Monte Carlo integration

integrals and averages

integral of a function over a domain

$$\int_{\mathbf{x}\in D} f(\mathbf{x})\,dA_{\mathbf{x}}$$

"size" of a domain

$$A_D = \int_{\mathbf{x}\in D} dA_{\mathbf{x}}$$

average of a function over a domain

$$\frac{\int_{\mathbf{x}\in D} f(\mathbf{x})\,dA_{\mathbf{x}}}{\int_{\mathbf{x}\in D} dA_{\mathbf{x}}} = \frac{\int_{\mathbf{x}\in D} f(\mathbf{x})\,dA_{\mathbf{x}}}{A_D}$$

integrals and averages examples

- average "daily" snowfall in Hanover last year
  - domain: year - time interval (1D)
  - integration variable: "day" of the year
  - function: snowfall of "day"

$$\frac{\int_{day \in year} s(day) dlength(day)}{lenght(year)}$$

integrals and averages examples

- "today" average snowfall in New Hampshire
  - domain: New Hamshire - surface (2D)
  - integration variable: "location" in New Hampshire
  - function: snowfall of "location"

$$\frac{\int_{location \in NewHampshire} s(location) darea(location)}{area(location)}$$

integrals and averages examples

- “today” average snowfall in New Hampshire
  - domain: New Hamshire x year – area x time (3D)
  - integration variables: “location” and “day” in New Hampshire this year
  - function: snowfall of “location” and “day”

$$\frac{\int_{day \in year} \int_{loc \in NewHampshire} s(loc, day) darea(loc) dlength(day)}{area(loc) length(day)}$$

discreet random variable

- random variable: x
- values: $x_0, x_1, \ldots, x_n$
- probabilities: $p_0, p_1, \ldots, p_n$ $\sum_{j=1}^{n} p_j = 1$

- example: tossing a die
  - values:

    $$x_1 = 1, x_2 = 2, x_3 = 3, x_4 = 4, x_5 = 5, x_6 = 6$$

  - probabilities:

    $$p_1 = 1/6, p_2 = 1/6, p_3 = 1/6, p_4 = 1/6, p_5 = 1/6, p_6 = 1/6$$

expected value and variance

- expected value: $$E[x] = \sum_{j=1}^{n} v_j p_j$$
  - average value of the variable
- variance: $$\sigma^2[x] = E[(x - E[x])^2]$$
  - how much different from the average
  - property: $$\sigma^2[x] = E[x^2] - E[x]^2$$
- example: tossing a die
  - expected value: $$E[x] = (1+2+3+4+5+6)/6 = 3.5$$
  - variance: $$\sigma^2[x] = \ldots = 0.916$$

estimating expected values

- to estimate the expected value of a variable
  - choose a set of random *values* based on the prob.
  - average their results

$$E[x] \approx \frac{1}{N}\sum_{i=1}^{N} x_i$$

- larger N give better estimate

- example: rolling a die
  - roll 3 times:
  - roll 9 times:

$$\{3,1,6\} \rightarrow E[x] \approx (3+1+6)/3 = 3.33$$

$$\{3,1,6,2,5,3,4,6,2\} \rightarrow E[x] \approx 3.51$$

law of large numbers

- by taking *infinitely* many samples, the error between the estimate and the expected value is *statistically* zero
  - the estimate will converge to the right value

$$\text{probability}\left[E[x] = \lim_{N \to \infty} \frac{1}{N} \sum_{i=1}^{N} x_i \right] = 1$$

continuous random variable

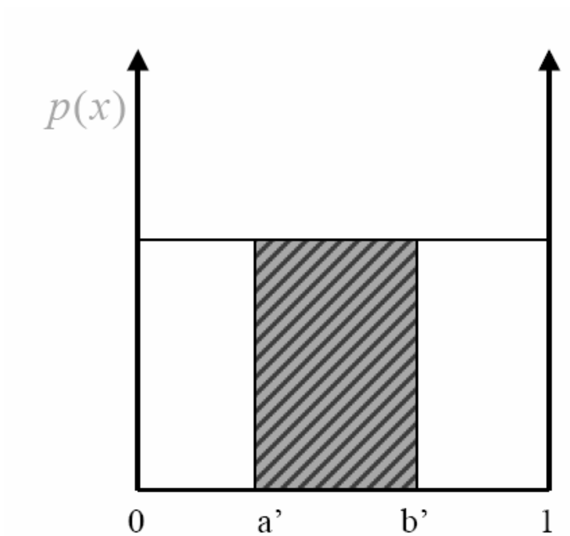
- random variable: x
- values: $x \in [a,b]$
- probability density function: $x \sim p$
  - property: $\int_a^b p(x)dx = 1$
- probability that var. has value *x*: $p(x)dx$

