B4M39RSO

Integrace a syntéza obrazu pomocí metody Monte Carlo

Vlastimil Havran, ČVUT v Praze
havran@fel.cvut.cz

Osnova

- Historie
- Výpočet integrálu metodou Monte Carlo
- Aplikace v syntéze obrazu
  - Antialiasing
  - Zobrazovací rovnice
  - Přímé osvětlení z plošných zdrojů světla
  - Form-faktory v radiozitě
- Vlastnosti metody Monte Carlo
- Postupy zefektivnění metody Monte Carlo
- Metody Quasi-Monte Carlo

Historie Monte Carlo (MC)

Buffon Needle Teorém – 18-té století

Vývoj atomové bomby, Los Alamos 1940, John von Neumann a Stanislav Ulam

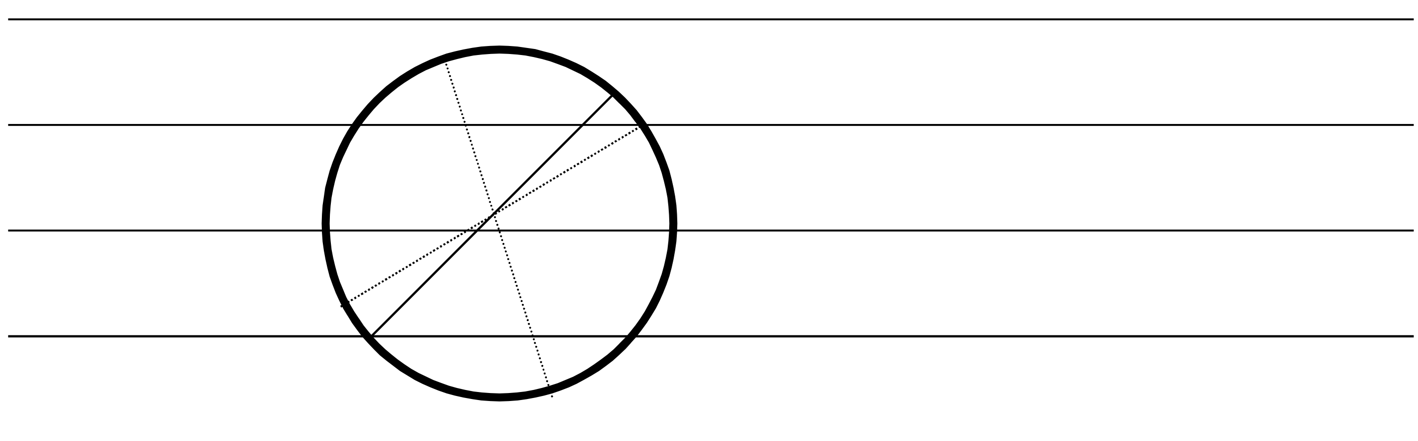
Rozvoj a aplikace metod od roku 1949 s využítím a rozvojem počítačů

“Buffon Needle” teorém

- Pravděpodobnost, že hozená jehla o délce ***L*** protne systém rovnoběžek, které jsou od sebe vzdáleny ***d*** je:

  $\boldsymbol{p} = 2*\boldsymbol{d}/(\boldsymbol{L}*\pi)$

- dá se užít k experimentálnímu zjištění hodnoty π



Metoda Monte Carlo

- Efektivní aplikace metody je podmíněna existencí výpočetní techniky
- Řešený problém musí mít hromadný charakter a je vyhodnocen s použitím statistických metod.
- V podstatě se jedná o diskrétní simulaci a to jak diskrétních tak i spojitých jevů – simuluje se mnoho případů daného děje, například:
  - Neurony - vznik, zánik, srážky s atomy vodíku
  - Úlohy hromadné obsluhy – chování počítačových sítí, dopravní situace
  - Sociologické a ekonomické modely – demografie, vývoj inflace, pojišťovnictví atd.

Integrování funkcí neurčených analyticky

- Kvadraturní metody výpočtu integrálu – Simpsonova metoda, vzorky na doméně integrandu jsou rozmístěny uniformně nezávisle na dimenzi problému. Kvadraturní metody integrace NEJSOU metody Monte Carlo!
- Metody Monte Carlo – vzorky jsou rozmístěny náhodně (nebo pseudonáhodně, pokud není k dispozici fyzikální generátor náhodných čísel)
- Speciální metody pro vzorkování domény integrandu, například Quasi-Monte Carlo, Randomized Quasi-Monte Carlo – umožňují rychlejší konvergenci = pro stejnou přesnost výsledku potřebujeme menší počet vzorků

Rovnice odrazu - zopakování

- Rovnice odrazu/reflectance equation *versus* zobrazovací rovnice/rendering equation
- “Kolik světla je odraženo do směru ω_o, je-li bod **x** osvětlen ze všech příchozích směrů?“
- Integrál přes všechny příchozí směry
  - hemisféra *H*(**x**) – pouze odraz nebo lom
  - sféra Ω – lom i odraz
- Pro výpočet obrazu rovnice vypočítávána rekurzivně

$$\boxed{L_o(\mathbf{x},\omega_o) = \underbrace{L_e(\mathbf{x},\omega_o)}_{\text{emise}} + \underbrace{\int_\Omega L_i(\mathbf{x},\omega_i) f_r(\omega_i,\mathbf{x},\omega_o)\cos\theta_i d\omega_i}_{\text{odraz + lom}}}$$

Diskrétní versus spojitá úloha

- Je zobrazovací úloha skutečně definována spojitě?
- 100 wattová žárovka emituje cca. 2.4×10^{19} fotonů za sekundu při 8 procentní účinnosti žárovky (Max-Planckův zákon E=hf=hc/lambda), Planckova konstanta $h = 6.626 \times 10^{-34}$ J.s
- Lidská schopnost rozlišit světlo:
  - Časově 25-100 Herzů
  - Logaritmická závislost pro okolní úroveň osvětlení: Weber-Fechnerův zákon
  - Extrémně široký rozsah vnímaných intensit
  - Práh citlivosti asi 10 fotonů na jednu tyčinku
- Z toho plynou nároky na počet částic pro výpočet osvětlení v běžné scéně interiéru na minimálně 10^{18} fotonů, pokud by mělo jít o skutečnou simulaci reality
- Z hlediska možné přesnosti a uskutečnitelnosti výpočtu lze zadání a řešení úlohy považovat za spojité

Aplikace Monte Carlo pro syntézu obrazu - Antialiasing

Potřebujeme ne jeden vzorek skrz středy pixelů obrázku, ale jako v reálné kameře integrujeme příchozí zář [$W.m^{-2}.sr^{-1}$] přes plochu celého pixelu:

