

# **993SM - Laboratory of Computational Physics week 7 November 4, 2024**

#### **Maria Peressi**

Università degli Studi di Trieste - Dipartimento di Fisica Sede di Miramare (Strada Costiera 11, Trieste) e-mail: [peressi@units.it](mailto:peressi@units.it) tel.: +39 040 2240242

1

# Monte Carlo methods

- Monte Carlo integration
- Metropolis method to generate non-uniform random number distributions

## "acceptance-rejection" or "hit or miss" Monte Carlo methods: **(to calculate areas)**

enclose the pond in a box of Area  $A_{\text{box}}$ 



- throw pebbles uniformly and randomly in the box
- count the number of pebbles felt in the pond with respect to the number felt in the box
- Assuming a uniform distribution, the number of pebbles falling into the ponds is proportional to the area of the pond:

$$
\frac{N_{\text{pond}}}{N_{\text{pond}} + N_{\text{box}}} = \frac{A_{\text{pond}}}{A_{\text{box}}} \n\Rightarrow A_{\text{pond}} = \frac{N_{\text{pond}}}{N_{\text{pond}} + N_{\text{box}}} A_{\text{box}}
$$

Monte Carlo methods: "acceptance-rejection" or "hit or miss" *x* - rejection [0*,* 1] **(to calculate areas)** *P*(*x, y*)  $\pi = ? ? ?$ *N* ï *N*  $\frac{1}{2} \text{Cov}(\mathbf{x}^2, \mathbf{u})$  $x_i, y_i$   $\bar{}$  $a$ *i*,  $b$ *i*,  $a$ *i*,  $c$  $\overline{a}$ *N<sup>c</sup> x*<sup>2</sup> + *y*<sup>2</sup> ⇥ 1 Then, the number of

 $\frac{1}{2}$ **B**<br> $\frac{1}{2}$ points  $N_c$  lying within the quarter circle (i.e. fulfilling the relation  $x^2 + y^2 \le 1$ <br>is compared to the total number  $N_c$  of points and the fraction will give us an is compared to the total number  $N$ <br>approximate value of  $\pi$ : approximate value of  $\pi$ :

$$
\pi(N) = 4\frac{N_c(N)}{N}
$$

## Monte Carlo methods: "acceptance-rejection" or "hit or miss" **(to calculate definite integrals)**

$$
\int W(x)dx = ?
$$



For  $W(x)$  positive in the integration interval, the value of the area under W(x) can be obtained by producing random points (i.e. (x,y) random pairs) uniformly distributed in a rectangle containing W(x).

For each point  $(x,y)$  compare y with  $W(x)$ : if  $y < W(x)$ , the point is accepted. The area under  $W(x)$  is the number of points accepted divided by the total number of points generated and multiplied by the area of the rectangle.

(remember: also used to generate random numbers  $x_i$  distributed according  $W(x)$ )

### Click slide for next, or go to previous, first, last slides or back to thumbnail layout. Other simple Monte Carlo methods  $I =$  $\int^b$  $\overline{a}$ We can always write:  $I = \int \phantom{a} f(x) dx = (b - a)\langle f \rangle$



i.e., the value of the integral of  $f(x)$ between a and b equals the length of the interval (b-a) times the average value of the function <f> over the same interval. (If  $f:[a,b] \rightarrow R$  is a continuous function, then there exists a number c in [a,b] such that  $f(c) =$ **(mean value theorem for integration))**

## **how to estimate <f> efficiently and accurately?**

## A simple Monte Carlo method: "sample mean" ⇤ *N*

 $f(x)$ 

 $\langle f \rangle$  f(xi)

 $\overline{a}$ 

$$
I = \int_{a}^{b} f(x)dx = (b - a)\langle f \rangle
$$

The sample mean can be calculated by sampling the function *(if smooth enough...)* with a sequence of N uniform random numbers in [a,b]:  $\mathbf{N}^T$ 

$$
\langle f \rangle \approx \frac{1}{N} \sum_{i=1}^{N} f(x_i)
$$

$$
\int_{a}^{b} f(x)dx \approx (b-a)\frac{1}{N}\sum_{i=1}^{N} f(x_{i}) = (b-a)\langle f \rangle
$$

 $\blacktriangleright x$ 

 $\boldsymbol{b}$ 

 $x_i$ 

#### Monte Carlo methods: error estimate of n = 10<sup>4</sup> random numbers using the sample mean method is F<sup>n</sup> = 3.1489. How does this result for Farlo methods.  $\blacksquare$ Fonce early meander. of n = 10<sup>4</sup> random numbers using the sample mean method is F<sup>n</sup> = 3.1489. How does this result **for Find Carlo methods**  $\mathbf{F} = \mathbf{F} \mathbf$  $\mathbf{F}_{\mathbf{a}} = \mathbf{F}_{\mathbf{a}} \mathbf{F}_{\mathbf{a}} + \mathbf{F}_{\mathbf{a}} \mathbf{F}_{\mathbf{a}}$  associated with n  $= 104$  trials is not in  $= 104$ approximately 0.0073. HONG CAN WINCHIVES. definitive if the actual error in Figure is the actual error in Figure Figure Figure F

Example: MC estimate of  $\pi$  (exact value known)

Since we know the "exact" result *I*, we can calculate the **error** in two ways: We can use either acceptance-rejection or sample mean method:  $I = 4 \int_1^1$  $\overline{0}$  $\sqrt{1-x^2} = \pi = 3.1416...$ **trainpic.** The estimate of  $m$  (exact value rilevily  $r^1$   $\overline{a}$ definitive can use either acceptance-rejection or sample mean method:  $I = 4 \int_0^{\infty} \sqrt{1 - x^2} = \pi = 3$  $\frac{1}{2}$  trainting to accuracy of  $\frac{1}{2}$  course, we cannot all  $\frac{1}{2}$   $\frac{1}{2}$ definitive can use either acceptance-rejection or sample mean method:  $I = 4 \int_0^{\infty} \sqrt{1 - x^2} = \pi =$ with the can use of their accontance rejection on  $\sim$  $\int_0^{\infty}$   $\int_0^{\infty}$  then  $\int_0^{\infty}$  then then the standard would be zero, the extendion of  $\int_0^{\infty}$  for  $\int_0^{\infty}$   $\int_0^{\infty}$   $\int_0^{\infty}$   $\int_0^{\infty}$   $\int_0^{\infty}$   $\int_0^{\infty}$   $\int_0^{\infty}$   $\int_0^{\infty}$   $\int_0^{\infty}$  ance we know the leader result i, we can calculate the **cribition** in two ways.

