Monte Carlo methods

© 1996-2017 Josef Pelikán

CGG MFF UK Praha

pepca@cgg.mff.cuni.cz

http://cgg.mff.cuni.cz/~pepca/

Monte Carlo quadrature

Integral to estimate:

$$I = \int_0^1 f(x)\,dx$$

Precondition: $f(x) \in L^2(0,1)$

If ξ is a random number from **R(0,1)** distribution, than **f(ξ)** is **primary estimate** of the integral:

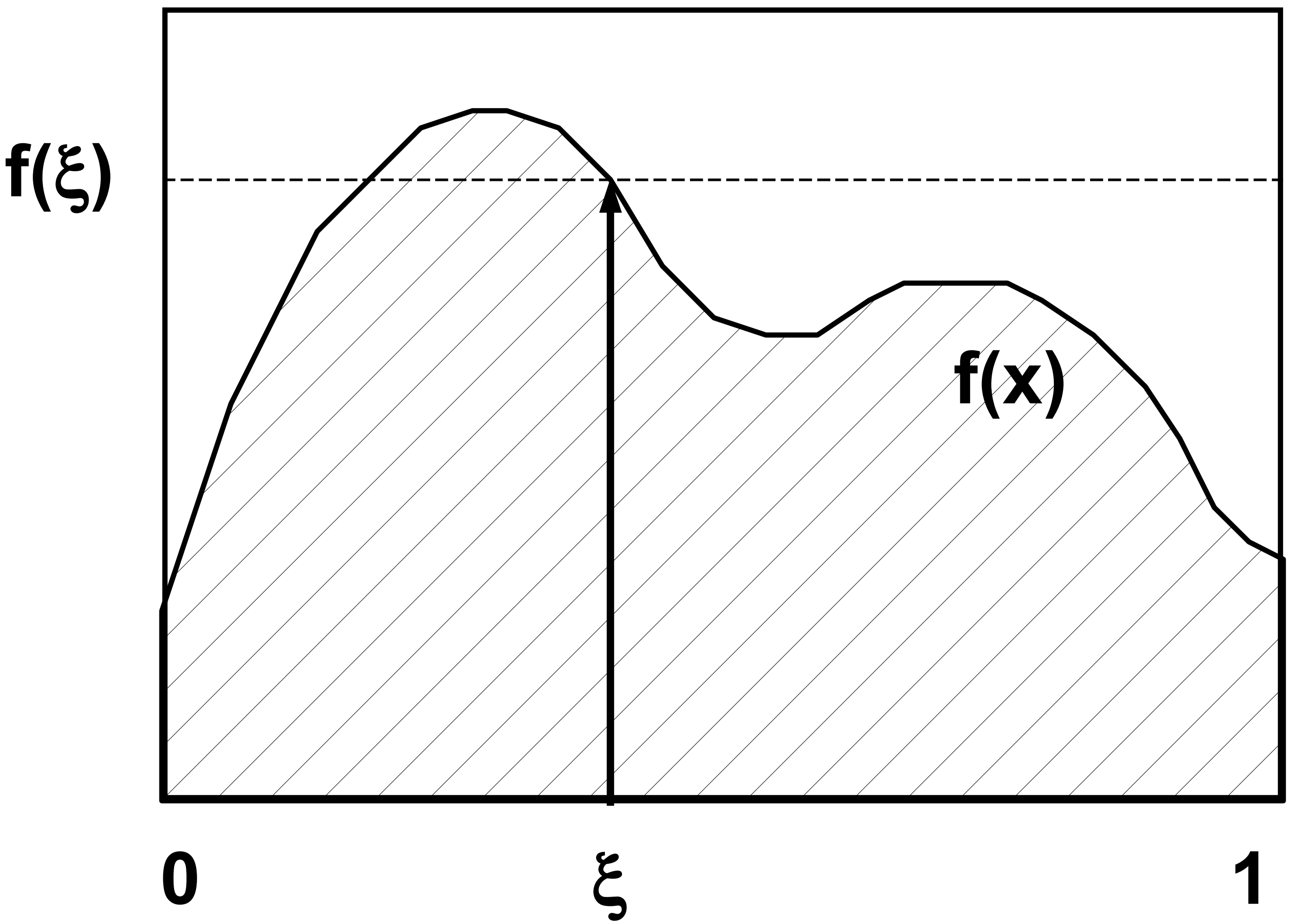
$$\langle I \rangle_{prim} = f(\xi)$$

Estimate is **unbiased**, because:

$$E\left(\langle I \rangle_{prim}\right) = \int_0^1 f(x)\,dx = I$$

© Josef Pelikán, http://cgg.ms.mff.cuni.cz/~pepca

Primary estimate



Variance of the primary estimate

Quality measure of an estimate is the **variance** (or standard deviation in statistics):

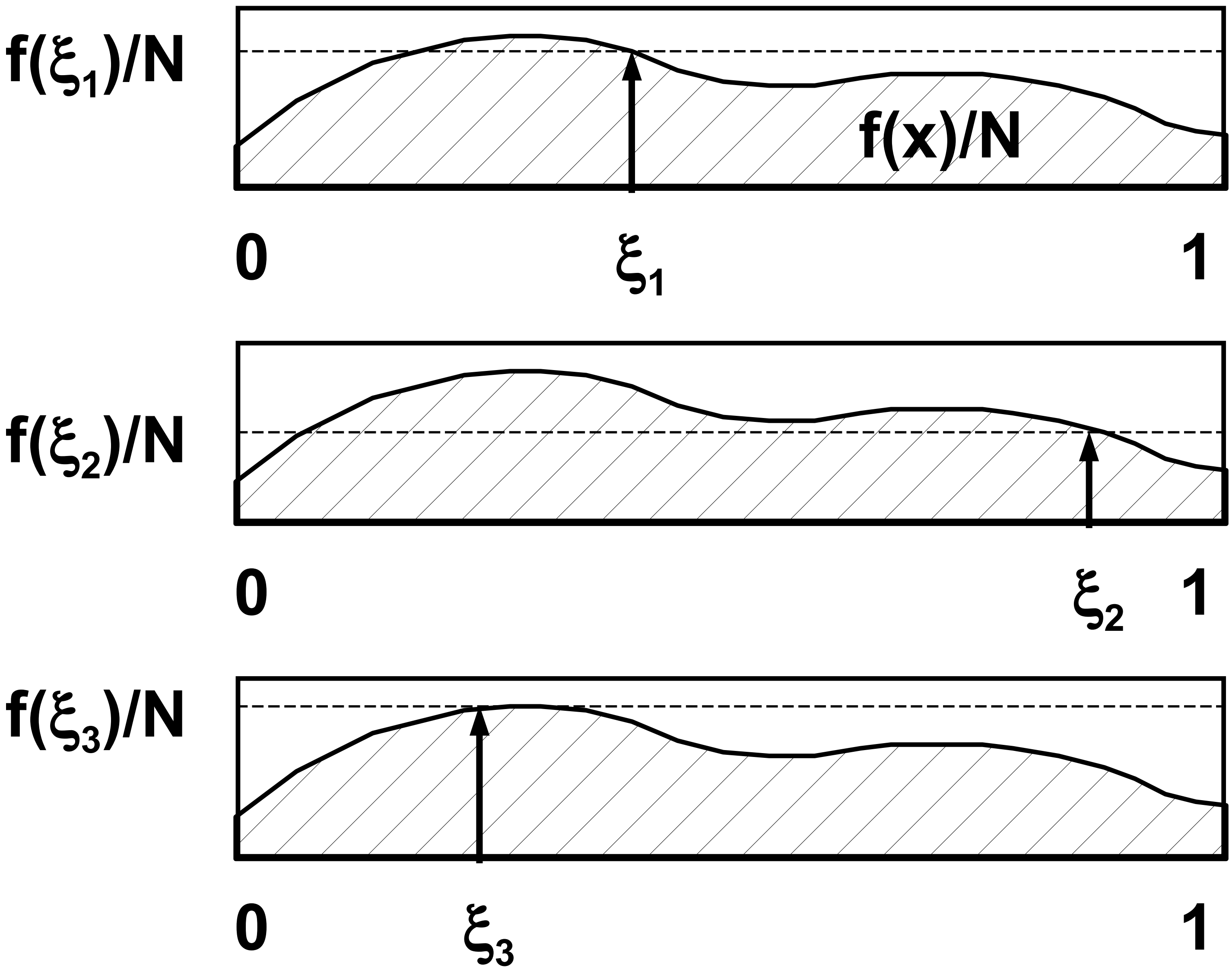
$$V\left(\langle I\rangle_{prim}\right) = \sigma^2_{prim} = \int_0^1 \left| f(x) - I \right|^2 dx = \int_0^1 f(x)^2 dx - I^2$$

(for unbiased estimate)

If we have **one sample only**, variance is too high!

© Josef Pelikán, http://cgg.ms.mff.cuni.cz/~pepca

Secondary estimate



Secodary estimate

The integral divided into **N addends**:

$$I = \int_0^1 f(x)\,dx = \sum_{i=1}^{N} \int_0^1 \frac{f(x)}{N}\,dx = \sum_{i=1}^{N} I_i$$

Secondary estimate of the integral:

$$\langle I \rangle_{sec} = \sum_{i=1}^{N} \langle I_i \rangle_{prim} = \frac{1}{N} \sum_{i=1}^{N} f(\xi_i)$$

Secondary estimate keeps to be **unbiased**.

Variance of the secondary estimate

$$\sigma^2_{sec} = \int_0^1 \cdots \int_0^1 \left[\frac{1}{N} \sum_{i=1}^{N} f(x_i) \right]^2 dx_1 \ldots dx_N - I^2 =$$

$$= \frac{1}{N} \int_0^1 f^2(x)\, dx - \frac{1}{N} I^2 =$$

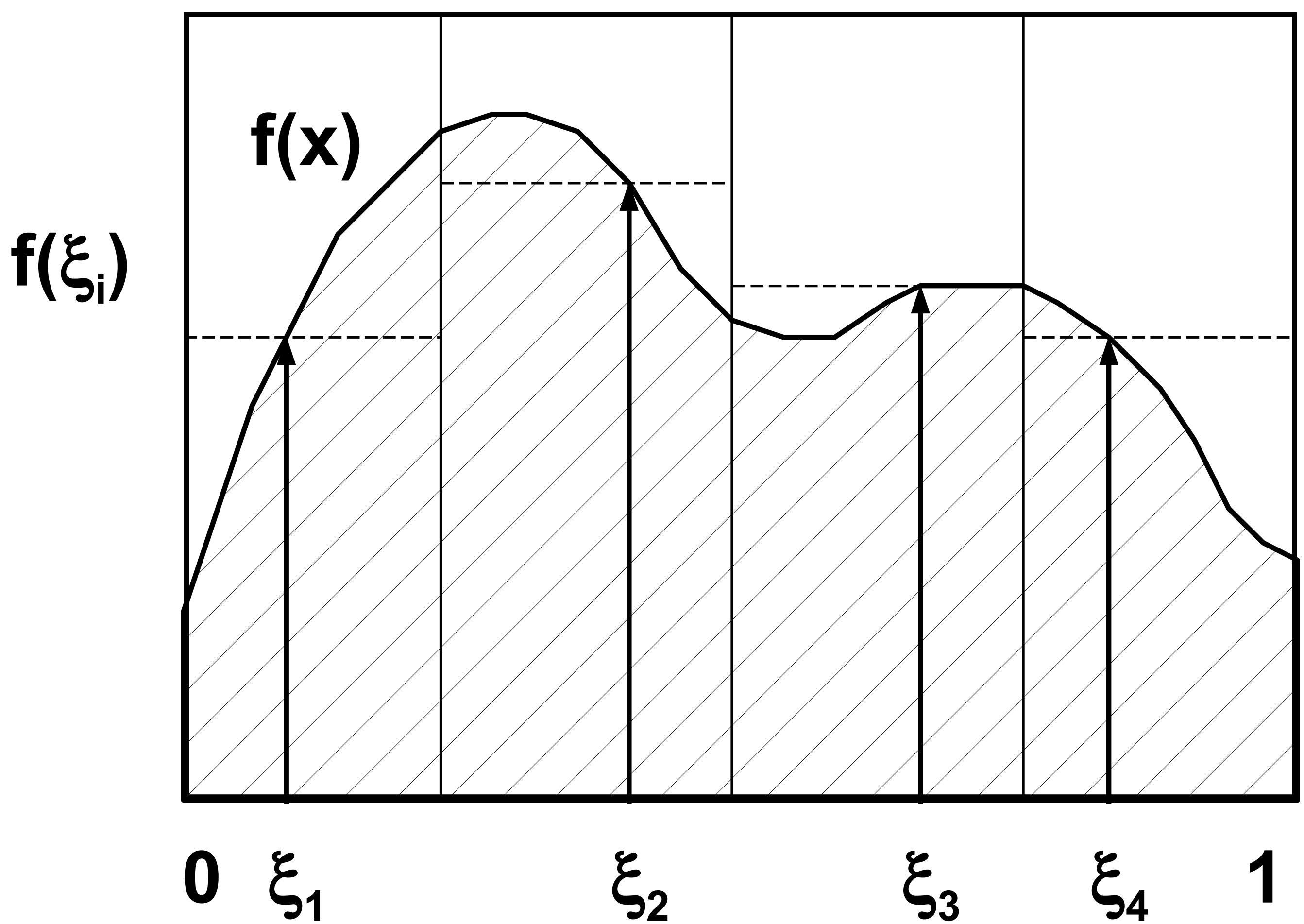
$$= \frac{\sigma^2_{prim}}{N}$$

… variance is √**N-times smaller**!
(convergence: **1/√N**)

Stratified sampling

- if we are sampling the whole set using constant distribution, **clustering is inevitable**
  - too high variance
- **stratified sampling**
  - reduces clustering
  - reduces variance
  - very popular in sampling discrete sets as well (statistics in sociology, etc.)
- domain is divided into intervals which are then sampled **independently**

Stratified sampling



Stratified sampling

Domain **(0,1)** is divided **into N parts A_i**:

$$I = \int_0^1 f(x)\,dx = \sum_{i=1}^{N} \int_{A_i} f(x)\,dx = \sum_{i=1}^{N} I_i$$

Estimate:

$$\langle I \rangle_{strat} = \sum_{i=1}^{N} \langle I_i \rangle_{prim} = \frac{1}{N} \sum_{i=1}^{N} f(\xi_i), \qquad f(\xi_i) \in A_i$$

Variance of stratified sampling

$$\underline{\sigma^2_{strat}} = \sum_{i=1}^{N}\left[\int_{A_i}\left[\frac{f(x_i)}{N}\right]^2 N\,dx_i - I_i^2\right] =$$

$$= \frac{1}{N}\int_0^1 f^2(x)\,dx - \sum_{i=1}^{N} I_i^2 \le \underline{\sigma^2_{sec}}$$

... variance cannot be higher than variance of **secondary estimate**!

