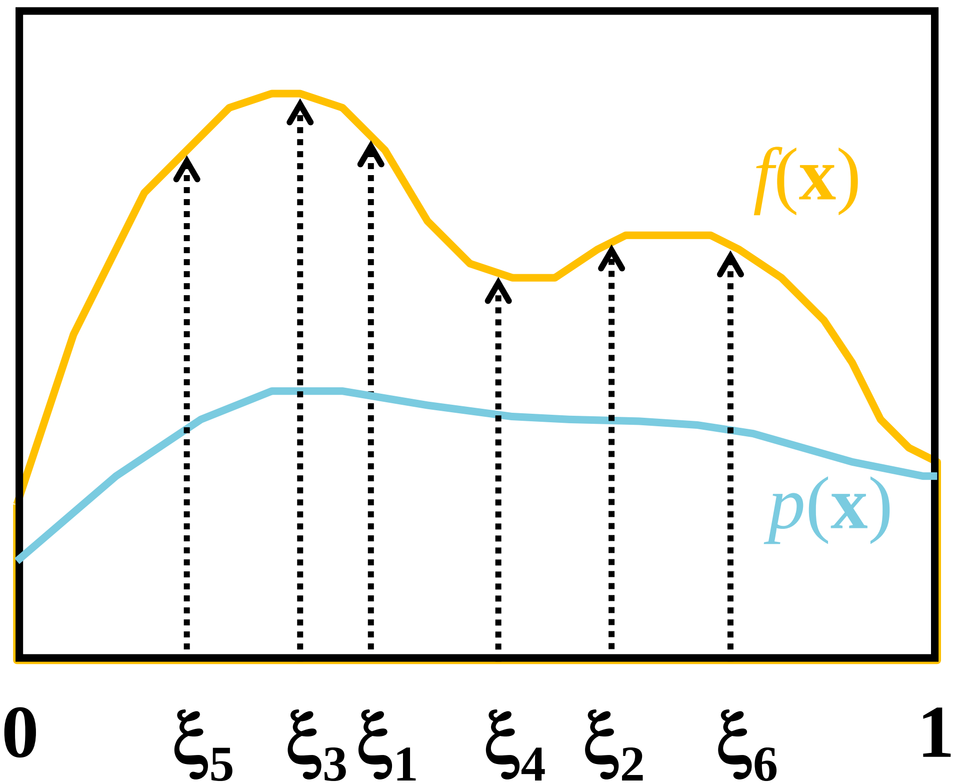
Počítačová grafika III – Path tracing

Jaroslav Křivánek, MFF UK

Jaroslav.Krivanek@mff.cuni.cz

Monte Carlo integrování

- Obecný nástroj k numerickému odhadu určitých integrálů



Integrál:

$$I = \int f(\mathbf{x}) \mathrm{d}\mathbf{x}$$

Monte Carlo odhad I:

$$\langle I \rangle = \frac{1}{N} \sum_{i=1}^{N} \frac{f(\xi_i)}{p(\xi_i)}; \quad \xi_i \propto p(\mathbf{x})$$

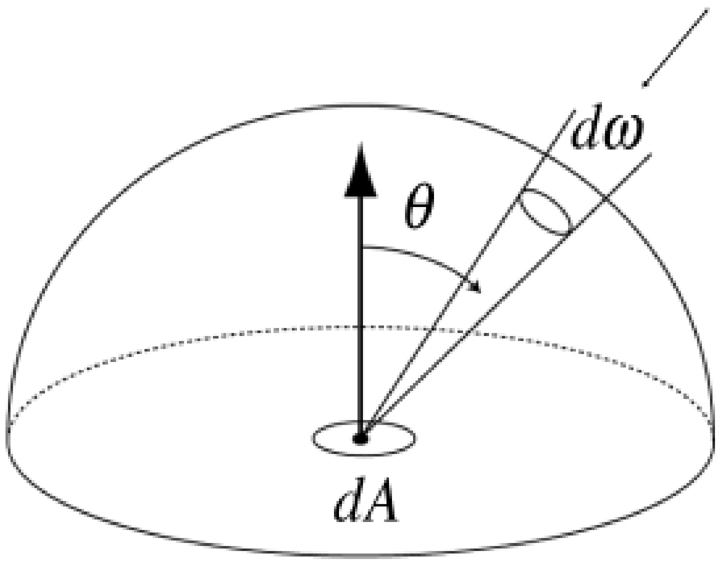
„V průměru“ to funguje:

$$E[\langle I \rangle] = I$$

Příklady MC estimátorů

Odhad irradiance – uniformní vzork.

$$E(\mathbf{x}) = \int_{H(\mathbf{x})} L_\mathrm{i}(\mathbf{x}, \omega_\mathrm{i}) \cdot \cos\theta_\mathrm{i} \, \mathrm{d}\omega_\mathrm{i}$$



- Uniformní vzorkování:

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

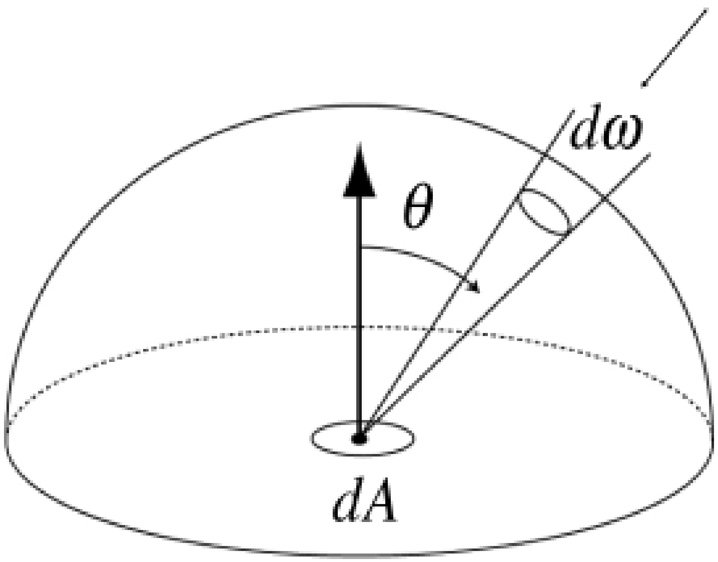
- Estimátor:

$$F_N = \frac{1}{N}\sum_{k=1}^{N}\frac{f(\omega_{\mathrm{i},k})}{p(\omega_{\mathrm{i},k})}$$

$$= \frac{2\pi}{N}\sum_{k=1}^{N} L_\mathrm{i}(\mathbf{x}, \omega_{\mathrm{i},k}) \cdot \cos\theta_{\mathrm{i},k}$$