1) the **actual error** from the difference with respect to the exact value: 2) the numerical error from the **variance of the data**  $\{f(x_i)\}$ :  $\Delta_n = |F_n - I|$ with  $\mathbf{r}$  $\Delta = |F - I|$  with  $F = (b - c)^{\frac{1}{n}} f(x)$  would  $\sum_{i=1}^{\infty} \frac{1}{n} \sum_{i=1}^{\infty} \frac{1}{n} \sum_{j=1}^{\infty} \frac{1}{j} \binom{u_j}{i}$ within a certain range centered on Fn. If the integrand were a constant, then the error would be zero, the extension of  $\frac{1}{n}$  for  $\alpha$ ,  $\alpha$  $a_n = |F_n - I|$  with  $F_n = (b-a)^{-1} \sum_{i=1}^{\infty} f(x_i)$ ,  $x_i$  rai  $\overline{a}$  $F_n = (b-a)^{\frac{1}{n}} \sum f(x_i), \quad x_i$  random 1  $\overline{n}$  $\sum$  $\frac{n}{n}$  $i=1$ with  $F_n = (b-a)^{\frac{1}{n}}$   $\sum f(x_i)$ ,  $x_i$  random

> $\sigma^2 = \langle f^2 \rangle - \langle f \rangle^2$ , and the contract of  $\mathcal{L}_1$  , and the contract of  $\mathcal{L}_1$ , and the contract of  $\mathbf{1}_{1}$

where

$$
\langle f \rangle = \frac{1}{n} \sum_{i=1}^{n} f(x_i)
$$
 and  $\langle f^2 \rangle = \frac{1}{n} \sum_{i=1}^{n} f(x_i)^2$ 

#### Monte Carlo methods: error estimate Monte Carlo methods: seeing how the actual error decreases as n increases. In Table 11.2 we see that as n goes from 10<sup>2</sup>  $2$  . However, we also see that  $\mathcal{L}$

Results:

**Results:** 
$$
I = 4 \int_0^1 \sqrt{1 - x^2} = \pi = 3.1416...
$$



interval [0, 1]. The actual error is given by the difference |F<sup>n</sup> − π|. The standard deviation σ<sup>n</sup> is 2) the numerical error from the variance of the data,  $\sigma_n$ , is roughly constant and is much larger than the actual error

## what is the correct error estimate?

# Monte Carlo methods: error estimate

...typically you do not know which is the "actual error" (you do not know the "true" value and you cannot compare your result with that!).... but we would like to give an error to our numerical estimate... (to which extent is our numerical estimate reliable?)

Two methods to estimate the error numerically from the variance of the data (**"reduction of variance"**):

1) average of the averages

II) block average

#### MC error handling: method 1 "average of the averages" MC arror handling n  $T_{\text{S}}$  Table 11.3: Example of the measurements of  $\epsilon$  $\lim_{\sigma} \cdot \text{match}$ 111 1 3.111 C U I U average of the averages  $\mathbf r$  hand ting mathod  $\mathbf{M}$  differences between the measurements is similar to the actual errors, and  $\mathbf{I}$ are a measurement of the error associated with a single measurement. The error and the entries of the average" the means  $\alpha$

 $\begin{array}{ccc} \text{N} & \text{N} & \text{N} \\ \text{N} & \text{N} & \text{N} \\ \text{N} & \text{N} & \text{N} \end{array}$ make additional runs of  $n$  trials each. Let  $M_{\alpha}$  be the average of each run:  $\sum_{n=1}^{\infty}$  make additional runs of *n* trials each.  $\mathcal{C}$  and  $\mathcal{C}$  and  $\mathcal{C}$  of n  $\mathcal{C}$  and  $\mathcal{C}$  of n  $\mathcal{C}$  and  $\mathcal{C}$ s of  $n$  that each. 1 :<br>D<br>D → Madei



Examples of Monte Carlo measurements of the mean value of  $f(x) = 4\sqrt{1 - x^2}$  in the interval [0, 1]. A total of 10 measurements of  $n = 10^4$  trials each were made. The mean value  $M_{\alpha}$ and the actual error  $|M_{\alpha} - \pi|$  for each measurement are shown. Examples of Monte Carlo measurements of the mean value of  $f(x) = 4\sqrt{1-x^2}$  in the  $\frac{1}{2}$  inter , (11.16), (11.16), (11.16), (11.16), (11.16), (11.16), (11.16), (11.16), (11.16), (11.16), (11.16), (11.16),  $^{\circ}$  and  $^{\circ}$  = shown. ne me  $=4$  $\frac{1}{11}\frac{M_{\alpha}}{M}$ 

Calculate: 
$$
\sigma_m^2 = \langle M^2 \rangle - \langle M \rangle^2
$$
 with  $\langle M \rangle = \frac{1}{m} \sum_{\alpha=1}^m M_{\alpha, \alpha}$ ,  $\langle M^2 \rangle = \frac{1}{m} \sum_{\alpha=1}^m M_{\alpha}^2$   
\n $\implies \sigma_m = 0.0068$ 

the differences between the measurements is similar to the actual errors, and hence the similar term of t  $m$  is consistent with the results for the actual err  $\overline{\phantom{a}}$  $\mathrm{s}\,\,\mathrm{for}\,\,\mathrm{th}$  $\sigma_m$  is consistent with the results for the actual errors

#### $M_{\odot}$  and mean the median trials to compute the mean only speaking, we have only speaking, we have  $\sim$ **1 independent trials remaining: method is remained to calculate or allows** manufacture  $\mathbf{r}$ of trials, we will use the relation (11.18b) and consider only this limit in  $\mathbf{C}$  $\left( \cdot\right)$  implies the most probable error  $\cdot\right)$ "block averages" MC error handling: method 11  $\mathbf{r}$  over the average value of  $\mathbf{r}$  over the 10 subsets is 3.14892, in agreement with the results is 3.14892, in agreement with the results is 3.14892, in agreement with the results is 3.14892, in agreement with t

Instead of doing additional measurements, divide them into "s SUBSETS" and let  $S<sub>x</sub>$  be the average within each subset: instead of doing additional measurements, divide<br>and let  $S_k$  be the average within each subset:  $\sim \kappa$  subset  $k$   $S_k$  $\mathcal{L}_k$  be the additional measurements.



