Course 09/2

Monte Carlo integration through uncorrelated sampling

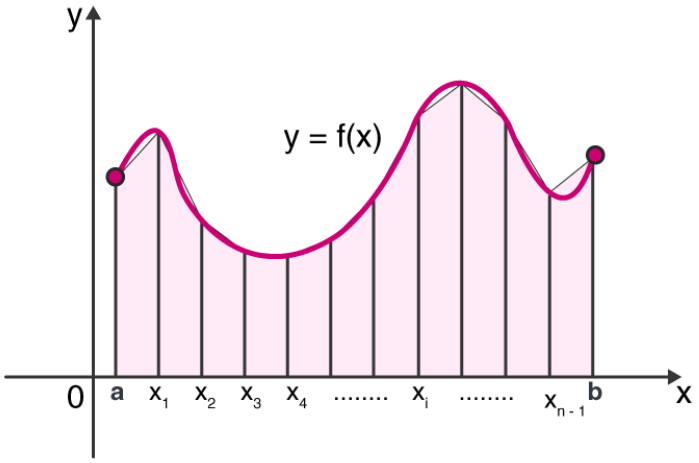
- Motivation
- Integration by Monte Carlo: Method
- Scaling of errors
  - Monte Carlo in 1D
  - Trapezoidal rule in 1D
  - Multidimensional integrals

Motivation

We would like to calculate integrals over many variables. For instance, consider the partition function for N particles:

$$\mathcal{Z} = \int \underbrace{d^3r_1 \ldots d^3r_N}_{3N \text{ variables}} \exp\left[-\frac{1}{k_B T} \sum_{i<j} V(r_{ij}) \right]$$

Suppose we want to use 10 points for every variable. Then, the cost scales like 10^{3N}.

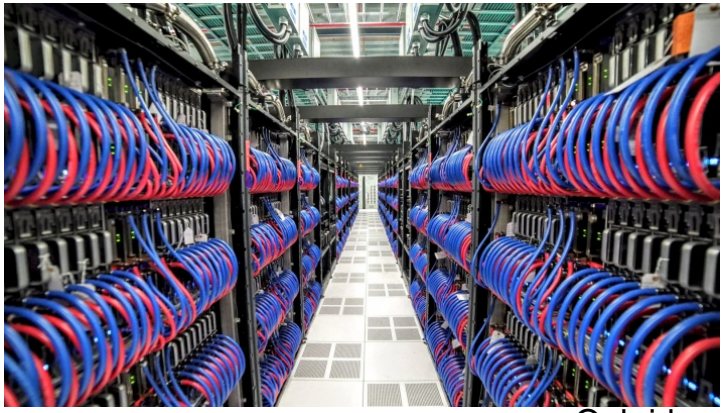


But, in practice, how much time does it take for the case of $N = 20$ particles ?