Odhad irradiance – cos vzorkování

$$E(\mathbf{x}) = \int_{H(\mathbf{x})} L_\mathrm{i}(\mathbf{x}, \omega_\mathrm{i}) \cdot \cos\theta_\mathrm{i} \, \mathrm{d}\omega_\mathrm{i}$$



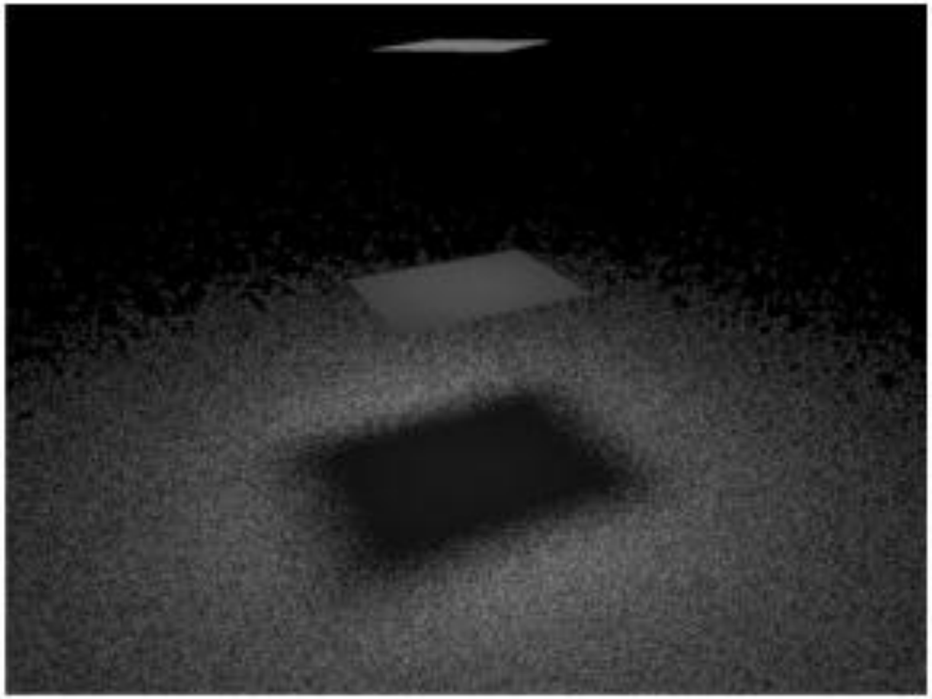
- Importance sampling:

$$\boxed{p(\omega) = \frac{\cos\theta}{\pi}}$$

- Estimátor:

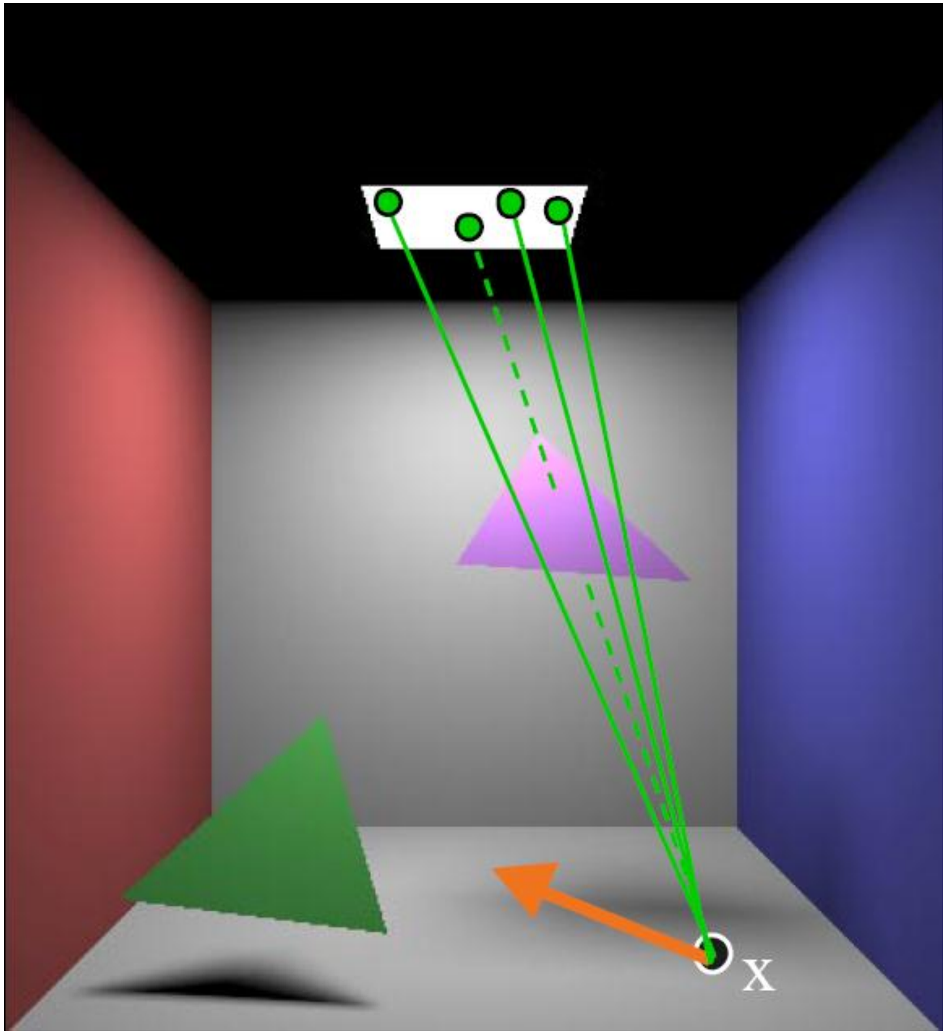
$$F_N = \frac{1}{N}\sum_{k=1}^{N}\frac{f(\omega_{\mathrm{i},k})}{p(\omega_{\mathrm{i},k})} = \frac{\pi}{N}\sum_{k=1}^{N} L_\mathrm{i}(\mathbf{x}, \omega_{\mathrm{i},k})$$