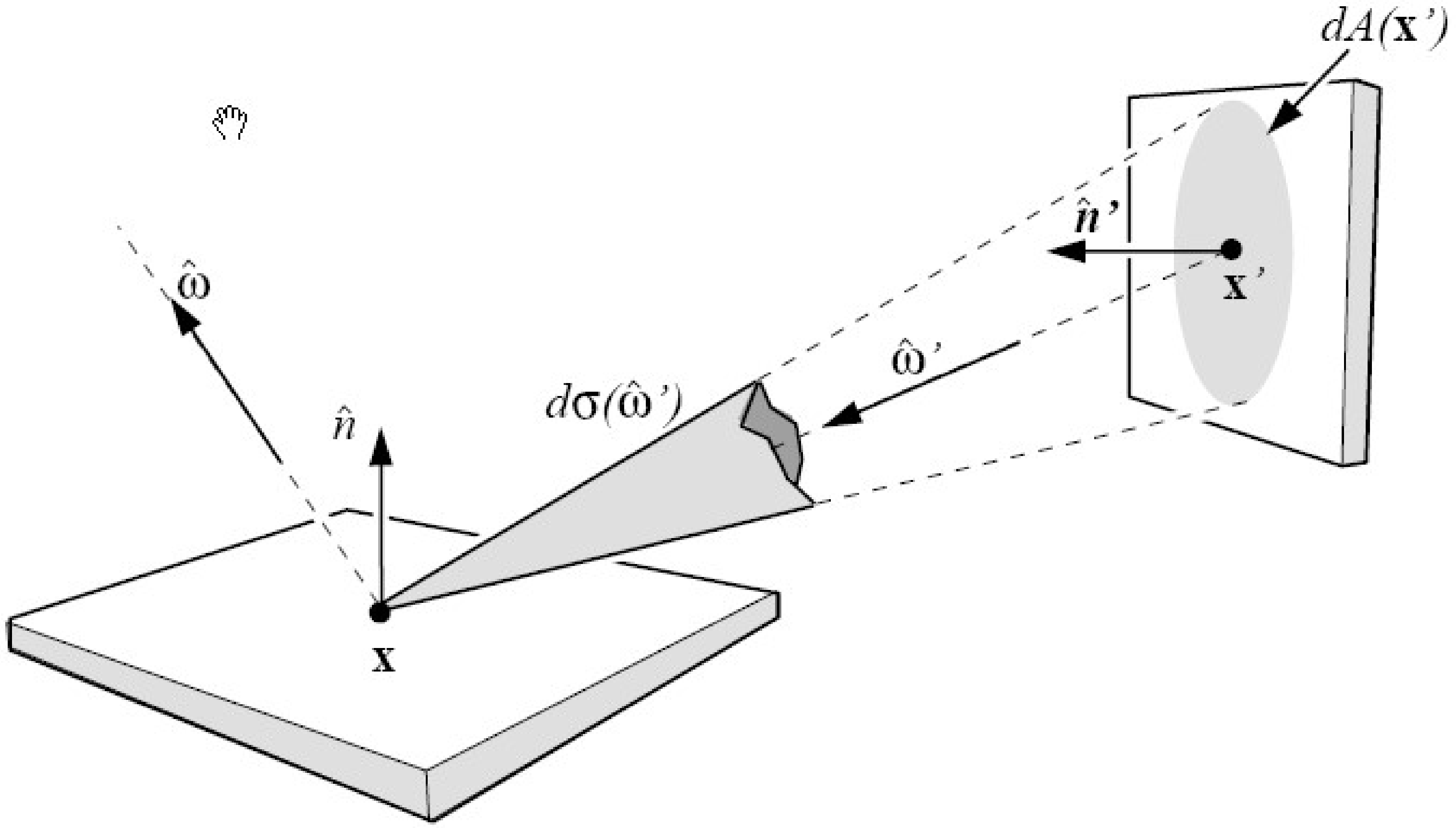
Bez antialiasingu

S antialiasingem

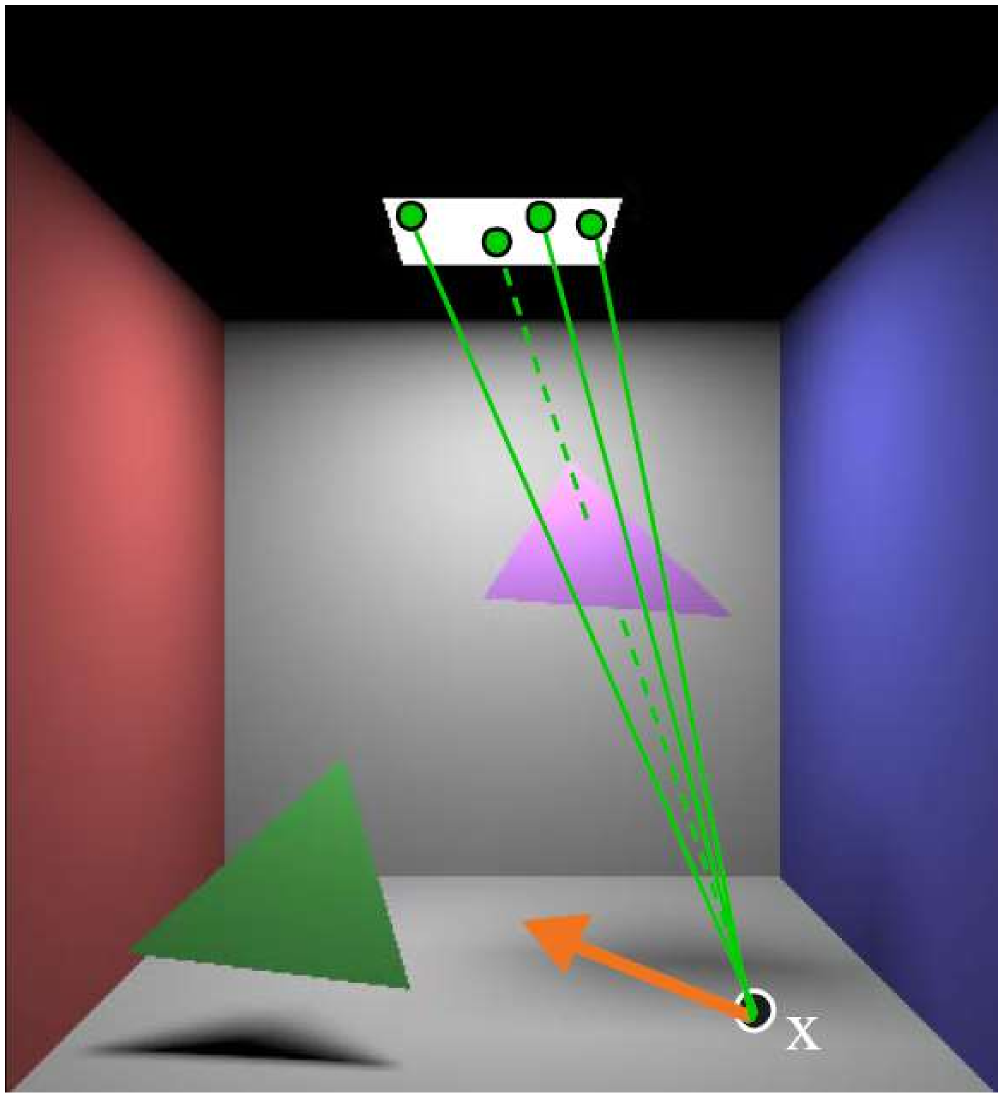
Aplikace Monte Carlo pro syntézu obrazu - Plošné zdroje světla

- Plošný (=nebodový) zdroj světla – přímé osvětlení musí být počítáno pomocí více vzorků
- Pro idealizované tvary lze úlohu spočítat analyticky, ale pak:
  - nebereme vliv zastínění dalšími objekty,
  - nemůžeme mít složitější tvar obrysu plošného světla,
  - co dělat, když světlo není plošné?
- Úloha se převádí na problém:
  - rozmísti body po povrchu světla,
  - spočti viditelnost a vzdálenost k bodu osvětlení,
  - zprůměruj hodnoty záře (angl. radiance)

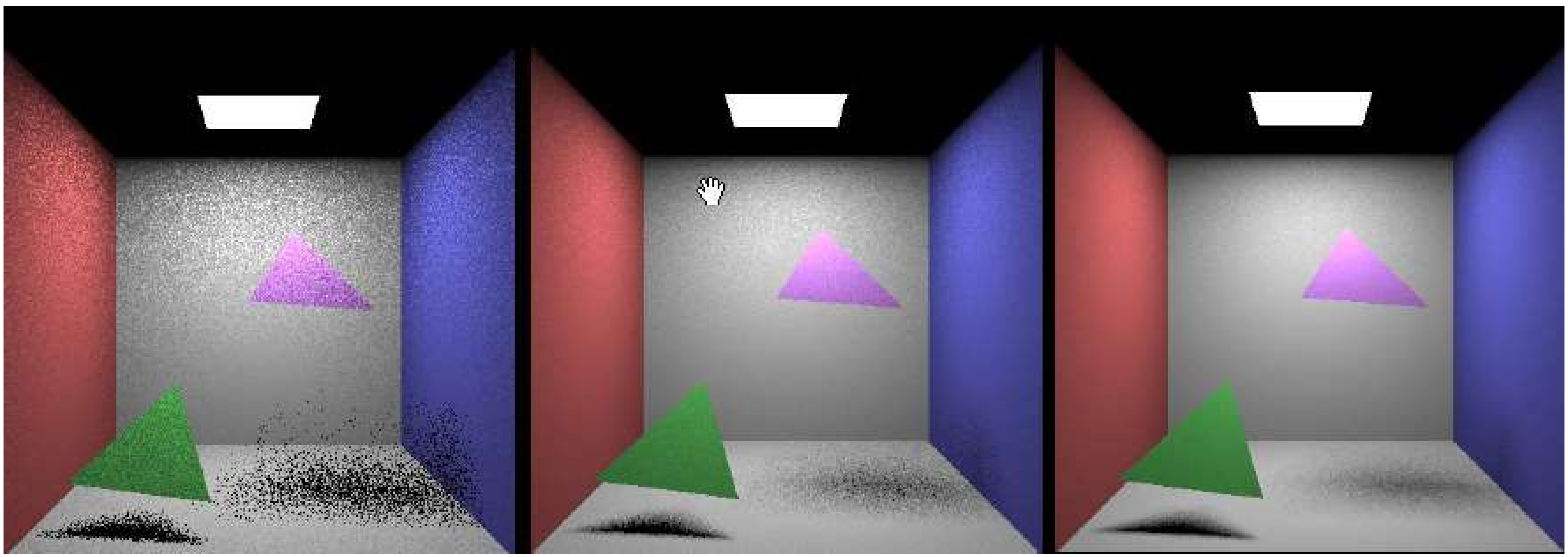
Aplikace Monte Carlo – Plošné zdroje světla