Performance of supercomputers

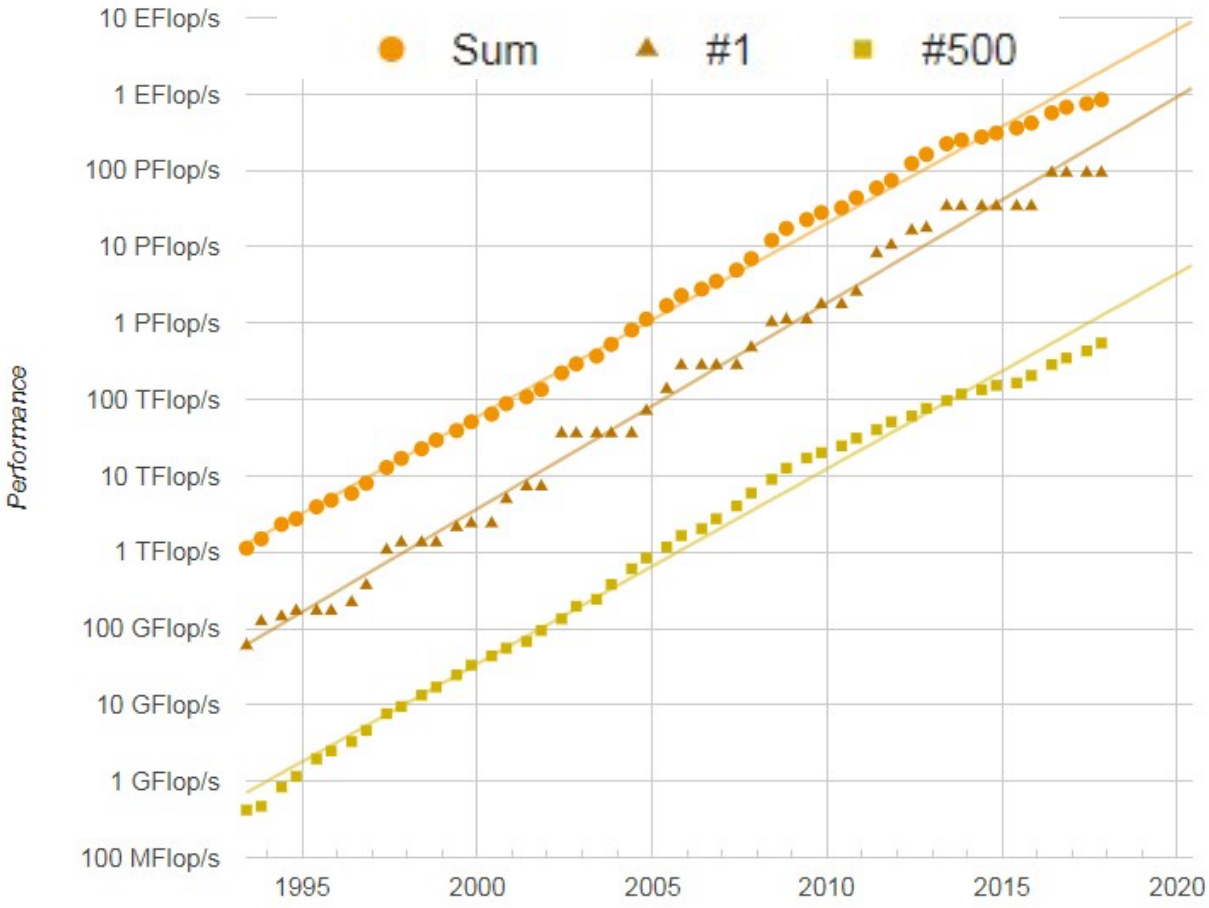
Let us use the fastest computer in the world. At this time (2022), the CRAY Frontier installed in Oakridge is the fastest and achieves:

~ 1.1 Eflop = 10^{18} evaluations per second.



Oakridge

Value		SI	
1000	10^3	k	kilo
1000^2	10^6	M	mega
1000^3	10^9	G	giga
1000^4	10^{12}	T	tera
1000^5	10^{15}	P	peta
1000^6	10^{18}	E	exa
1000^7	10^{21}	Z	zetta
1000^8	10^{24}	Y	yotta



Motivation: Our problem with $N = 20$ particles

Calculation of required time for integrating the partition function Z

$$\text{Time} = \frac{\#\ \text{operations}}{\#\ \text{operations/second}} = \frac{10^{3N}}{1\ \text{Eflop}} = \frac{10^{60}}{10^{18}\ /\text{s}} = 10^{42}\ \text{seconds}$$

$$= \quad 2 \cdot 10^{24} \quad \text{age of the universe}$$

Faster algorithms are needed for multidimensional integrals e.g. in

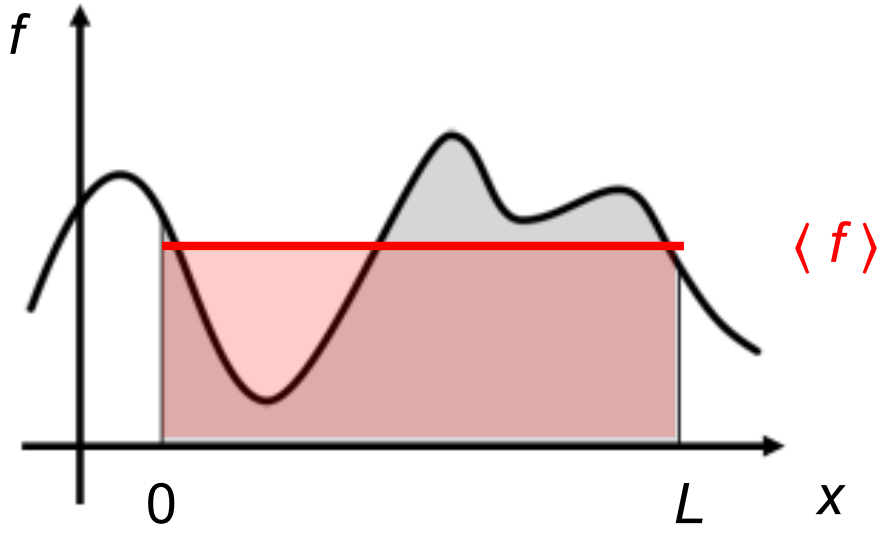
– statistical mechanics

– quantum mechanics

Integration by Monte Carlo: Method

Let us focus on integrals in one dimension, even if, at the end, it will turn out that Monte Carlo methods are more convenient than conventional schemes only for integrals in higher dimensions.

