is sometimes desirable?**
  - The second estimator's expected value

$$0.5\frac{N}{N+1} \neq 0.5$$

  - It is biased, but it is variance is $O(N^{-2})$
  - For large value of N, the second estimator is preferred

Sampling bias

- **Consider again image reconstruction**
  - Consider a Monte-Carlo estimate of

$$I(x, y) = \iint f(x - x', y - y') L(x', y') \, \mathrm{d}x' \, \mathrm{d}y'$$

  - Assume we take image samples uniformly: constant probability density (unbiased, larger variance):

$$I(x, y) \approx \frac{1}{N p_c} \sum_{i=1}^{N} f(x - x_i, y - y_i) L(x_i, y_i)$$

  - The practical realization (biased, less variance):

$$I(x, y) = \frac{\sum_i f(x - x_i, y - y_i) L(x_i, y_i)}{\sum_i f(x - x_i, y - y_i)}$$

5. Importance sampling

Importance sampling

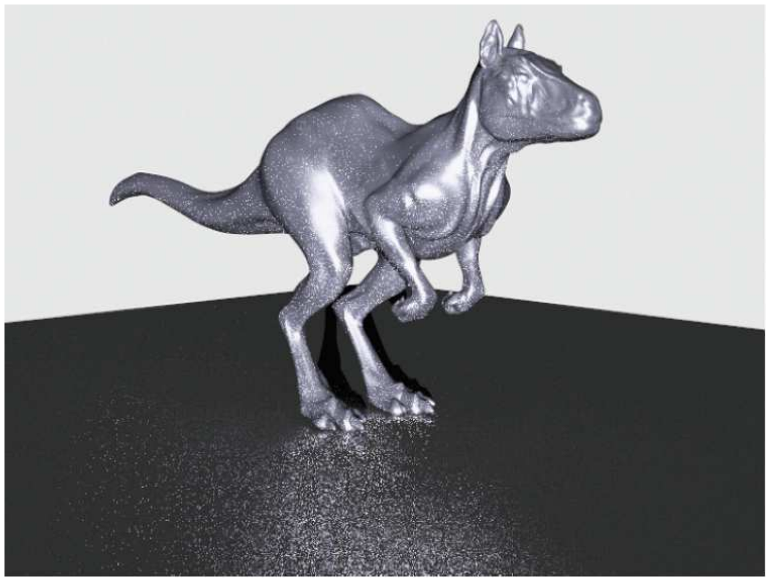
- **Importance sampling is a variance reduction technique**
  - Monte-Carlo estimator

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

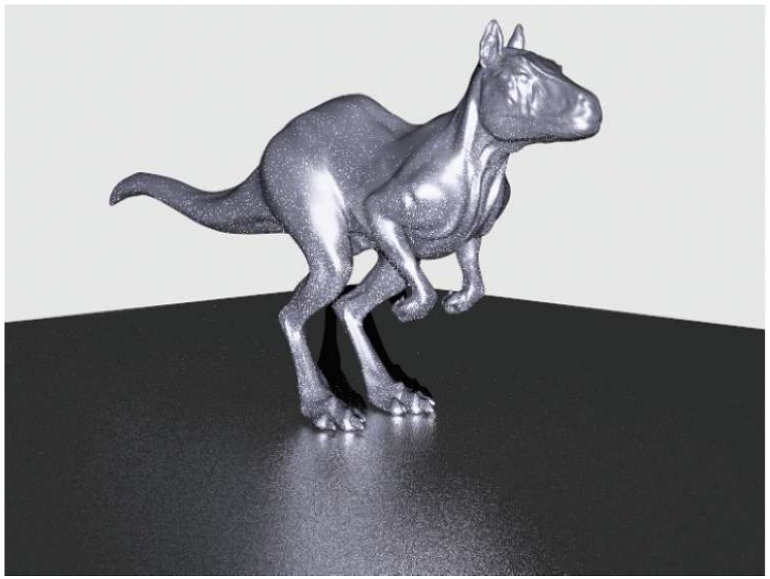
  - The fact
    - If samples are taken from distribution p(x) that is similar to function f(x), the convergence will be much faster
    - Can increase variance if p(x) is bad
  - Basic idea
    - Concentrate work where the values of the integrand is relatively high