uniformly distributed random variable

- p is the same everywhere in the interval

$p(x) = \text{const}$ and $\int_a^b p(x)dx = 1$ implies

$$p(x) = 1/(b-a)$$



[Bala]

expected value and variance

- expected value:

$$E[x] = \int_a^b x p(x) dx$$

$$E[g(x)] = \int_a^b g(x) p(x) dx$$

- variance:

$$\sigma^2[x] = \int_a^b (x - E[x])^2 p(x) dx$$

$$\sigma^2[g(x)] = \int_a^b (g(x) - E[g(x)])^2 p(x) dx$$

- estimating expected values:

$$E[g(x)] \approx \frac{1}{N} \sum_{i=1}^N g(x_i)$$

multidimensional random variables

- everything works fine in multiple dimensions
  - but it is often hard to precisely define domain
    - except in simple cases

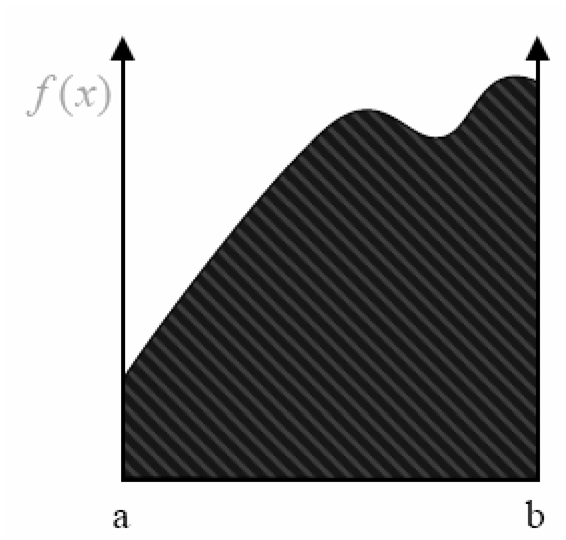
$$E[g(\mathbf{x})] = \int_{\mathbf{x} \in D} g(\mathbf{x})\, p(\mathbf{x})\, dA_{\mathbf{x}}$$

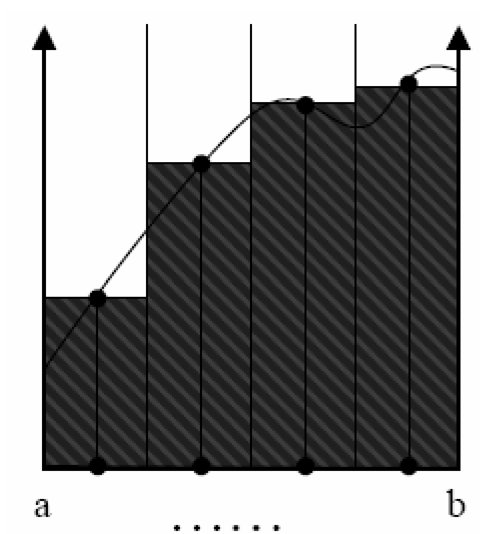
deterministic numerical integration

- split domain in set of fixed segments
- sum function values times size of segments

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

$$I \approx \sum_j f(x_j)\Delta x$$





[Bala]