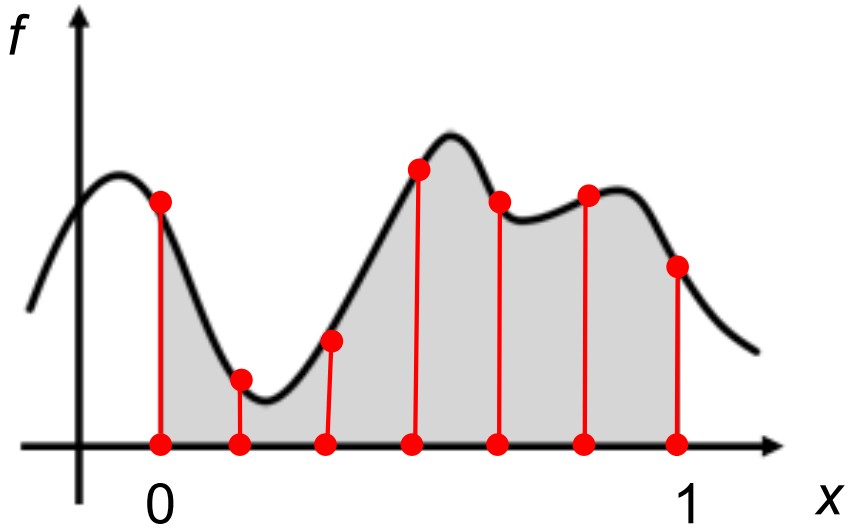
$$I = \int_0^L f(x)\,dx = L \cdot \frac{1}{L} \int_0^L f(x)\,dx = L \cdot \langle f \rangle$$



Integration by Monte Carlo: Method

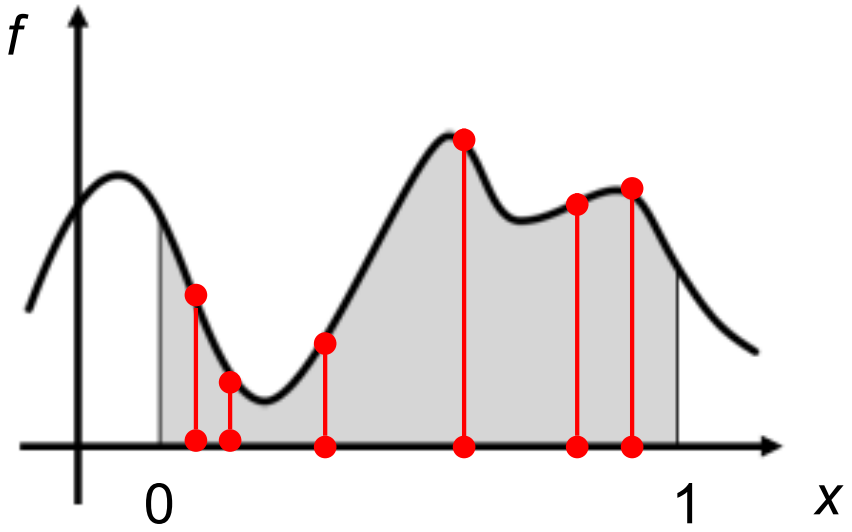
Without loss of generality, let us take $L = 1$.

$$I = \int_0^1 f(x)\,dx = \langle f \rangle \approx \frac{1}{N}\sum_{i=1}^{N} f(x_i)$$



Regular grid integration

x_i regularly distributed



Monte Carlo integration

x_i distributed according to a uniform random distribution

Integration by Monte Carlo: Error

$$I = \int_0^1 f(x)\,dx = \langle f \rangle \approx \frac{1}{N}\sum_{i=1}^{N} f(x_i)$$

We can obtain an expression for the error by realizing that

x_i is a random number → $f(x_i)$ is a random number → I is an average of random numbers

