Počítačová grafika III – Monte Carlo rendering 3

Jaroslav Křivánek, MFF UK
Jaroslav.Krivanek@mff.cuni.cz

Metody snížení rozptylu MC estimátorů

Metody snížení rozptylu

- Importance sampling
  a) Podle BRDF (nejčastější)
  b) Podle L_i (pokud známo: přímé osvětlení)
  - V syntéze obrazu je IS nejčastěji používaná metoda

- Řídící funkce (control variates)

- Lepší rozložení vzorků
  - Stratifikace
  - quasi-Monte Carlo (QMC)

Řídící funkce

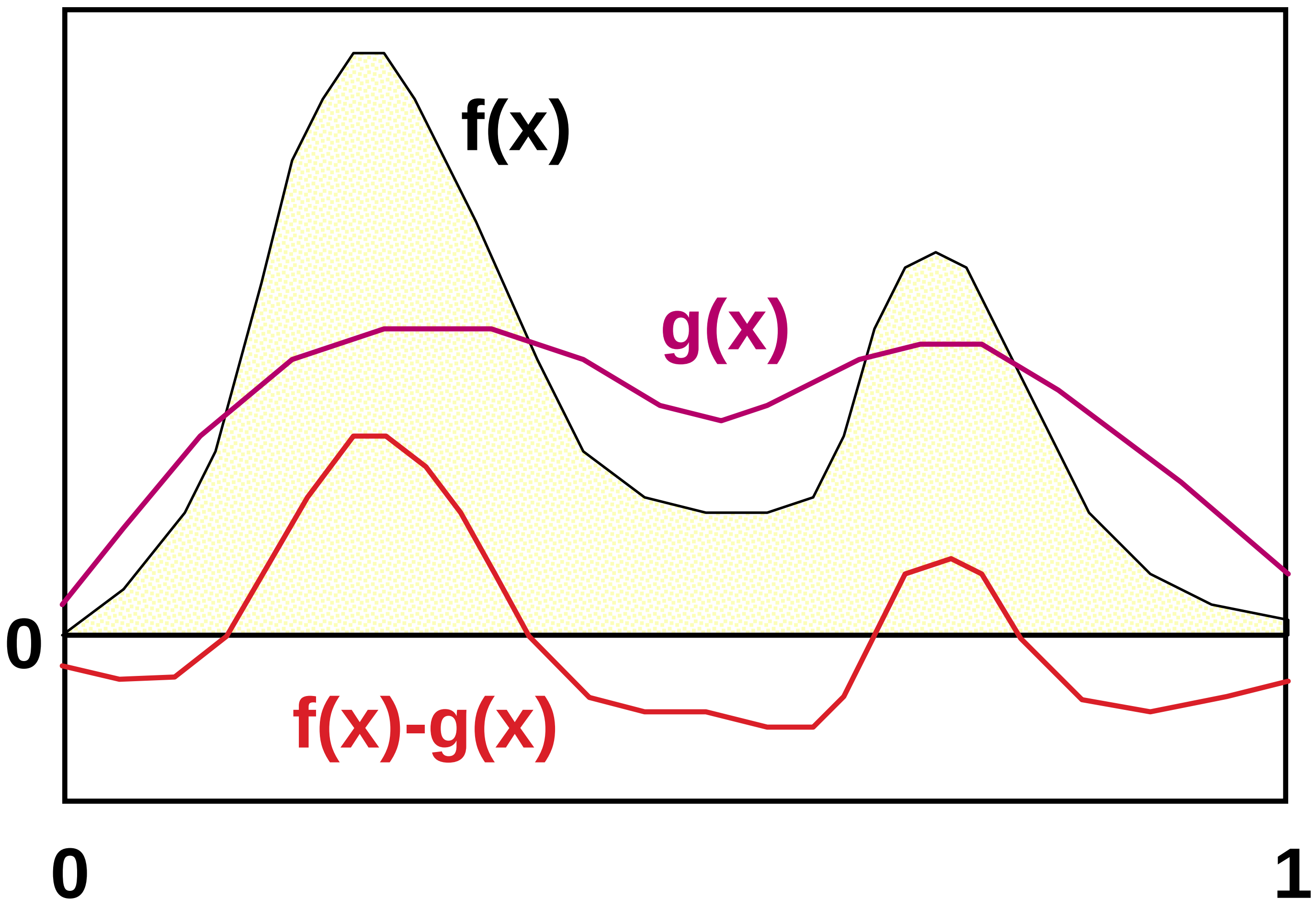
Funkce **g(x)**, která **aproximuje integrand** a dokážeme ji **analyticky zintegrovat**:

$$I = \int f(\mathbf{x})\,\mathrm{d}\mathbf{x} = \underline{\int \left[f(\mathbf{x}) - g(\mathbf{x}) \right] \mathrm{d}\mathbf{x}} + \underline{\int g(\mathbf{x})\,\mathrm{d}\mathbf{x}}$$

numerické integrování (MC)
menší rozptyl než $f(\mathbf{x})$

umíme analyticky
integrovat

Transformace řídící funkcí



Řídící funkce vs. Importance sampling

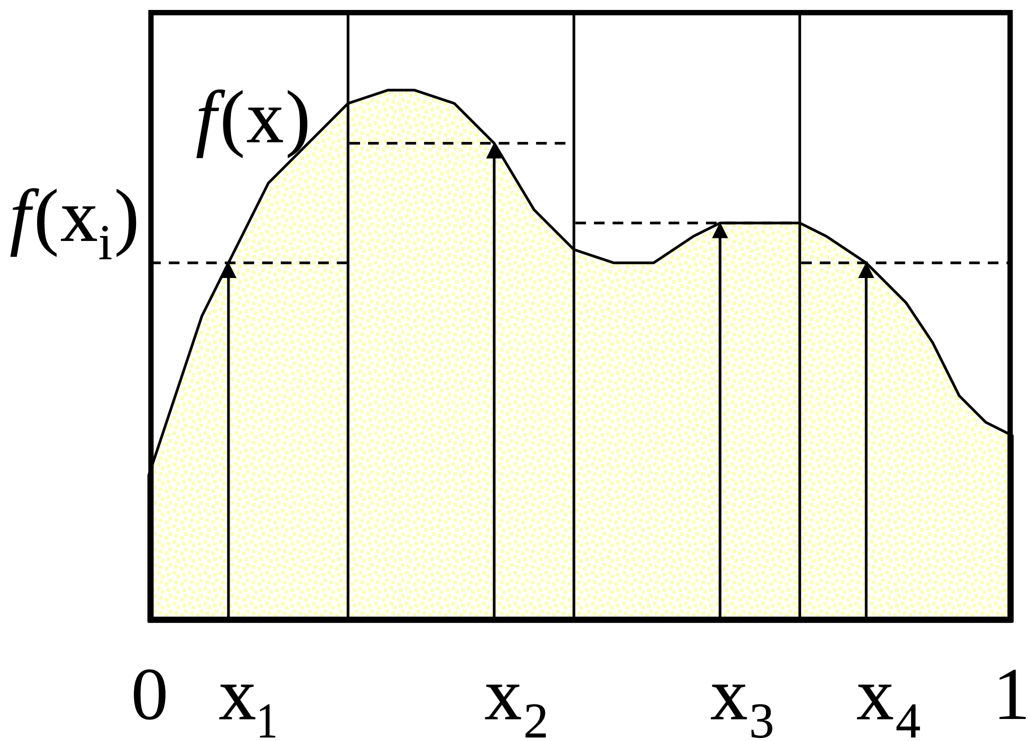
- Importance sampling
  - Lepší pokud se funkce, podle níž umíme vzorkovat, vyskytuje v integrandu jako multiplikativní člen (rovnice odrazu, zobrazovací rovnice).
- Řídící funkce
  - Lepší pokud se funkce, kterou umíme analyticky integrovat, vyskytuje v integrandu jako aditivní člen.
- Proto v se v syntéze obrazu téměř vždy používá importance sampling.