Vizualizace algoritmu pro výpočet přímého osvětlení



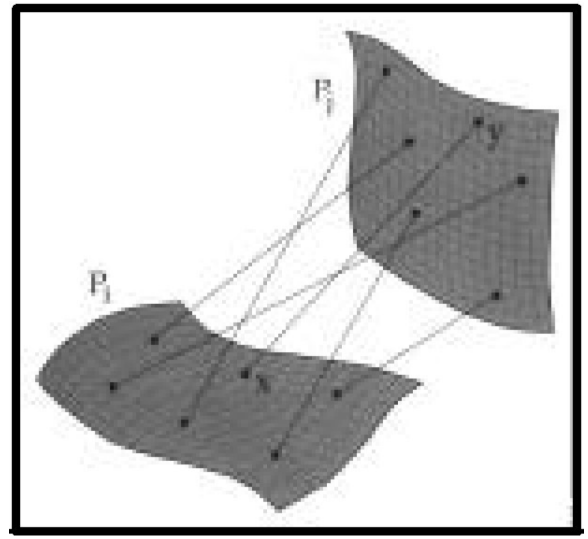
Ukázka pro plošný zdroj světla



1 vzorek na pixel 9 vzorků na pixel 36 vzorků na pixel

Aplikace metody Monte Carlo – Form Faktory pro radiozitu

- Vyber body na
  - první plošce
  - druhé plošce
- Spočti viditelnost mezi dvojicemi bodů
- Spočítej form faktor F_{ij}



$$B(\mathbf{x}) = B_e(\mathbf{x}) + \rho(\mathbf{x}) \int_S B(\mathbf{y}) \underbrace{\frac{G(\mathbf{x},\mathbf{y})V(\mathbf{x},\mathbf{y})}{\pi}}_{G'(\mathbf{x},\mathbf{y})} dA$$

$$G(x,y) = \frac{\cos\theta_x \cos\theta_y}{\cancel{\pi} r_{xy}^2}$$

$$B_i = \frac{1}{A_i} \int_{A_i} B(\mathbf{x}) dA_i =$$

$$= B_{e,i} + \rho_i \sum_{j=1}^{N} B_j \underbrace{\frac{1}{A_i} \int_{A_i} \int_{A_j} G'(\mathbf{x},\mathbf{y}) dA_j}_{F_{ij}}$$

F_{ij} … form faktor

Vlastnosti integrace vzorkovacích metod

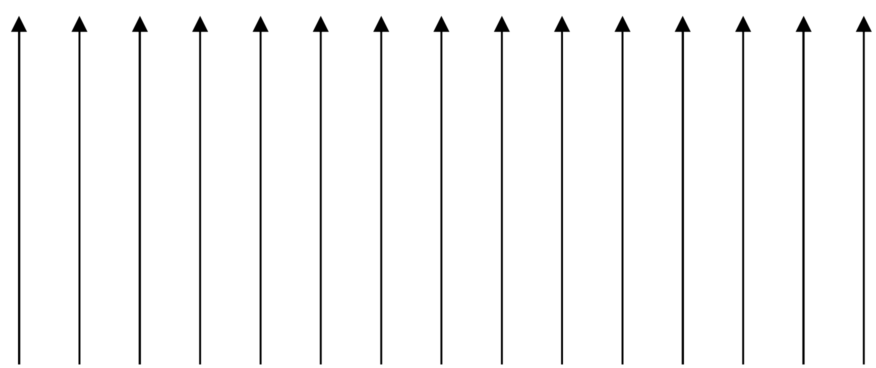
- Správnost odhadu integrálu – ačkoliv používáme jen konečný (malý) počet vzorku a výsledek zahrnuje statistickou chybu, je metoda výpočtu korektní a neobsahuje systematickou chybu
  - říkáme, že odhad integrálu je *nestranný* (anglicky *unbiased*)
  - v opačném případě (výpočet je zatížený chybou) říkáme, že odhad integrálu je *stranný* nebo *vychýlený*. Pak chybu výpočtu metodu označujeme *vychýlení odhadu* (anglicky se nazývá *bias*).
- Rychlost konvergence – jak rychle se zmenšuje rozptyl či skutečná chyba výpočtu s počtem vzorků N, lze vyjádřit buď asymptoticky nebo experimentálně

Odbočka: Kvadraturní metody integrace

$$\hat{I} = \sum_{i_1=1}^{n}\sum_{i_2=1}^{n}\sum_{i_3=1}^{n}\cdots\sum_{i_s=1}^{n} w_{i_1} w_{i_2} w_{i_3}\cdots w_{i_s} f(x_{i_1}, x_{i_2}, x_{i_3},\cdots,x_{i_s})$$

- Obdélníková, rovnoběžníková, Simpsonova metoda:

  rychlost konvergence pro s-dimenzionální integrál je $O(1/N^{1/s})$
- Používají se maximálně pro jedno či dvojrozměrné integrály
- Vzorky jsou rozmístěny v pravidelné mřížce na doméně integrálu



Monte Carlo Integrace

- Rychlost konvergence pro s-dimenzionální integrál je $O(1/N^{1/2})$, tedy nezávislá na počtu dimenzí!
- Podmínkou úspěšné aplikace MC metody je:
  - možnost opakovaně počítat pokusy pro daný jev
  - možnost rychle a kvalitně generovat náhodná čísla pro realizaci pokusů. Náhodná čísla nesmí být snadné předvídat.
  - aplikace racionálního algoritmu integrace s co nejrychlejší konvergencí

Monte Carlo Integrace - Shrnutí

- Výhody metody MC
  - Jednoduché na implementaci
  - Jednoduché na představu
  - Robustní řešení pro různé tvary domén a integrandů
  - Efektivní pro vícerozměrné integrály
  - Efektivní i pro malý počet vzorků a funkce se singularitami
  - Rychlost konvergence $O(N^{1/2})$, chyba se zmenšuje $O(1/N^{1/2})$
  - Možnost kontroly a odhadu přesnosti výpočtu pomocí rozptylu
- Nevýhody
  - Relativně pomalá konvergence – zmenšení statistické chyby o polovinu vyžaduje zvětšit počet vzorků čtyřikrát
  - Důsledek pro syntézu obrazu: obrázek obsahuje šum
  - Výpočet musí mít hromadný charakter, N vzorků

Náhodná veličina

- Vzorky náhodné veličiny X
  - generované náhodným procesem
  - generované podle pravděpodobnostní hustoty
  - Funkce Y(X) je rovněž náhodná veličina
- Střední hodnota náhodné veličiny:

$$E[f] \equiv \int_{\Omega} f(x)p(x)d\mu(x)$$

- Vlastnosti

$$E[\sum_i f_i] = \sum_i E[f_i]$$

$$E[af] = aE[f]$$

Náhodná veličina - zopakování

- Definice: Rozptyl

$$
\begin{aligned}
V[X] &= E[(X - E[X])^2] \\
&= E[X^2 - 2XE[X] + E[X]^2] \\
&= E[X^2] - E[X]^2
\end{aligned}
$$

- Vlastnosti $V[\sum_i X_i] = \sum_i V[X_i]$

$$V[aX] = a^2 V[X]$$

- Rozptyl se zmenšuje s počtem vzorků

$$V[\frac{1}{N}\sum_{i=1}^{N} X_i] = \frac{1}{N^2}\sum_i V[X_i] = \frac{1}{N} V[X]$$

Nestranný odhad

- Konečný integrál

$$I = \int_{\Omega} f(x) d\mu(x)$$

- Střední hodnota

$$E[f] = \int_{\Omega} f(x) p(x) d\mu(x)$$

- Definice odhadu (estimátoru) pro konstantní pravd. hustotu:

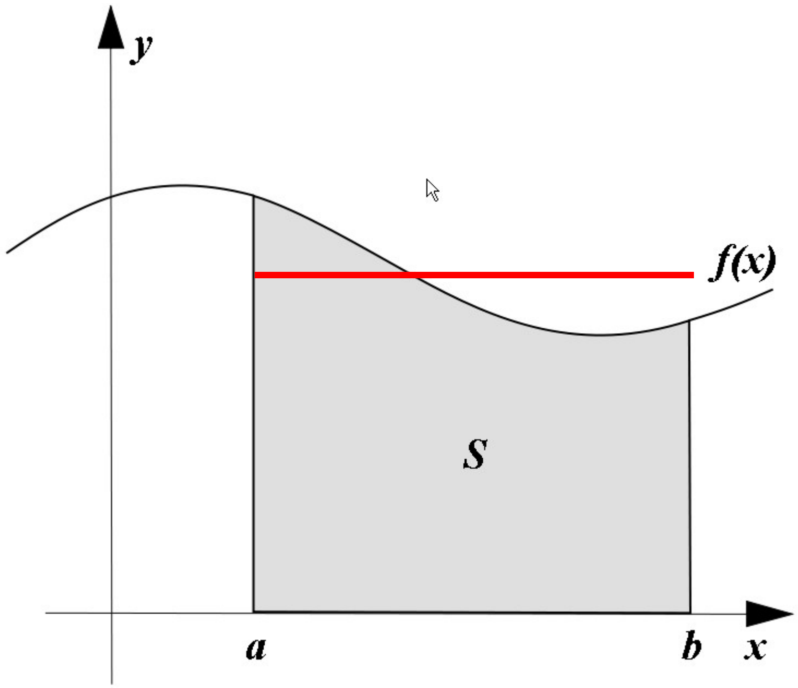
$$F_N = \frac{1}{N} \sum_{i=1}^{N} Y_i = \frac{1}{N} \sum_{i=1}^{N} f(X_i)$$

- Vlastnosti: odhad je nestranný

$$E[F_N] = I(f)$$

$$\begin{aligned}
E[F_N] &= E[\frac{1}{N} \sum_{i=1}^{N} Y_i] \\
&= \frac{1}{N} \sum_{i=1}^{N} E[Y_i] \\
&= \frac{1}{N} \sum_{i=1}^{N} E[f(X_i)] \\
&= \frac{1}{N} \sum_{i=1}^{N} \int_{\Omega} f(x) p(x) d\mu(x) \\
&= \frac{1}{N} \sum_{i=1}^{N} \int_{\Omega} f(x) d\mu(x) \\
&= \int_{\Omega} f(x) d\mu(x)
\end{aligned}$$