© Josef Pelikán, http://cgg.ms.mff.cuni.cz/~pepca

Division of the domain

- **uniform** division of the interval **(0,1)**
  - natural approach for unknown function **f**
- if we know more about **f waveform**, our goal is to use a division induced by a shape of the function (to minimize partial variances)
- division of **d-dimensional interval** leads to $\mathbf{N}^d$ samples
  - **"N rooks"** sampling is smarter

Importance sampling

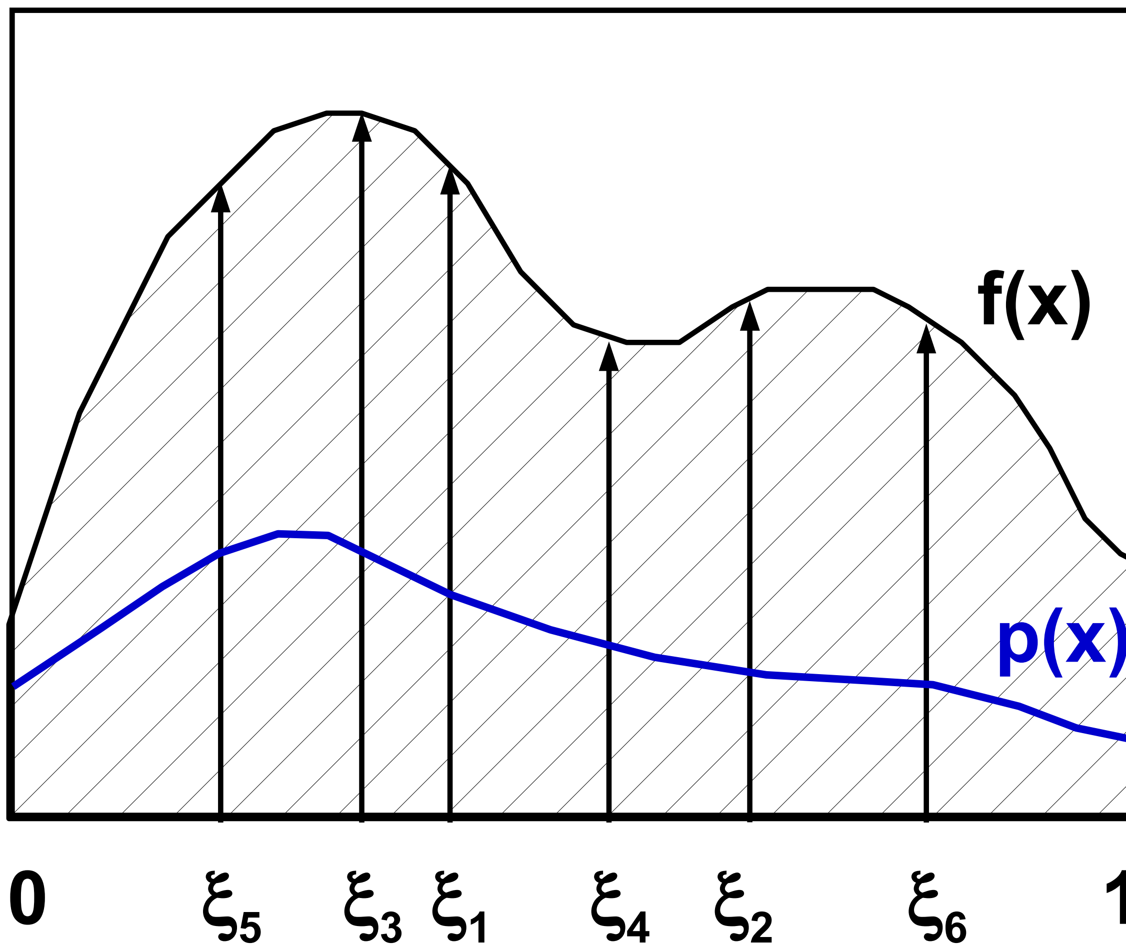
- sampling is more important in parts with higher value of the function **f**
  - samples from that regions have higher impact (relative contribution to the result)

- **“importance sampling”** prioritizes such samples
  - formally: sampling is controlled by custom **probability distribution function** (density) **p(x)**

- **lower variance** but still unbiased estimate

© Josef Pelikán, http://cgg.ms.mff.cuni.cz/~pepca

Importance sampling



Importance sampling

Estimated integral is transformed:

$$I = \int_0^1 f(x)\, dx = \int_0^1 \frac{f(x)}{p(x)}\, p(x)\, dx$$

If PDF of random variable ξ is **p(x)**, we use an estimate:

$$\langle I \rangle_{imp} = \frac{f(\xi)}{p(\xi)}$$

(unbiased)

Variance of importance sampling

$$\sigma^2_{imp} = \int_0^1 \left[\frac{f(x)}{p(x)}\right]^2 p(x)\,dx - I^2 =$$

$$= \int_0^1 \frac{f^2(x)}{p(x)}\,dx - I^2$$

If PDF **p(x)** is similar to an integrand **f(x)**, we are estimating integral of function with lower **variance** than **f(x)**.

PDF (density) p(x)

- $\mathbf{p(x) \geq 0}$, $\mathbf{p(x) > 0}$ where $\mathbf{f(x) \neq 0}$
- $\int \mathbf{p(x)\,dx = 1}$
- it is possible to **efficiently generate** samples from the distribution
  - easiest way: cummulative distribution function **P(x)** is known and invertible ($\mathbf{P^{-1}(x)}$)

$$\underline{P(x)} = \int_0^x p(t)\,dt$$

Algorithm

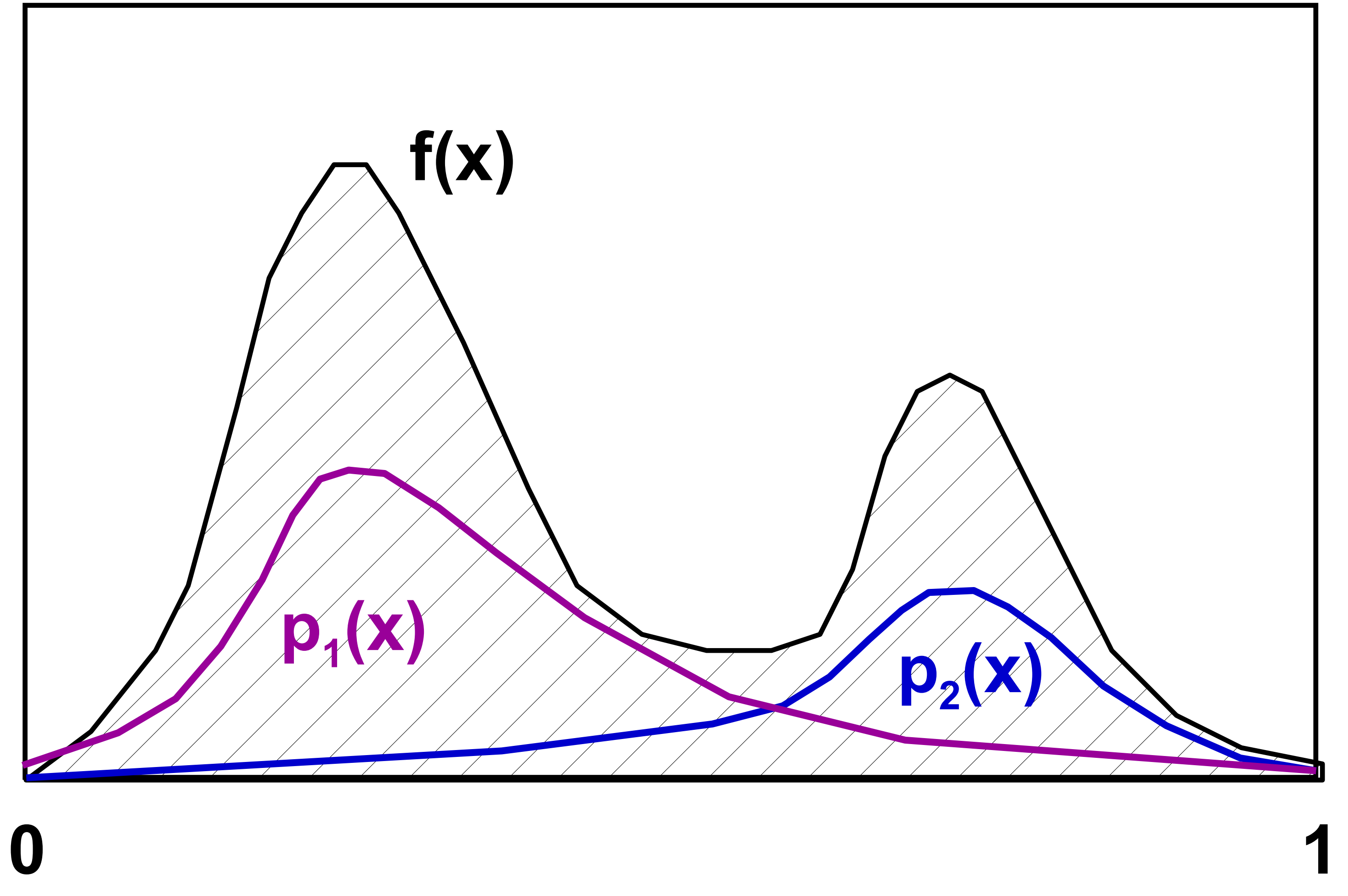
Instead of sampling from **p(x)** we compute **τ** from **uniform distribution R(0,1)** and transform it:

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

Estimate: $$\langle I \rangle_{imp} = \frac{f\left(P^{-1}(\tau)\right)}{p\left(P^{-1}(\tau)\right)}$$

$$I = \int_0^1 f(x)\,dx = \int_0^1 f\left(P^{-1}(t)\right) \frac{dP^{-1}(t)}{dt}\,dt = \int_0^1 \frac{f\left(P^{-1}(t)\right)}{p\left(P^{-1}(t)\right)}\,dt$$

Combined sampling



Combined sampling

Presumption: **n** random variables $\xi_1, .. \xi_n$ with PDFs (densitioes) $\mathbf{p_1(x)}, .. \mathbf{p_n(x)}$.

Combined estimate of the integral is:

$$\langle I \rangle_{comb} = \sum_{i=1}^{n} w_i(\xi_i) \frac{f(\xi_i)}{p_i(\xi_i)}$$

where $\mathbf{w_i(x)}$ are nonnegative **weight functions**.

Eliminating bias

$$E\left(\langle I\rangle_{comb}\right) = \sum_{i=1}^{n} \int_0^1 \left[w_i(x_i) \frac{f(x_i)}{p_i(x_i)} \right] p_i(x_i)\, dx_i =$$

$$= \int_0^1 \left[\sum_{i=1}^{n} w_i(x) \right] f(x)\, dx \equiv \int_0^1 f(x)\, dx$$

The only condition:

$$\forall x: \quad \sum_{i=1}^{n} w_i(x) = 1$$

Variance of combined estimate

$$\sigma^2_{comb} = \sum_{i=1}^{n} \left\{ \int_0^1 \left[w_i(x_i) \frac{f(x_i)}{p_i(x_i)} \right]^2 p_i(x_i)\, dx_i - \left[\int_0^1 w_i(x_i) \frac{f(x_i)}{p_i(x_i)} p_i(x_i)\, dx_i \right]^2 \right\} =$$

$$= \int_0^1 \left[\sum_{i=1}^{n} \frac{w_i^2(x)}{p_i(x)} \right] f(x)\, dx - \sum_{i=1}^{n} \left[\int_0^1 w_i(x)\, f(x)\, dx \right]^2$$

© Josef Pelikán, http://cgg.ms.mff.cuni.cz/~pepca

Mean, maximum (cutoff)

$$w_i(x) = \frac{1}{n}$$

$$\langle I \rangle_{average} = \frac{1}{n} \sum_{i=1}^{n} \frac{f(\xi_i)}{p_i(\xi_i)}$$

$$w_i(x) = \begin{cases} 1 & \text{for } p_i(x) = \max_j\{p_j(x)\} \\ 0 & \text{otherwise} \end{cases}$$

$$\langle I \rangle_{max} = \sum_{i=1}^{n} \left(p_i(\xi_i) = \max_j\{p_j(\xi_i)\} \right) ? \frac{f(\xi_i)}{p_i(\xi_i)} : 0$$