Lepší rozmístění vzorků

- Při výběru množiny nezávislých vzorků se stejnou hustotou pravděpodobnosti dochází ke shlukování
  - velký rozptyl odhadu
- Lepší rozmístění vzorků = integrační oblast je pravidelněji pokryta
  - snížení rozptylu
- Metody
  - Vzorkování po částech (stratifikace, **stratified sampling**)
  - quasi-Monte Carlo (QMC)

Vzorkování po částech

- Interval se rozdělí na části, které se odhadují samostatně



Vzorkování po částech

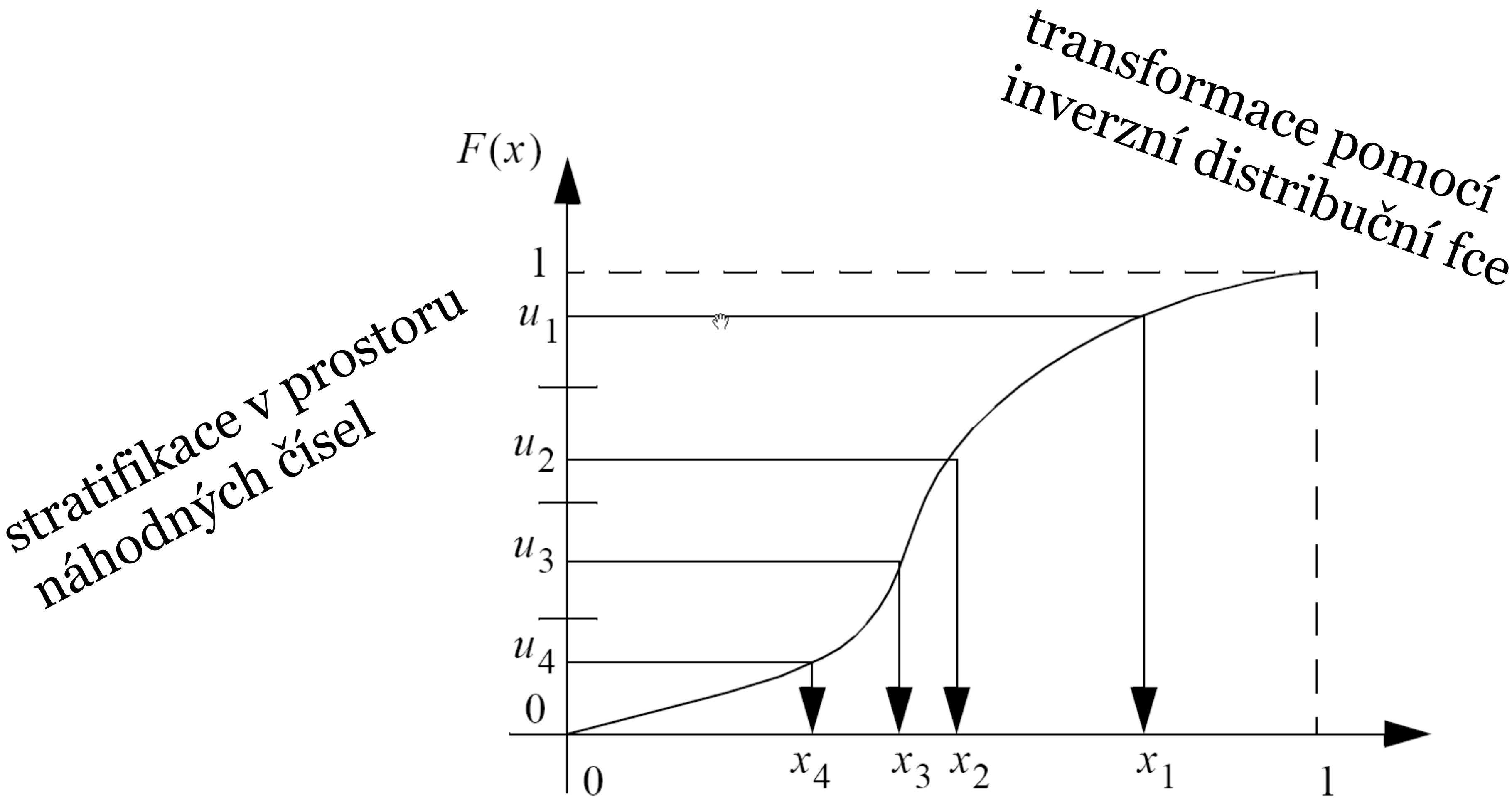
Rozdělení intervalu Ω na N částí Ω_i:

$$I = \int_{\Omega} f(x)\,\mathrm{d}x = \underline{\sum_{i=1}^{N} \int_{\Omega_i} f(x)\,\mathrm{d}x} = \sum_{i=1}^{N} I_i$$

Estimátor:

$$\hat{I}_{\text{strat}} = \underline{\frac{1}{N} \sum_{i=1}^{N} f(X_i),} \quad X_i \in \Omega_i$$

Kombinace vzorkování po částech s Importance Sampling



Vzorkování po částech

- Potlačuje shlukování vzorků
- Redukuje rozptyl odhadu
  - Rozptyl menší nebo roven rozptylu sekundárního estimátoru
- Účinné jen pro nízkou dimenzi integrandu

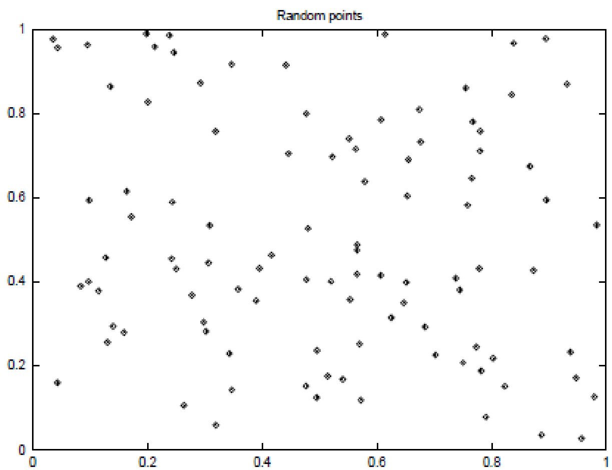
Rozklad intervalu na části

- **uniformní** rozklad intervalu **(0,1)**
  - přirozená metoda pro zcela neznámou funkci ***f***
- známe-li alespoň přibližně **průběh funkce *f***, snažíme se o takový rozklad, aby byl rozptyl funkce na subintervalech co nejmenší
- rozklad ***d*-rozměrného intervalu** vede na N^d výpočtů
  - úspornější metodou je vzorkování **"*N* věží"**

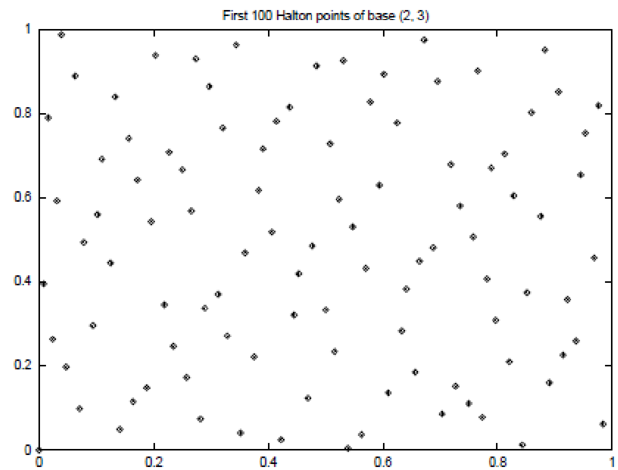
Metody Quasi Monte Carlo (QMC)