Computing Irradiance



4 eye rays per pixel
100 shadow rays

Odhad irradiance – vzrokování zdroje



Odhad irradiance – vzrokování zdroje

$$E(\mathbf{x}) = \int_{H(\mathbf{x})} L_\mathrm{i}(\mathbf{x}, \omega_\mathrm{i}) \cdot \cos\theta_\mathrm{i} \, \mathrm{d}\omega_\mathrm{i}$$

$$= \int_A L_\mathrm{e}(\mathbf{y} \to \mathbf{x}) \cdot V(\mathbf{y} \leftrightarrow \mathbf{x}) \cdot \underbrace{\frac{\cos\theta_\mathbf{y} \cdot \cos\theta_\mathbf{x}}{\|\mathbf{y} - \mathbf{x}\|^2}}_{G(\mathbf{y} \leftrightarrow \mathbf{x})} \mathrm{d}A$$

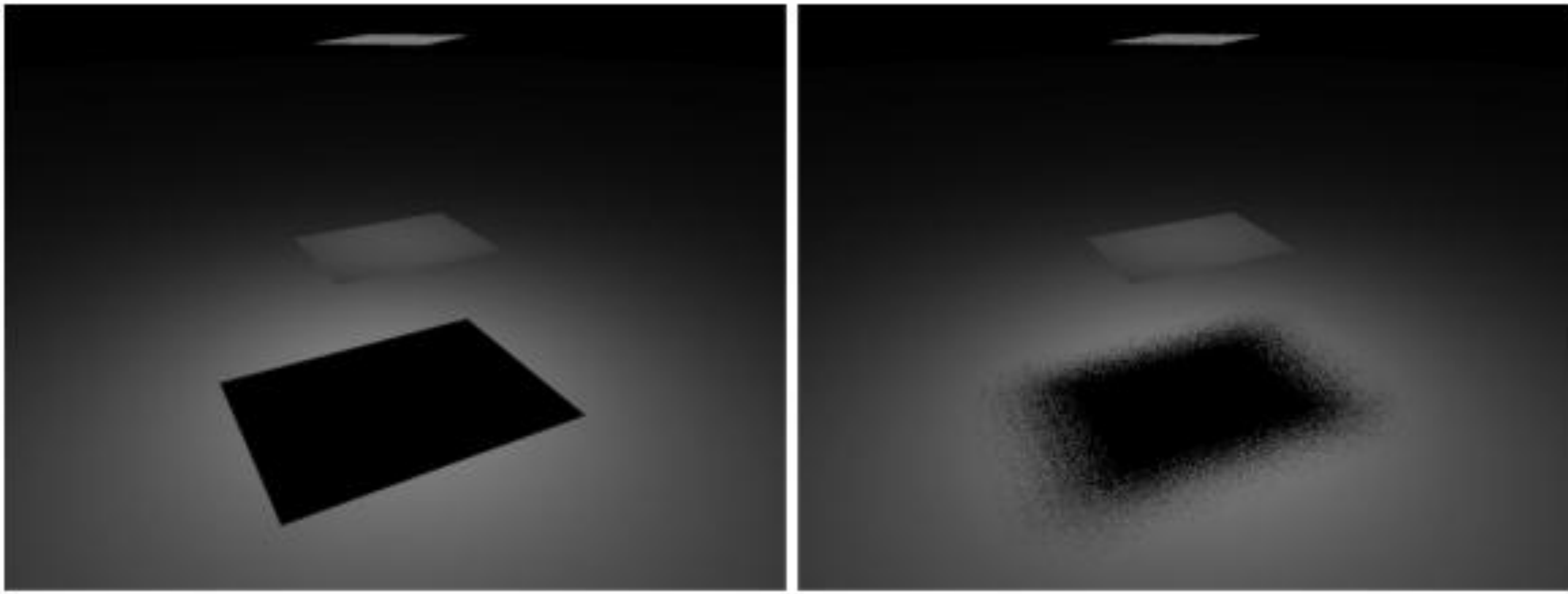
- Uniformní vzorkování plochy zdroje:

$$p(\mathbf{y}) = \frac{1}{|A|}$$

- **Estimátor**

$$F_N = \frac{|A|}{N} \sum_{k=1}^{N} L_\mathrm{e}(\mathbf{y}_k \to \mathbf{x}) \cdot V(\mathbf{y}_k \leftrightarrow \mathbf{x}) \cdot G(\mathbf{y}_k \leftrightarrow \mathbf{x})$$