Monte Carlo numerical integration

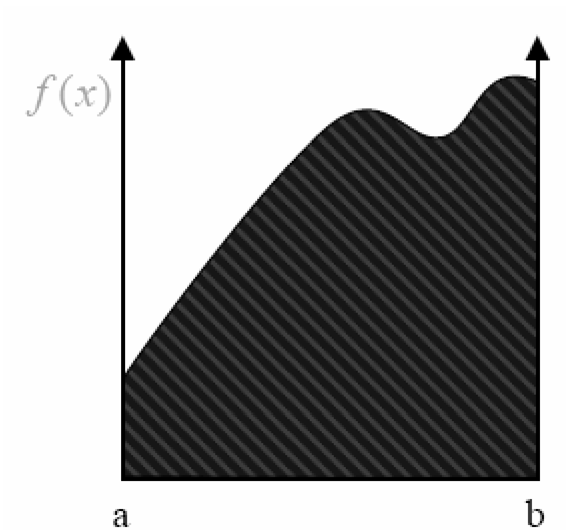
- need to evaluate $$I = \int_a^b f(x)dx$$
- by definition $$E[g(x)] = \int_a^b g(x)\,p(x)dx$$
- can be estimated as $$E[g(x)] \approx \frac{1}{N}\sum_{i=1}^{N} g(x_i)$$
- by substituting $$g(x) = f(x)/p(x)$$

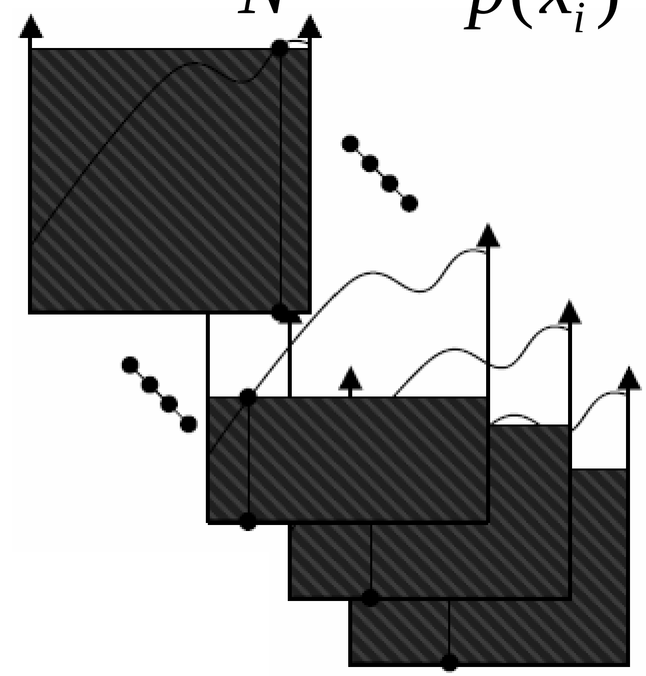
$$I = \int_a^b \frac{f(x)}{p(x)}\,p(x)dx \approx \frac{1}{N}\sum_{i=1}^{N}\frac{f(x_i)}{p(x_i)}$$

Monte Carlo numerical integration

- intuition: compute the area randomly and average the results

$$I = \int_a^b f(x)dx \qquad I \approx \frac{1}{N}\sum_{i=1}^{N} \frac{f(x_i)}{p(x_i)}$$





[Bala]

Monte Carlo numerical integration

- formally, we can prove that

$$\bar{I} = \frac{1}{N}\sum_{i=1}^{N}\frac{f(x_i)}{p(x_i)} \Rightarrow E[\bar{I}] = E[g(x)]$$

- meaning that if we were to try multiple times to evaluate the integral using our new procedure, we would get, on average, the same result

- variance of the estimate: $\sigma^2[\bar{I}] = \frac{1}{N}\sigma^2[g(x)]$

example: integral of constant function

- analytic integration

$$I = \int_a^b f(x)dx = \int_a^b kdx = k(b-a)$$

- Monte Carlo integration

$$I = \int_a^b f(x)dx = \int_a^b kdx \approx$$

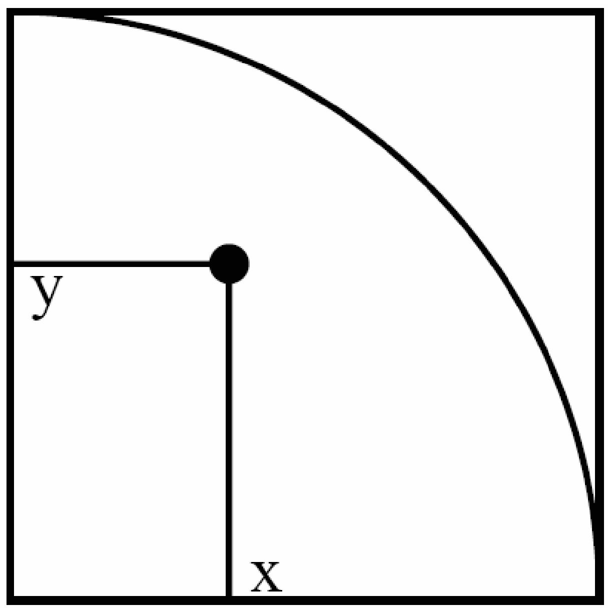
$$\approx \frac{1}{N}\sum_{i=1}^{N}\frac{f(x_i)}{p(x_i)} = \frac{1}{N}\sum_{i=1}^{N}k(b-a) =$$