Shrnutí výpočtu střední hodnoty



- Střední hodnota funkce
- Rozptyl funkce

- Odhad střední hodnoty
- Odhad rozptylu

Diskrétní pravděpodobnostní hustoty

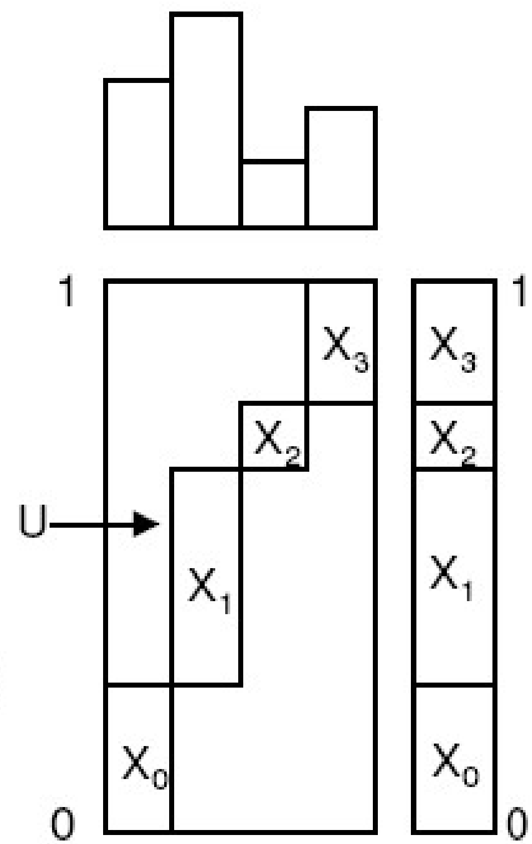
- Diskrétní události X_i s pravděpodobností p_i

  $p_i > 0 \qquad \sum_{i=1}^{n} p_i = 1$

- Distribuční funkce:

  $$P_j = \sum_{i=1}^{j} p_i$$

- Výběr vzorků
  - U: vyber uniformní náhodnou veličinu
  - Vyber $X_i \sim p_i$ aby odpovídalo $P_{i-1} < U < P_i$
  - Není potřeba lineární interpolace pro výpočet polohy vzorku ze sousedů, vybíráme diskrétní hodnotu



Spojité pravděpodobnostní distribuce

- Pravděpodobnostní hustota

  p(x), p(x) > 0
- Distribuční funkce P(X)

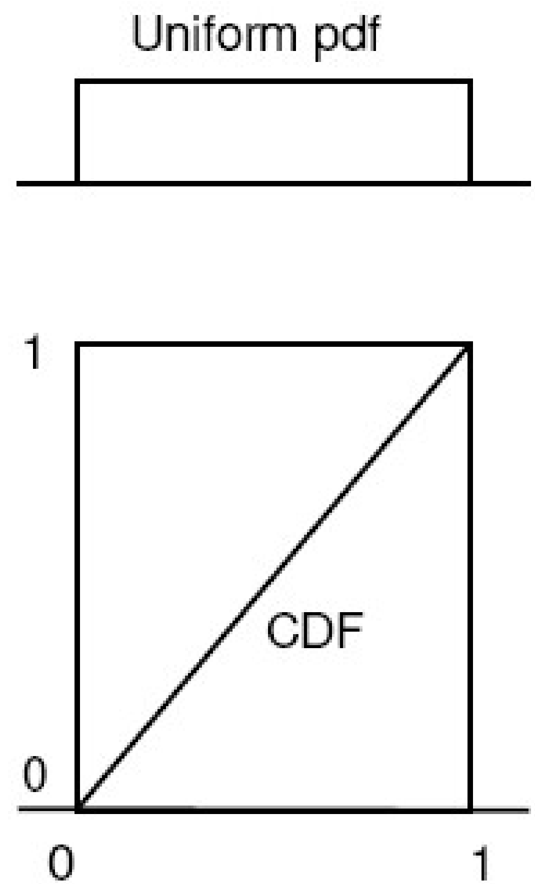
$$P(x) = \int_0^x p(x)dx$$

$$P(x) = \Pr(X < x)$$

$$P(1) = 1$$

$$\Pr(a < X < b) = \int_a^b p(x)dx$$

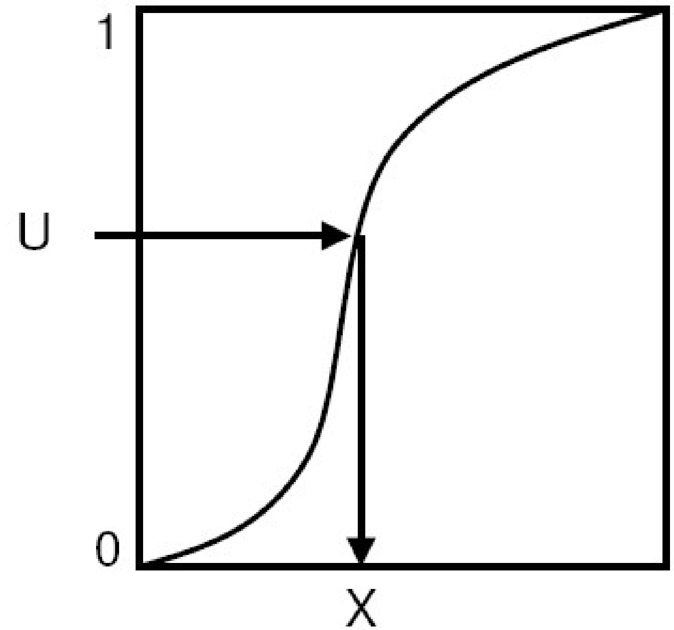
$$= P(b) - P(a)$$



Vzorkování pro spojité distribuce (inverse transform method)

- Distribuční funkce

$$P(x) = \Pr(X < x)$$



- Algoritmus výběru vzorků
  - Zjisti $X = P^{-1}(U)$
  - 2 kroky k dosažení výsledku:
    - spočti distribuční funkci P(x) z pravděpodobnostní hustoty p(x)
    - spočti inverzní funkci $P^{-1}(x)$
    - najdi hodnotu pro náhodné číslo „U“
    - Pro tabulku je třeba lineární interpolace pro výpočet polohy vzorku ze sousedů, vybíráme spojitou hodnotu X
    - Poznámka: distribuční funkce P(x) se počítá jako integrál z hustoty pravděpodobnosti p(x).

Příklad: analytická funkce

- Zadání funkce

$$p(x) = (n+1)x^n$$

$$P(x) = x^{n+1}$$

$$X \sim p(x) \Rightarrow X = P^{-1}(U) = \sqrt[n+1]{U}$$

- Integrál polynomiální funkce je:

$$\int_0^1 x^n dx = \left. \frac{x^{n+1}}{n+1} \right|_0^1 = \frac{1}{n+1}$$

Příklad 2: Vzorkování v kruhu, náhodná veličina v R^2

$$A = \int_0^{2\pi}\int_0^1 r\,dr\,d\theta = \int_0^1 r\,dr \int_0^{2\pi} d\theta = \left.\frac{r^2}{2}\right|_0^1 \left.\theta\right|_0^{2\pi}$$