- Použití striktně deterministických sekvencí místo náhodných čísel
- Vše funguje jako v MC, důkazy se ale nemohou opírat o statistiku (nic není náhodné)
- Použité sekvence čísel s nízkou dikrepancí (**low-discrepancy sequences**)

Diskrepance



High Discrepancy
(clusters of points)



Low Discrepancy
(more uniform)

Defining discrepancy

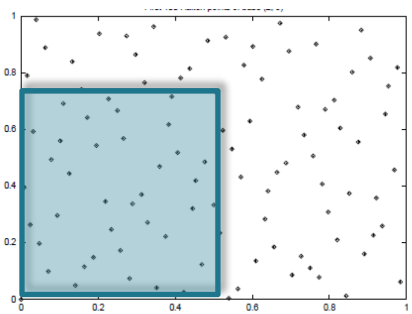
- *s*-dimensional "brick" function:

$$\mathcal{L}(\mathbf{z}) = \begin{cases} 1 \text{ if } 0 \le \mathbf{z}|_1 \le v_1, 0 \le \mathbf{z}|_2 \le v_2, \ldots, 0 \le \mathbf{z}|_s \le v_s \\ 0 \text{ otherwise.} \end{cases}$$

- True volume of the "brick" function:

$$V(A) = \prod_{j=1}^{s} v_j$$

- MC estimate of the volume of the "brick":



$$\frac{1}{N} \sum_{i=1}^{N} f(\mathbf{z}_i) = \frac{m(A)}{N}$$

total number of sample points

number of sample points that actually fell inside the "brick"

Discrepancy

- Discrepancy (of a point sequence) is the maximum possible error of the MC quadrature of the "brick" function over all possible brick shapes:

$$\boxed{\mathcal{D}^*(\mathbf{z}_1, \mathbf{z}_2, \ldots \mathbf{z}_N) = \sup_A \left| \frac{m(A)}{N} - V(A) \right|.}$$

  - serves as a measure of the uniformity of a point set
  - must converge to zero as N -> infty
  - the lower the better (cf. **Koksma-Hlawka Inequality**)

Koksma-Hlawka inequality

- Koksma-Hlawka inequality

„variation“ of *f*

$$\left|\int_{\mathbf{z}\in[0,1]^s} f(\mathbf{z})\,d\mathbf{z} - \frac{1}{N}\sum_{i=1}^{N} f(\mathbf{z}_i)\right| \le \mathcal{V}_{\mathrm{HK}} \cdot \mathcal{D}^*(\mathbf{z}_1, \mathbf{z}_2, \ldots \mathbf{z}_N)$$

- the KH inequality only applies to *f* with finite variation
- QMC can still be applied even if the variation of f is infinite

Van der Corput Sequence

- b ... **base**, must be relative prime (2,3,5,7,....)
- radical inverse

$$\begin{aligned} \Phi_b : \mathbb{N}_0 &\rightarrow \mathbb{Q} \cap [0,1) \\ i = \sum_{j=0}^{\infty} a_j(i) b^j &\mapsto \Phi_b(i) := \sum_{j=0}^{\infty} a_j(i) b^{-j-1} \end{aligned}$$

Van der Corput Sequence (base 2)

i	binary form of i	radical inverse	H_i
1	1	0.1	0.5
2	10	0.01	0.25
3	11	0.11	0.75
4	100	0.001	0.125
5	101	0.101	0.625
6	110	0.011	0.375
7	111	0.111	0.875

- point placed in the middle of the interval
- then the interval is divided in half
- has low-discrepancy

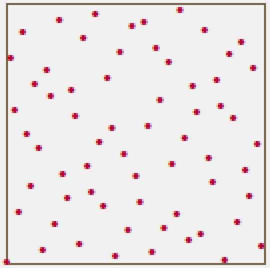
Table credit: Laszlo Szirmay-Kalos

Van der Corput Sequence (base b)

```
double RadicalInverse(const int Base, int i)
{
        double Digit, Radical, Inverse;
        Digit = Radical = 1.0 / (double) Base;
        Inverse = 0.0;
        while(i)
        {
                Inverse += Digit * (double) (i % Base);
                Digit *= Radical;
                i /= Base;
        }
        return Inverse;
}
```

Radical inversion based points in higher dimension

Halton sequence $x_i := \left(\Phi_{b_1}(i), \dots, \Phi_{b_s}(i)\right)$ where b_i is the i-th prime number



Hammersley point set $x_i := \left(\frac{i}{n}, \Phi_{b_1}(i), \dots, \Phi_{b_{s-1}}(i)\right)$