© Josef Pelikán, http://cgg.ms.mff.cuni.cz/~pepca

Balance heuristics

$$w_i(x) = \frac{p_i(x)}{\sum_{j=1}^{n} p_j(x)}$$

$$\langle I \rangle_{bal} = \sum_{i=1}^{n} \frac{f(\xi_i)}{\sum_{j=1}^{n} p_j(\xi_i)}$$

$$\sigma_{bal}^2 = \int_0^1 \frac{f^2(x)}{\sum_{i=1}^{n} p_i(x)}\, dx - \sum_{i=1}^{n} \left[\int_0^1 \frac{p_i(x)}{\sum_{j=1}^{n} p_j(x)} f(x)\, dx \right]^2$$

$$\sigma_{comb}^2 \geq \sigma_{bal}^2 - \left(1 - \frac{1}{n}\right) \cdot I^2$$

Power heuristics

Generalization: $$w_i(x) = \frac{p_i^\beta(x)}{\sum_{j=1}^{n} p_j^\beta(x)}$$

$$\langle I \rangle_{power} = \sum_{i=1}^{n} \frac{p_i^{\beta-1}(\xi_i)}{\sum_{j=1}^{n} p_j^\beta(\xi_i)} f(\xi_i)$$

$\beta = 1$ .. balance, $\beta = \infty$ .. maximum

© Josef Pelikán, http://cgg.ms.mff.cuni.cz/~pepca

Integrand transformation

We can interpret the combined estimate as **a transformation of the integrand**:

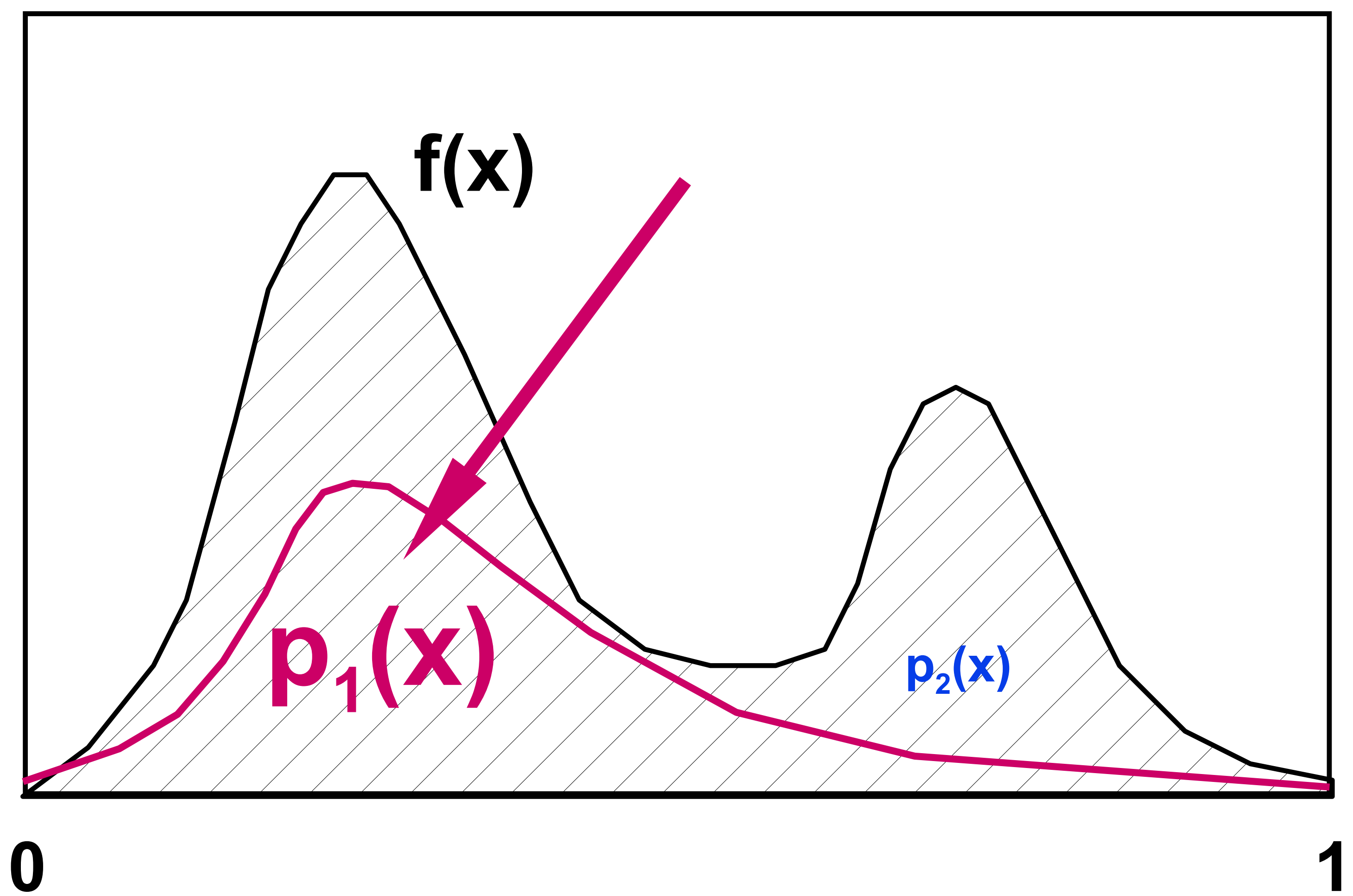
$$I = \int_0^1 f(x)\,dx = \sum_{i=1}^{n} \int_0^1 w_i(x)\cdot f(x)\,dx$$

Combination of importance samplings:

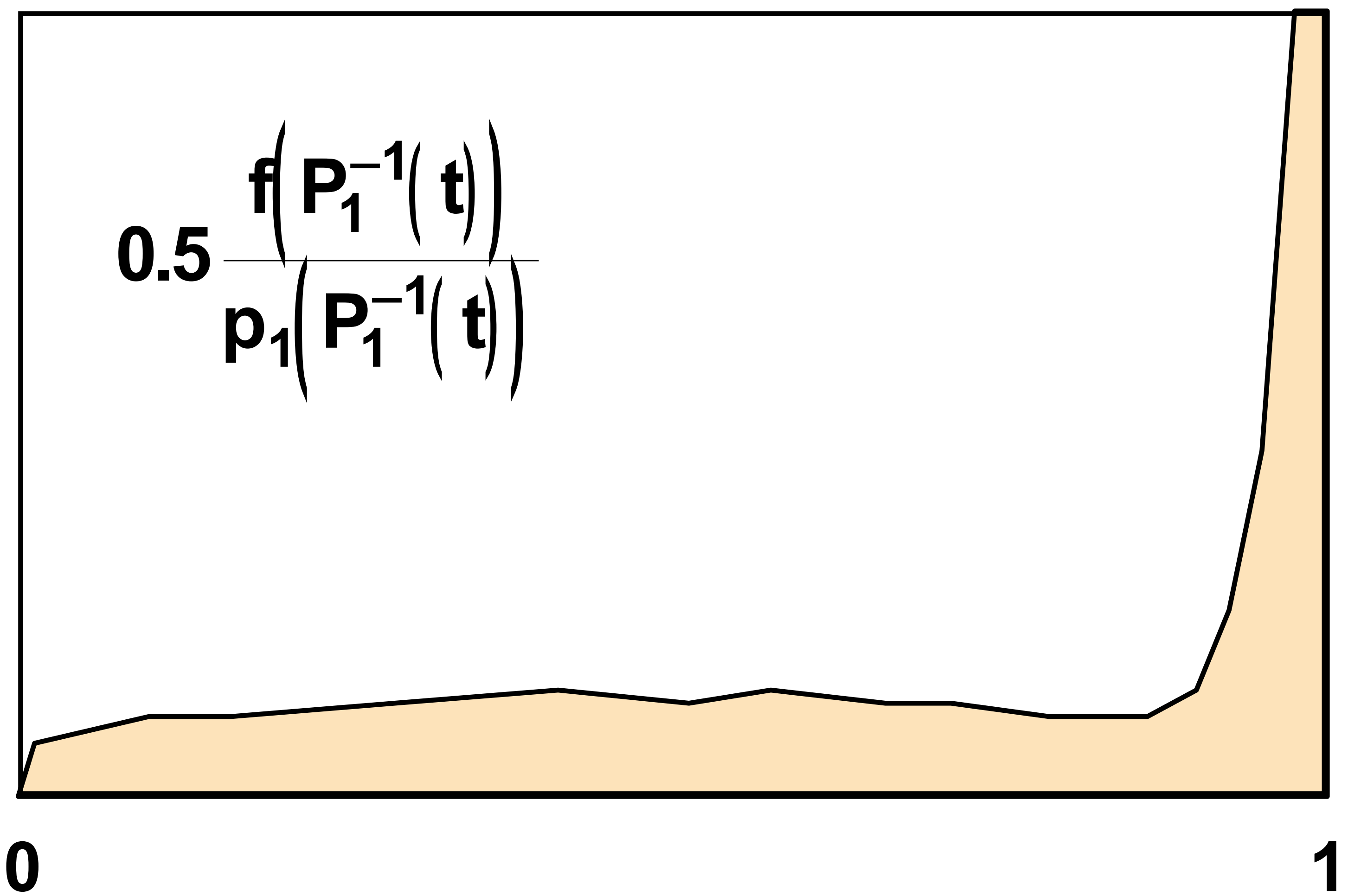
$$I = \sum_{i=1}^{n} \int_0^1 \frac{w_i\left(P_i^{-1}(t)\right)}{p_i\left(P_i^{-1}(t)\right)}\, f\left(P_i^{-1}(t)\right)\,dt$$

© Josef Pelikán, http://cgg.ms.mff.cuni.cz/~pepca

Example: one term of a combination



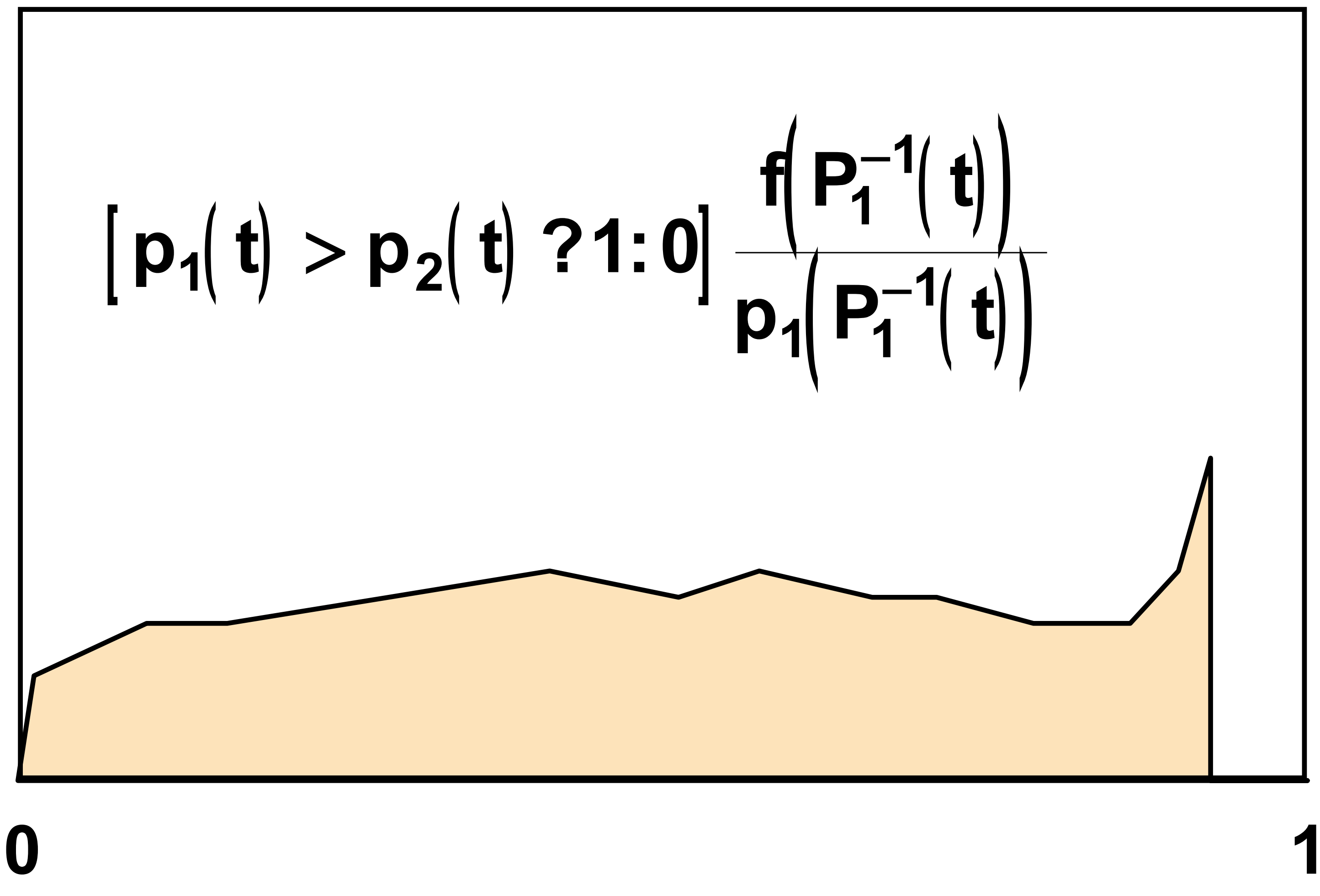
Arithmetic average (mean)