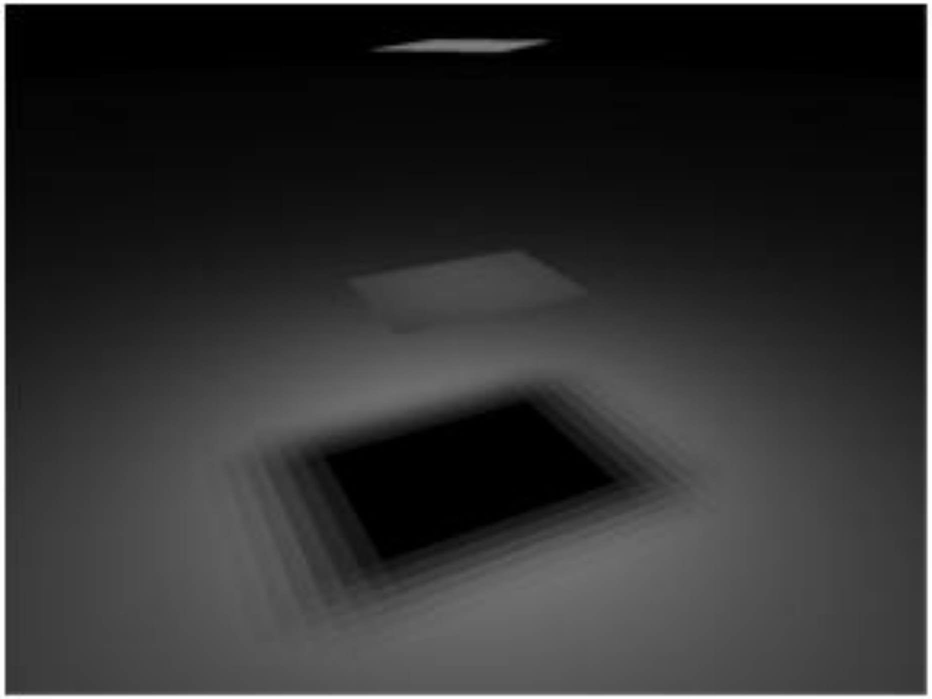
Example – Area Sampling

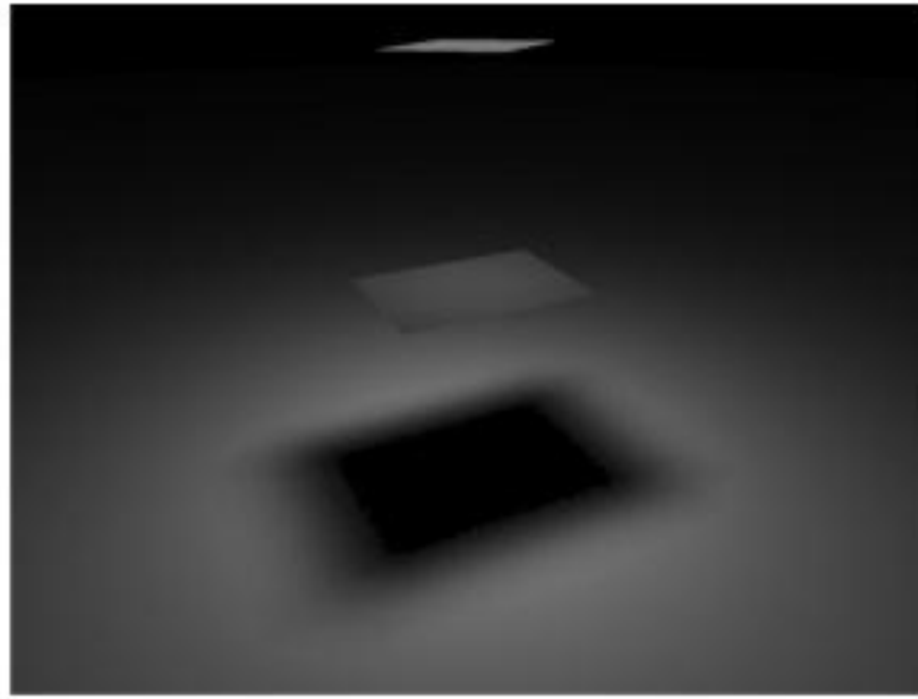


1 shadow ray per eye ray

Center Random

Example – Area Sampling



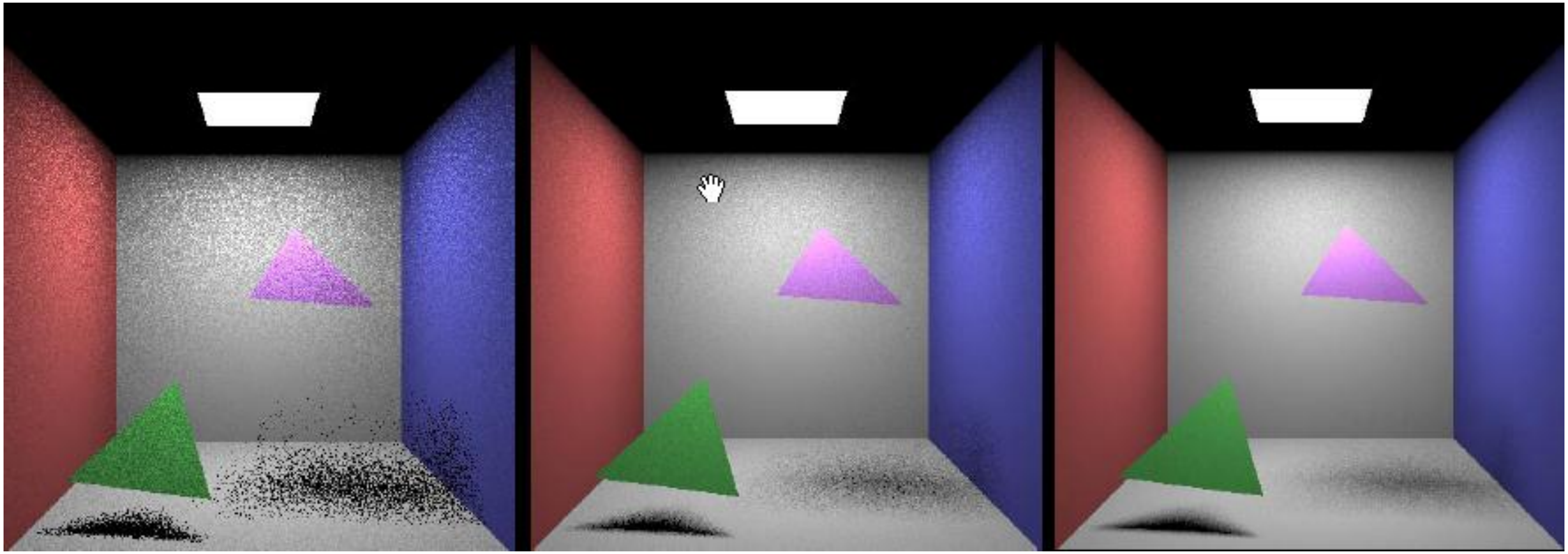


16 shadow rays per eye ray

Uniform grid

Stratified random

Plošné zdroje světla



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

Přímé osvětlení na ploše s obecnou BRDF

- Odhadovaný integrál

$$L_{\mathrm{o}}(\mathbf{x},\omega_{\mathrm{o}}) = \int_A L_{\mathrm{e}}(\mathbf{y}\to\mathbf{x})\cdot f_r(\mathbf{y}\to\mathbf{x}\to\omega_{\mathrm{o}})\cdot V(\mathbf{y}\leftrightarrow\mathbf{x})\cdot G(\mathbf{y}\leftrightarrow\mathbf{x})\,\mathrm{d}A$$

- **Estimátor** (uniformní vzorkování povrchu zdroje)

$$F_N = \frac{|A|}{N}\sum_{k=1}^{N} L_{\mathrm{e}}(\mathbf{y}_k\to\mathbf{x})\cdot f_r(\mathbf{y}_k\to\mathbf{x}\to\omega_{\mathrm{o}})\cdot V(\mathbf{y}_k\leftrightarrow\mathbf{x})\cdot G(\mathbf{y}_k\leftrightarrow\mathbf{x})$$

Nepřímé osvětlení na ploše s obecnou BRDF

- Odhadovaný integrál

$$L_o^{\mathrm{ind}}(\mathbf{x},\omega_o) = \int_{H(\mathbf{x})} L_r(\mathrm{r}(\mathbf{x},\omega_i),-\omega_i)\cdot f_r(\mathbf{x},\omega_i\to\omega_o)\cdot\cos\theta_i\,\mathrm{d}\omega_i$$

- **Estimátor** pro vzorkování směrů s obecnou pdf p(ω)

$$L_o^{\mathrm{ind}}(\mathbf{x},\omega_o) = \frac{1}{N}\sum_{k=1}^{N}\frac{L_r(\mathrm{r}(\mathbf{x},\omega_{i,k}),-\omega_{i,k})\cdot f_r(\mathbf{x},\omega_{i,k}\to\omega_o)\cdot\cos\theta_{i,k}}{p(\omega_{i,k})}$$

PDF úměrná nebo velmi podobná BRDF

Distribution Ray Tracing
Path tracing

Sledování cest od kamery

```
renderImage()
{
  for all pixels
  {
    Color pixelCol = (0,0,0);
    for k = 1 to N
    {
      wk := náhodný směr skrz k-tý pixel
      pixelCol += getLi(camPos,wk)
    }
    return Lo / N
  }
}
```

Distribution Ray Tracing (Cook 84)

```
getLi(x, wi)
{
   hit := NearestIntersect(x, wi)
   wo  := -wi;
   y   := hit.pos;
   if no intersection
        return backgroundCol;
   else
   {
        Lo = (0,0,0)
        for k = 1 to N
        {
         wk := náhodný směr na hemisféře s hustotou p(w)
         Lo += getLi(y, wk) * fr(y, wk, wo) * dot(hit.n,wk) / pdf(wk)
         }
        return Lo / N + directLighting (y, wo);
   }
}
```

Distribution Ray Tracing