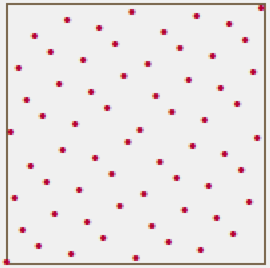


Image credit: Alexander Keller

Transformace náhodných čísel

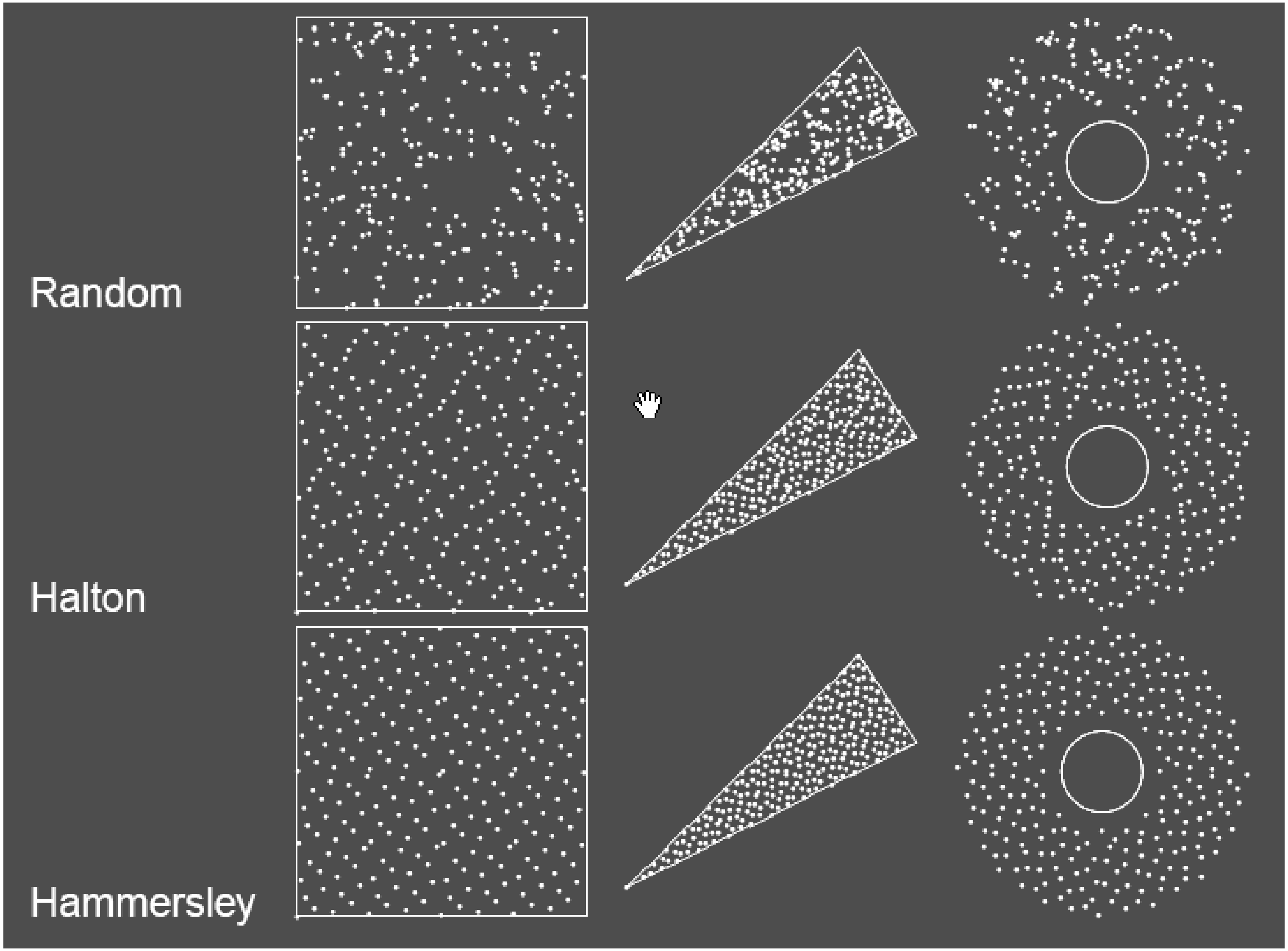


Image credit: Alexander Keller