When the random numbers are uncorrelated (independent), the variance of I is related to the variance of f. For N points:

$$\sigma_I^2 = \frac{1}{N}\sigma_f^2 = \frac{1}{N}\left[\langle f^2 \rangle - \langle f \rangle^2\right] = \frac{1}{N}\left[\frac{1}{N}\sum_{i=1}^{N} f^2(x_i) - \left(\frac{1}{N}\sum_{i=1}^{N} f(x_i)\right)^2\right]$$

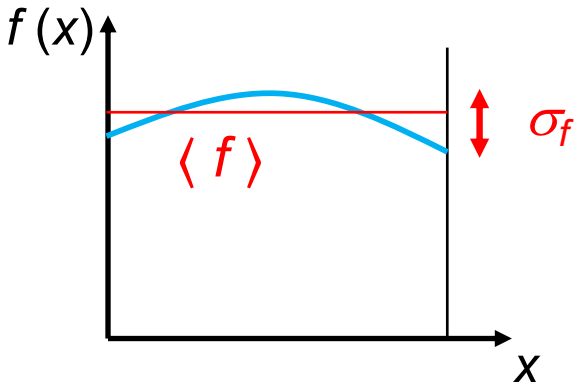
Integration by Monte Carlo: Error

$$\sigma_I^2 = \frac{1}{N}\,\sigma_f^2$$

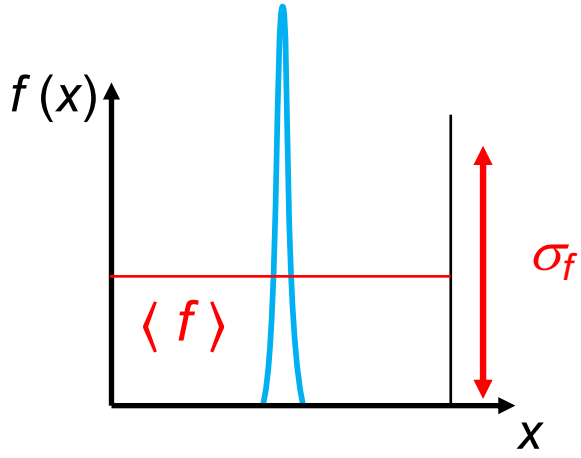
NB

1. $\sigma_I \propto \frac{1}{\sqrt{N}}$ Scaling of error with N : 4× more points to reduce the error by a factor of 2 !

2. $\sigma_I \propto \sigma_f$ The smoother f , the smaller σ_I !

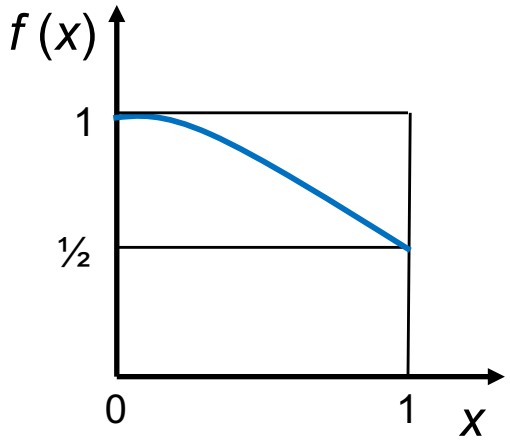


good case for
Monte Carlo integration



bad case for
Monte Carlo integration

Integration by Monte Carlo: Example



$$I = \int_0^1 \frac{1}{1+x^2}\,dx$$

$$= \text{atan } x\Big|_0^1 = \frac{\pi}{4}$$

$$\cong 0.78540$$

N	I	σ_I
10	0.81491	0.04638
20	0.73535	0.03392
50	0.79606	0.02259
100	0.79513	0.01632
200	0.78677	0.01108
500	0.78242	0.00719
1000	0.78809	0.00508
2000	0.78790	0.00363
5000	0.78963	0.00227

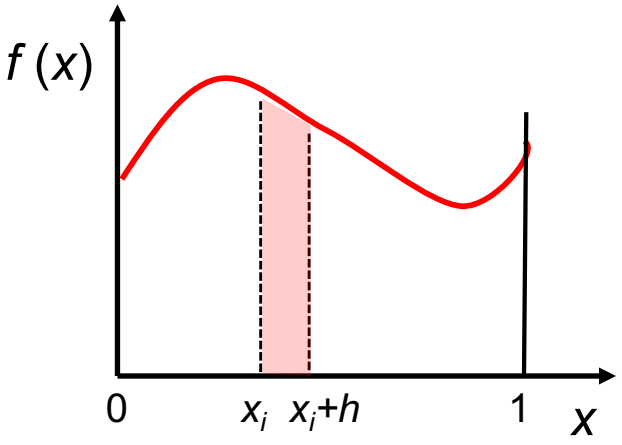
$\frac{\pi}{4}$ 0.78540

Scaling of error with trapezoidal-rule integration (1D)

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

To calculate the integral we use N equidistant points, separated by $h = 1 / N$.