$$p(r,\theta) = p(r)\,p(\theta)$$

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

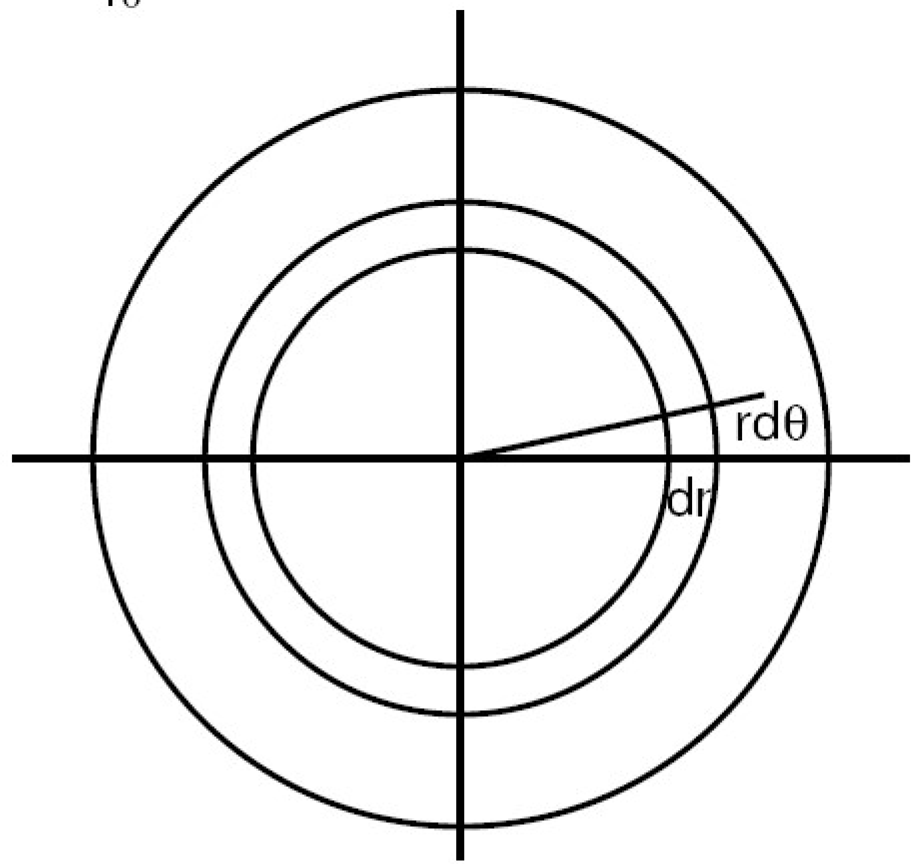
$$\theta = 2\pi U_1$$

$$P(\theta) = \frac{1}{2\pi}\theta$$

$$p(r) = 2r$$

$$r = \sqrt{U_2}$$

$$P(r) = r^2$$

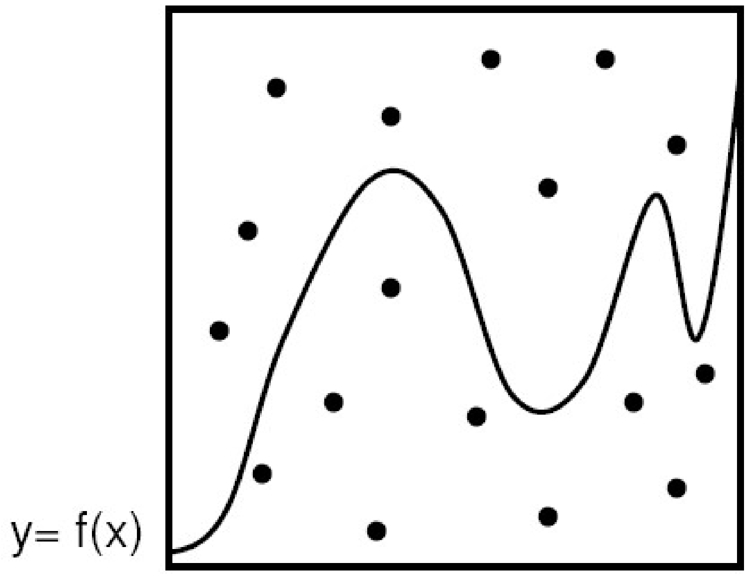


Zamítací metoda (Rejection Sampling)

- Monte Carlo integrace

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

$$= \int_0^1 \int_0^{f(x)} dy\, dx$$

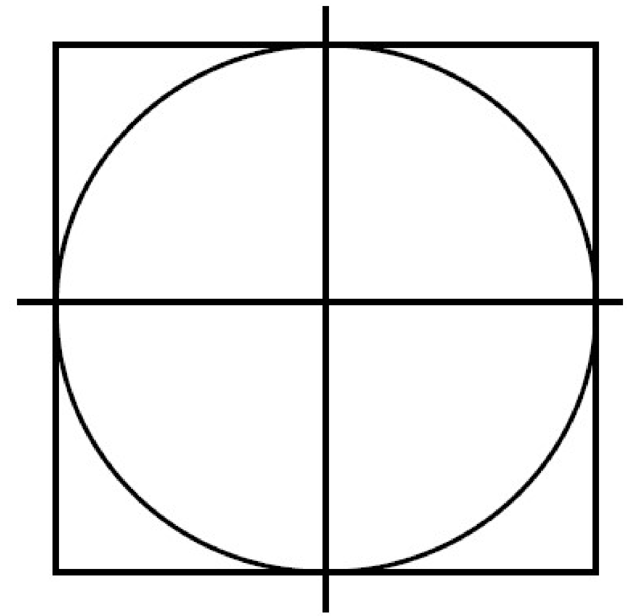


- Algoritmus
  - Vyber náhodné U1 a U2
  - Přijmi vzorek, pokud f(U1) < U2
- Efektivita
  - Plocha funkce pod křivkou/plocha obdélníka
  - Analytická integrace je vždy efektivnější

Zamítací metoda pro kružnici

- Algoritmus

```
do {
  X= 1-2*U1;
  Y= 1-2*U2;
} while (X² + Y² > 1);
```



- Rozšíření pro kouli (3D)
  - Vyber vzorek v kouli
  - Vyber náhodný směr
- Není příliš efektivní

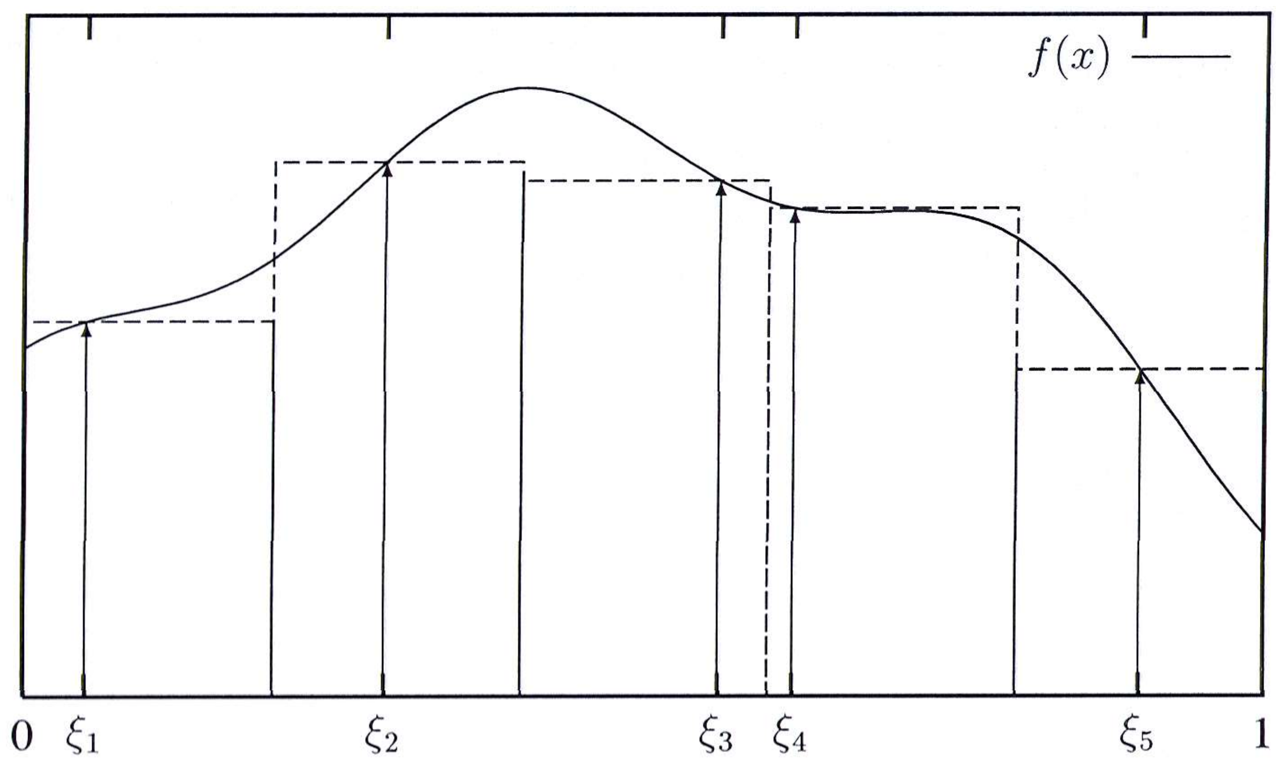
Metody zefektivnění výpočtu

- Metody
  - neinformované (blind Monte Carlo)
  - informované (informed Monte Carlo)

- Některé metody zefektivnění výpočtu
  - Vymezení hlavní části (“control variates”)
  - Výběr na podintervalech
  - Podstatný výběr

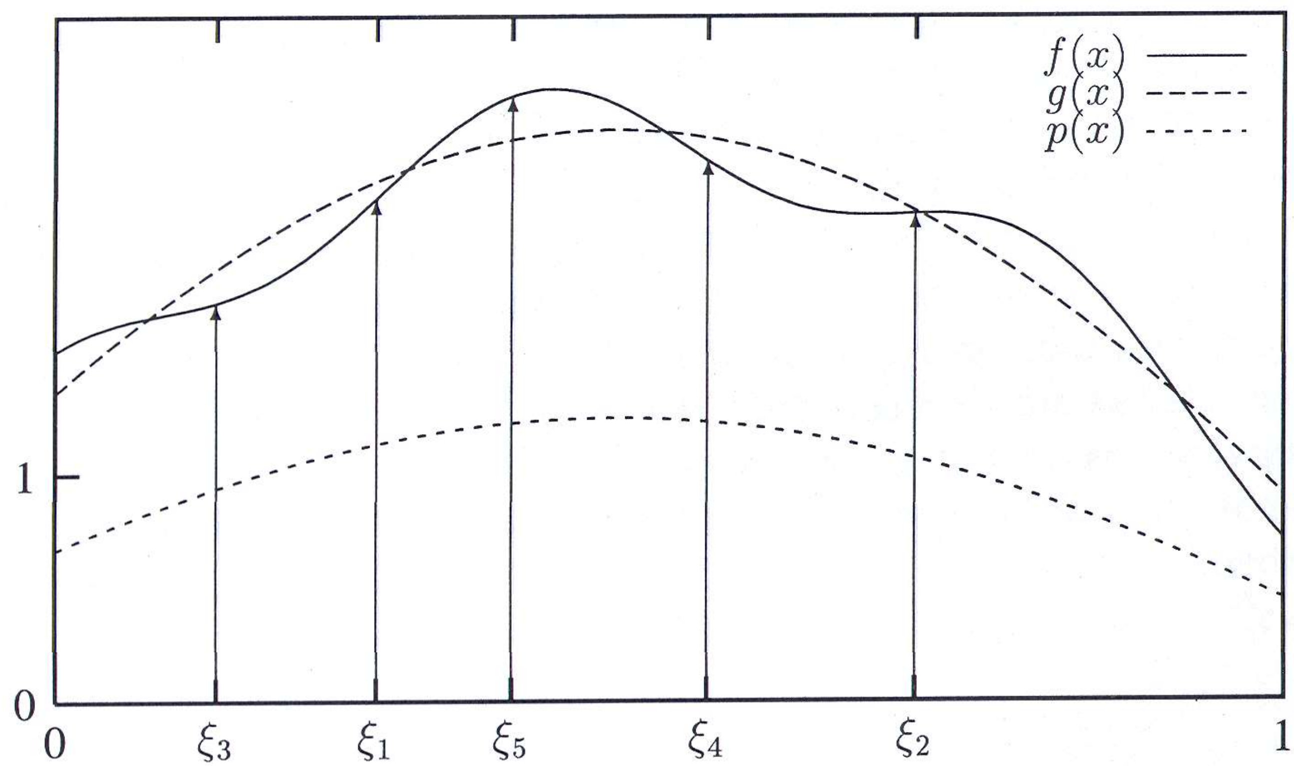
Výběr na podintervalech (Stratified Sampling)