Ukázka výsledků pro MC a QMC

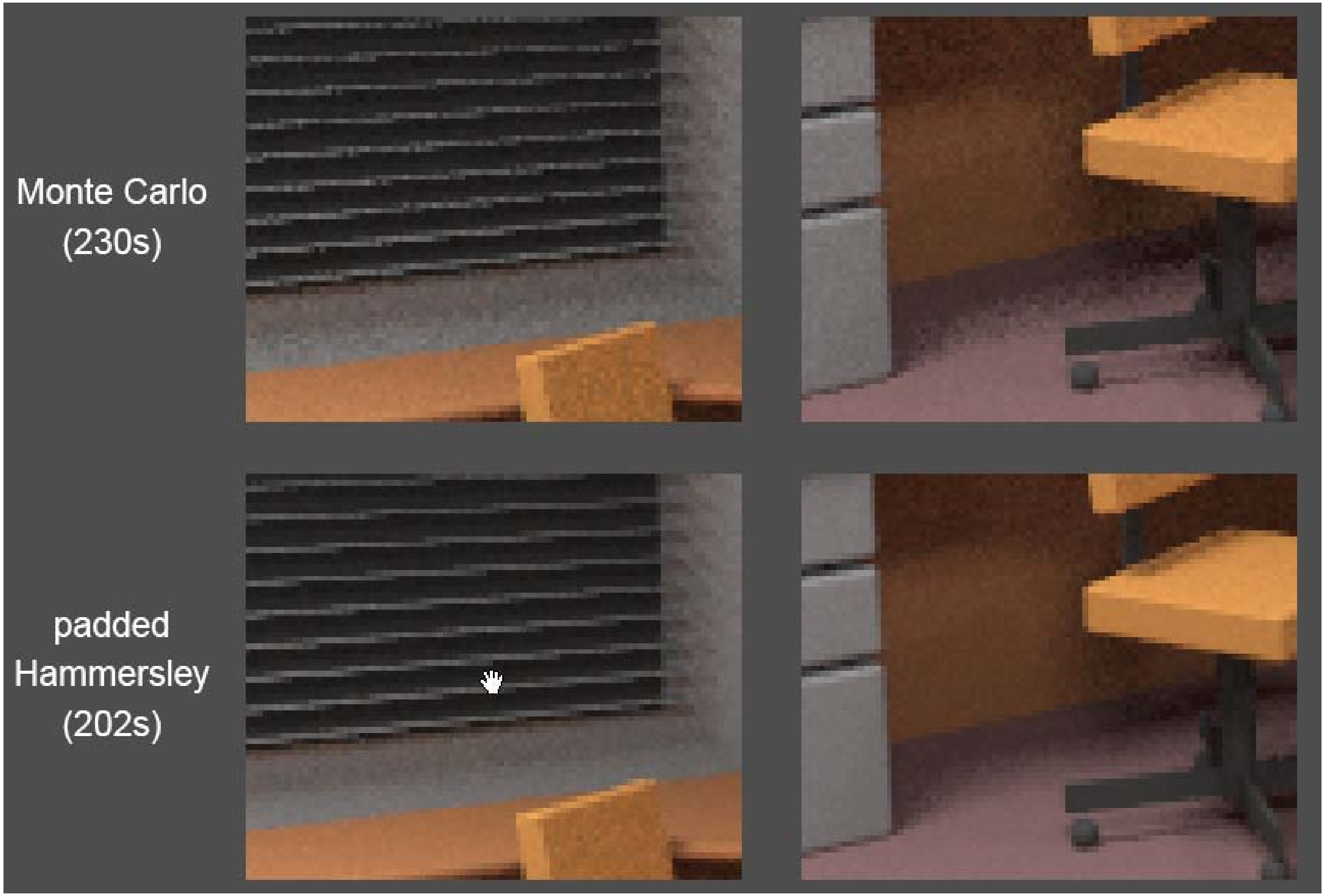


Image credit: Alexander Keller

Přímé osvětlení

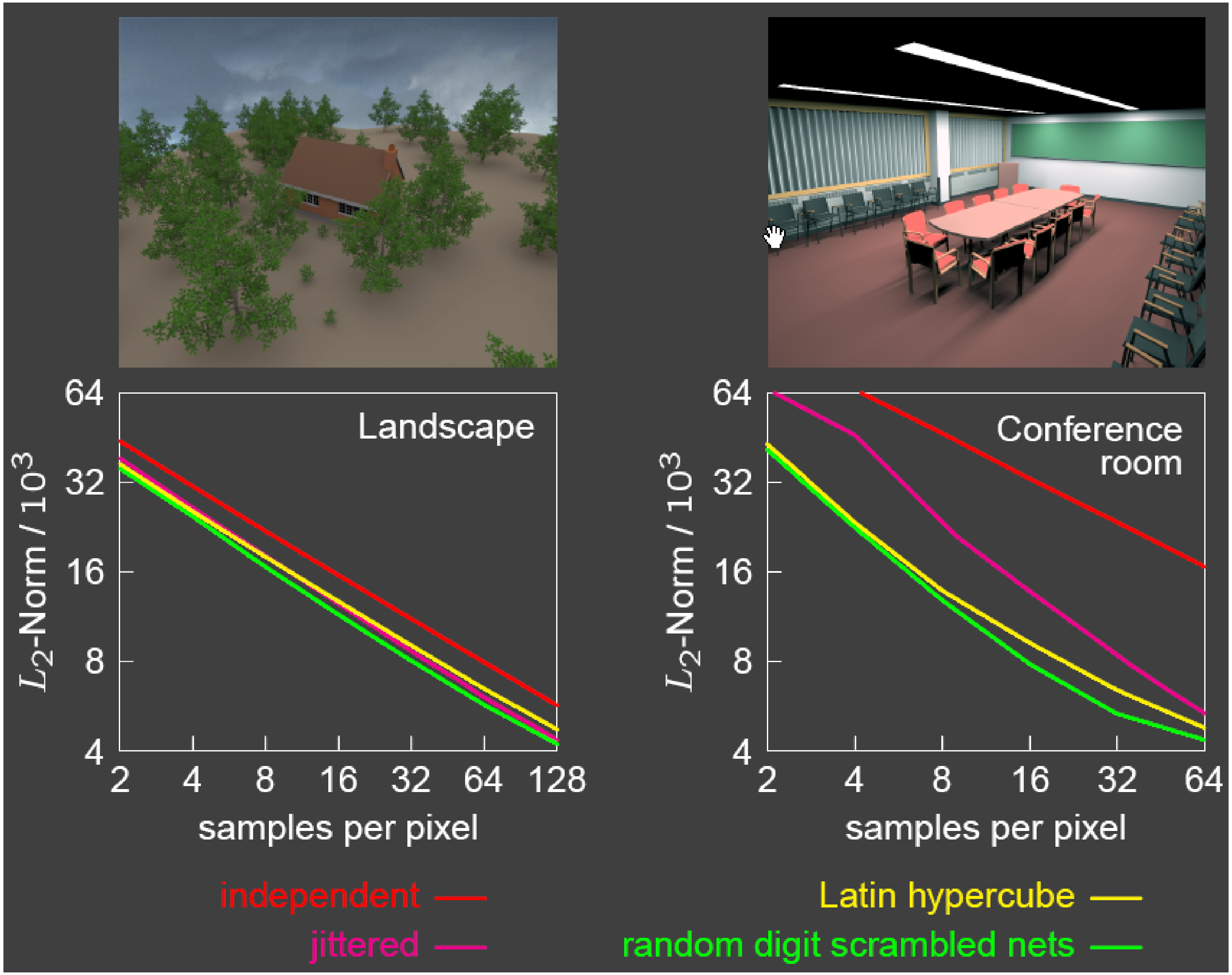
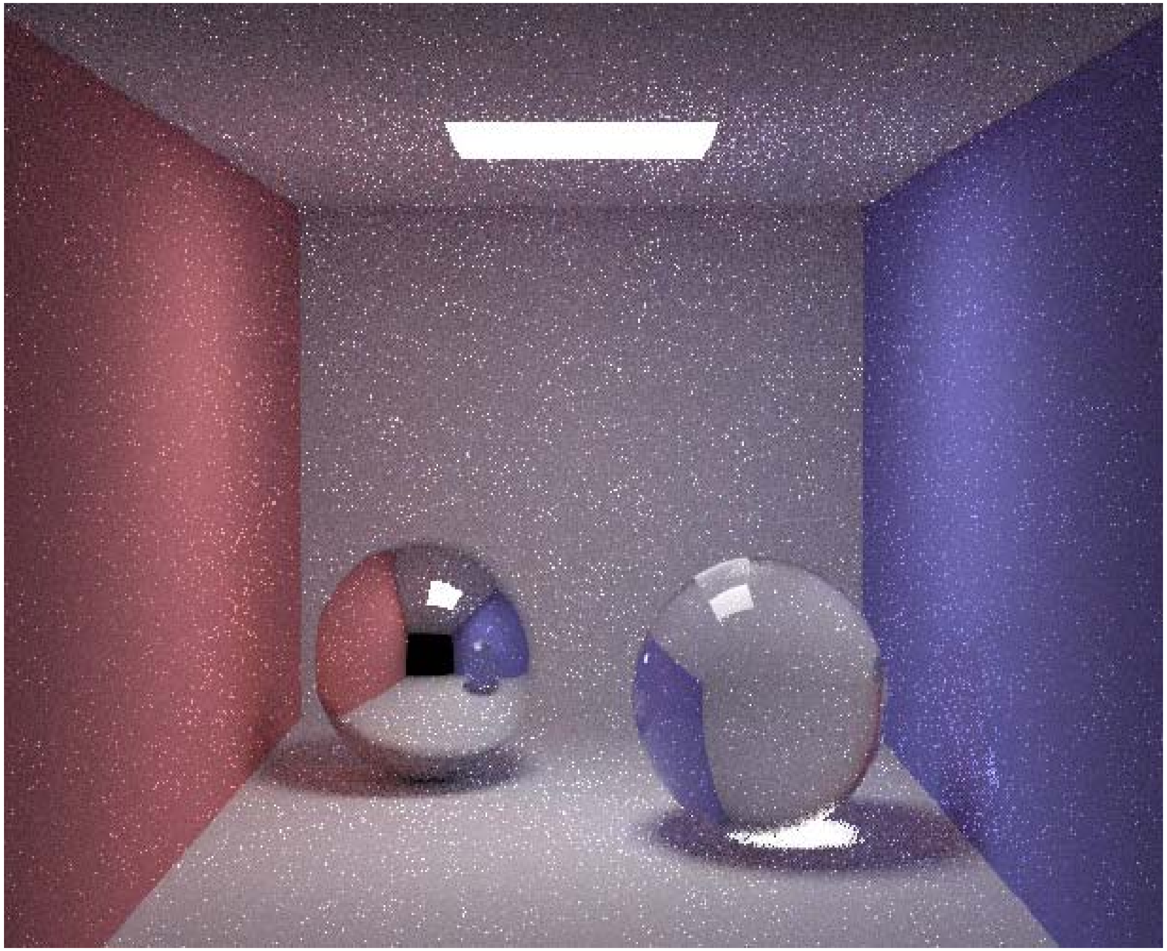