- Ad hoc ukončení rekurze
  1. maximální povolená hloubka
  2. minimální povolený příspěvek
  - Oba tyto přístupy vedou k systematické chybě (bias)

- Zásadní problém
  - Exponenciální růst počtu paprsků s hloubkou rekurze

Sledování cest (Path tracing, Kajiya86)

- Pouze jeden sekundární paprsek
  1. Náhodný výběr interakce (ideální lom, difúzní odraz, …)
  2. Importance sampling podle vybrané interakce

- Přímé osvětlení
  - Doufej, že náhodně vygenerovaný paprsek trefí zdroj, anebo
  - Vyber náhodně jeden vzorek na jednom zdroji světla

- Trasuj stovky cest přes každý pixel a zprůměruj výsledek
- Výhoda: žádná exploze počtu paprsků kvůli rekurzi

Path Tracing – Implicitní osvětlení

```
getLi(x, w)
{
   Color thrput = (1,1,1)
   Color accum  = (0,0,0)
   while(1)
   {
        hit = NearestIntersect(x, w)
        if no intersection
                return accum + thrput * bgRadiance(x, w)
        if isOnLightSource(hit)
                accum += thrput * Le(hit.pos, -w)
        ρ = reflectance(hit.pos, -w)
        if rand() < ρ // russian roulette - survive (reflect)
                wi  := SampleDir(hit)
                thrput *= fr(hit.pos, wi, -w) * dot(hit.n, wi) / (ρ*pdf(wi))
                x   := hit.pos
                w   := wi
        else // absorb
            break;
   }
   return accum;
}
```

Ukončení rekurze – Ruská ruleta

- Pokračuj v rekurzi s pravděpodobností q
- Uprav váhu faktorem 1 / q

$$Z = \begin{cases} Y/q & \text{pokud } \xi < q \\ 0 & \text{jinak} \end{cases}$$

$$E[Z] = \frac{E[Y]}{q} \cdot q + 0 \cdot \frac{1}{q-1} = E[Y]$$

Výběr náhodného směru – Importance Sampling

```
getLi(x, w)
{
   Color thrput = (1,1,1)
   Color accum  = (0,0,0)
   while(1)
   {
        hit = NearestIntersect(x, w)
        if no intersection
                return accum + thrput * bgRadiance(x, w)
        if isOnLightSource(hit)
                accum += thrput * Le(hit.pos, -w)
        ρ = reflectance(hit.pos, -w)
        if rand() < ρ // russian roulette – survive (reflect)
                wi  := SampleDir(hit)
                thrput *= fr(hit.pos, wi, -w) * dot(hit.n, wi) / (ρ * pdf(wi))
                x   := hit.pos
                w   := wi
        else // absorb
            break;
   }
   return accum;
}
```