- Rozděl doménu na stejně velké části
- V každé části náhodně vyber jeden vzorek
- Známé také jako “jittered sampling” (pro více dimenzí)
- Nevýhoda: nelze jednodušše přidávat vzorky

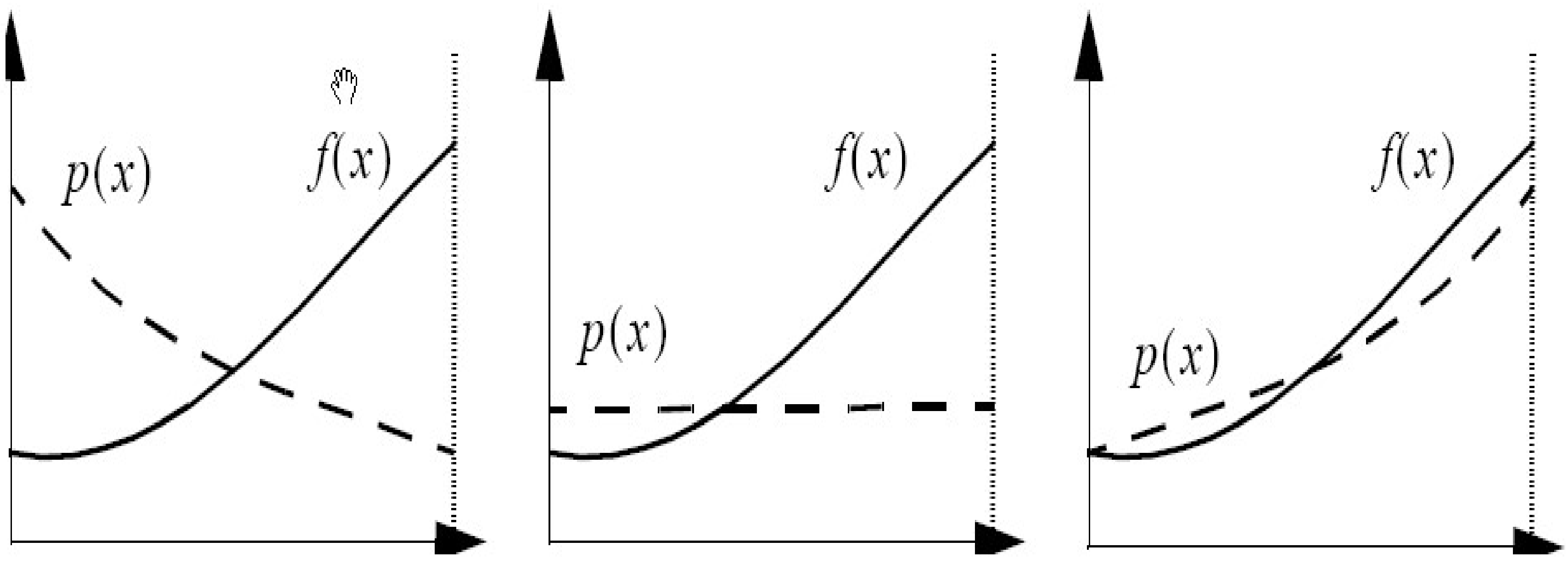


Podstatný výběr (Importance Sampling)

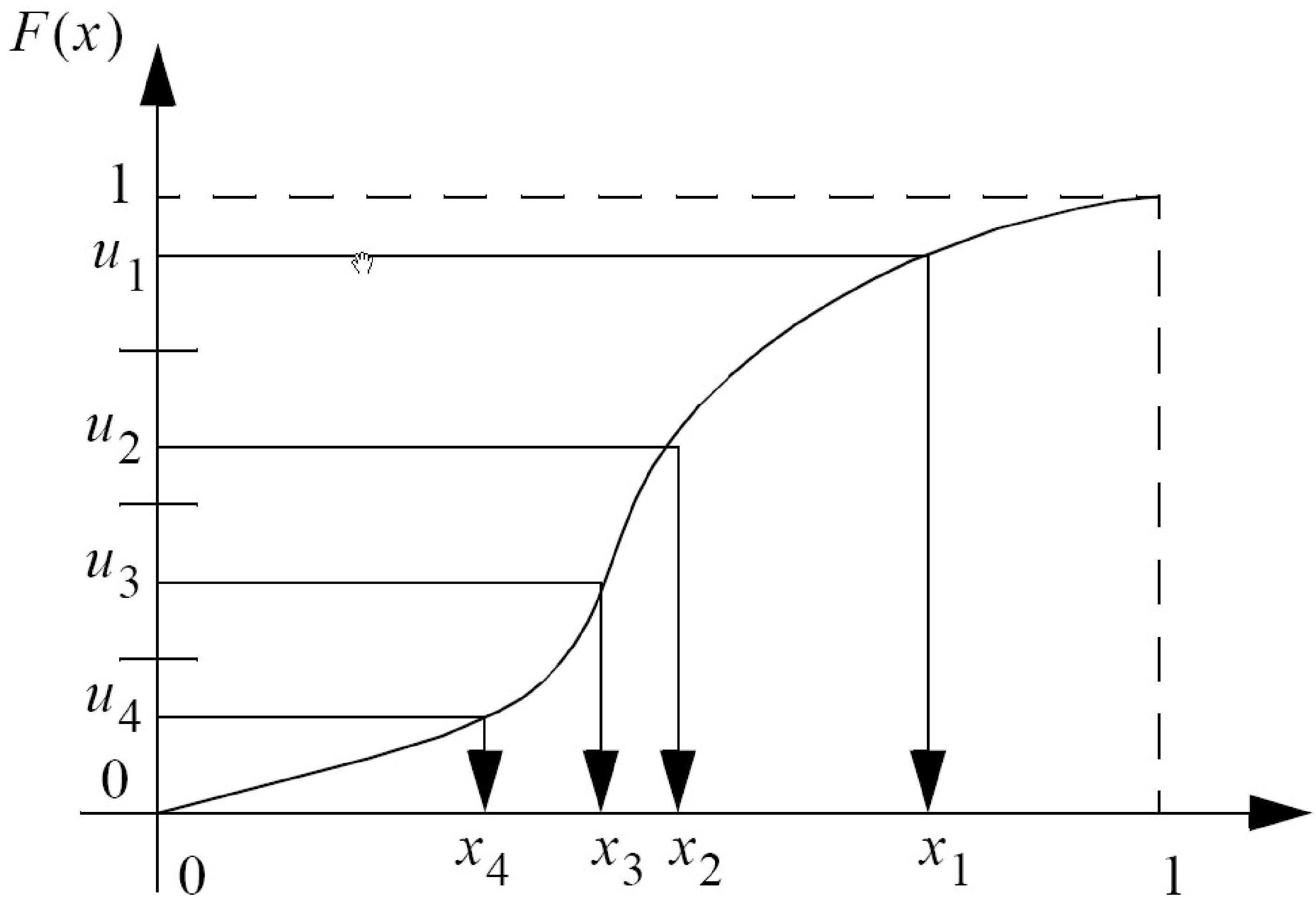
- V oblastech, kde je očekávaná hodnota funkce vyšší, vybíráme více vzorků
- Je nutné mít představu o přibližném tvaru funkce (funkce důležitosti)



Vliv funkce důležitosti na rozptyl střední hodnoty



Kombinace metody podstatného výběru a výběru na podintervalech



Částečné shrnutí statistiky

- Odhad rozptylu střední hodnoty slouží jako míra stochastické chyby odhadu
- Pokud použijeme efektivnější metodu integrace, rozptyl odhadu se zmenší
- Výsledek integrálu = odhad střední hodnoty
- Vzorce pro rozptyl pro efektivnější metody
  - dají se odvodit
  - jsou v literatuře či na webu

  http://en.wikipedia.org/wiki/Importance_sampling

Důležité vzorce a postupy

- Odhad integrálu: zvolíme funkci hustoty pravděpodobnosti $p(X_i)$, podle které vybíráme vzorky

$$I = \int_{\Omega} f(x)\,d\mu(x)$$

- Nestranný odhad střední hodnoty
  - téměř náhodná pravd. hustota p(x)
  - funkce p(x) nesmí být nulová, kde je funkce f(x) nenulová

použijeme tento vzorec:

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

$$E[F_N] = E\left[\frac{1}{N}\sum_{i=1}^{N}\frac{f(X_i)}{p(X_i)}\right]$$

$$= \frac{1}{N}\sum_{i=1}^{N} E\left[\frac{f(X_i)}{p(X_i)}\right]$$

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

$$= \int_{\Omega} f(x)\,d\mu(x)$$

Generátor náhodných čísel

- Univerzální test generátoru náhodných čísel je obtížný problém
- Základní požadavky na generátor
  - test dobré shody – rovnoměrné rozdělení na interval (0,1)
  - test náhodnosti – jsou získaná náhodná čísla na sobě nezávislá (frekvence výskytu číslic v čísle)
  - Rychlost a paměťová nenáročnost
- Praktické doporučení:
  - Nepoužívejte funkci rand() v knihovně ANSI C
  - Použijte buď Mersenne Twister (v nové verzi C++ v knihovně)

    http://www.math.sci.hiroshima-u.ac.jp/~m-mat/MT/emt.html
  - Nebo aspoň funkci drand48() pod UNIXem

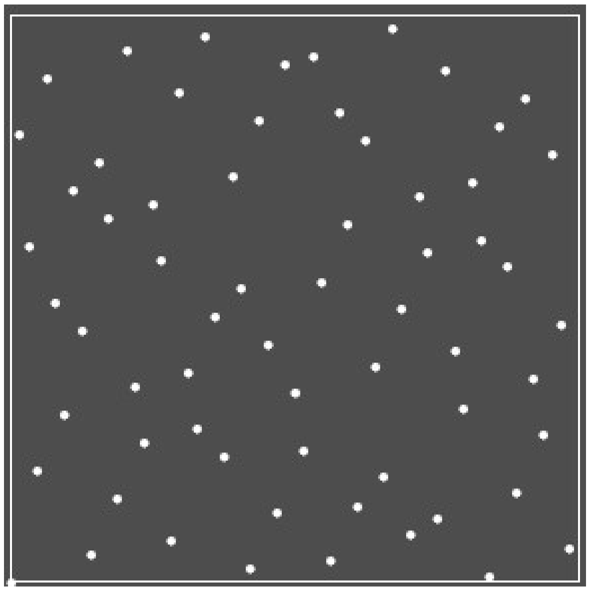
Metody Quasi Monte Carlo (QMC)

- Rychlost konvergence lze změnit z $O(1/N^{1/2})$ pro Monte Carlo na $O((\log N)^s/N)$ pro Quasi Monte Carlo, kde "*s*" je dimenze domény integrálu
- To umožňuje ještě rychlejší konvergenci odhadu než kvadraturní metody integrace a Monte Carlo, ale jen pro malý a střední počet dimenzí (10 až 20)
- To přesně vyhovuje problému syntézy obrazu

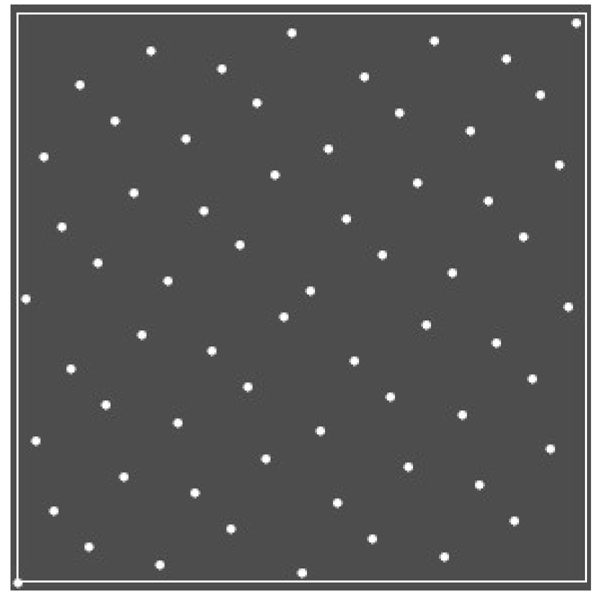
Důležité názvy generátorů QMC

- van der Corput
- Halton
- Hammersley
- Sobol
- Niederreiter
- Niederreiter-Xing

Ukázky výsledku 2D generátorů

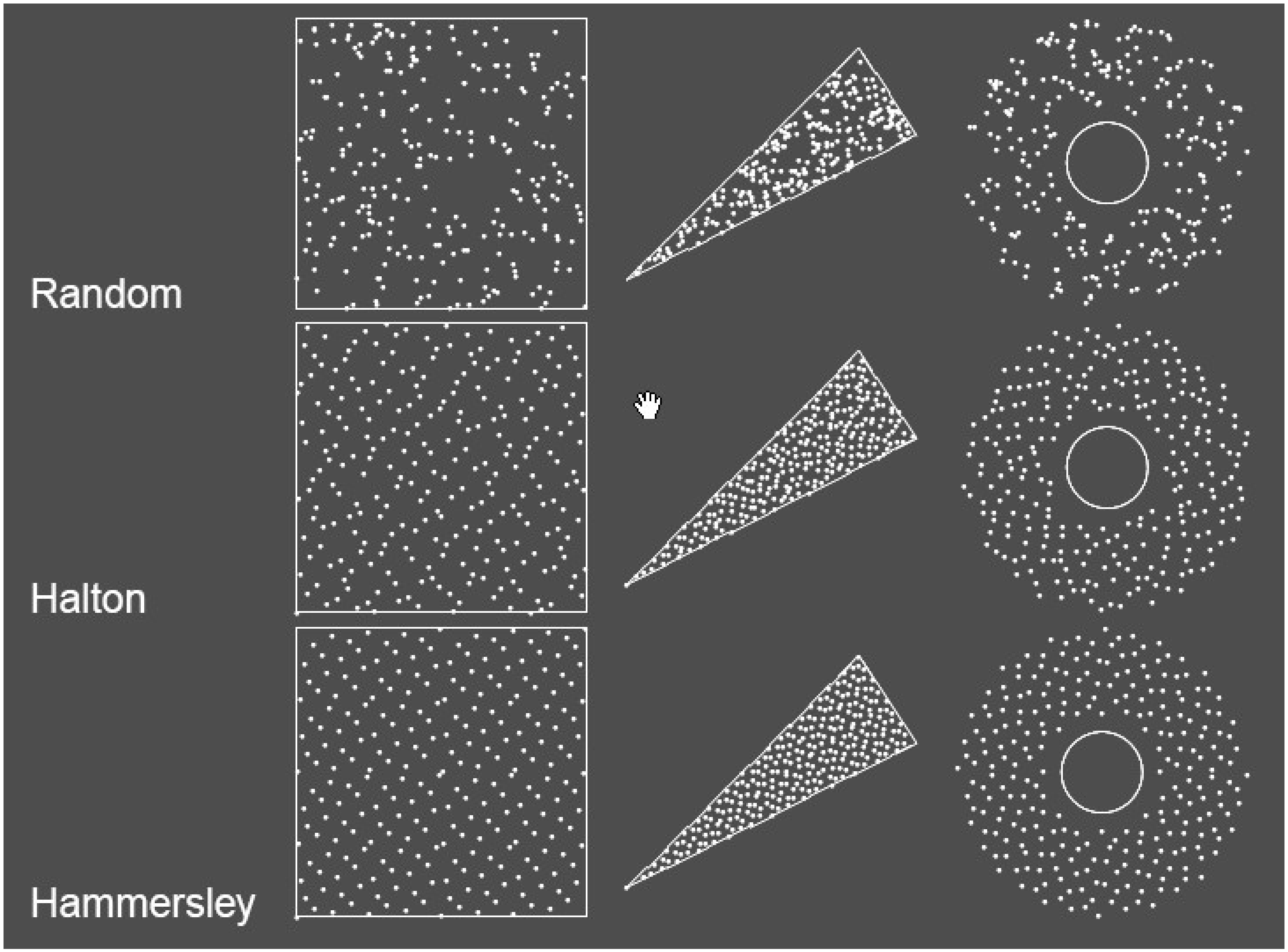


Halton



Hammersley
(1. dimenze pravidelně
2. a vyšší Halton, je nutné
vědět dopředu počet vzorků)

Ukázky rozmístění vzorků



Algoritmus generátoru Halton

- přímý algoritmus výpočtu
- b ... prvočíslo (2,3,5,7,....)
- i ... semínko (seed), větší než 0