The variance associated to the average of the subsets  $\sigma_s^2 = \langle S^2 \rangle - \langle S \rangle^2$  $\tau = 0.025$  but  $f(x) = f(x) - f(x)$  shown in Table 11.3.3.  $\sigma_s/\sqrt{s}$ , which for our example is approximately  $0.025/\sqrt(10) \approx 0.008$ . The variance associated to the average of the subsets  $\sigma_s^2 = \langle S^2 \rangle - \langle S \rangle^2$ gives  $\sigma_s = 0.025$ , but  $\frac{1}{\sqrt{2}}$  and  $\frac{1}{\sqrt{2}}$  and  $\frac{1}{\sqrt{2}}$  and  $\frac{1}{\sqrt{2}}$  interpret the n trials either the n t

is consistent with the actual error **approximately** is consistent with the actual error  $\frac{1}{\text{is consistent with the actual error}}$ as considered with the accuracy of s measurements with no second such as  $\frac{1}{\sqrt{2}}$ .

Monte Carlo methods: error estimate - variance reduction summary  $\sigma_n/$  $\sqrt{n} \approx \sigma_m \approx \sigma_s / \sqrt{s}$ from the variance of the whole set of data the variance of the average of (proof) **the averages** it does not change from the variance of the block averages the most convenient! but: change block size and check that Note: for uncorrelated data !

# Monte Carlo methods: summary

We have introduced :

\* "acceptance-rejection"

 $*$  "sample mean" to estimate  $\langle f \rangle \approx \frac{1}{\Delta t}$ N  $\sum$  $\overline{N}$  $i=1$  $f(x_i)$ 

both OK for smoothly varying functions, but not very efficient for rapidly varying functions

**How to improve the efficiency of MC integration?**

# A trick for numerical integration: "reduction of variance"

(Note: same word, but different meaning w.r.t. previous slides on error handling)

Given a function  $f(x)$ to integrate, suppose that  $g(x)$  exists, whose integral is known and such that:

$$
|f(x) - g(x)| < < \varepsilon
$$

Therefore:

$$
F = \int_a^b f(x)dx = \int_a^b ((f(x) - g(x)) + g(x))dx = \int (f(x) - g(x))dx + \int g(x)dx
$$

easy to calculate

#### Another simple Monte Carlo method: "importance sampling" *N* = 10*,* 000 *N<sup>c</sup>* = 7*,* 854 (10*,* 000) = 3*.*1416 ⇤  $\overline{4}$ *N*  $\Omega$ *N x<sup>i</sup> g*(*xi*) *N* = 10*,* 000 *N<sup>c</sup>* = 7*,* 854 (10*,* 000) = 3*.*1416 ⇤ 1 *g*(*x*) *N x<sup>i</sup> g*(*xi*)

Mean value: easy to calculate for smoothly varying functions. But not for functions rapidly varying. <u>UILLE</u> not for functions **OHS** *N* ⇥



### **How to manage such cases?**

# Another simple Monte Carlo method: "importance sampling"

 $\overline{\mathbf{N}}$ Mean value: easy to calculate for smoothly varying functions. Idea: in order to calculate:

$$
\langle f \rangle \approx \frac{1}{N} \sum_{i=1}^{N} f(x_i)
$$

consider a distribution function  $p(x)$  easy to integrate analytically and close to  $f(x)$ :

$$
F = \int_{a}^{b} f(x)dx = \int_{a}^{b} \left[ \frac{f(x)}{p(x)} \right] p(x)dx = \left\langle \frac{f(x)}{p(x)} \right\rangle \int_{a}^{b} p(x)dx
$$
  
\nwhere  $\left\langle \frac{f(x)}{p(x)} \right\rangle \approx \frac{1}{N} \sum_{i=1}^{N} \left[ \frac{f(x_i)}{p(x_i)} \right]$   
\nwith  $\left\{ x_i \right\}$  distributed according to  $p(x)$   
\nwith  $\left\{ x_i \right\}$  distributed according to  $p(x)$ 

#### Monte Carlo methods: "importance sampling" Because we cannot evaluate σ analytically in general, we determine σ a posteriori and choose a form of possible, mathods f(x) as much as possible, particularly when  $\mathbf{r}$  is large. **FIGILE CAIR MAKE THE INTEGRAND.** reduced. As an example, we again consider the integral (see Problem ee) <u>" 1</u> e−x<sup>2</sup> dex. (11.47) dex. ( Because Cause mathade  $\blacksquare$  force candidate minimized  $\blacksquare$ choice of provided make the integrand suppliers, and hence the integral of the variance will be variance with  $\sigma$ reduced. As an example, we are integral (see Problem et al.,  $\alpha$

Calculate:

Calculate: 
$$
F = \int_0^1 e^{-x^2} dx.
$$

with "sample mean" with random numbers with uniform<br> $\mathbf{u} = \mathbf{u} + \mathbf{u}$ the computation time per trial for the nonuniform case is larger, the smaller value of σ makes the distribution or using the "importance sampling" with  $p(x) = e^{-x}$ 



(pay attention to the normalization of  $p(x)...$ ) (pay attention to the normalization of  $p(x)...$ ) σ 0.2010 0.0550

## $error(MC)~1/\sqrt{N}$  => see log(error) vs. log(N) but with different prefactors for sample means vs importance sampling



#### choice of p(x) would make the integrand f(x)/p(x) slowly varying, and hence the variance will be Choice of the importance sampling function



(pay attention to the normalization of  $p(x)...$ )



on https://moodle2.units.it/

**pi.f90** Monte Carlo integration for the calculation of <sup>π</sup>

For the other exercises: write yourself the codes

## $error(MC)~1/\sqrt{N}$  => see log(error) vs. log(N)



#### $\arccos(MC) \sim 1/\sqrt{N} \cdot$ "true" arror an <code>error(MC)~1/ $\sqrt{\mathsf{N}}$  :"true" error and statistical error</code>