Výběr náhodného směru – Importance Sampling

- Obyčejně vzorkujeme s hustotou „co nejpodobnější" součinu

$$f_r(\omega_\mathrm{i}, \omega_\mathrm{o}) \cos \theta_\mathrm{i}$$

  - Ideálně bychom chtěli vzorkovat podle

$$L_\mathrm{i}(\omega_\mathrm{i})\, f_r(\omega_\mathrm{i}, \omega_\mathrm{o}) \cos \theta_\mathrm{i},$$

  ale to neumíme, protože neznáme L_i

- Co když bude hustota přesně úměrná $f_r(\omega_\mathrm{i}, \omega_\mathrm{o}) \cos \theta_\mathrm{i}$?

„Ideální“ BRDF Importance Sampling

$$p(\omega_\mathrm{i}) \propto f_r(\omega_\mathrm{i} \to \omega_\mathrm{o}) \cdot \cos\theta_\mathrm{i}$$

- Normalizace (integrál pdf musí být = 1)

$$p(\omega_\mathrm{i}) = \frac{f_r(\omega_\mathrm{i} \to \omega_\mathrm{o}) \cdot \cos\theta_\mathrm{i}}{\boxed{\int_{H(\mathbf{x})} f_r(\omega_\mathrm{i} \to \omega_\mathrm{o}) \cdot \cos\theta_\mathrm{i} \, \mathrm{d}\omega_\mathrm{i}}}$$

odrazivost ρ

„Ideální“ BRDF IS v Path Traceru

- Obecná hustota (pdf)

```
...
thrput *= fr(.) * dot(.) / ( ρ * p(wi) )
```

- „Ideální“ BRDF importance sampling

$$p(\omega_{\mathrm{i}}) = f_r(\omega_{\mathrm{i}} \to \omega_{\mathrm{o}}) \cdot \cos\theta_i \;/\; \rho$$

```
...
thrput *= 1
```

Pravděpodobnost přežití cesty

```
getLi(x, w)
{
   Color thrput = (1,1,1)
   Color accum  = (0,0,0)
   while(1)
   {
        hit = NearestIntersect(x, w)
        if no intersection
                return accum + thrput * bgRadiance(x, w)
        if isOnLightSource(hit)
                accum += thrput * Le(hit.pos, -w)
        ρ = reflectance(hit.pos, -w)
        if rand() < ρ // russian roulette – survive (reflect)
                wi  := SampleDir(hit)
                thrput *= fr(hit.pos, wi, -w) * dot(hit.n, wi) / (ρ * p(wi))
                x   := hit.pos
                w   := wi
        else // absorb
            break;
   }
   return accum;
}
```

Pravděpodobnost přežití cesty

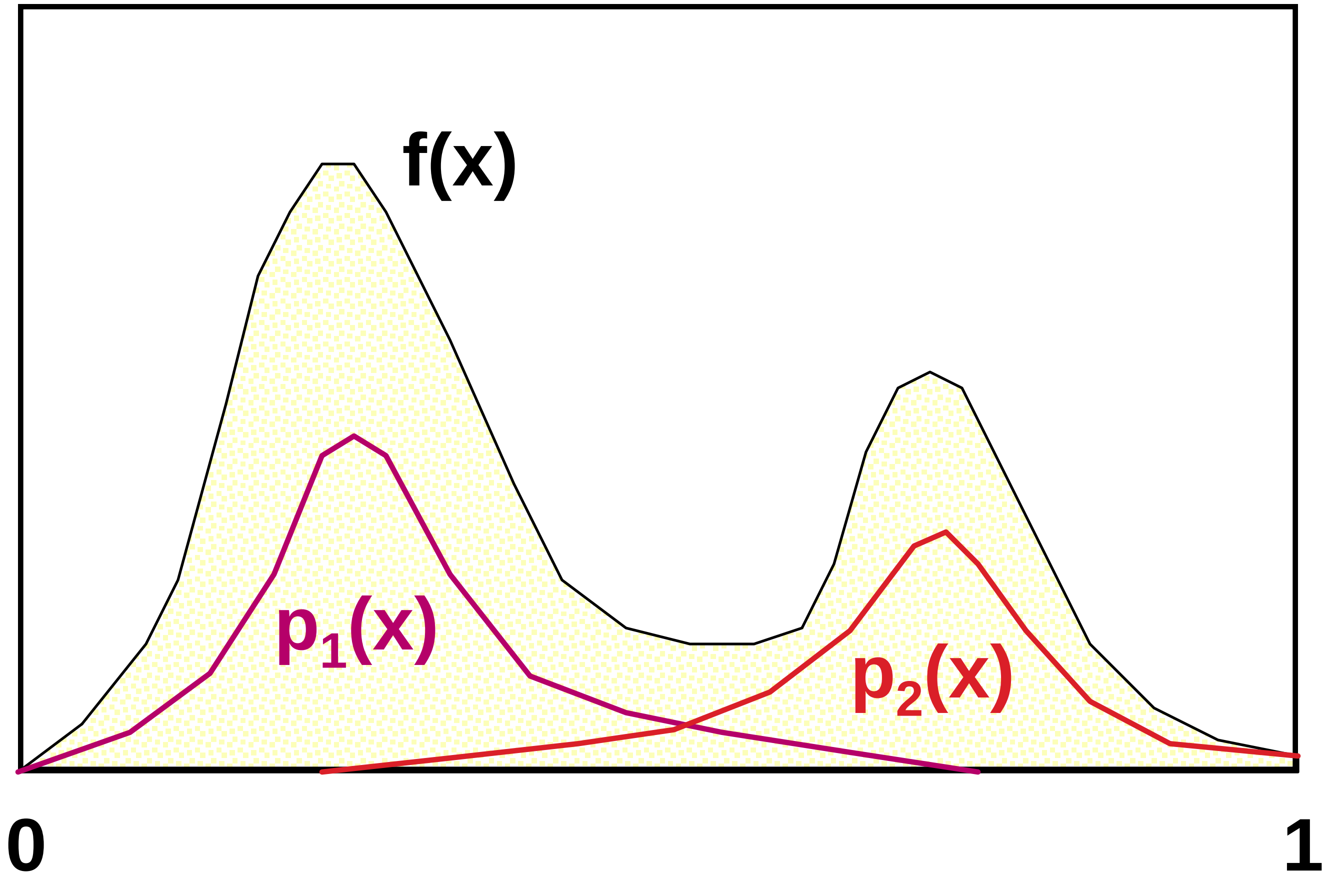
- Použití odrazivosti ρ jako p-nosti přežití dává smysl
  - Pokud plocha odráží jen 30% energie, pokračujeme pouze s 30% pravděpodobností.