Importance sampling

- **The practical case**
  - The integrand is the product of more than one function
  - Finding p(x) similar to one of the multiplicands can be helpful
  - Especially important in rendering



Stratified uniform sampling



Importance sampling based on BRDF

Multiple importance sampling

- **We are frequently faced with integrals with two or more function**

$$\int f(x)g(x)\,\mathrm{d}x$$

  – Importance sampling strategy for both f(x) and g(x), which to choose?
  – Assume we are not able to combine two sampling to compute a PDF proportional to f(x)g(x)

  – A bad choice of sampling distribution can be much worse than uniform distribution

Multiple importance sampling

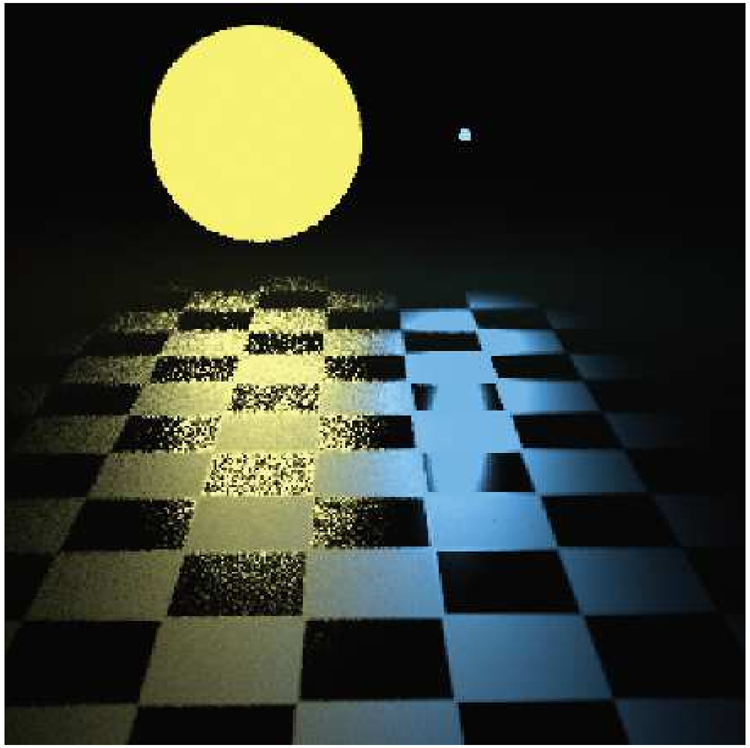
- **Consider direct lighting integral evaluation**

$$L_o(p, \omega_o) = \int_{\mathcal{S}^2} f(p, \omega_o, \omega_i)\, L_d(p, \omega_i)\, |\cos\theta_i|\, d\omega_i$$

  – We can perform importance sampling based on either L_d or f, one of these will often perform poorly