Sample mean

**Figure 5.**  $G.$  **Laurizi**, a.y.  $2019-20$ (credits: G. Lautizi, a.y. 2019-20)

## Summary of numerical integration (MC and deterministic) methods

#### **MC sample mean**

 $\int^b$ a  $f(x)dx = (b-a) < f > \approx (b-a)\frac{1}{\lambda^2}$ N  $\sum$ N  $i=1$  $f(x_i)$  with  $\{x_i\}$  randomly uniformly distributed in  $[a, b]$  $\int it$  can be considered as Importance sampling with  $p(x) = \frac{1}{x}$  $\frac{1}{b-a}$  in [a, b]  $\setminus$ 

#### **MC importance sampling**

$$
\int_{a}^{b} f(x)dx = \int_{a}^{b} \frac{f(x)}{p(x)} p(x)dx = \frac{f(x)}{p(x)} > \int_{a}^{b} p(x)dx \approx \frac{1}{N} \sum_{i=1}^{N} \frac{f(x_i)}{p(x_i)} \int_{a}^{b} p(x)dx
$$

with  $\{x_i\}$  randomly distributed according  $p(x)$ 

#### **Deterministic, equispaced points**

$$
\int_{a}^{b} f(x)dx \approx \sum_{i=1}^{N} v_{i}f(x_{i}) \quad \text{ with } x_{i} = a + \frac{b-a}{N}i \text{ , } v_{i} \text{ to be determined}
$$

#### **Deterministic, non equispaced points**

 $\int^b$ a  $f(x)dx \approx \sum$ N  $i=1$  $\overline{v_{i}}f(x_{i})\ \ \ \ with\ \{x_{i}\}\ ,\ \{v_{i}\} \ to\ be\ determined$ 

Error estimate: comparison between deterministic and MC methods in d-dimension

#### Error estimate for numerical integration with deterministic methods  $\Gamma$  is statistically independent (that is, if  $\Gamma$  i EIT OF ESUITIALE TOF HUITIEFICAL INC into 20 sets of 10 sets of 10 sets of 10 sets of 10 sets of 100 points each. If the sets of  $\mathbf{r}$ Error estimate for numerical integration

(Reminder from previous slides)

$$
\int_{\text{previous slides}} f(x)dx = F_n + error
$$

How to evaluate the error? Consider the Taylor expansion of the integrand function and then integrate: We derive the dependence of the truncation error estimates on the number of intervals for the

$$
f(x) = f(x_i) + f'(x_i)(x - x_i) + \frac{1}{2}f''(x_i)(x - x_i)^2 + \dots,
$$

$$
\int_{x_i}^{x_{i+1}} f(x) dx = f(x_i) \Delta x + \frac{1}{2} f'(x_i) (\Delta x)^2 + \frac{1}{6} f''(x_i) (\Delta x)^3 + \dots
$$
 (\*)

$$
\Delta x \equiv x_{i+1} - x_i
$$

### Error estimate for numerical integration: Rectangular approximation  $\mathbb{R}^n$  derive the dependence of the truncation error error estimates on the number of intervals for the n the assumed adequacy of the assumed adequacy of the Taylor series expansion of the integrand for the integral for

(Reminder from previous slides)

$$
\int_{x_i}^{x_{i+1}} f(x) dx \approx f(x_i) \Delta x
$$

Compare 
$$
\Box
$$
 with (\*):  
\n
$$
\int_{x_i}^{x_{i+1}} f(x) dx = f(x_i) \Delta x + \frac{1}{2} f'(x_i) (\Delta x)^2 + \frac{1}{6} f''(x_i) (\Delta x)^3 + \dots
$$
\nerror  
\n(leading order in  $\Delta x$ )

 $\lim_{u \to 0} (\Delta u - (v - u)/\hbar)$ . error is  $\hbar$ For  $n$  intervals  $(\Delta x = (b-a)/n)$ : error is  $n (\Delta x)^2 \sim 1/n$ 

 $\sim$ <sup>2</sup>. (11.61) (...and similarly for 27 higher-order approximations)

Numerical integration: multidimensional integrals To illumerical integration multi  $F =$ :<br>|-<br>| 1 R  $f(x, y) dx dy$ 

the number of parts (or pairs of points) of the integration domain: The rectangular approximation gives  $\Delta x \Delta y \sim (\Delta x)^2 \sim 1/n$ , being  $n$ 

$$
\int_{x_i}^{x_{i+1}} \int_{y_i}^{y_{i+1}} f(x, y) dx dy \approx f(x_i, y_i) \Delta x \Delta y \qquad (*)
$$

The Taylor expansion of the integrand function gives:

$$
f(x,y) = f(x_i, y_i) + f'_x(x_i, y_i)(x - x_i) + f'_y(x_i, y_i)(y - y_i) + \dots
$$

$$
\int_{x_i}^{x_{i+1}} \int_{y_i}^{y_{i+1}} f(x, y) dx dy = f(x_i, y_i) \Delta x \Delta y + \int_{x_i}^{t} (x_i, y_i) \frac{(\Delta x)^2}{2} \Delta y + f'_y(x_i, y_i) \Delta x \frac{(\Delta y)^2}{2} + \dots
$$
  
\n
$$
(*) \text{ against } (*) = > \text{error}
$$
  
\n
$$
n \text{ intervals: error is } n(\Delta x)^3 \sim 1/n^{1/2}
$$
  
\n
$$
(*) \text{ leading order in } \Delta x
$$

#### Numerical integration: multidimensional integrals Many problems in physics involve averaging over many variables. For example, suppose we know  $1.3 - 1.3$   $1.3 - 1.3$   $1.3 - 1.3$ Many problems in physics involve averaging over many variables. For example, suppose we know the position and velocity dependence of the total energy of the total energy of the total energy of the total dimensions each particle has three velocity components and three position components. Hence

Therefore for rectangular approx.:  $\sum_{i=1}^n$ 

involves computing a d  $\sim$  60 dimensional integral. (More accurately, the total energy is a function  $\sim$ d=1:  $error \sim 1/n$  d=2:  $error \sim 1/n^{1/2}$  $c_{\rm eff}$  into p intervals, there we point standard numerical methods to sum. Clearly, standard numerical methods to sum. Clearly, standard numerical methods of  $\alpha$  $\frac{1}{\sqrt{2}}$ such as Simpson's rule would be impractical for this example.