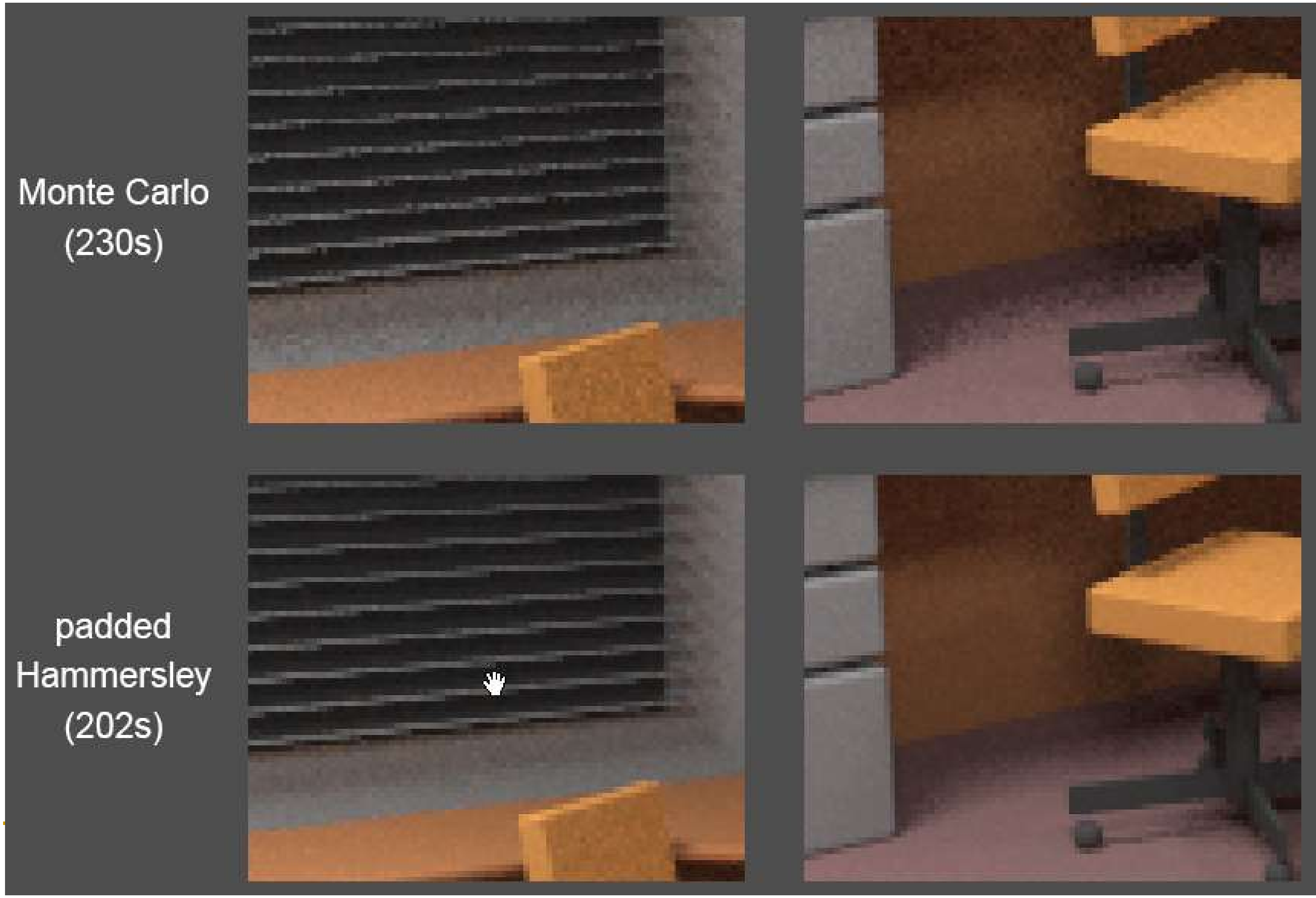
```
double RadicalInverse(const int Base, int i)
{
  double Digit, Radical, Inverse;

  Digit = Radical = 1.0 / (double) Base;
  Inverse = 0.0;

  while(i)
  {
    Inverse += Digit * (double) (i % Base);
    Digit *= Radical;
    i /= Base;
  }

  return Inverse;
}
```

Inkrementální algoritmus pro Haltonův generátor

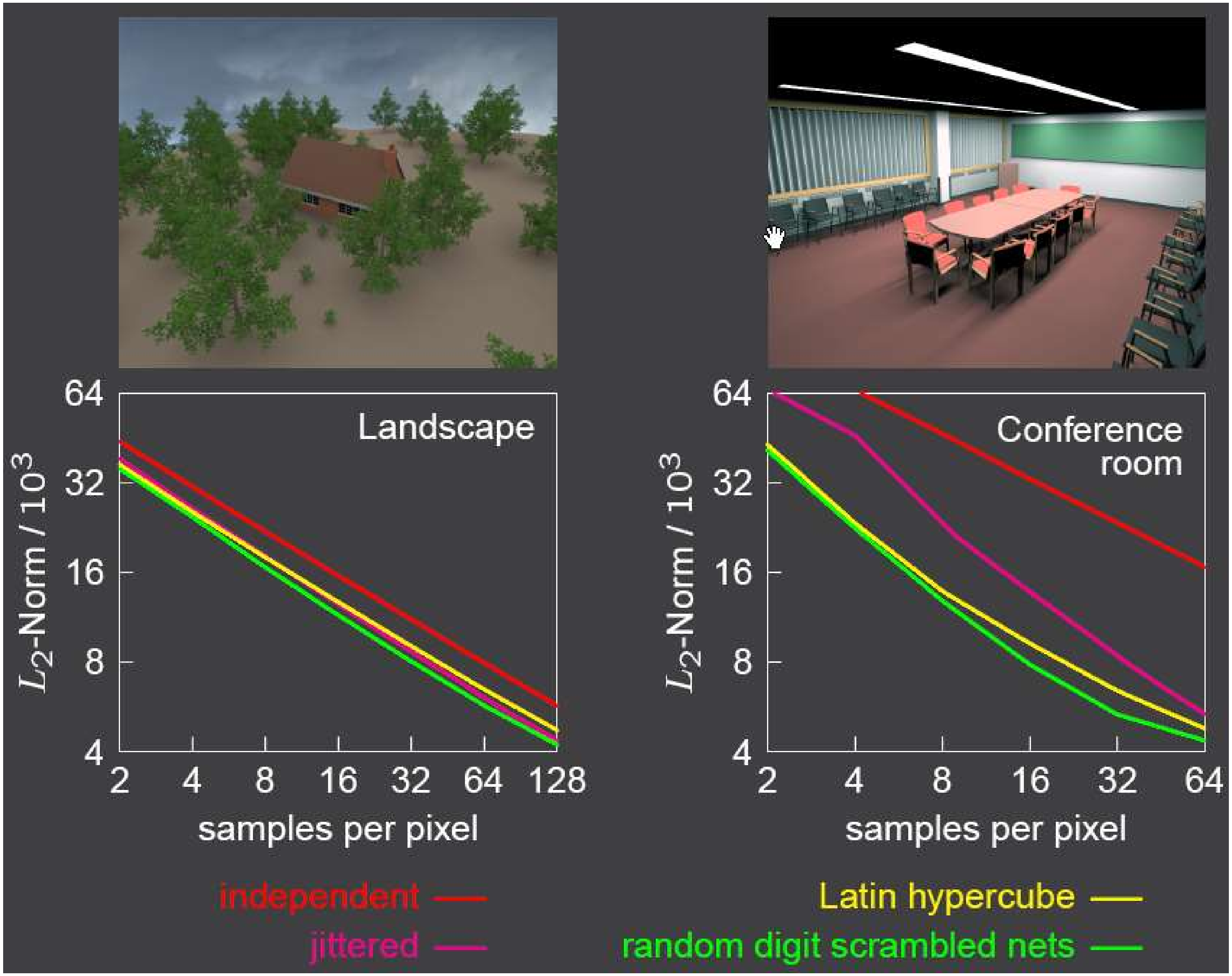
- Trik – musíme znát vypočtenou hodnotu pro stejnou dimenzi a semínko o jedničku menší

```
double NextRadicalInverse(const double Radical, double Inverse)
// Radical = 1.0 / Base
{
   const double AlmostOne = 1.0 - 1e-10;
   double NextInverse, Digit1, Digit2;

   NextInverse = Inverse + Radical;

   if(NextInverse < AlmostOne)
      return NextInverse;
   else
   {
      Digit1 = Radical;
      Digit2 = Radical * Radical;

      while(Inverse + Digit2 >= AlmostOne)
      {
         Digit1 = Digit2;
         Digit2 *= Radical;
      }

      return Inverse + (Digit1 - 1.0) + Digit2;
   }
}
```

Ukázka obrázků pro MC a QMC



Přímé osvětlení



Literatura MC obecně

- F. Fabian, Z. Kluiber: Metoda Monte Carlo a možnosti jejího uplatnění, Praha 1998, ISBN 80-7175-058-1
- M. Virius, Aplikace Matematické Statistiky, skriptum ČVUT, FJFI 1998
- Kalos, Whitlock: Monte Carlo Methods, Volume I: Basics, John Wiley and Sons, 1986
- G.S. Fishman: Monte Carlo, Concepts, Algorithms, and Applications, Springer Verlag, 1995

Literatura pro syntézu obrazu

- Eric Lafortune: Mathematical Models and Monte Carlo Algorithms for Physically Based Rendering, 1997
- Szirmay Kalos, skripta v angličtině, 1997: Monte-Carlo Methods in Global Illumination (ke stáhnutí na webu)
- Efficient Monte Carlo and Quasi-Monte Carlo Rendering Techniques, Eurographics 2003 Tutorial T2
- Monte Carlo Ray Tracing, SIGGRAPH 2003 Course 44