$$= \frac{N}{N}k(b-a) = k(b-a)$$

example: computing pi

- take the square $[0,1]^2$, with a quarter-circle in it

$$A_{circle} = \int_0^1 \int_0^1 f(x,y)\,dx\,dy \qquad f(x,y) = \begin{cases} 1 & (x,y) \in circle \\ 0 & \text{otherwise} \end{cases}$$

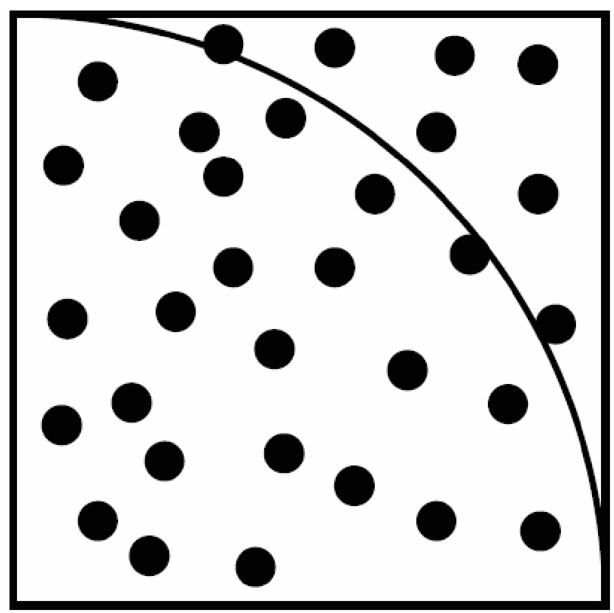


[MIT OpenCourseWare]

example: computing pi

- estimate area of circle by tossing point in the plane and evaluating *f*

$$A_{circle} \approx \frac{1}{N}\sum_{i=1}^{N} f(x_i, y_i)$$



[MIT OpenCourseWare]

example: computing pi

- by definition $A_{circle} = \pi / 4$

- numerical estimation of pi

$$\pi \approx \frac{4}{N} \sum_{i=1}^{N} f(x_i, y_i)$$

Monte Carlo numerical integration

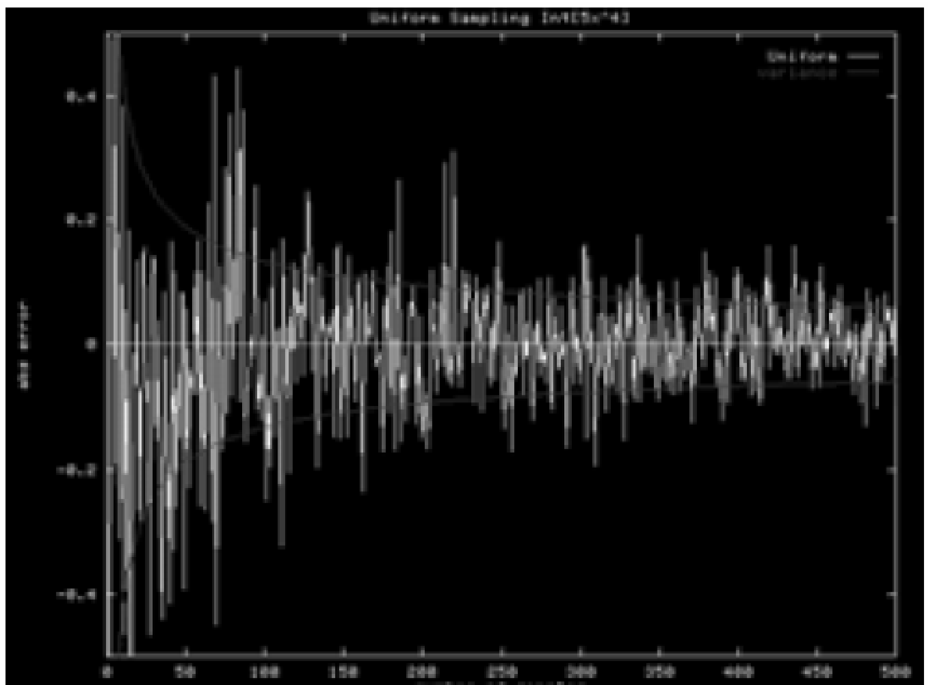
- works in any dimension!
  - need to carefully pick the points
  - need to properly define the pdf
    - hard for complex domain shapes
    - e.g. how to uniformly sample a sphere?