$$0.5\,\frac{f\left(P_1^{-1}(t)\right)}{p_1\left(P_1^{-1}(t)\right)}$$

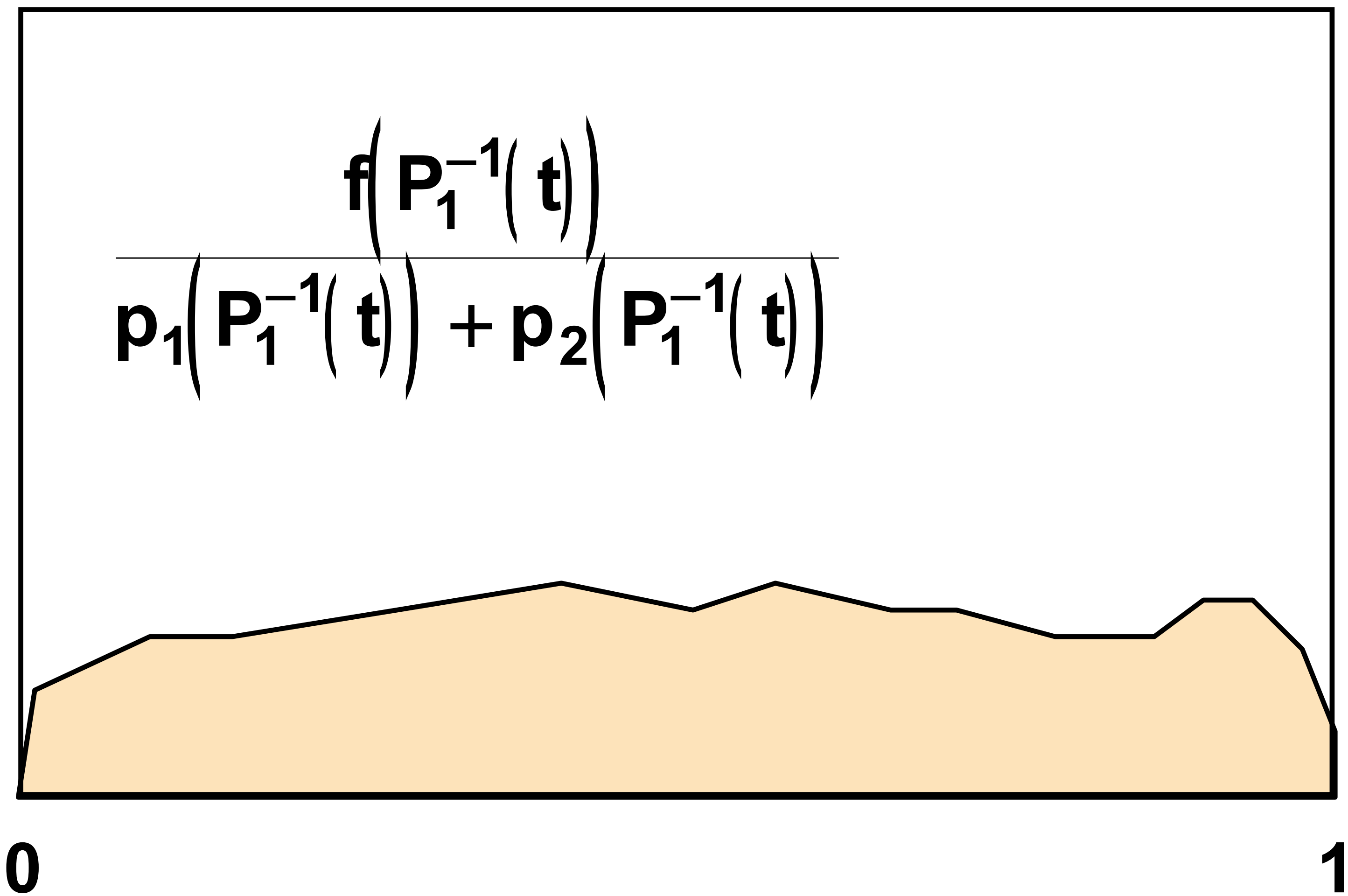
Maximum

$$\left[p_1(t) > p_2(t)\ ?\ 1 : 0\right] \frac{f\left(P_1^{-1}(t)\right)}{p_1\left(P_1^{-1}(t)\right)}$$



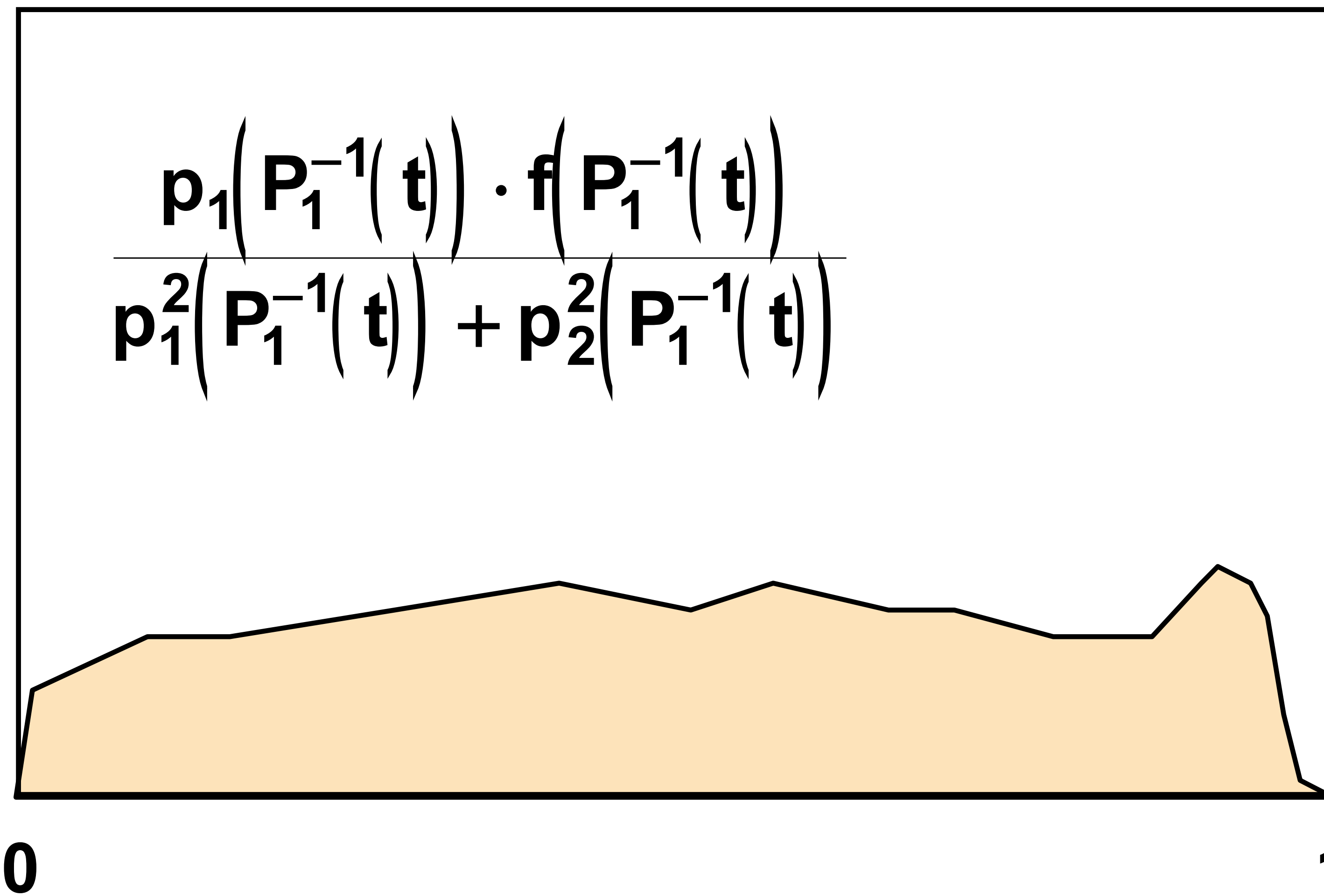
Balance heuristics

$$\frac{f\left(P_1^{-1}(t)\right)}{p_1\left(P_1^{-1}(t)\right)+p_2\left(P_1^{-1}(t)\right)}$$

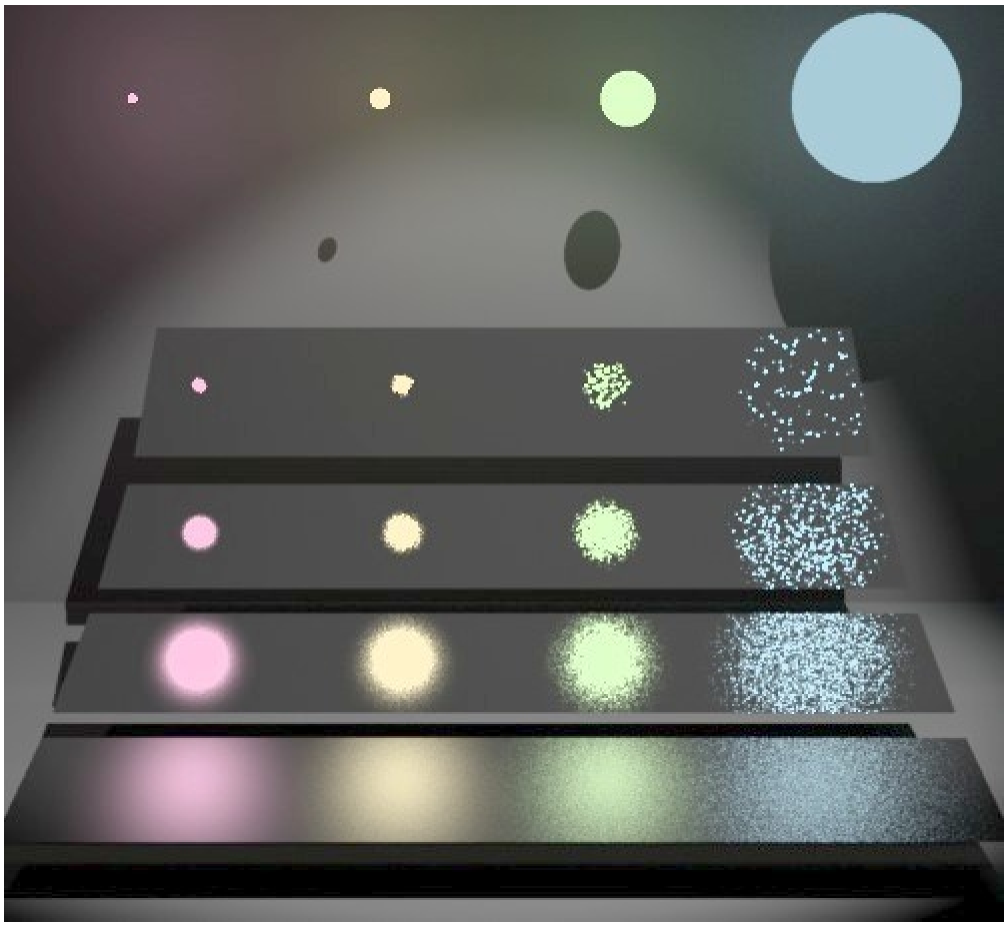


Power heuristics for β=2

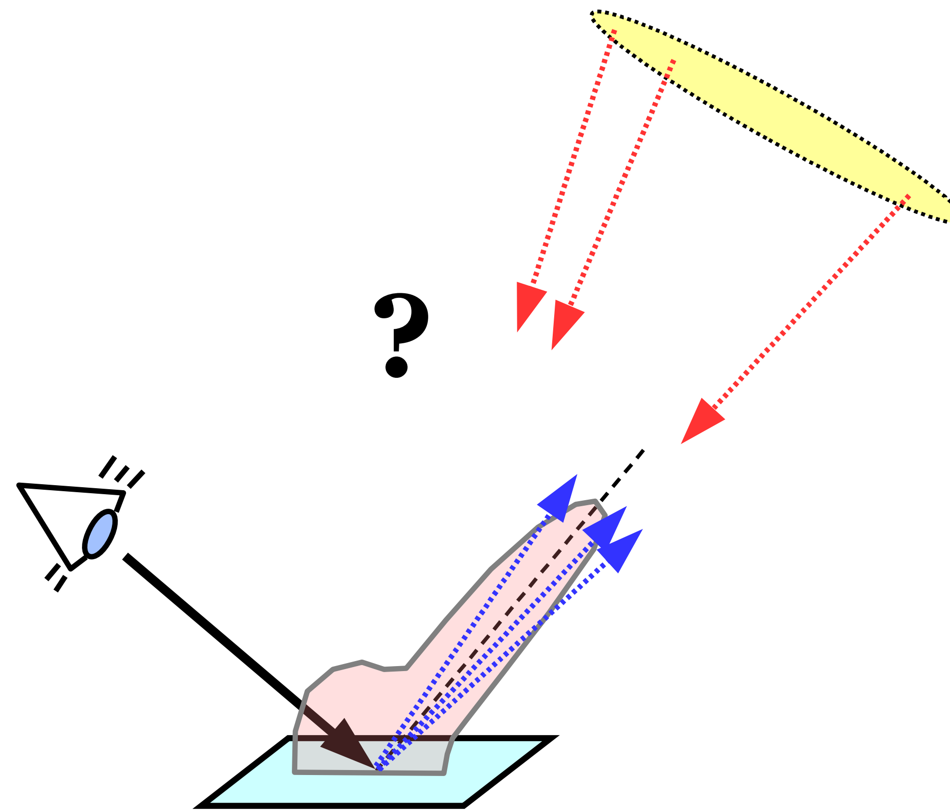
$$\frac{p_1\left(P_1^{-1}(t)\right)\cdot f\left(P_1^{-1}(t)\right)}{p_1^2\left(P_1^{-1}(t)\right)+p_2^2\left(P_1^{-1}(t)\right)}$$