#### $\overline{\phantom{a}}$  discussion of the error associated with the standard numerical methods  $\overline{\phantom{a}}$ In general:

**if** the error decreases as  $n^{-a}$  for  $d = 1$ , then the error decreases as  $n^{-a/d}$  in d dimensions. d = 1, then the error decreases as n−a/d in d dimensions. In contrast, we find (see Section 11.4) that

#### **Classical formulas with equispaced points:** slowly decreasing error for multidimensional integration ! such as Simpson's rule are preferable to Monte Carlo methods unless the domain of integration is very complicated. However, the error in the error in the error in the conventional methods increases with  $\sim$

## Numerical integration: error in MC methods  $\sigma_n/$  $\sqrt{n} \approx \sigma_m \approx \sigma_s/\sqrt{s}$

( $\sigma_n$  is roughly constant with n; for uncorrelated points, the variance of the averages goes like  $\sim 1/n^{1/2}$ )

The average function value

$$
\langle f \rangle \equiv \frac{1}{N} \sum_{i=1}^{N} f(x_i)
$$

The average squared function value

$$
\langle f^2 \rangle \equiv \frac{1}{N} \sum_{i=1}^{N} f^2(x_i)
$$

Estimate of the integrand (+/- standard error)

$$
\int f \ dV \approx V \langle f \rangle \ \pm V \sqrt{\frac{\langle f^2 \rangle - \langle f \rangle^2}{N}}
$$

#### Numerical integration: errors in multidimensional integrals c. Determine the computational time per trial using the two Monte Carlo methods. Which Monte Carlo methods. Which Mo 11.3 ∗Numerical Integrals Integrals Integration of Multidimension<br>11.3 ∗Numerical Integrals Integrals Integrals Integrals Integrals Integrals Integrals Integrals Integrals Inte ltidimensional integrals **Carlo Municipal** 11.3 <sup>∗</sup>Numerical Integration of Multidimensional Integrals c. Determine the computational time per trial using the two Monte Carlo methods. Which Monte Carlo mari errors in multidimensional Integrals



if the error decreases as  $n^{-a}$  for  $d = 1$ , then the error decreases as  $n^{-a/d}$  in d dimensions. d = 1, then the error decreases as n−a/d in dimensions. In dimensions. In dimensions. In dimensions. In dimensio<br>In dimensions. In  $\frac{d}{dx}$  is the error decreases as n $\frac{d}{dx}$  in dimensions. In dimensions, we find  $\frac{d}{dx}$ 

d = 1, then the error decreases as n−a/d in d dimensions. In contrast, we find (see Section 11.4) that the error for all Monte Carlo integration methods decreases as  $n^{-1/2}$  independently of the integral. the error for all Monte Carlo integration methods decreases as n−1/2 independently of the dimensional dimension<br>The dimension methods decreases as n−1/2 independently of the dimensional dimensional dimensional dimensional

#### **Monte Carlo convenient for multidimensional integration!** dimension (see Appendix 11A), and Monte Carlo methods are essential for higher dimensional is very complicated. However, the error in the error in the error in the conventional methods increases with  $\alpha$ such as Simpson's rule are preferable to Monte Carlo methods unless the domain of integration is very complicated. However, the error in the conventional numerical methods increases with **Monte Carlo convenient for multidimensional integration !**

and  $M$  and  $M$  conclude that for low dimensions, the classical methods, the classical methods, the classical methods,  $\alpha$ 

## Summary: advantages of MC integration methods

- convergence as  $\sim N^{1/2}$  in any dimension regardless of the smoothness of the integrand
- simplicity: only two simple steps required (namely, producing a set of sampling points and evaluating the integrand function over such points)
- generality: sampling can be used even on domains that do not have a natural correspondence with the 'standard' domain  $[0,1]$ <sup>d</sup> and thus are not well-suited to numerical quadrature
- better suited than quadrature for integrands with singularities (importance sampling can handle this problem)
- flexibility: easy to add more points as needed (in the Gaussian quadrature, increasing the accuracy implies doing calculations from scratch)

## Metropolis Algorithm Metropolis Algo

by Metropolis, Rosenbluth, Rosenbluth, Teller and Teller (1953)

to generate random points with a given distribution rejected (see Appendix 11C). The Metropolis method is useful for computing averages of the form

method)  $\lim_{s \to 0}$  in  $\lim_{s \to 0}$  in Chapter  $\lim_{s \to 0}$  $mce$  sality.  $The "imp"$  $\epsilon$  related to use importance sampling to use  $\epsilon$ according to an arbitrary probability density p(x). The Metropolis algorithm produces a random walk of points  ${x}$  whose asymptotic probability distribution approaches p(x) after a large number  $a$ (motivation: related to the "importance sampling" integration method)

#### Metropolis Algorithm Another way of generating an arbitrary nonuniform probability distribution was introduced by Metropolis, Rosenbluth, Rosenbluth, Teller and Teller in 1953. The Metropolis algorithm is a  $\mathbf{M}$  of an  $\mathbf{m}$  in  $\mathbf{N}$  in which certain possible sampling at the sample sample sample. r rege appendix Appendium

how to generate random points with a given distribution? according to an arbitrary probability density p(x). The Metropolis algorithm produces a random  $p(x)$ rate ra<br> ndom points w n points with a given distribution: $p(x)$ how to generate random points with a given distribution?