- Co když neumím spočítat ρ ?
- Alternativa
  1. Nejdříve vygeneruj náhodný směr podle $p(\omega_i)$
  2. $$q_{\text{survival}} = \min\left\{ 1, \; \frac{f_r(\omega_i \to \omega_o)\cos\theta_i}{p(\omega_i)} \right\}$$
  - Pro „ideální“ BRDF IS stejné jako původní metoda

Zpět k obecnému MC integrování – „Multiple Importance Sampling“

Multiple Importance Sampling

(Veach & Guibas, 95)



Multiple importance sampling

- Máme dáno *n* vzorkovacích „technik" (hustot pravděpodobnosti) $p_1(x), \ldots, p_n(x)$
- Z každé techniky (hustoty) vybereme n_i vzorků $X_{i,1}, \ldots, X_{i,n_i}$

- **Kombinovaný estimátor**

kombinační váhy
(mohou být různé pro každý vzorek)

$$\boxed{F = \sum_{i=1}^{n} \frac{1}{n_i} \sum_{j=1}^{n_i} w_i(X_{i,j}) \frac{f(X_{i,j})}{p_i(X_{i,j})}}$$

vzorkovací techniky

vzorky z jednotlivých technik

Nestrannost kombinovaného odhadu

$$E[F] = \ldots = \int \left[\sum_{i=1}^{n} w_i(x) \right] f(x)\,\mathrm{d}x \equiv \int f(x)$$

- Podmínka pro váhové funkce

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

Volba váhových funkcí

- **Cíl:** minimalizovat rozptyl kombinovaného estimátoru

1. Aritmetický průměr (velmi špatná kombinace)

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

2. Vyrovnaná heuristika (velmi dobrá kombinace)
    - ....

Vyrovnaná heuristika (Balance heurist.)

- Kombinační váhy

$$\hat{w}_i(\mathbf{x}) = \frac{n_i p_i(\mathbf{x})}{\sum_k n_k p_k(\mathbf{x})}$$

- Výsledný estimátor (po dosazení vah)

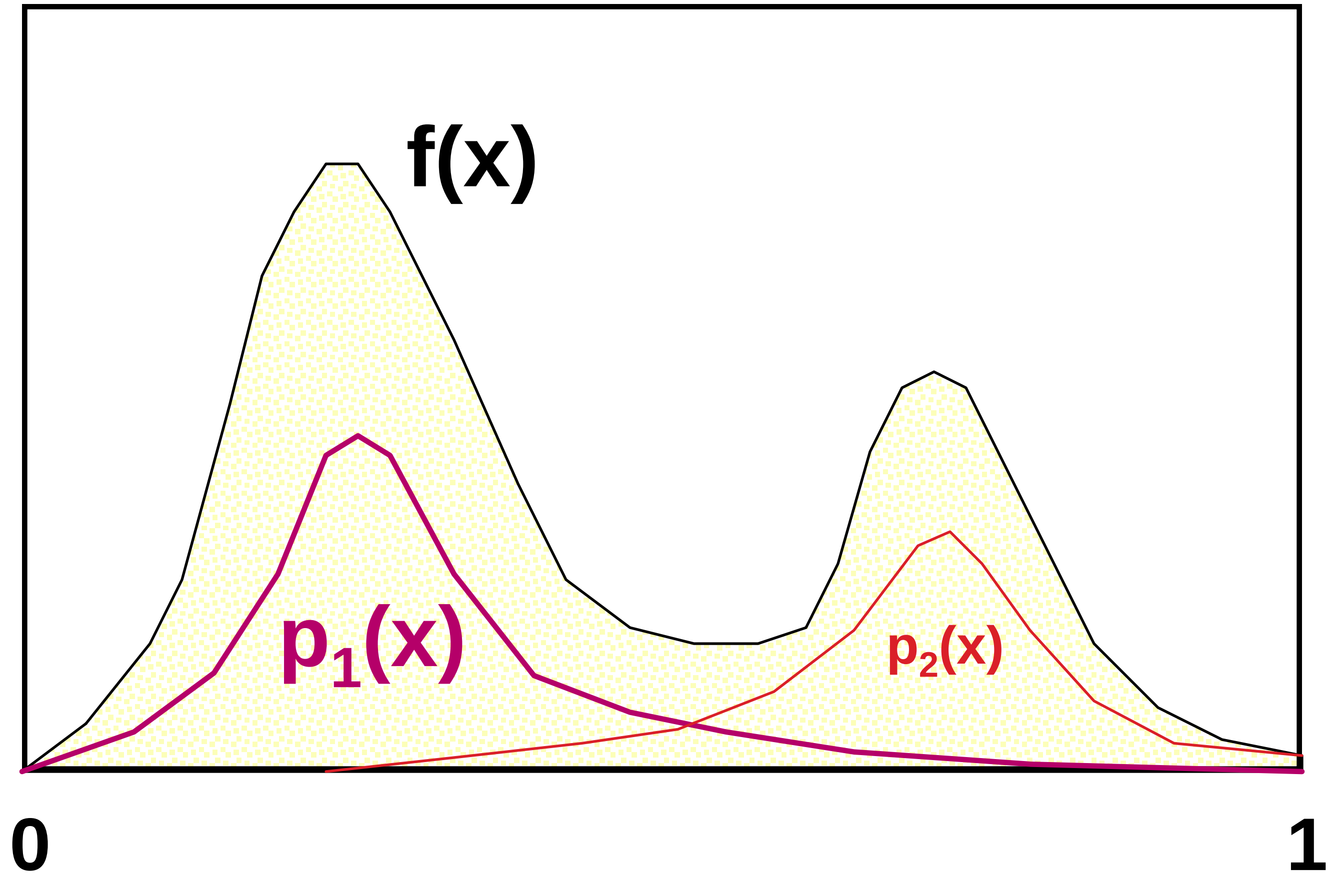
$$\boxed{F = \sum_{i=1}^{n} \sum_{j=1}^{n_i} \frac{f(X_{i,j})}{\sum_k n_k p_k(X_{i,j})},}$$