Let us focus on one slice corresponding to the integral between x_i and $x_i + h$.



$$\int_{x_i}^{x_i+h} f(x)\, dx = \int_0^h f(x_i + x')\, dx'$$

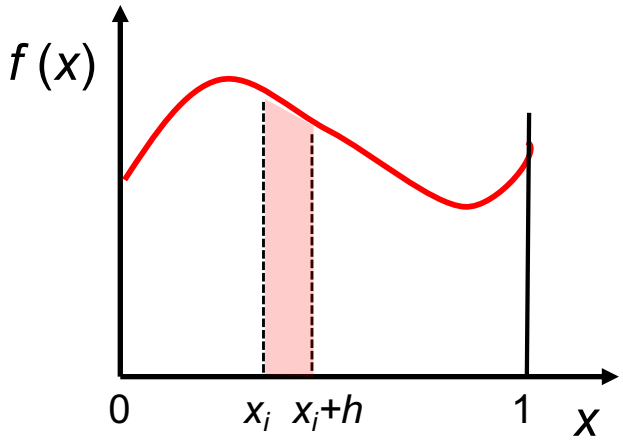
$$\cong \int_0^h [\, f(x_i) + x' f'(x_i) + \ldots \,]\, dx'$$

$$\cong f(x_i) \cdot h + f'(x_i) \cdot \frac{h^2}{2} + o(h^3)$$

Scaling of error with trapezoidal-rule integration (1D)

Integral between x_i and $x_i + h$

$$\int_{x_i}^{x_i+h} f(x)\,dx \quad \cong \quad f(x_i)\cdot h \;+\; f'(x_i)\cdot \frac{h^2}{2} \;+\; o(h^3)$$



Expression for the derivative

$$f'(x_i) \;=\; \frac{f(x_{i+1}) - f(x_i)}{h} + o(h)$$

By combining these expressions, we then obtain

$$\int_{x_i}^{x_i+h} f(x)\,dx \quad \cong \quad \frac{f(x_{i+1}) + f(x_i)}{2} \cdot h \;+\; o(h^3)$$

error on integral over [x_i , $x_i + h$]

The integral I corresponds to N slices, so the global error for I is N times larger:

$$\sigma_I \;\propto\; N\cdot h^3 \;\propto\; N\cdot\left(\frac{1}{N}\right)^3 \;\propto\; \frac{1}{N^2}$$

Scaling of errors for multidimensional integrals

$$I = \int_0^1 dx_1 \ \ldots\ldots \int_0^1 dx_d \quad f(x_1, x_2, \ldots x_d) \quad = \int_D d^d\vec{x}\ f(\vec{x})$$

Estimate of integral in Monte Carlo integration

$$I \cong \frac{1}{N} \sum_{i=1}^{N} f(\vec{x}_i)$$

where $\vec{x}^{\,i} = (x_1^i, x_2^i, \ldots x_d^i)$ is chosen randomly in the domain of integration.

Estimate of error in Monte Carlo integration

$$\sigma_I = \frac{1}{\sqrt{N}}\, \sigma_f$$

The error scales with N in the same way as in the one-dimensional case.