**Idea: produce a random walk with points**  $\{x_i\}$  **whose asymptotic probability of**  $\{x_i\}$  **and**  $\{x_i\}$ **whose asymptotic probability distribution p<sub>N</sub>(x)**<br>of the essupied pesitions approaches  $p(x)$  ofter **of the occupied positions approaches**  $p(x)$  after  $\blacksquare$ **For simplicity a large number N of steps For simplicity and Metropolis algorithm in the context of estimating one-dimensional context of estimating one-dimensional context of estimating one-dimensional context of estima** proacnes  $p(x)$  arter It can be shown that it is sufficient (but not necessary) to satisfy the "detailed balance" condition ∑y dist<br>n<sup>i</sup>maac whose asymptotic probability distribution police (x) whose asymptotic probability distribution p<sub>N</sub>(x)<br>of the occupied positions approaches  $p(x)$  after where policy is an arbitrary probability distribution that need not be not be not be not be normalized. In Cha<br>In Chapter ?? we have a stribution that need not be no

#### Metropolis Algorithm Metropolis Algorithn where policy distribution that need not be not  $\mathbf{ifhm}$ !  $m<sub>1</sub>$ rejected (see Appendix 11C). The Metropolis method is useful for computing averages of the form  $\mathbf{p}(\mathbf{x}) = \begin{cases} \mathbf{p}(\mathbf{x}) & \text{if } \mathbf{x} \in \mathbb{R}^n, \mathbf{p}(\mathbf{x}) = \mathbf{p}(\mathbf{x}) & \text{if } \mathbf{x} \in \mathbb{R}^n, \mathbf{p}(\mathbf{x}) = \mathbf{p}(\mathbf{x}) & \text{if } \mathbf{x} \in \mathbb{R}^n, \mathbf{p}(\mathbf{x}) = \mathbf{p}(\mathbf{x}) & \text{if } \mathbf{x} \in \mathbb{R}^n, \mathbf{p}(\mathbf{x}) = \mathbf{p}(\mathbf{x}) & \text{if } \mathbf{x} \$ Another way of generating an arbitrary nonuniform probability distribution was introduced by Metropolis, Rosenbluth, Rosenbluth, Teller and Teller in 1953. The Metropolis algorithm is a  $\mathbf{M}$  of an  $\mathbf{m}$  in  $\mathbf{N}$  in which certain possible sampling at the sample sample sample sample. r rege appendix Appendium

with a given distribution? according to an arbitrary probability density p(x). The Metropolis algorithm produces a random how to generate random points with a given dist will discuss the application of the Metropolis algorithm to problems in statistical mechanics. The mechanics in statistical mechanics. The mechanics is not the mechanics of the mechanics. The mechanics is not the mechanic listribution?  $\overline{\phantom{a}}$ is a generate random points with a given distribution:<br> $n(r)$ how to generate random points with a given distribution? where  $p(x)$  is an arbitrary probability distribution that need not be no  $p(x)$ rate ra<br> ndom points w n points with a given distribution: $p(x)$ how to generate random points with a given distribution?

**Idea: produce a random walk with points**  $\{x_i\}$  **whose asymptotic probability of**  $\{x_i\}$  **and**  $\{x_i\}$ **whose asymptotic probability distribution p<sub>N</sub>(x)**<br>of the essupied pesitions approaches  $p(x)$  ofter of the occupied positions approaches  $p(x)$  after **a large number N of steps** proacnes  $p(x)$  arter It can be shown that it is sufficient (but not necessary) to satisfy the "detailed balance" condition whose asymptotic probability distribution  $p_N(x)$ **a large number N of steps and a large number N of steps and a large number of steps**  $\frac{d}{d\mathbf{S}}\left\{ \frac{d}{d\mathbf{r}}\right\}$ ion n. $\left(\mathbf{x}\right)$  $\mathbf{p}(x)$  after walk of points  ${x_i}$  whose asymptotic probability distribution approaches p(x) after a large number  $a$  $W^{(w)}$ Idea: produce a random walk **Millian discussi the application of the Metropolis algorithm to problems in statistical methods in statistical** For the occupied positions approaches  $p(x)$ a large number N of steps in the Metropolis algorithm in the Metropolis algorithm in the Metropolis algorithm i ∑y dist<br>n<sup>i</sup>maac whose asymptotic probability distribution p<sub>N</sub>(x)<br>of the occupied positions approaches  $p(x)$  after

ertain  $p(x)$ A random walk in general is defined by specifying a transition probability  $T(x_i \rightarrow x_j)$  from one value  $x_i$ to another value  $x_j$  and the distribution of points  $x_0, x_1, x_2, \ldots$  converges to a certain  $x_0, x_1, x_2, \ldots$  converges to a certain  $p(x)$ ecifying a ie value  $x_i$ oints sufficient (but not not not not needed balance "detailed balance") to satisfy the "detailed balance" con<br>Separate satisfy the "detailed balance" conditions in the "detailed balance" conditions in the "detailed balan  $x_0, x_1, x_2, \ldots$  converges to a certain  $p(x)$ A **reaccording to an arbitrary probability** probability produces and the Metropolis and produces a random produces where pointing is an arbitrary probability distribution that need not be not be not be not be not be not be no<br>In Chapter ?? We have the notified of the normalized of the need not be not be not be not be not be not be no A random walk in general is defined by specifying a **EVALUSILION PRODUCE THE METROPOLIS ALGORITHM IN THE METAL ONE-VALUE OF ALGORITHM**  $\mathcal{L}(w_i - w_j)$  in the value  $w_i$ definitive integrals. Suppose the use integration of portal variables  $\omega_0, \omega_1, \omega_2, \ldots$  converges is a cordin  $p(\omega)$ .  $\Delta$  reproduce wells in concretic defined by epecific  $\alpha$  $\Gamma$  cording to an arbitrary probability density probability density produces a random produces a random produces  $T(x) = x$  random produces  $x$ . to another value  $x_i$  and the distribution of points  $x_0, x_1, x_2, \ldots$  converges to a certain  $p(x)$  $\mathcal{L}(v, v_1, v_2, \ldots, v_n)$ 

### Comment:

need to consider a RW more general than the 'standard' RW with length and probability fixed for each step.

<u>Remind</u>: a RW with fixed length and  $p_{\text{left}} = p_{\text{right}}$  gives *P<sub>N</sub>(x)* that for large N tends to a gaussian distribution with a standard deviation that depends on N:

$$
\sigma^2 = Dt; \quad D = \frac{\ell^2}{2\Delta t}; \quad \Delta t = \frac{t}{N} \Longrightarrow \sigma^2 = \ell^2 N/2 \qquad P(x, N\Delta t) = \sqrt{\frac{2}{\pi N}} e^{-x^2/(2N)}
$$

(here  $l=1$ ; remind also the factor of 2 due to discretization)

The recipe to obtain a gaussian distribution with given σ from simple RWs was to generate *several* RWs with the same N and do the histogram of their end-points. The approach we are going to discuss now is something **different**, the focus being **one RW**.