  - příspěvek vzorku nezávisí na tom, ze které byl pořízen techniky (tj. pdf)

Vyrovnaná heuristika (Balance heurist.)

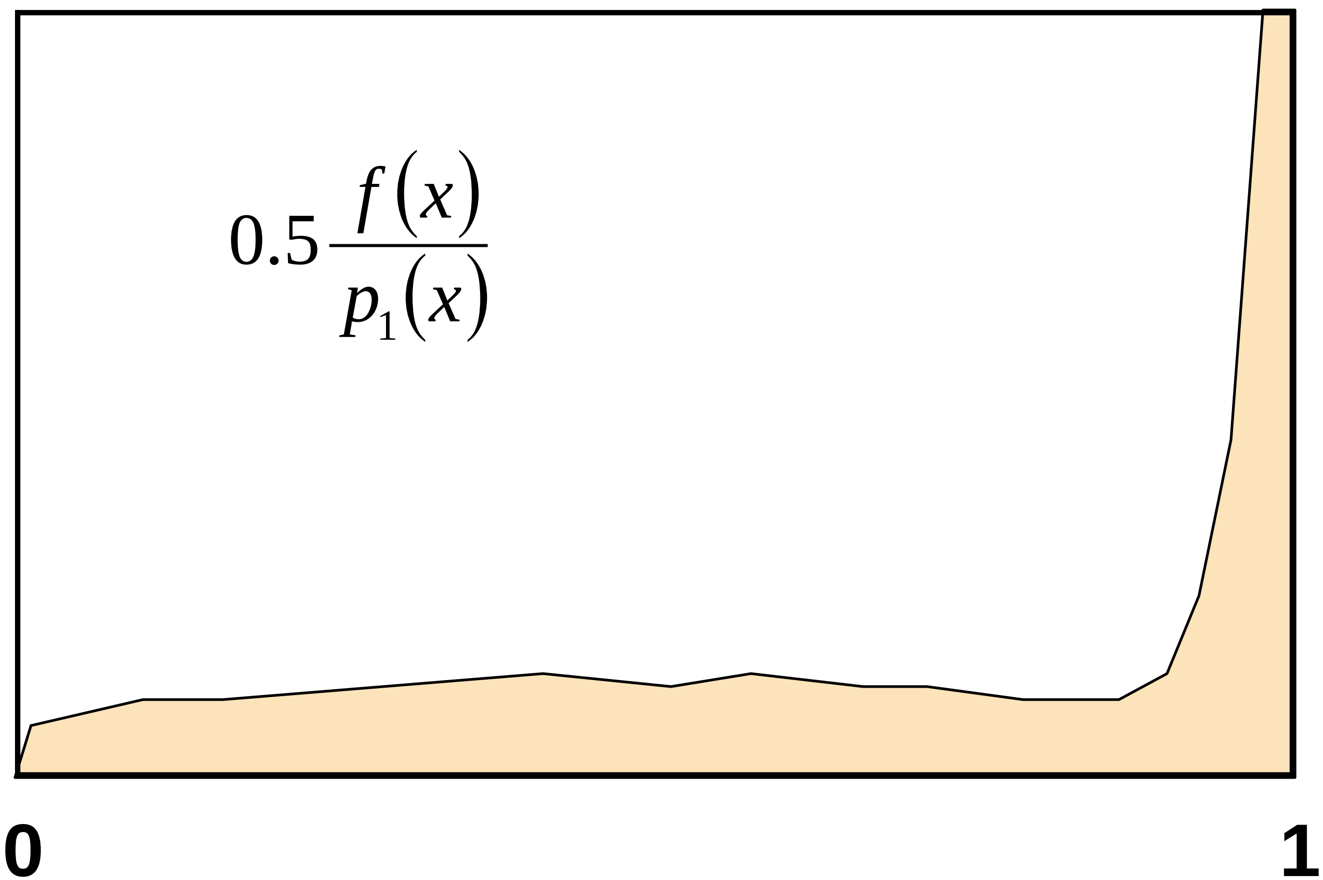
- Vyrovnaná heuristika je téměř optimální
  - Žádný kombinovaný estimátor nemůže mít rozptyl „o mnoho“ menší než vyrovnaná heuristika
- Další možné kombinační heuristiky
  - Maximální heuristika
  - Mocninná heuristika
  - viz. Veach 1997

Jeden člen kombinovaného odhadu



Aritmetický průměr

$$0.5\frac{f(x)}{p_1(x)} + 0.5\frac{f(x)}{p_2(x)}$$



Vyrovnaná heuristika