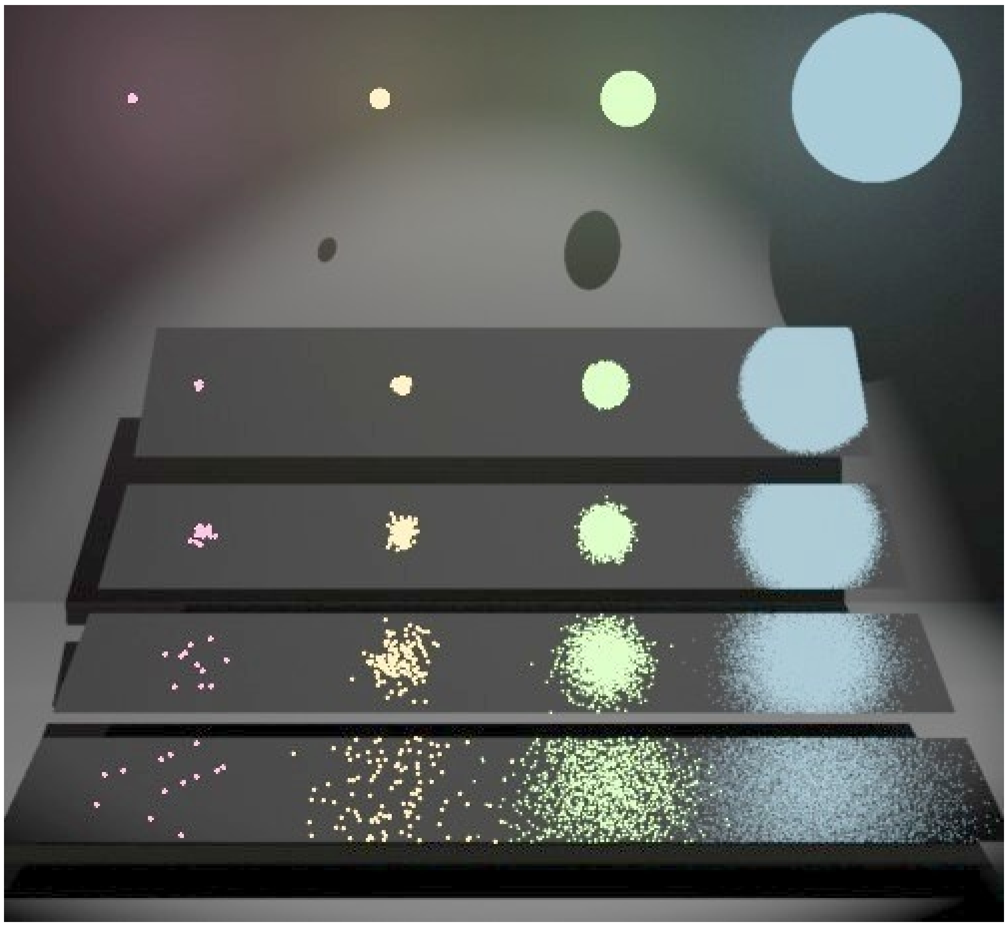
Rendering example



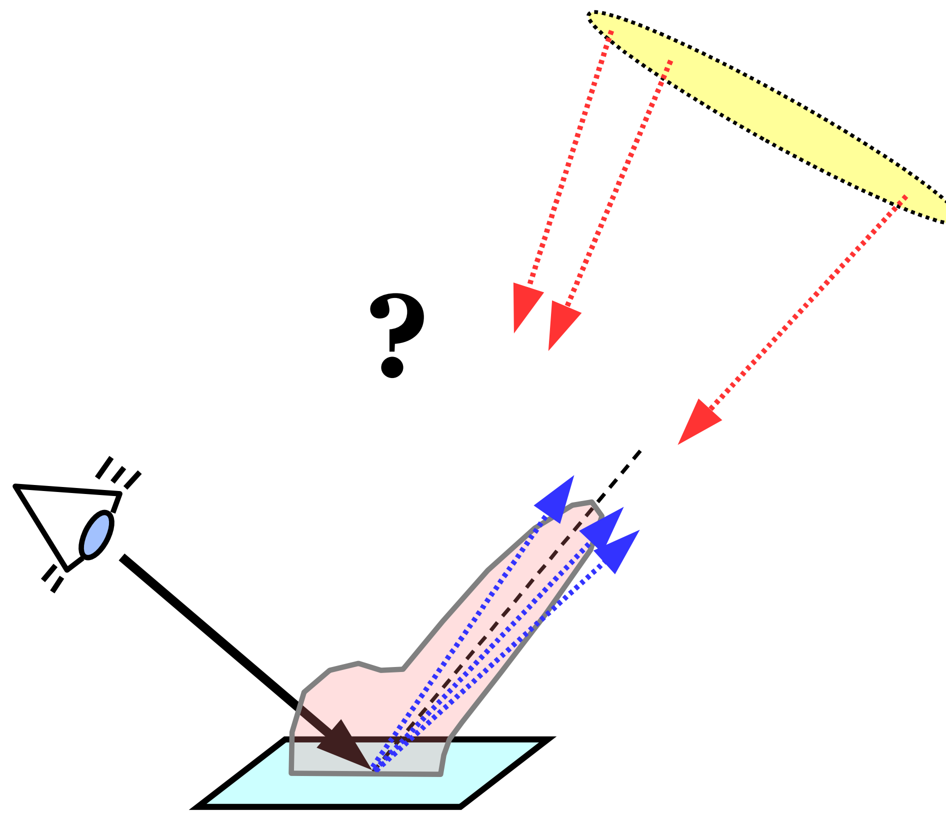
© 1995 Eric Veach, Leonidas J. Guibas



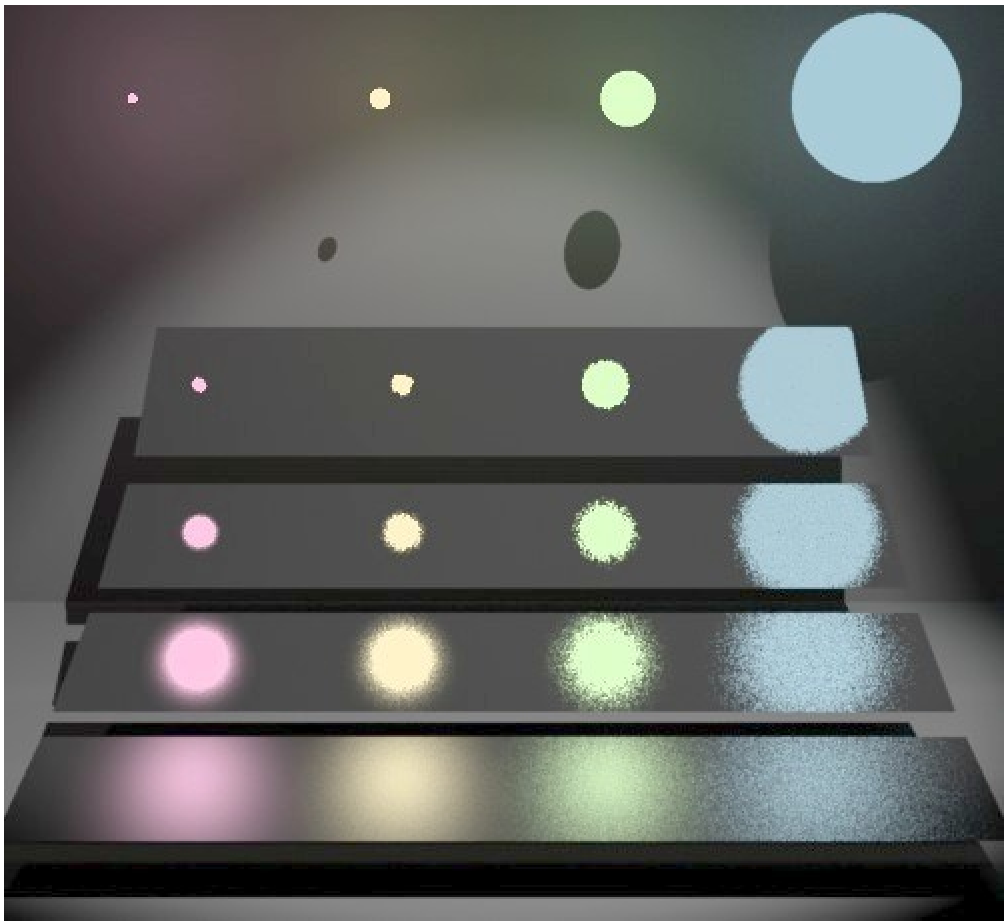
Rendering example



© 1995 Eric Veach, Leonidas J. Guibas



Combined sampling

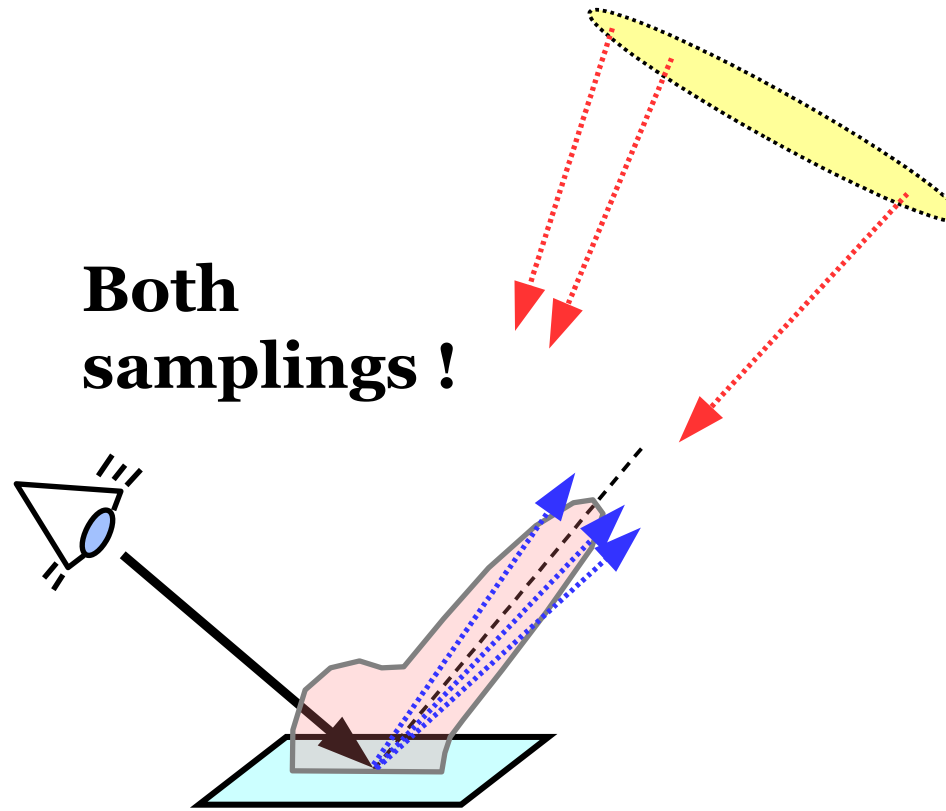


© 1995 Eric Veach, Leonidas J. Guibas

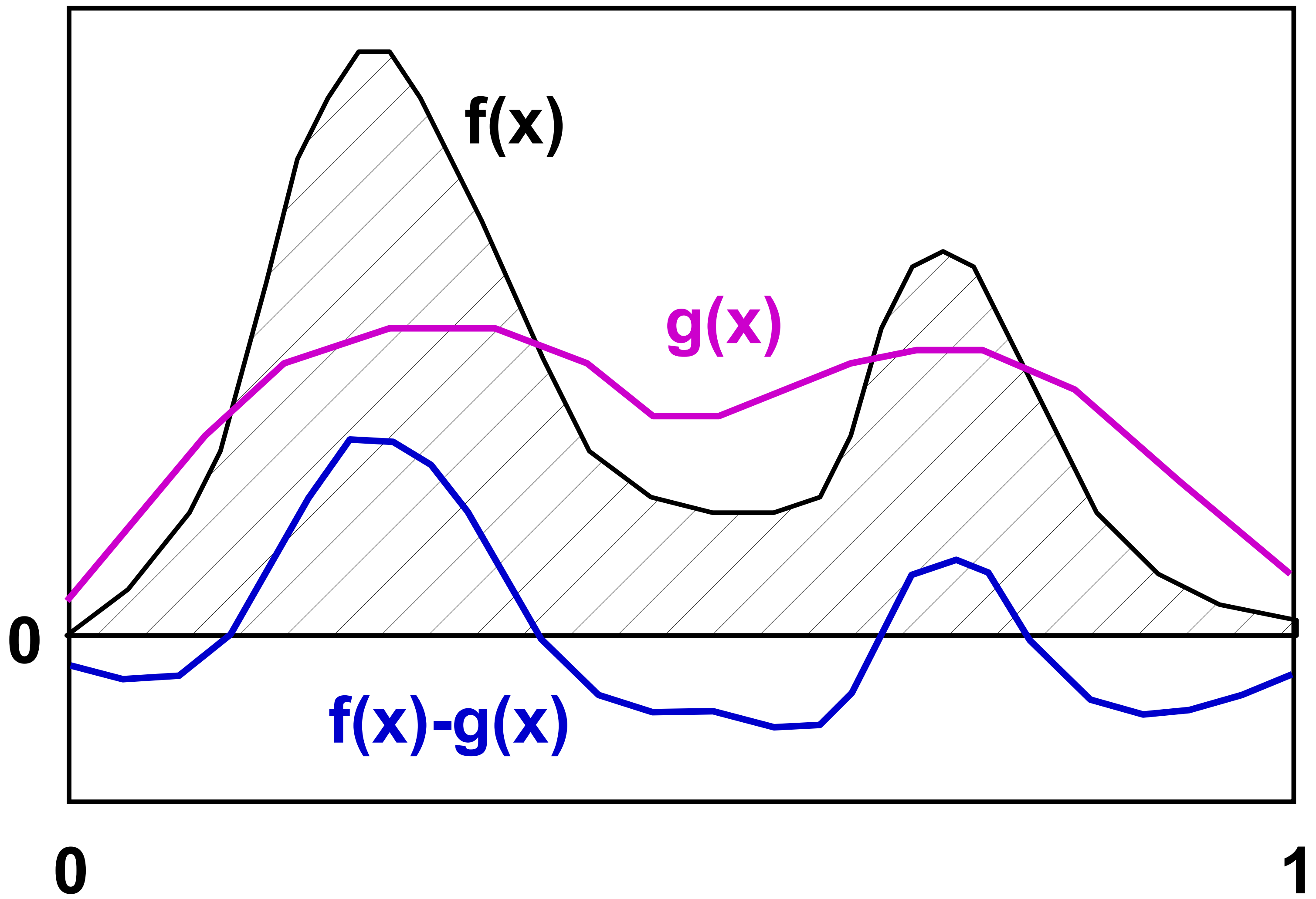
Power heuristics

β = 2

Both samplings !