$$I = \int_{\mathbf{x}\in D} f(\mathbf{x})\,dA_{\mathbf{x}} \approx \frac{1}{N}\sum_{i=1}^{N}\frac{f(\mathbf{x})}{p(\mathbf{x})}$$

- works for badly-behaving functions!

Monte Carlo numerical integration

- expected value of the error is $O\left(1/\sqrt{N}\right)$
  - convergence does not depend on dimensionality
  - deterministic integration is hard in high dimensions



[Bala]

importance sampling principle

- how to minimize the noise?
- pick samples in area where function is large

$$I \approx \frac{1}{N} \sum_{i=1}^{N} \frac{f(\mathbf{x})}{p(\mathbf{x})}$$

- pick a distribution similar to the function

$$p_{\text{optimal}} \propto f(\mathbf{x})$$

1D interval - uniform sampling

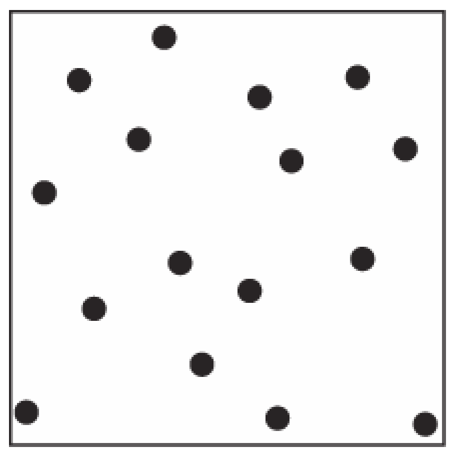
- call random function in [0,1), rescale if necessary
  - in reality only pseudorandom
    - relies on good generator

$$r = rand() \Rightarrow x = r$$

2D square – uniform sampling

- pick two *independent* random values in [0,1)

$$\mathbf{r} = \left[rand() \;\; rand() \right]^T \Rightarrow \mathbf{x} = \mathbf{r}$$

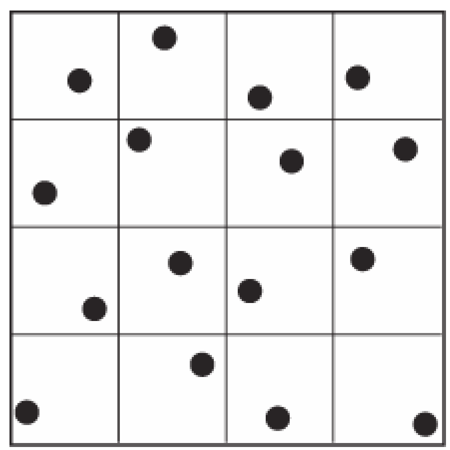


[Shirley]

2D square – stratified sampling

- divide domain in smaller domains, then pick random points in each
  - better variance than normal sampling

$$\mathbf{x} = \begin{bmatrix} \frac{i + r_x}{n_i} & \frac{j + r_y}{n_j} \end{bmatrix}^T$$