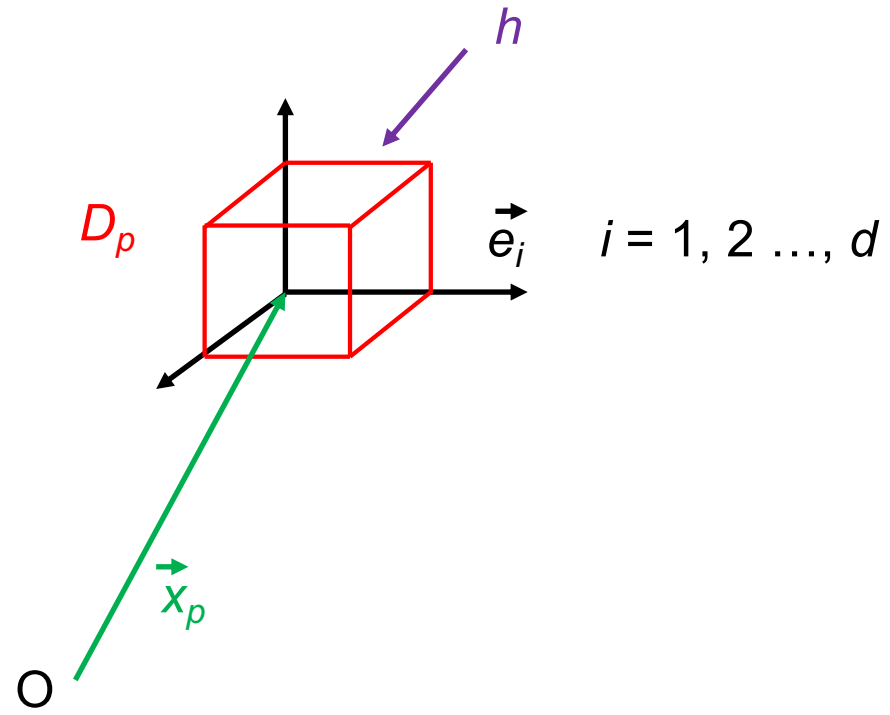
This result should be contrasted with the regular grid integration.

Scaling of error for grid integration in d dimensions

$$I = \int_0^1 dx_1 \ \ldots\ldots \int_0^1 dx_d \quad f(x_1, x_2, \ldots x_d) \quad = \int_D d^d\vec{x}\ f(\vec{x})$$

We assume that the points are regularly distributed in space.
How does the error scale with the number of points N used?

We separate the domain D in subdomains D_p, one for every grid point $\vec{x}_p$.



Number of subdomains: N

Linear dimension of subdomain: $h = \dfrac{1}{N^{1/d}}$

Volume of one subdomain: $h^d = 1 / N$

Check total volume of D : $N \cdot h^d = 1$

Scaling of error for grid integration in d dimensions

Integral over one subdomain D_p

$$\int_{D_p} d^d\vec{x}\; f(\vec{x}) = \int_0^h dx_1' \ldots\ldots \int_0^h dx_d' \;\; f(\vec{x}_p + \vec{x}')$$

$$= \int_0^h dx_1' \;\ldots\; \int_0^h dx_d' \left(f(\vec{x}_p) + \sum_{i=1}^{d} \left.\frac{\partial f}{\partial x_i}\right|_{\vec{x}_p} x_i' + \ldots \right)$$

$$= f(\vec{x}_p)\, h^d + \sum_{i=1}^{d} \left.\frac{\partial f}{\partial x_i}\right|_{\vec{x}_p} \frac{h^{d+1}}{2} + \ldots\; o(h^{d+2})$$

$$= f(\vec{x}_p)\, h^d + \frac{1}{2}\sum_{i=1}^{d} [\, f(\vec{x}_p + h\vec{e}_i) - f(\vec{x}_p)\,]\, h^d + \ldots\; o(h^{d+2})$$

error on integral over D_p

where we used the two-point formula for the gradient:

$$\left.\frac{\partial f}{\partial x_i}\right|_{\vec{x}_p} = \frac{f(\vec{x}_p + h\vec{e}_i) - f(\vec{x}_p)}{h} + o(h)$$

Scaling of error for grid integration in d dimensions

Errror of integral over one subdomain D_p

$$\propto \quad h^{d+2}$$

Errror of integral over full domain D

$$\sigma_I \quad \propto \quad N \cdot h^{d+2} \quad \propto \quad N \cdot \left(\frac{1}{N^{1/d}}\right)^{d+2} \quad \propto \quad N^{-2/d}$$

$$h = \frac{1}{N^{1/d}}$$

Scaling of error: Summary

Monte Carlo

$$\sigma_I \propto N^{-1/2}$$

The Monte Carlo scaling is independent of dimension.

Grid integration

$$\sigma_I \propto N^{-2/d}$$

The grid integration scaling worsens with dimension.

Assuming that the prefactors are all of order unity, Monte Carlo integration is more efficient for $d \geq 4$.

Prefactors can affect exact crossover, but different scaling behaviour remains.

Elements affecting prefactors are the quadrature formula in grid integration and the weight function in Monte Carlo integration.

Course 09/2

Monte Carlo integration through uncorrelated sampling

- Motivation
- Integration by Monte Carlo: Method
- Scaling of errors
  - Monte Carlo in 1D
  - Trapezoidal rule in 1D
  - Multidimensional integrals