Control function



Control function

Function **g(x)** is **approximating the integrand**.
If we can compute its integral easily (close form!):

$$I = \int_0^1 f(x)\,dx = \int_0^1 [f(x) - g(x)]\,dx + \int_0^1 g(x)\,dx =$$

$$= \int_0^1 [f(x) - g(x)]\,dx + J = \int_0^1 [f(x) - g(x) + J]\,dx$$

Unbiased estimate: $\langle I \rangle_{con} = f(\xi) - g(\xi) + J$

Integral equations

Fredholm integral equation of the second kind:

$$\underline{f(x)} = g(x) + \int_0^1 K(x,y) \cdot \underline{f(y)}\, dy$$

unknown — f

known functions — g, K

- **finite element methods** FEM (the whole function)
- **Monte-Carlo methods** (local)

Recursive Monte Carlo estimate

The right side is estimated **stochastically** (importance sampling) using PDFs $\mathbf{p_i(x)}$:

$$\langle f(x) \rangle_r = g(x) + \frac{K(x,\xi_1)}{p_1(\xi_1)} \cdot \langle f(\xi_1) \rangle_r =$$

$$= g(x) + \frac{K(x,\xi_1)}{p_1(\xi_1)} \cdot \left[g(\xi_1) + \frac{K(\xi_1,\xi_2)}{p_2(\xi_2)} \cdot \langle f(\xi_2) \rangle_r \right]$$

$$= g(x) + \frac{K(x,\xi_1)}{p_1(\xi_1)} g(\xi_1) + \frac{K(x,\xi_1)}{p_1(\xi_1)} \frac{K(\xi_1,\xi_2)}{p_2(\xi_2)} g(\xi_2) + \ldots$$

Recursive Monte Carlo estimate

$$\langle f(x) \rangle_r = \sum_{i=0}^{\infty} \left[\prod_{j=1}^{i} \frac{K(\xi_{j-1}, \xi_j)}{p_j(\xi_j)} \right] g(\xi_i), \qquad \xi_0 = x$$

$\{\xi_1, \xi_2, \xi_3, \ldots\}$ is a **Markov chain**, if the probability $p_i(x)$ is stationary – depends only on ξ_{i-1} (and not on i)

Functional formulation: $f = g + \mathrm{T} f$

Solution (Neumann series): $f = g + \mathrm{T} g + \mathrm{T}^2 g + \ldots$

Russian roulette

- while evaluation a **Neumann series** only finite number of terms can be considered
  – fixed cutoff would introduce **bias** (reduction)
- more correct random cutoff using **Russian roulette**
  – estimate keeps **unbiased**
- can be applied to single integral estimate
  – but is more useful for **infinite series cutoff**

 © Josef Pelikán, http://cgg.ms.mff.cuni.cz/~pepca

Russian roulette for single integral

Integral transformation:

$$I = \int_0^1 f(x)\,dx = \int_0^P \frac{1}{P} f\left(\frac{t}{P}\right) dt \qquad 0 < P \leq 1$$

Unbiased estimate with one random sample:

$$\langle I \rangle_{Russ} = \begin{cases} \frac{1}{P} f\left(\frac{\xi}{P}\right) & \text{for } \xi < P \\ 0 & \text{otherwise} \end{cases}$$

Russian roulette for integral equations

$$\left\langle f(x) \right\rangle_{Russ,r} = \sum_{i=0}^{k} \left[\prod_{j=1}^{i} \frac{K(\xi_{j-1}, \xi_j)}{P_j \cdot p_j(\xi_j)} \right] g(\xi_i), \qquad \xi_0 = x$$

$\{\xi_1, \xi_2, \ldots \xi_k\}$ is **finite** random walk, because estimate $\langle f(\xi_k) \rangle = 0$.

Every sample (step) ξ_i is chosen with **probability** P_i and from PDF (density) $p_i(x)$.

If random value $\tau_{i+1} > P_{i+1}$, the whole process ends; otherwise a new step ξ_{i+1} is computed (new term).

Probabilities

In physical contexts often: $\int_0^1 K(x, y)\, dy < 1$

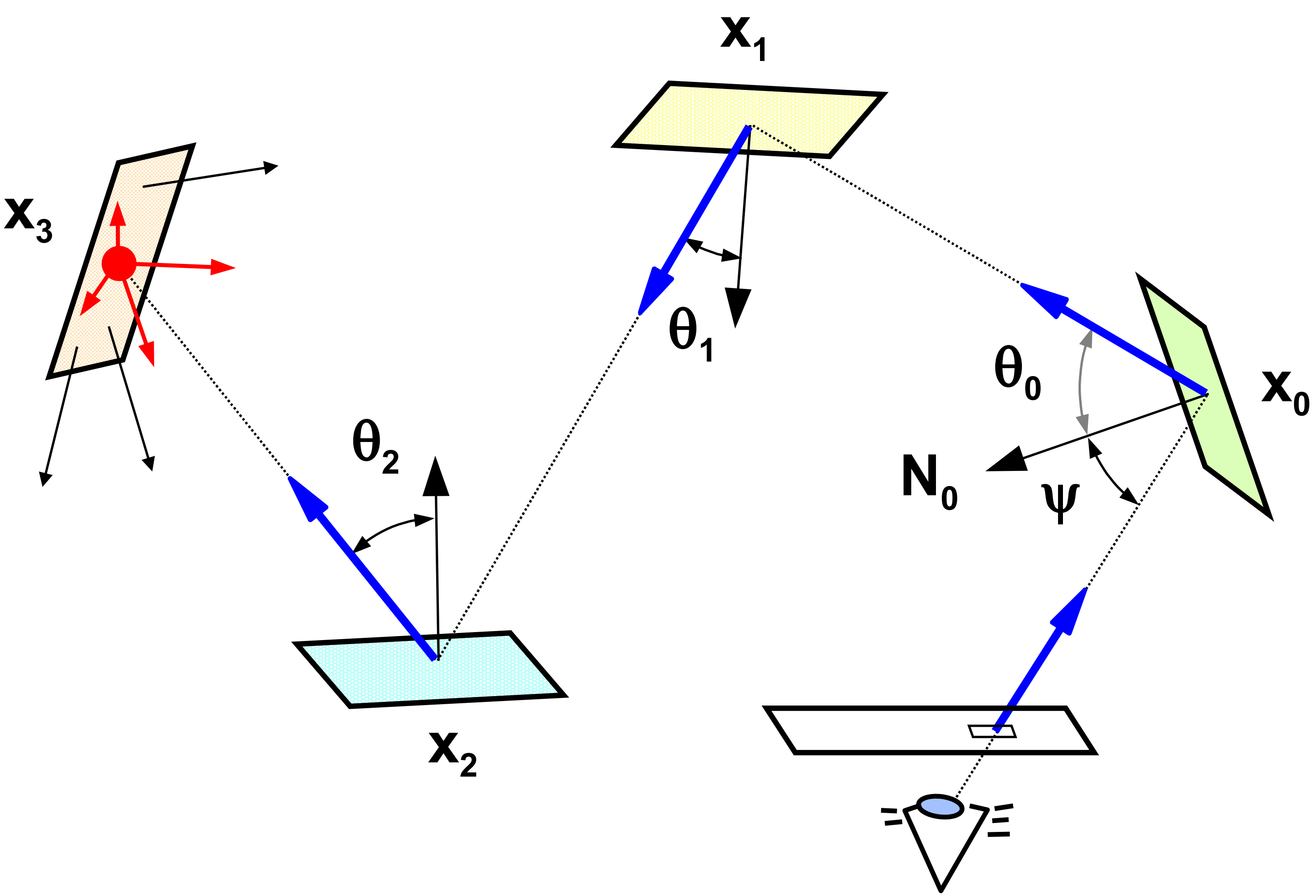
Than the kernel **K** can be used for construction of **subcritical probability setup**:

$$P_i = \int_0^1 K(\xi_{i-1}, y)\, dy, \qquad p_i(x) = \frac{K(\xi_{i-1}, x)}{P_i}$$

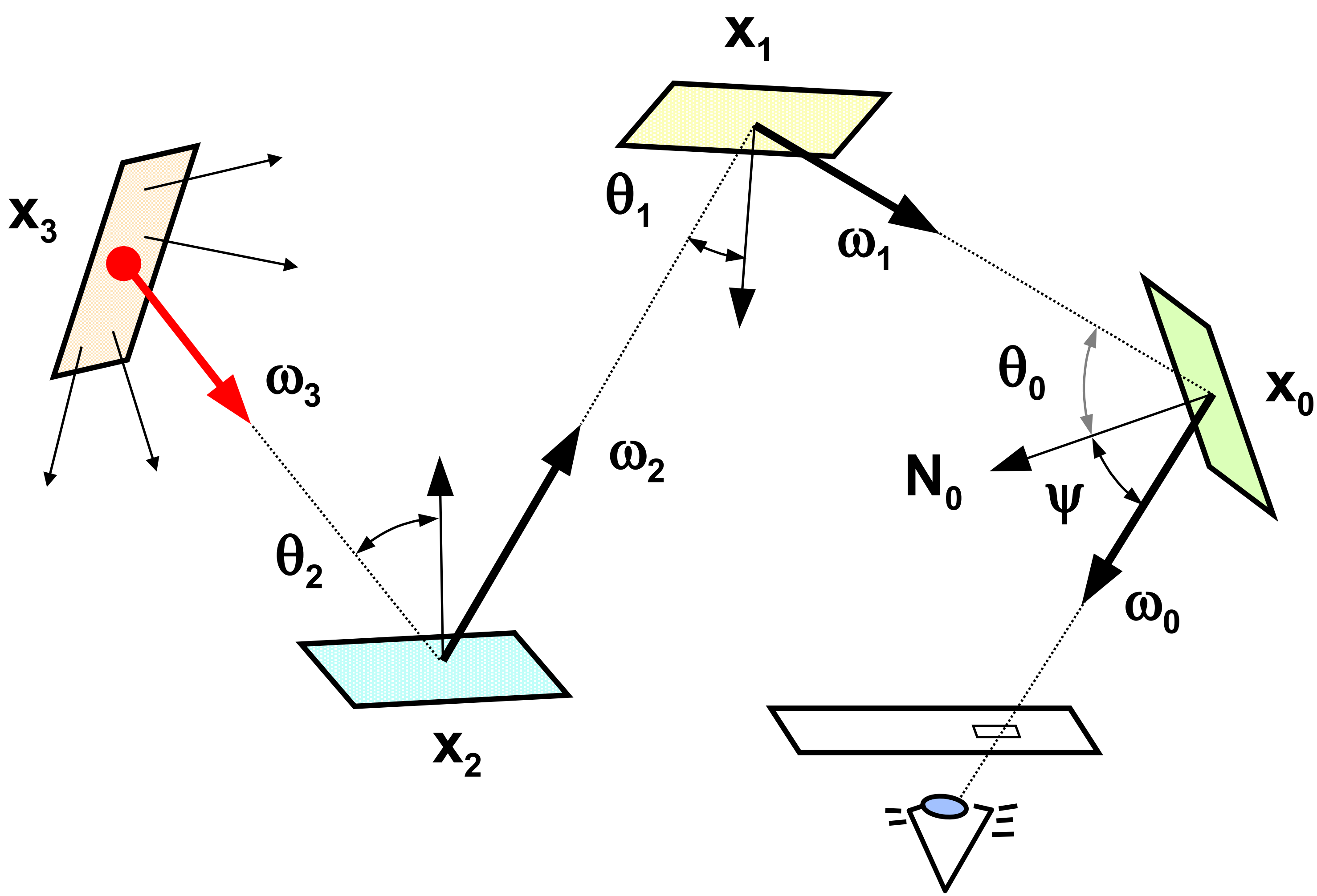
Estimate is reduced to: $\langle f(x) \rangle_{subcrit} = \sum_{i=1}^{k} g(\xi_i)$

© Josef Pelikán, http://cgg.ms.mff.cuni.cz/~pepca

Path Tracing – walk from camera



Path Tracing – light propagation



Next event estimation (NEE)

Previous estimate can use to have **high variance** (only small number of terms are nonzero). Better way is to estimate **g(x)** using "one step ahead" approach:

$$f(x) = g(x) + h(x)$$

$$h(x) = \int_0^1 K(x,y) \cdot f(y)\, dy =$$

$$= \int_0^1 K(x,y) \cdot g(y)\, dy + \int_0^1 K(x,y) \cdot h(y)\, dy$$

Next event estimation

- **the 1st integral** can be estimated using PDF similar to **g(x) – no recursion!**
  - random value ζ_i from PDF $\mathbf{p_i(x)}$
- **the 2nd integral** (recursive) is estimated using importance sampling with subcritical probability (kernel **K**)
  - random value ξ_i from PDF $\mathbf{K(\xi_{i-1},x)/P_i}$

$$\langle h(x) \rangle_{\mathbf{nextev}} = \frac{K(x,\zeta_1)\, g(\zeta_1)}{p_1(\zeta_1)} + \langle h(\xi_1) \rangle_{\mathbf{nextev}}$$