Image credit: Alexander Keller

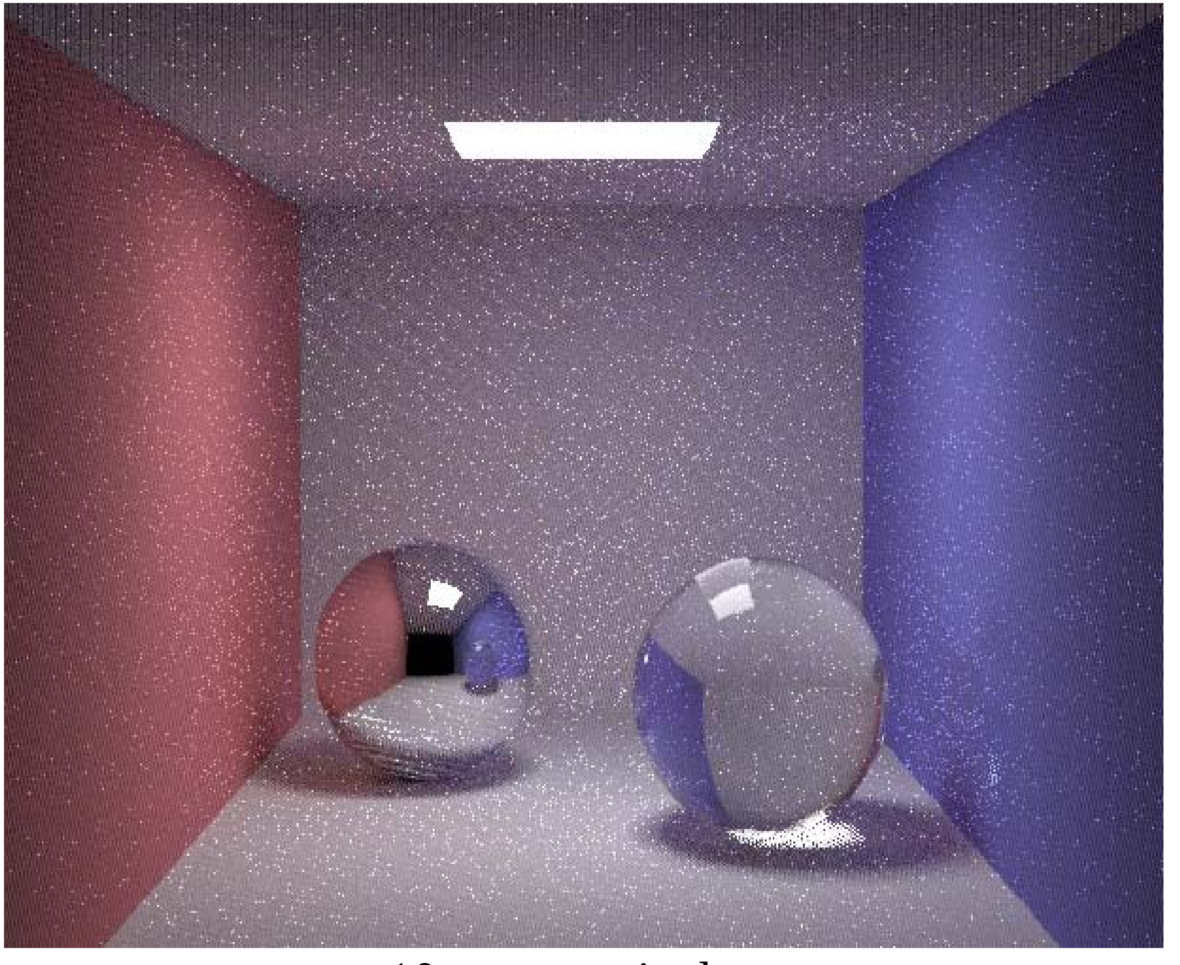
Stratified sampling



Henrik Wann Jensen

10 cest na pixel

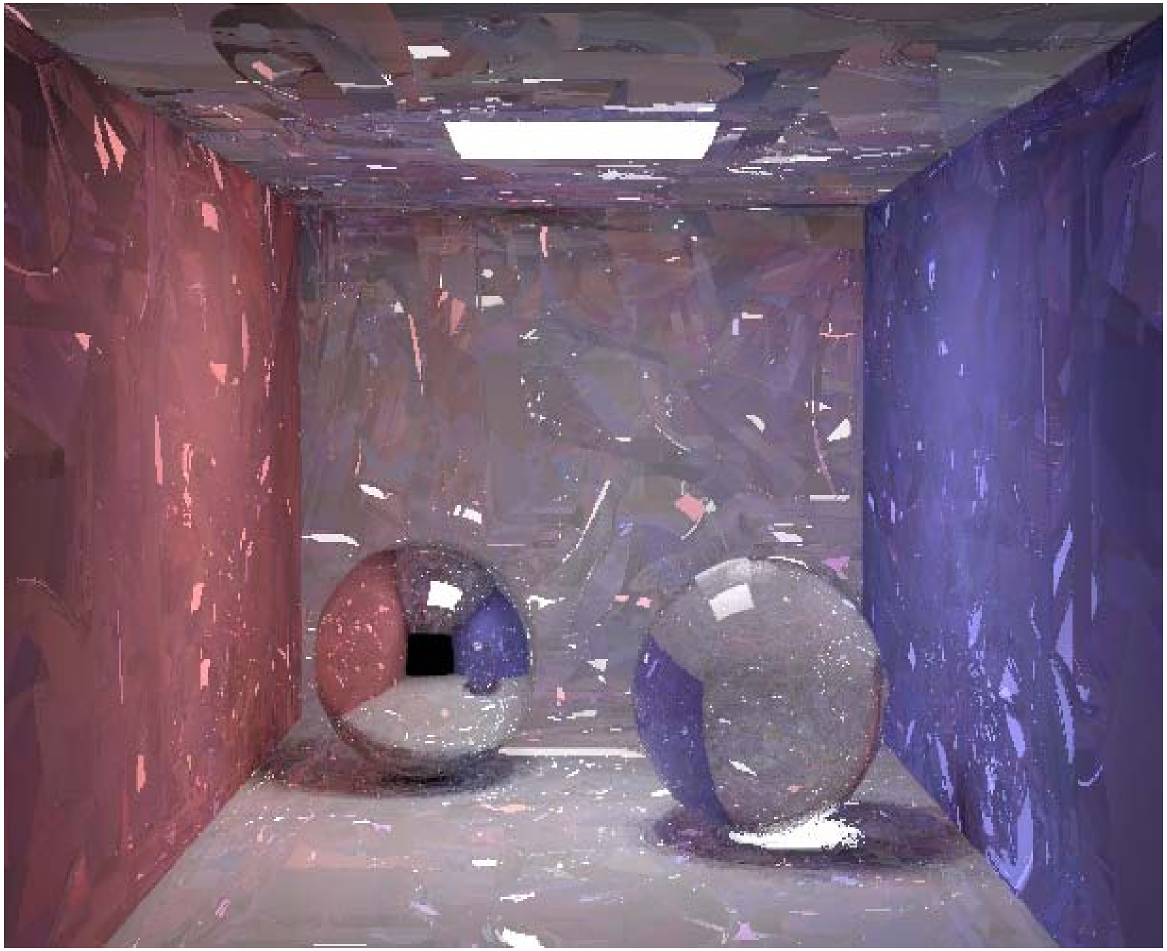
Quasi-Monte Carlo



Henrik Wann Jensen

10 cest na pixel

Fixní náhodná sekvence



10 cest na pixel

Henrik Wann Jensen

Metody Quasi Monte Carlo (QMC)

- Nevýhody QMC:
  - V obrázku mohou vzniknout viditelné „vzory“ (místo šumu v MC)

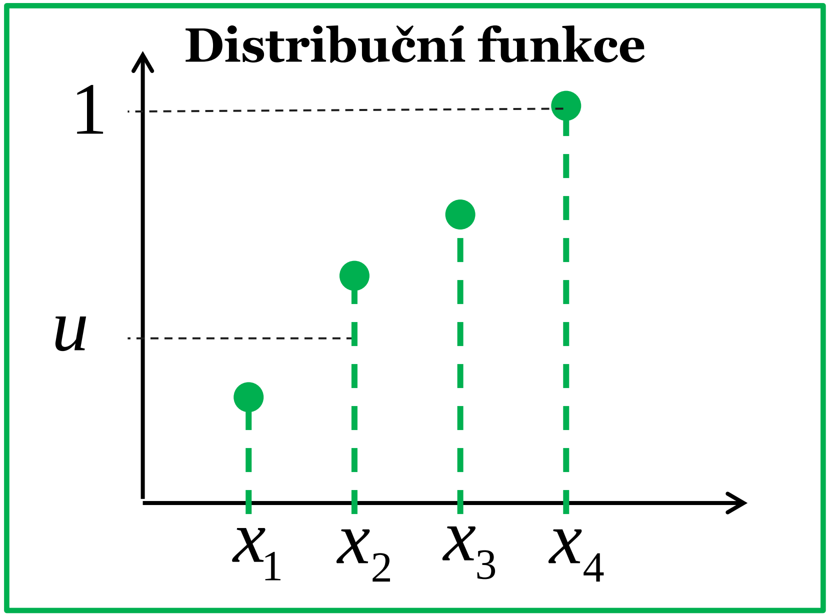
Generování vzorků z distribuce

1D diskrétní náhodná veličina

- Dána p-nostní fce $p(i)$, distribuční fce $P(i)$
- Postup
  1. Vygeneruj u z $R(0,1)$
  2. Vyber x_i pro které

$$\boxed{P(i-1) < u \leq P(i)}$$

(definujeme $P(0) = 0$)



- Nalezení i se provádí půlením intervalu

2D diskrétní náhodná veličina

- Dána p-nostní fce $p(i, j)$
- Možnost 1:
  - Interpretovat jako 1D vektor pravděpodobností
  - Vzorkovat jako 1D distribuci

2D diskrétní náhodná veličina

- Možnost 2 (lepší)
  1. „Řádek" i_{sel} vybrat podle marginálního rozdělení, popsaného 1D p-nostní fcí

$$p_I(i) = \sum_{j=1}^{n_j} p(i, j)$$

  2. „Sloupec" j_{sel} vybrat podle podmíněného rozdělení příslušejícího vybranému „řádku" i_{sel}

$$p_J(j \mid I = i_{\text{sel}}) = \frac{p(i_{\text{sel}}, j)}{p_I(i_{\text{sel}})}$$

Vzorkování směrů podle mapy prostředí

- Intenzita mapy prostředí definuje hustotu (pdf) na jednotkové kouli

- Pro účely vzorkování ji aproximujeme jako 2D diskrétní distribuci nad pixely mapy

- Pravděpodobnost výběru pixelu je dána součinem
  - Intenzity pixelu
  - Velikostí pixelu na jednotkové kouli (závisí na mapování)

- Detaily viz. writeup

Vzorkování 1D spojité náhodné veličiny

- Transformací rovnoměrné náhodné veličiny

- Zamítací metoda (rejection sampling)