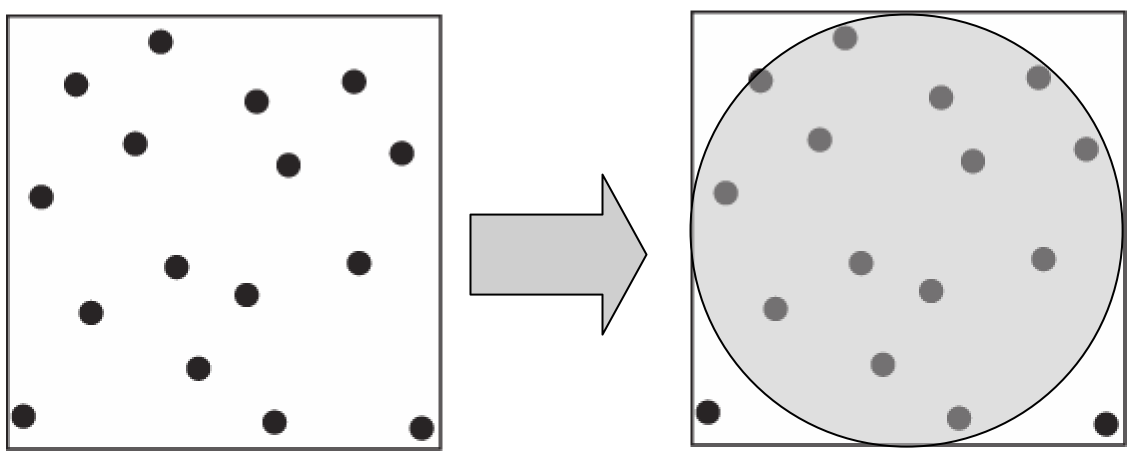


[Shirley]

2D circle – rejection sampling

- pick random points in the uniform square and discard the ones outside the domain

$$\begin{cases} \mathbf{r} \in [0,1]^2 \\ \mathbf{r} \sim 1 \end{cases} \Rightarrow \mathbf{x} = \begin{cases} 2\mathbf{r}-1 & |2\mathbf{r}-1| \leq 1 \\ discard & otherwise \end{cases}$$

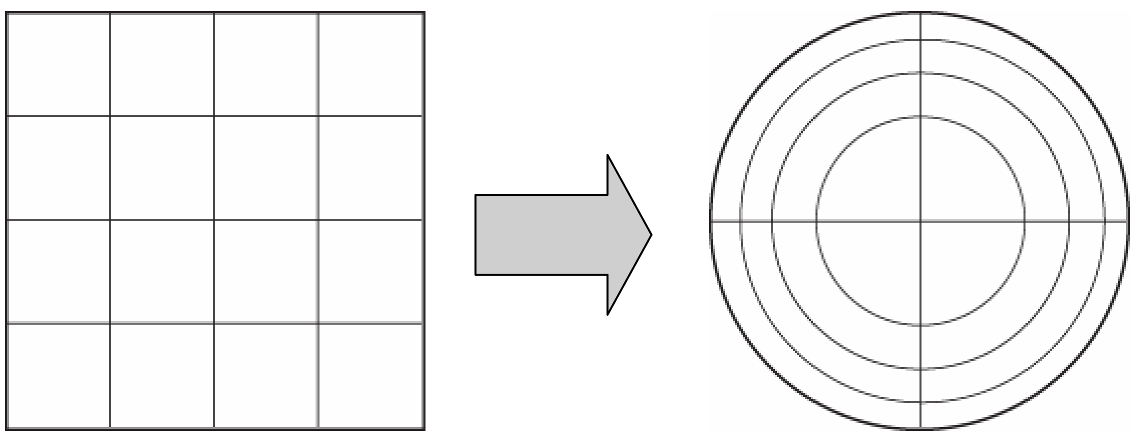


[Shirley]

2D circle – remapping from square

- pick random points in the uniform square and remap them on the circle using polar coords.

$$\begin{cases} \mathbf{r} \in [0,1]^2 \\ \mathbf{r} \sim 1 \end{cases} \Rightarrow \begin{cases} \varphi = 2\pi r_x \\ r = r_y \end{cases} \Rightarrow \begin{cases} \mathbf{x} = \left(r_y \cos(2\pi r_x), r_y \sin(2\pi r_x)\right) \\ \mathbf{x} \sim non\ uniform \end{cases}$$