Next event estimation

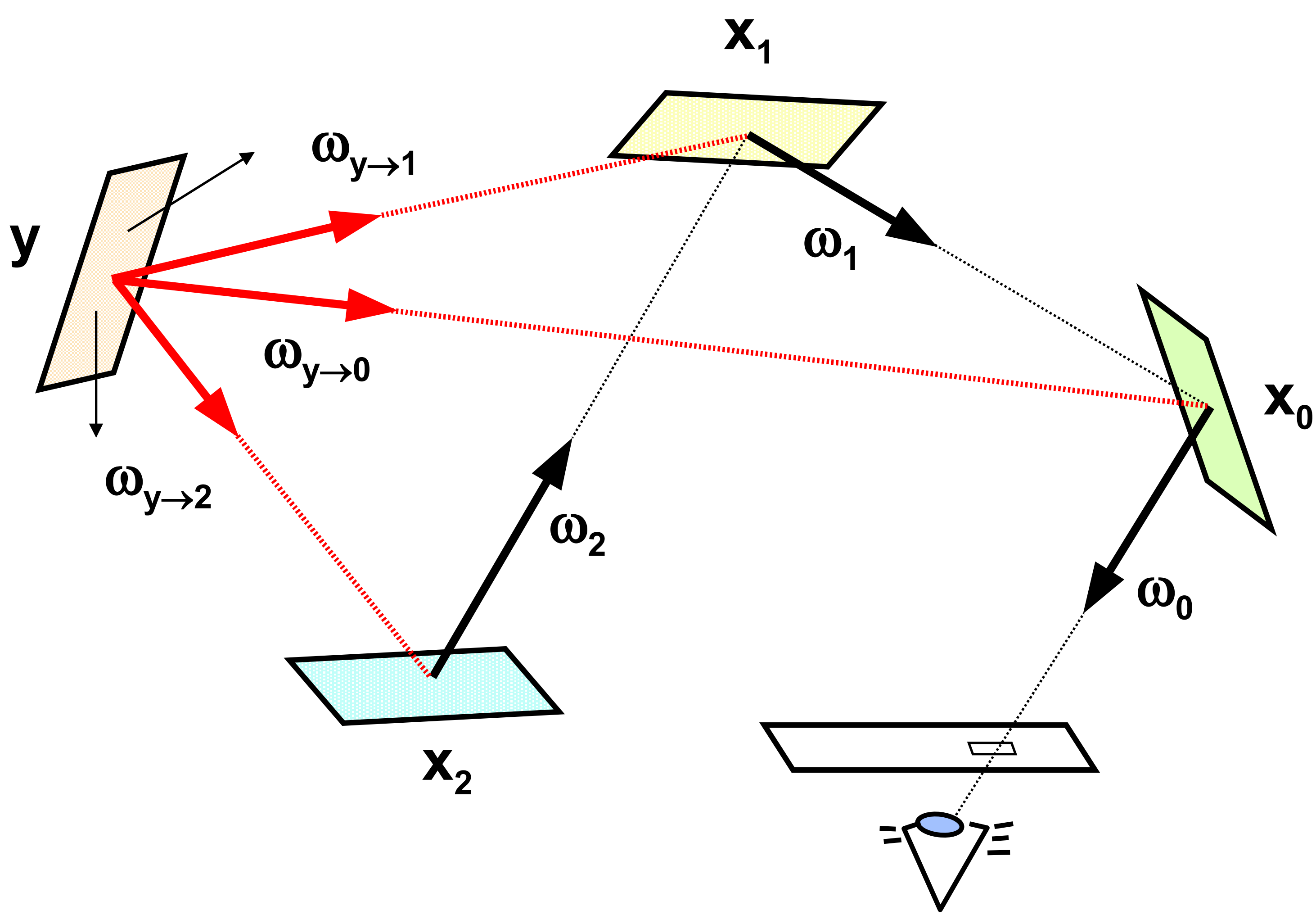
Estimation of the function **h**:

$$\langle h(x) \rangle_{nextev} = \sum_{i=1}^{k} \frac{K(\xi_{i-1}, \zeta_i)\, g(\zeta_i)}{p_i(\zeta_i)}$$

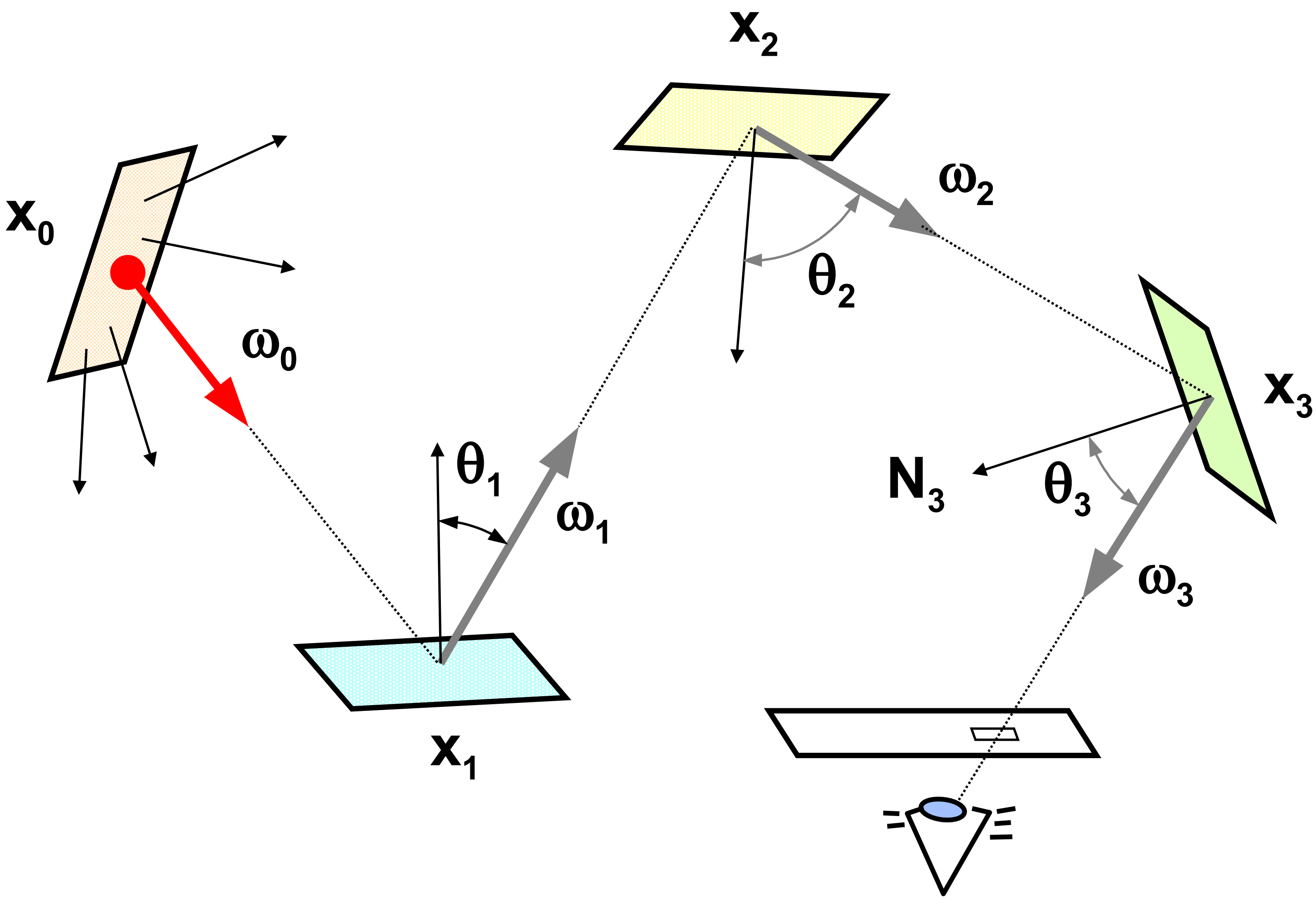
Integral systém estimate:

$$\langle f(x) \rangle_{nextev} = g(x) + \sum_{i=1}^{k} \frac{K(\xi_{i-1}, \zeta_i)\, g(\zeta_i)}{p_i(\zeta_i)}$$