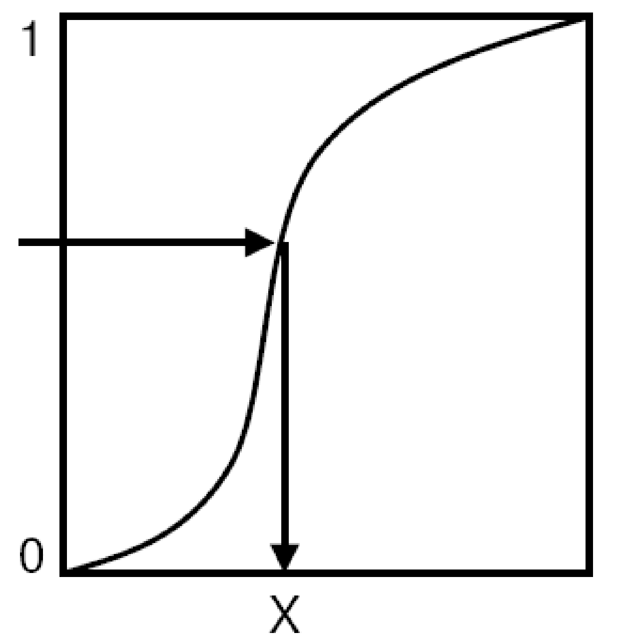
Vzorkování 1D spojité náhodné veličiny transformací

- Je-li U je náhodná veličina s rozdělením R(0,1), pak náhodná veličina X

$$X = P^{-1}(U)$$

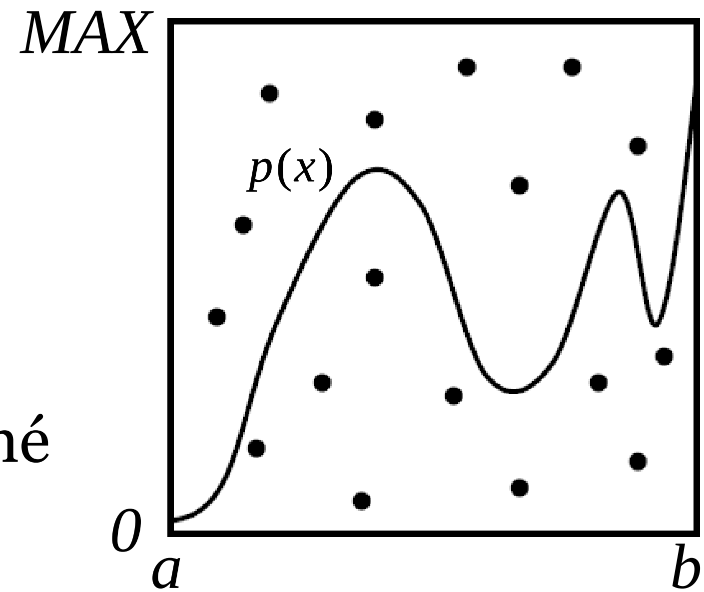
  má rozdělení popsané distribuční funkcí P.

- Pro generování vzorků podle hustoty p potřebujeme
  - Spočítat cdf $P(x)$ z pdf $p(x)$
  - Spočítat inverzní funkci $P^{-1}(x)$



Vzorkování 1D spojité náhodné veličiny zamítací metodou

- Algoritmus
  - Vyber náhodné u_1 z $R(a, b)$
  - Vyber náhodné a u_2 z $R(0, MAX)$
  - Přijmi vzorek, pokud $p(u_1) < u_2$



- Přijaté vzorky mají rozložení dané hustotou $p(x)$

- Účinnost = % přijatých vzorků
  - Plocha funkce pod křivkou / plocha obdélníka
  - Transformační metoda vždy efektivnější (ale vyžaduje integrovat hustotu a invertovat distribuční fci)

Vzorkování 2D spojité náhodné veličiny

- Jako pro 2D diskrétní veličinu
- Dána hustota $p(x, y)$
- Postup

1. Vyber x_{sel} z marginální hustoty

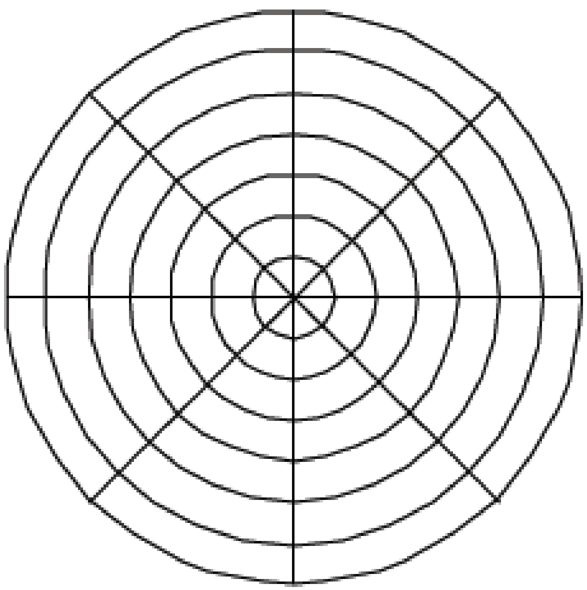
$$p_X(x) = \int p(x, y)\,\mathrm{d}y$$

2. Vyber y_{sel} z podmíněné hustoty

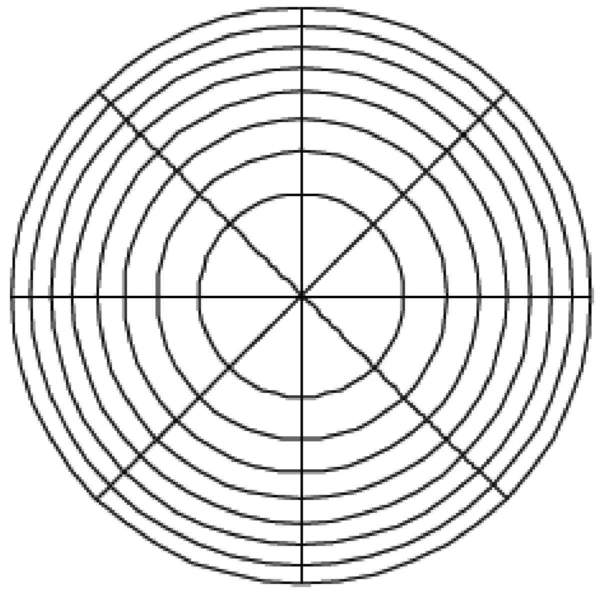
$$p_Y(y \mid X = x_{\mathrm{sel}}) = \frac{p(x_{\mathrm{sel}}, y)}{p_X(x_{\mathrm{sel}})}$$