Multiple importance sampling

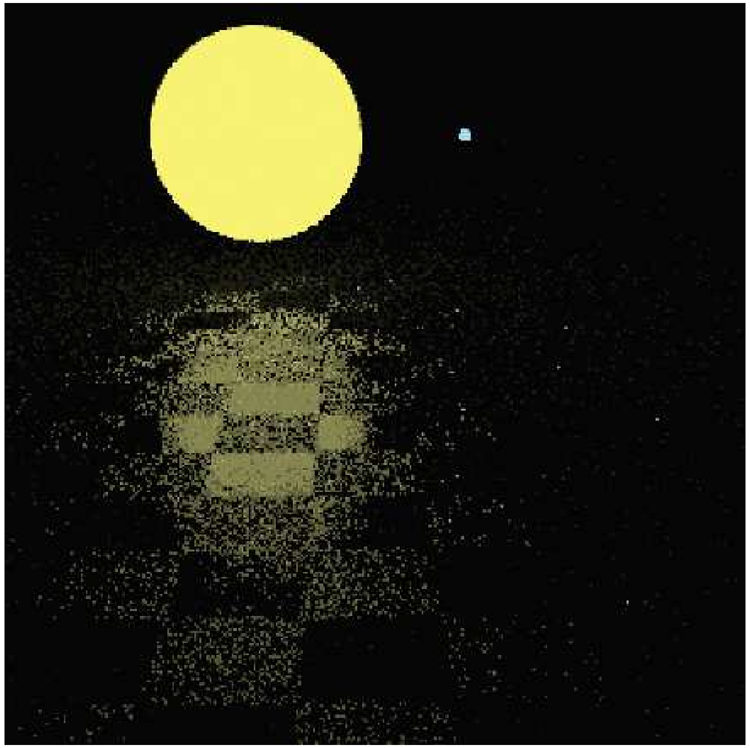
- **Consider a near-mirror BRDF**
  - The value of integrand will be close to 0 for angles off the reflection angle
  - Sampling L_d will lead to large variance



Sampling from light distribution

Multiple importance sampling

- **Consider a near-mirror BRDF**
  - Sampling BRDF could be much better
  - However, for diffuse and glossy BRDFs, sampling form BRDF will lead to similar problem



Sampling from BRDF

Multiple importance sampling

- **How to solve?**
  - Try to match either of them
  - Weighting scheme
    - If two sampling distributions p_f and p_g are used to estimate the value of

$$\int f(x)g(x)\,\mathrm{d}x$$

    - The new Monte-Carlo estimator is given by

$$\frac{1}{n_f}\sum_{i=1}^{n_f}\frac{f(X_i)g(X_i)w_f(X_i)}{p_f(X_i)}+\frac{1}{n_g}\sum_{j=1}^{n_g}\frac{f(Y_j)g(Y_j)w_g(Y_j)}{p_g(Y_j)}$$

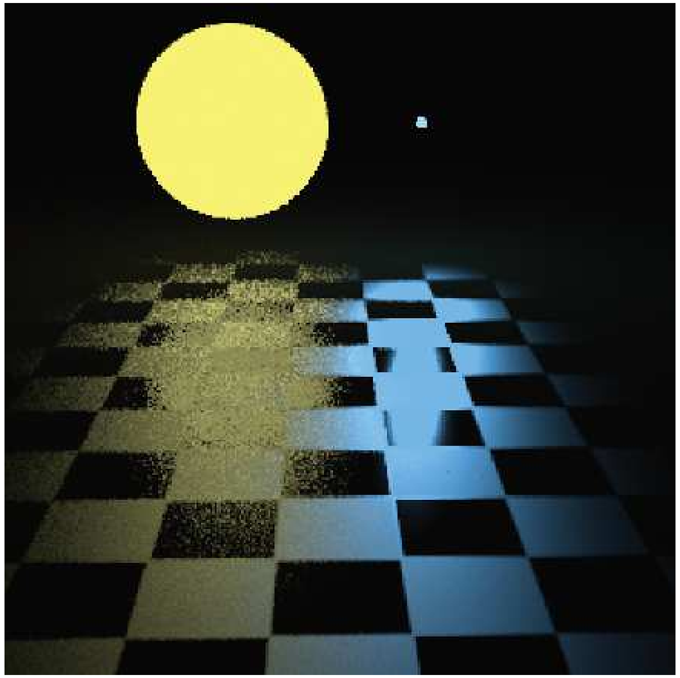
    - n_f: number of samples taken from p_f distribution
    - n_g: number of samples taken from p_g distribution

Multiple Importance Sampling

- **Weighting function**
  - Balance heuristic

$$w_s(x) = \frac{n_s p_s(x)}{\sum_i n_i p_i(x)}$$

    - Effectively proven to reduce variance



Next lecture: Sampling and reconstruction