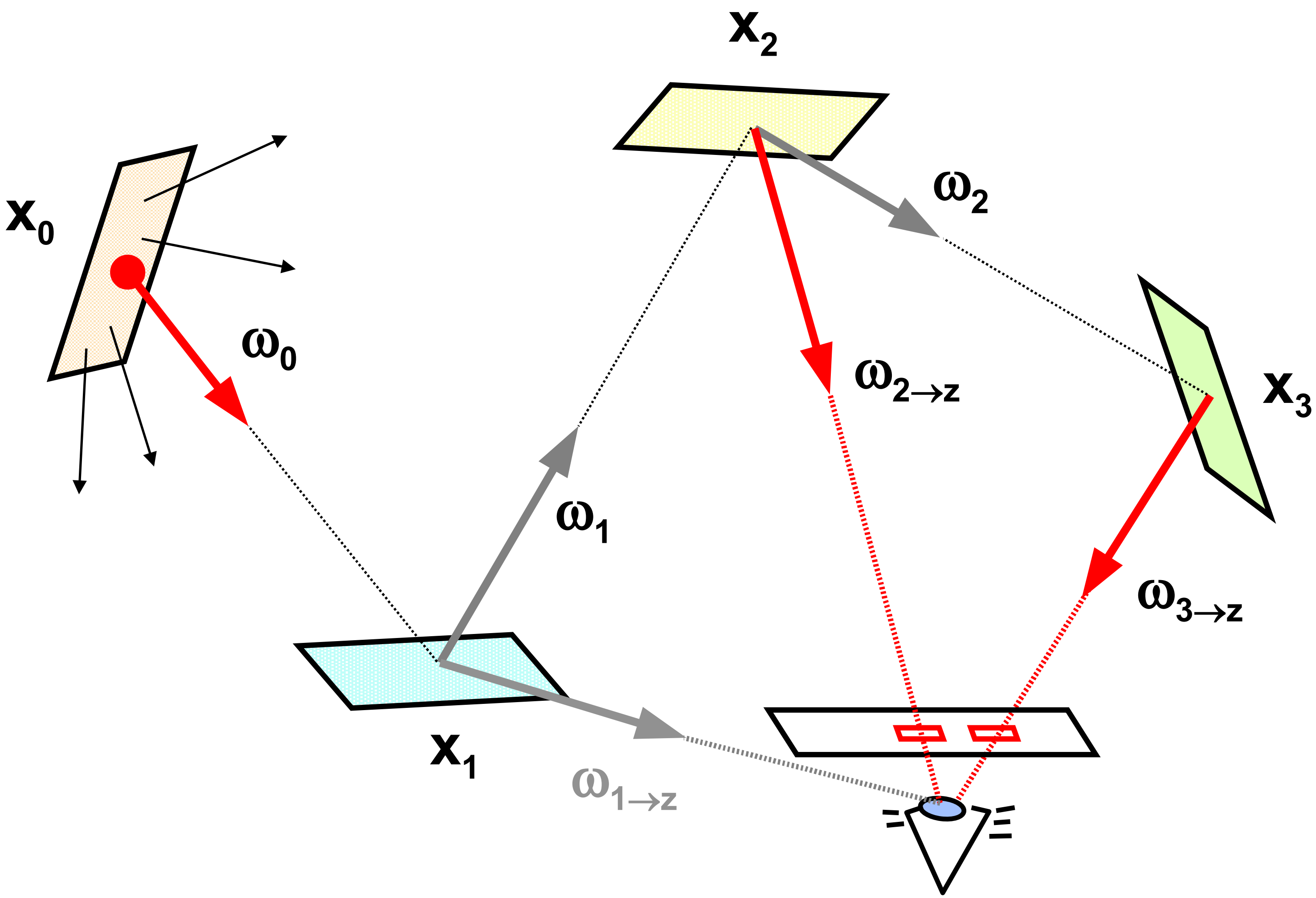
NEE for Path Tracing



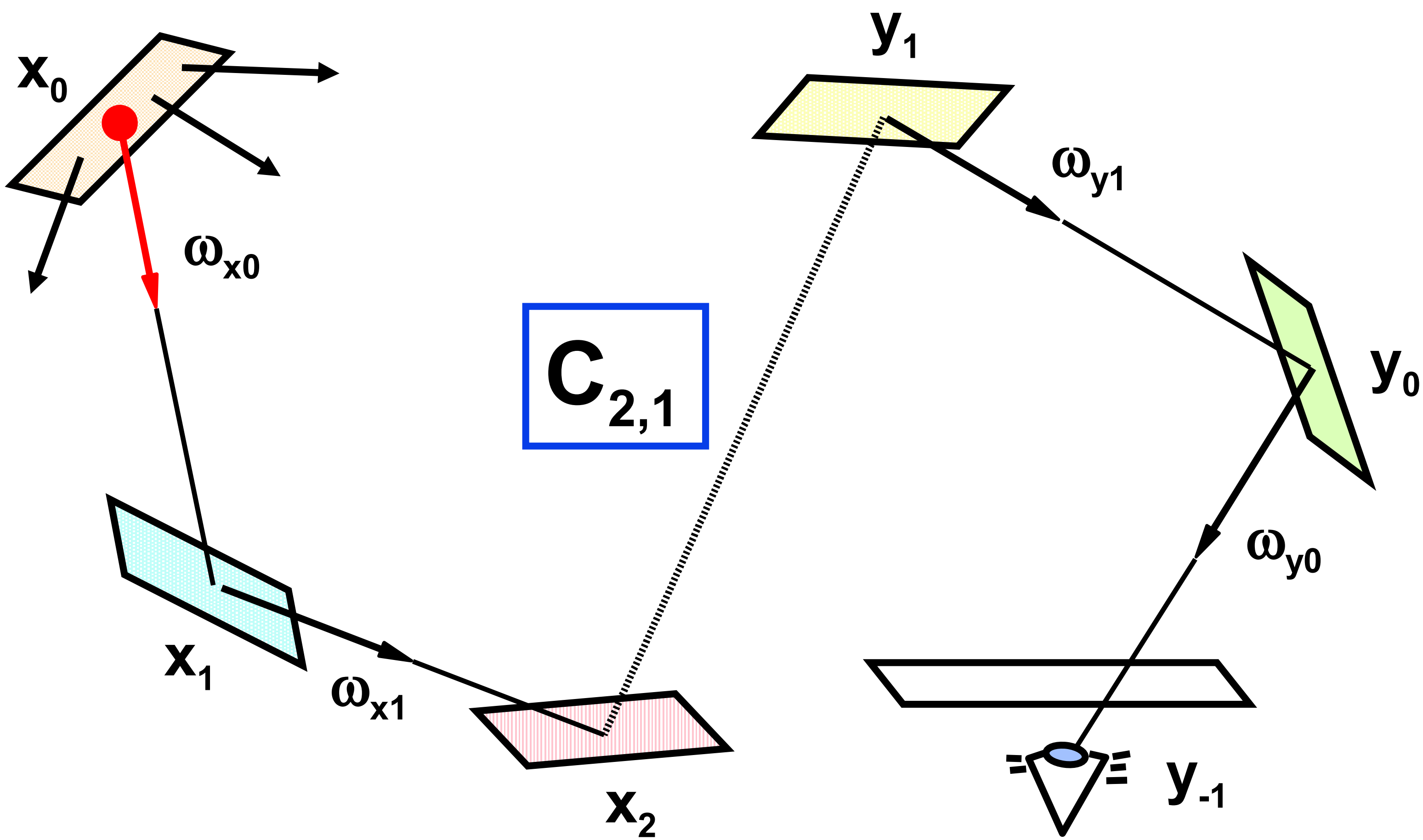
Light Tracing – light propagation



NEE for Light Tracing



Bidirectional Path Tracing

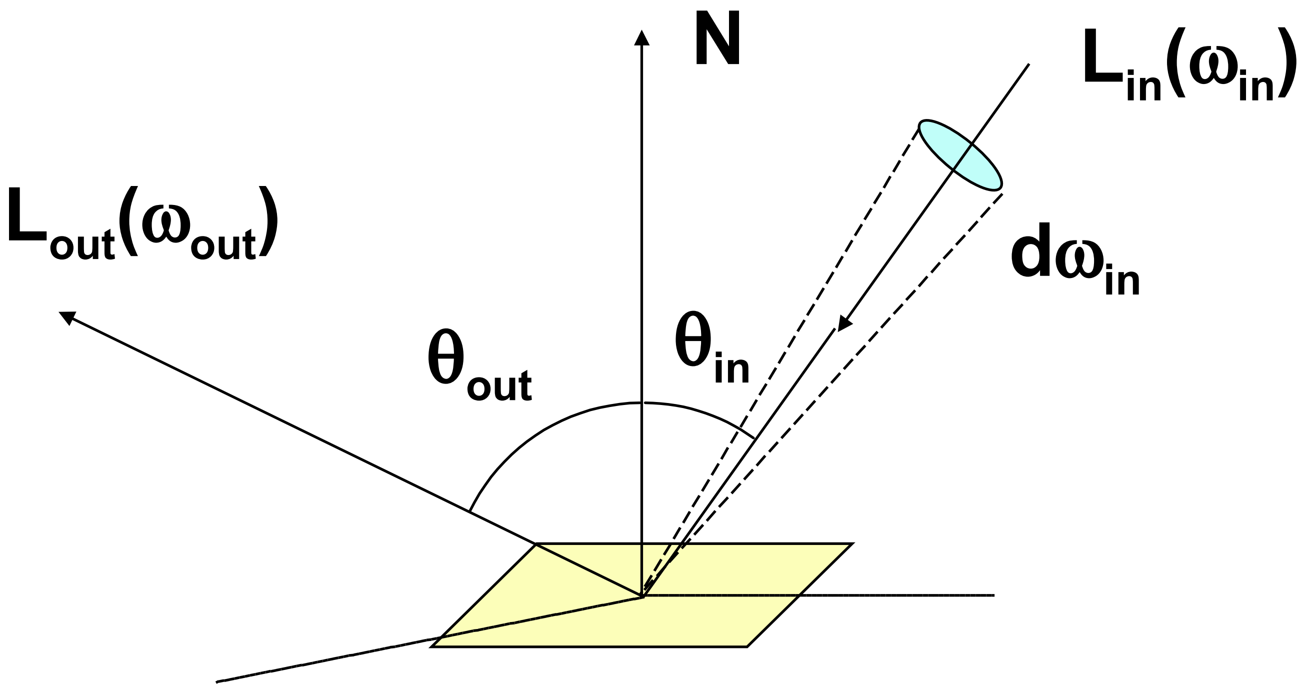


Bidir PT example

© 1995 Eric Veach, Leonidas J. Guibas

Monte Carlo in rendering: BRDF

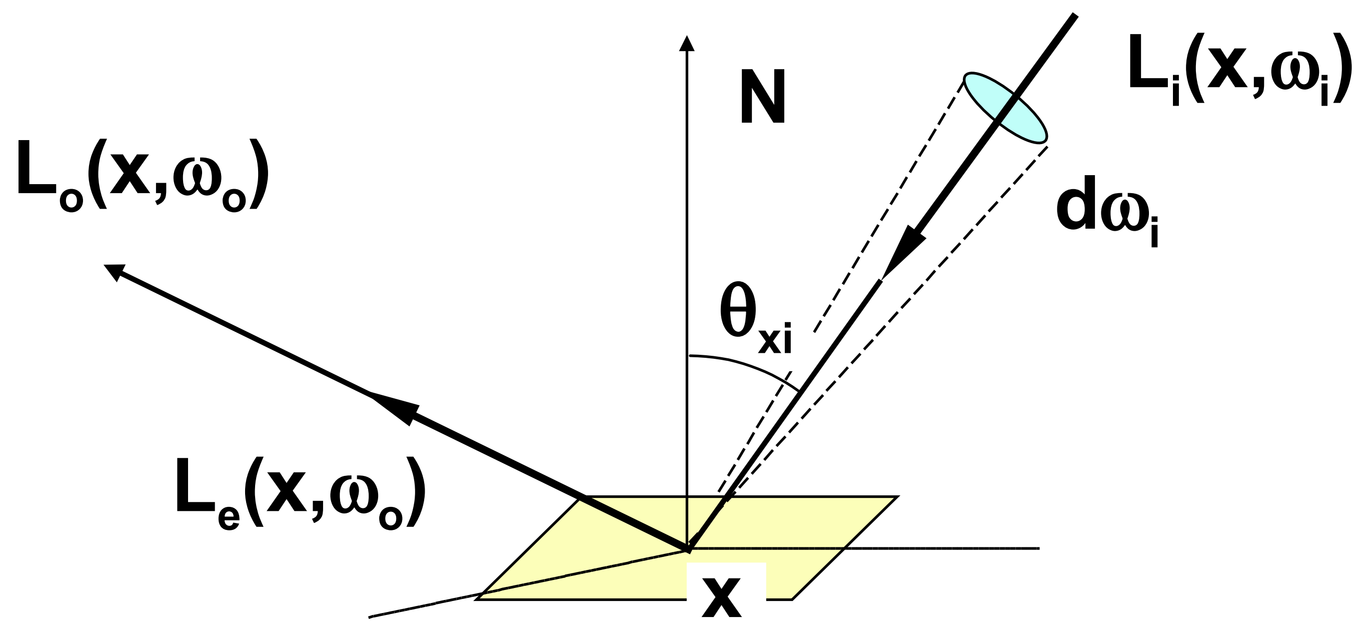
"Bidirectional Reflectance Distribution Function"



$$f(\omega_{in} \rightarrow \omega_{out}) = \frac{L_{out}(\omega_{out})}{L_{in}(\omega_{in}) \cdot \cos\theta_{in} \cdot d\omega_{in}} \quad [sr^{-1}]$$

Local rendering equation (OVTIGRE)

"Outgoing, Vacuum, Time-Invariant, Gray Radiance Equation"



own radiant exitance

$$L_o(x,\omega_o) = L_e(x,\omega_o) + \int f(x,\omega_i \to \omega_o) \cdot L_i(x,\omega_i) \cdot \cos\theta_{xi} \, d\omega_i$$

Light propagation operators

Rendering equation for **radiance** (operators):

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

$$\mathbf{L} = \mathbf{e} + \mathbf{T}\mathbf{e} + \mathbf{T}^2\mathbf{e} + \mathbf{T}^3\mathbf{e} + \ldots$$

Integral **operator** **T** can be decomposed into diffuse (**D**) and specular (**S**) components:

$$\mathbf{T} = \mathbf{D} + \mathbf{S}$$

$$\mathbf{L} = \mathbf{e} + (\mathbf{D} + \mathbf{S})\mathbf{e} + (\mathbf{D} + \mathbf{S})^2\mathbf{e} + \ldots$$

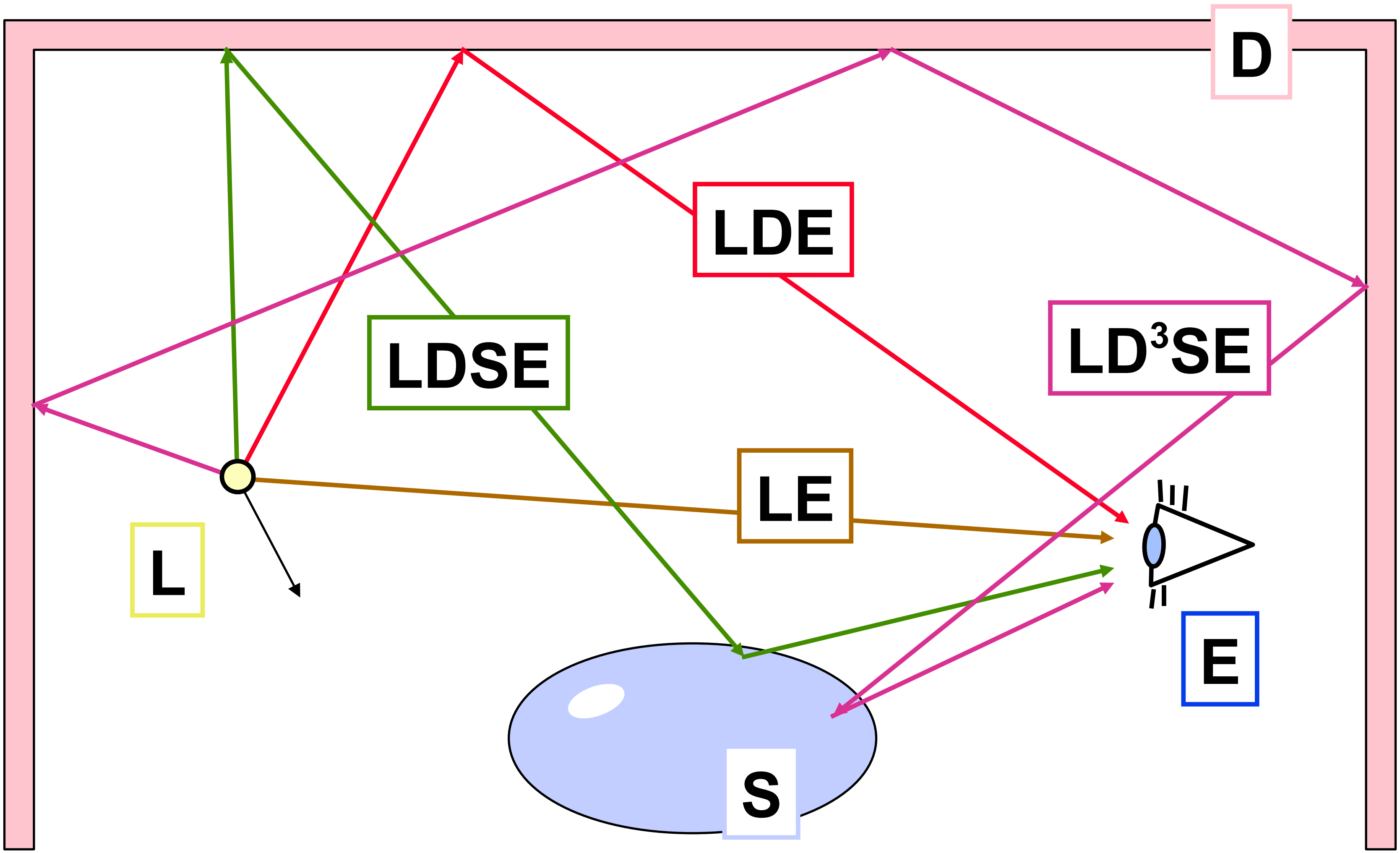
$$\mathbf{L} = \mathbf{e} + \mathbf{D}\mathbf{e} + \mathbf{S}\mathbf{e} + \mathbf{D}\mathbf{D}\mathbf{e} + \mathbf{D}\mathbf{S}\mathbf{e} + \mathbf{S}\mathbf{D}\mathbf{e} + \mathbf{S}\mathbf{S}\mathbf{e} + \ldots$$

Regular expression alphabet

- **light source L**
- **diffuse reflection D**
  - Lambertian reflection (omnidirectional)
- **specular reflection S**
  - directional reflection, highlight – directional part of a BRDF
  - idealized **mirror reflection**: $\mathbf{S_M}$
- **viewer's eye E**
  - contribution to the result image

© Josef Pelikán, http://cgg.ms.mff.cuni.cz/~pepca

Light propagation paths



Classical rendering methods I

- **shading** with highlights and **shadows** (e.g. Phong model): **L(D|S)E**
  - shadow casting is often ignored

- **recursive ray-tracing** (Whitted): **L[D|S]S_M*E**
  - the first specular reflection is accurate (reflectance model from a light source), the rest is replaced by mirror reflections

Classical rendering methods II

- **distributed ray-tracing** (Cook): **L[D]S*E**
  - all specular reflections are estimated correctly
- basic **radiosity**: **LD*E**
  - diffuse materials (reflections) only
- **all** possible light paths: **L(D|S)*E**
  - correct solution of rendering equation (Kajiya – Path tracing)

References

- **E. Lafortune**: ***Mathematical Models and Monte Carlo Algorithms for Physically Based Rendering***, PhD thesis, KU Leuven, 29-63
- **M. Kalos, P. Whitlock**: ***Monte Carlo Methods***, John Wiley & Sons, 1986, 89-116
- **A. Glassner**: ***Principles of Digital Image Synthesis***, Morgan Kaufmann, 1995, 840-864