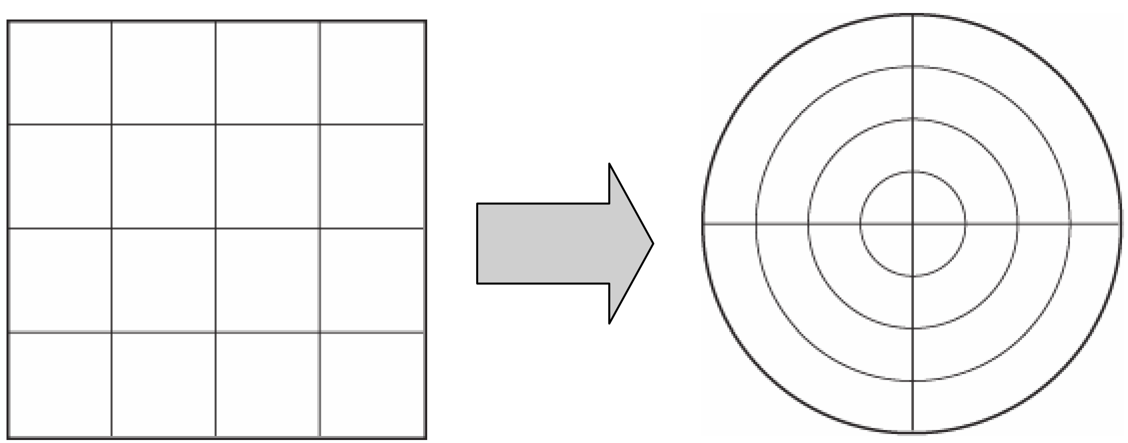


[Shirley]

2D circle – remapping from square

- make sampling uniform by computing proper pdf
  - Sh. 14.4.1

$$\begin{cases} \mathbf{r} \in [0,1]^2 \\ \mathbf{r} \sim 1 \end{cases} \Rightarrow \begin{cases} \varphi = 2\pi r_x \\ r = \sqrt{r_y} \end{cases} \Rightarrow \begin{cases} \mathbf{x} = \left(\sqrt{r_y}\cos(2\pi r_x), \sqrt{r_y}\sin(2\pi r_x)\right) \\ \mathbf{x} \sim 1/\pi \end{cases}$$



[Shirley]

3D direction – uniform distribution

- uniformly distribution with respect to solid angle

$$\begin{cases} \mathbf{r} \in [0,1]^2 \\ \mathbf{r} \sim 1 \end{cases} \Rightarrow \begin{cases} \varphi = 2\pi r_x \\ \theta = \arccos(r_y) \end{cases} \Rightarrow$$

$$\Rightarrow \begin{cases} \mathbf{d} = \left(\sqrt{1-r_y^2}\cos(2\pi r_x), \sqrt{1-r_y^2}\sin(2\pi r_x), r_y\right) \\ \mathbf{d} \sim \dfrac{1}{2\pi} \end{cases}$$

3D direction – cosine distribution

- cosine distribution with respect to solid angle

$$\begin{cases} \mathbf{r} \in [0,1]^2 \\ \mathbf{r} \sim 1 \end{cases} \Rightarrow \begin{cases} \varphi = 2\pi r_x \\ \theta = \arccos\left(\sqrt{r_y}\right) \end{cases} \Rightarrow$$

$$\Rightarrow \begin{cases} \mathbf{d} = \left(\sqrt{1-r_y}\cos(2\pi r_x), \sqrt{1-r_y}\sin(2\pi r_x), \sqrt{r_y}\right) \\ \mathbf{d} \sim \dfrac{\cos(\theta)}{\pi} \end{cases}$$

3D direction – cosine power distribution

- cosine power distribution wrt solid angle

$$\begin{cases} \mathbf{r} \in [0,1]^2 \\ \mathbf{r} \sim 1 \end{cases} \Rightarrow \begin{cases} \varphi = 2\pi r_x \\ \theta = \arccos\left(r_y^{1/(n+1)}\right) \end{cases} \Rightarrow$$

$$\Rightarrow \begin{cases} \mathbf{d} = \left(\sqrt{1 - r_y^{\frac{2}{n+1}}} \cos(2\pi r_x), \sqrt{1 - r_y^{\frac{2}{n+1}}} \sin(2\pi r_x), r_y^{\frac{1}{n+1}} \right) \\ \mathbf{d} \sim \frac{n+1}{2\pi} \cos^n(\theta) \end{cases}$$