Sampling a Circle

WRONG ≠ Equi-Areal | **RIGHT = Equi-Areal**



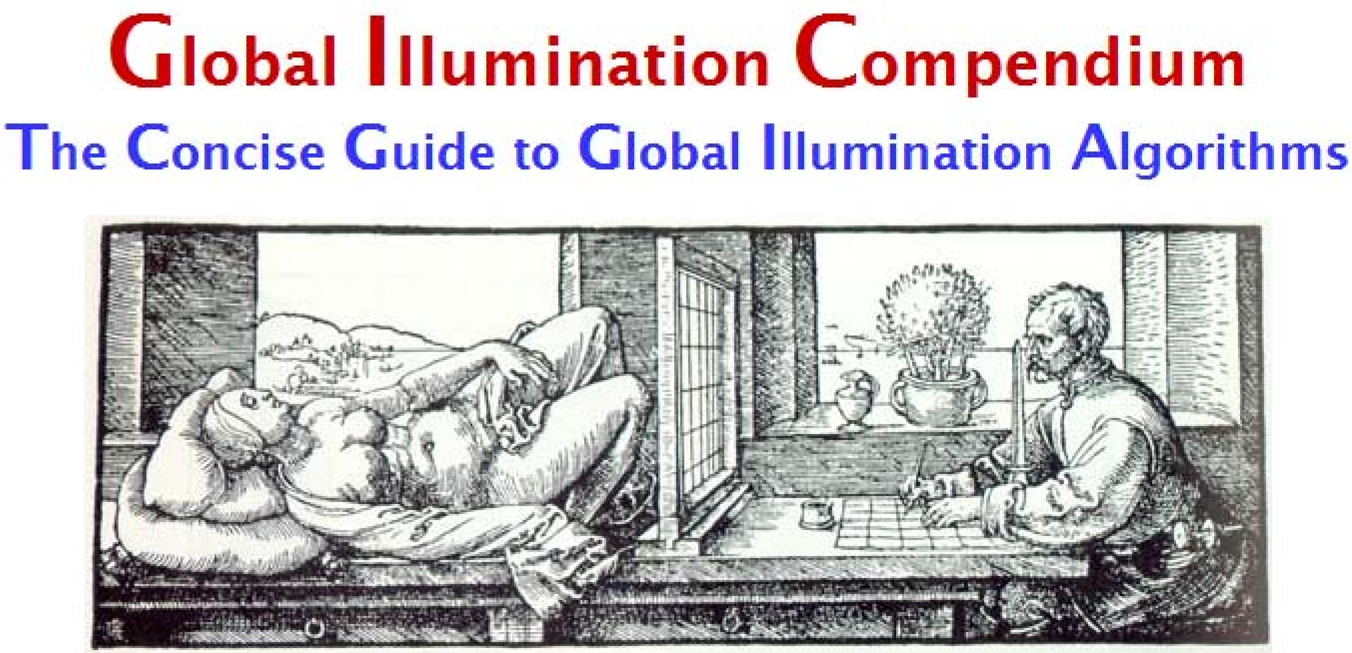
$$\theta = 2\pi U_1$$
$$r = U_2$$



$$\theta = 2\pi U_1$$
$$r = \sqrt{U_2}$$

Transformační vzorce

- P. Dutré: **Global Illumination Compendium**, http://people.cs.kuleuven.be/~philip.dutre/GI/



Albrecht Duerer, *Underweysung der Messung mit dem Zirkel und Richtscheyt* (Nurenberg, 1525), Book 3, figure 67.

Importance sampling Phongovy BRDF

- Paprsek dopadne na plochu s Phongovou BRDF. Jak vygenerovat sekundární paprsek pro vzorkování nepřímého osvětlení?

- Path tracing
  - Pouze 1 sekundární paprsek – je třeba zvolit komponentu BRDF (druh interakce)
  - Postup:
    1. Vyber komponentu BRDF (difúzní odraz / lesklý odraz / lom)
    2. Vzorkuj vybranou komponentu

Fyzikálně věrohodná Phongova BRDF

$$f_r^{\text{Phong}}(\omega_\text{i} \to \omega_\text{o}) = \frac{\rho_\text{d}}{\pi} + \frac{n+2}{2\pi}\rho_\text{s}\cos^n\theta_\text{r}$$

- Kde:

$$\cos\theta_\text{r} = \omega_\text{o} \cdot \omega_\text{r}$$

$$\omega_\text{r} = 2(\omega_\text{i} \cdot \mathbf{n})\mathbf{n} - \omega_\text{i}$$

- Zachování energie: $\rho_d + \rho_s \leq 1$

Výběr interakce

```
pd = max(rho_d.r, rho_d.g, rho_d.b); // prob. of diffuse
ps = max(rho_s.r, rho_s.g, rho_s.b); // prob. of specular

u = rand(0, pd + ps);

Vec3 dir, float pdf, Col3 brdfVal;
if (u <= pd)
  {dir, pdf, brdfVal} = sampleDiffuse();
  return {dir, pdf * pd, brdfVal}

else // if (u <= pd+ps)
  {dir, pdf, brdfVal} = sampleSpecular();
  return {dir, pdf * ps, brdfVal}
```

Vzorkování difúzního odrazu

- Importance sampling s hustotou $p(\theta) = \cos(\theta) / \pi$
  - θ…úhel mezi normálou a vygenerovaným sekundárním paprskem
  - Generování směru:

$$\varphi = 2\pi r_1 \qquad \theta = \operatorname{acos}(\sqrt{r_2})$$

$$x = \cos(2\pi r_1)\sqrt{1-r_2}$$
$$y = \sin(2\pi r_1)\sqrt{1-r_2}$$
$$z = \sqrt{r_2}$$

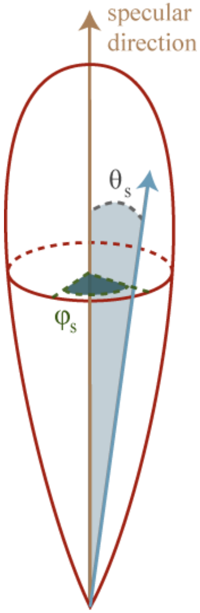
    - r1, r2 … uniformní na <0,1>
  - Zdroj: Dutre, Global illumination Compendium (on-line)
  - Odvození: Pharr & Huphreys, PBRT

sampleDiffuse()

```
// build the local coordinate frame with N = z-axis
Vec3 U = arbitraryNormal(N);  // U is perpendicular to the normal N
Vec3 V = crossProd(N, U);     // orthonormal base with N and U

// generate direction in the local coordinate frame
float r1 = rand(0,1), r2 = rand(0,1);
float sin_theta = sqrt(1 – r1);
float cos_theta = sqrt(r1);
float phi       = 2.0*PI*r2;
float pdf       = cos_theta/PI;
// to Cartesian coordinates
Vec3 ldir (cos(phi)*sin_theta, sin(phi)*sin_theta, cos_theta);

// transform to global coordinate frame
Vec3 gdir = ldir.x * U + ldir.y * V + ldir.z * Z

// evaluate BRDF component
Col brdfVal = rho_d / PI;

return {gdir, pdf, brdfVal}
```

Vzorkování lesklého odrazu

- Importance sampling s hustotou $p(\theta) = (n+1)/(2\pi) \cos^n(\theta)$
  - θ…úhel mezi ideálně zrcadlově odraženým ω_o a vygenerovaným sekundárním paprskem



Generován

$$\varphi = 2\pi r_1$$

$$\theta = \operatorname{acos}\left(r_2^{\frac{1}{n+1}} \right)$$

$$x = \cos(2\pi r_1)\sqrt{1 - r_2^{\frac{2}{n+1}}}$$

$$y = \sin(2\pi r_1)\sqrt{1 - r_2^{\frac{2}{n+1}}}$$

$$z = r_2^{\frac{1}{n+1}}$$

- r1, r2 … uniformní na <0,1>

SampleSpecular()

```
// build the local coordinate frame with R = z-axis
Vec3 R = 2*dot(N,wi)*N – wi;      // ideal reflected dir
Vec3 U = arbitraryNormal(R);      // U is perpendicular to R
Vec3 V = crossProd(R, U);         // orthonormal base with R and U

// generate direction in local coordinate frame
{Vec3 ldir, float pdf} = rndHemiCosN (n);  // formulas form prev. slide

// transform to global coordinate frame
Vec3 gdir = ldir.x * U + ldir.y * V + ldir.z * R

// reject if direction under the tangent plane
float cos_theta_i = dot(N, gdir);
if(cos_theta_i <= 0)  return {gdir, pdf, Col3(0)};

// evaluate BRDF component
Col brdfVal = rho_s * (n+2)/(M_PI*2) * pow(dir.z, n);//dir.z=cos_theta_r

return {gdir, pdf, brdfVal}
```

Alternativní strategie pro výběr komponenty BRDF

- Předchozí příklad vybere komponentu podle odrazivosti ρ

- Druhá možnost (embree)
  1. Vyber směr podle každé BRDF komponenty
  2. Vyber komponentu s p-ností danou hodnotou BRDF komponenty ve vygenerovaném směru

  - Nepotřebuje odrazivosti ρ komponenty
  - Může být neefektivní pro mnoho BRDF komponent