## Markov chains p(x)f(x) dx

Consider a sequence of "configurations"  $C = \{C_1, C_2, ...$  $CN$  } stochastically generated, i.e.  $C_{k+1}$  is obtained from the previous one,  $C_k$ , by making some random changes on the former.  $F$  and  $F$  is the Metropolis of the Metropolis algorithm in the context of  $C$  on  $F$  and  $F$  one-dimensional one-dimensional one-dimensional one-dimensional one-dimensional or  $F$ 

The sequence is a **Markov chain** if the probability of making a transition from C<sub>k</sub> to C<sub>k+1</sub> is not dependent on how we arrived at  $C_k$  (its history), i.e. no memory.  $\frac{d}{dt}$  definition. Suppose that we wish that  $\frac{d}{dt}$  is a more random variability of  $\overline{\phantom{a}}$  according to an arbitrary probability density of  $\overline{\phantom{a}}$ . The Metropolis and the Metropolis

The sequence of points  $x_0, x_1, x_2, \ldots$  of a simple RW is a Markov chain. That is sufficient (but not necessary) to satisfy the "detailed balance" conditions of the of steps. The random walk is defined by specific a transition probability to the random walk is defined by specific<br>The random walk is defined by specific a transition probability T(xi → xj ) from one one one one one one

#### The detailed balance  $\mathbf{L}$  and  $\mathbf{L}$  and  $\mathbf{L}$  and  $\mathbf{L}$ . The Metropolis algorithm produces a random probability density  $\mathbf{L}$ walk of points and points and points a large number of points  $x_i$  and  $y_i$  an according to an arbitrary probability density p(x). The Metropolis algorithm produces a random according to an arbitrary probability density probability density probability density probability of the Metropolis and the Met walk of the points and parameters as we are the parameters of the set of the set of the set of the set of the s For simplicity, we introduce the Metropolis algorithm in the context of estimating one-dimensional **definition integrated integrated variables** For simplicity, we interest the Metropolis and the Metropolis simulation of the context of the Metropolis and the context of the context of the context of the con definite integrals. Suppose that we wish to use importance sampling to generate random variables For simplicity, we interest the detailed balance of  $\mathcal{L}_1$ The detailed helenes according to an arbitrary parameters and the Metropolis and The Metropolis and Metropolis and the Metropolis

Choose a transition probability  $T(x_i \rightarrow x_j)$  from one value  $x_i$  to another value  $x_j$  (from one configuration  $C_i$  to another one  $\tilde{C}_{i+1}$ ) such that the distribution of points  $x_0, x_1, x_2, \ldots$  (of configurations) converges to the desired  $p(x)$ . configuration  $C_i$  to another one  $\check{C}_{i+1}$ ) such that the  $\mathbf f$   $\mathbf v$  the condition. according to an arbitrary probability density p(x). The Metropolis algorithm produces a random configuration  $C_i$  to another one  $C_{i+1}$ ) such that the  $\alpha$ CHOOSE A CHAISICION PRODADING  $\mathcal{L}(\mathcal{U}_l, \rightarrow \mathcal{U}_j)$ . IFOIH converges to the desired  $p(x)$ . It is **sufficient (not necessary)** to satisfy the condition: Choose a transition probability  $T(x_i \rightarrow x_j)$  from

It is **sufficient** (not necessary) to satisfy the condition:  $\mathbf x$  ondition:

$$
p(x_i)T(x_i \to x_j) = p(x_j)T(x_j \to x_i)
$$

A simple choice (not unique!) is:  $T$  as a simple choice (not uniquely is:<br> $T=\frac{1}{\sqrt{2\pi}}\left[\frac{1}{\sqrt{2\pi}}\right]$ A simple choice (not unique!) is:

$$
T(x_i \to x_j) = \min\left[1, \frac{p(x_j)}{p(x_i)}\right]
$$

The state of t  $\overline{\phantom{0}}$ (We can easily v as <mark>lv verif</mark> p(xi)  $\left(\ldots\right)$  38 (We can easily verify...) 38 If the "walker" is at position x<sup>i</sup> and we wish to generate xi+1, we can implement this choice of

<sup>T</sup>(x<sup>i</sup> <sup>→</sup> <sup>x</sup><sup>j</sup> ) <sup>=</sup> min "

#### Example:



















#### The Metropolis algorithm p(xi)T(xi)T(xi)T(xi), (11.55), (11.55), (11.55), (11.55), (11.55), (11.55), (11.55), (11.55), (11.55), (11.55), The retropolis algorithm  $\cdots$  is consistent with  $\cdots$  $T_{h}$  Matropolic olgarithm i ne Metropolis algorithm Another way of generating an arbitrary nonuniform probability distribution was introduced by Metropolis, Rosenbluth, Rosenbluth, Teller and Teller in 1953. The Metropolis algorithm is a will discuss the application of the Metropolis algorithm to problems in statistical mechanics. The Metropolis a<br>The Metropolis algorithm to problems in statistical mechanics. The Metropolis algorithm to problems in statist Fire Pieu opolis algorithm in

#### $p(x)$  is given.  $p(x)$  is given.  $p(x)$  is given.

If the "walker" is at position  $x_i$  and we wish to generate  $x_{i+1}$ , we can implement this choice of  $T(x_i \rightarrow x_j)$  by the following steps: If the "walker" is at position x<sup>i</sup> and we wish to generate xi+1, we can implement this choice of The wanter is at position  $x_i$  and  $T(x_i \rightarrow x_i)$  by the following steps If the "walker" is at position  $x_i$  and we wish to generate  $x_{i+1}$ , we can implement this choice of  $T(x_{i+1}, x_i)$  by the following steps: d we wish to generate  $x_{i+1}$ , we can implement this choice of If the "walker" is at position  $x_i$  and we wish to generate  $x_{i+1}$ , we can implement this cho  $T(x_i \rightarrow x_j)$  by the following steps:

- 1. Choose a trial position  $x_{\text{trial}} = x_i + \delta_i$ , where  $\delta_i$  is a random number in the interval  $[-\delta, \delta]$ .  $\ddot{\phantom{a}}$ of steps. The random walk is defined by specific in the interval in the interval of  $\delta_i$  is a random number in the interval interval in the interval in the interval in the interval interval in the interval interval in th
- 2. Calculate  $w = p(x_{\text{trial}})/p(x_i)$ .  $= p(x<sub>trial</sub>)$  $\frac{1}{2}$ value  $\frac{1}{2}$  Calculate  $w = n(x_{\text{train}})/n(x_i)$
- 3. If  $w \ge 1$ , accept the change and let  $x_{i+1} = x_{\text{trial}}$ .  $p(\mathbf{x})$  distribution of  $\mathbf{x}$  ,  $\mathbf{y}$  distribution of  $\mathbf{x}$  ,  $\mathbf{y}$  distribution of  $\mathbf{x}$ where  $p(x)$  is an arbitrary probability distribution that need not be not It can be shown that  $P(\alpha_t | \text{data}) / P(\alpha_t)$ . else
	- 4. If  $w < 1$ , generate a random number  $r$ . 4. If  $w < 1$ , generate a random number r.<br>
	5. If  $r < w$  accept the change and let  $x_{i+1} = x_{i+1}$ .
	- 5. If  $r \leq w$ , accept the change and let  $x_{i+1} = x_{\text{trial}}$ . will discuss the application of the Metropolis algorithm to problems in statistical mechanics. The  $\mathcal{N}$
- 6. If the trial change is not accepted, then let  $x_{i+1} = x_i$ .  $\frac{1}{2}$  in the trial enange is not accepted, then ict  $x_{i+1} - x_i$ . For simplicity, we introduce the Metropolis algorithm in the  $\frac{1}{2}$  one-dimensional context of estimating one-dimensional context of estimating one-dimensional context of estimating one-dimensional context of estimatin definite integrals. Suppose that we wish to use integrals. Supportance sampling to generate random variables t<br>The contract of the contract o

#### It is necessary to sample many points of the random walk before the random walk before the asymptotic probability  $\mathbf{r}$ The algorithm from T) to 6) has to be repeated until  $\mathbf{a}$  small percentage of the sample  $\mathbf{a}$  is weaked the distribution  $p(x)$  of the points  $\{x_i\}$  is reached. The algorithm from  $\mathbb{I}$  to 6) has to be repeated until inc algorithm nonly to by has to be repeated then  $\tau$  is too small, a large percentage of trial steps will be accepted, but again the accepted, but ag The algorithm from 1) to 6) has to be repeated until  $T = \frac{1}{2}$  $\mathbf u$ uuon  $\mu$

## note:

it's important how to handle the rejected attempts for the generation of the random walk:

in case of a rejected attempt, the walker does not move, and we have to consider again the point where we tried to move from;

in the integration with importance sampling, a point which is unchanged after a rejected attempt does enter again in the average, i.e. its weight in the sum increases



- how to choose  $x_0$ ?  $\bullet\;$  how to choose  $\;x_0$
- how to choose  $\delta$  ? (if too small, most trial steps accepted, but the walker moves too slowly; if too large, only a few trial steps are accepted...)  $\bullet$  how to choose  $\delta$  ?<br>(if too small most trial s  $s_{\text{max}}$ • how to choo

• equilibration is necessary (how many steps?)

## Answers:  $\Delta$  he wish to generate  $\Delta$  he wish to generate  $\Delta$

- how to choose  $x_0$  ? Convenient to start from a maximum  $\bullet\;$  how to choose  $\;x_0$ Such that the start convenient to start  $\sim$
- how to choose  $\delta$  ? (if too small, most trial steps accepted, but the walker moves too slowly; if too large, only a few trial steps are accepted...) A good compromise is a choice accepting from  $\sim$  1/3 to  $\sim$  1/2 of the trial steps  $\bullet$  how to choose  $\delta$  ?<br>(if too small most trial s • how to choo
	- equilibration is necessary (how many steps?) A possible criterion based on error estimate

## **Some programs:**

in moodle2.units.it:

## **gauss\_metropolis.f90**

```
! gauss metropolis.f90
! METROPOLIS generation of random numbers with a Gaussian distribution
P(x) = exp(-x**2/(2*signa**2))/sqrt(2*pirsigna**2).... start from a given x=x0 .....
do i=1,n
```


endif

enddo

```
! gauss metropolis.f90
! METROPOLIS generation of random numbers with a Gaussian distribution
P(x) = exp(-x**2/(2*signa**2))/sqrt(2*pirsigna**2).... start from a given x=x0 .....
do i=1,nx1 = x1 + xx2 = x2 + x**2x3 = x3 + x**3x4 = x4 + x**4! cccccccccccccccccccccccccccccccc
    expx = - x^{**2} / (2*signa^{**2})call random number(rnd)
    xp = x + delta * (rnd-0.5)expxp = - xp**2 /(2*signa**2) ! metropolis
     w = exp (expxp-expx) ! algorithm
    call random number(rnd)
    if (w > rnd) then
       x = xp! ccccccccccccccccccccccccccccccccc
        acc=acc+1. 
     endif
    ibin = nint(x/\text{delta})if (abs(ibin) < maxbin/2) istog(ibin) = istog(ibin) + 1
  enddo
                                   calculate some momenta
                                      calculate the histogram
                                calculate the acceptance ratio
```
Metropolis generation of random numbers distribution

1) let's use the Metropolis method to generate a gaussian distribution  $0.45$ quuss metrooclis.diit'

# example of application:

 $(n=1000, x0=0, \delta=5, \sigma=1)$ 



(with **gauss\_metropolis.f90**)

**If the "Walker" is an intervalled "walker" is a summer is a significant to the "Analytical"** Answers from numerical experiments:

- how to choose  $x_0$ ? Convenient to start from a maximum  $\bullet\;$  how to choose  $\;\;x_0$ Such that the start is start that the asymptotic distribution as possible.
- $\bullet$  how to choose  $\overrightarrow{O}$ ? (if too small, most trial steps accepted, but the walker moves too slowly; if too large, only a few trial steps are accepted...) A good compromise is a choice accepting from  $\sim$  $1/3$  to  $\sim$  1/2 of the trial steps: depends on  $\sigma$ • how to choose  $\delta$ ?<br>(if too small most trial st • how to choo
	- equilibration is necessary (how many steps?) A possible criterion based on error estimate: Consider when  $\langle x^2 \rangle \approx \sigma^2$