Computational Astrophysics I: Introduction and basic concepts

Helge Todt

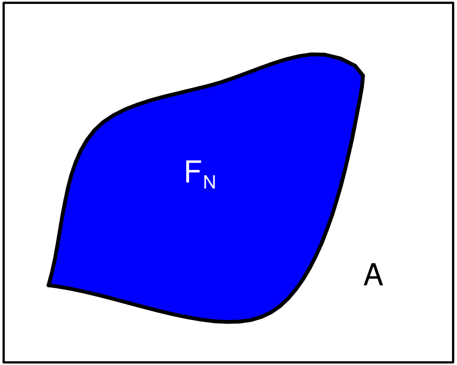
Astrophysics
Institute of Physics and Astronomy
University of Potsdam

SoSe 2024, 26.6.2024

Monte-Carlo integration

MC integration I

Idea: Can the area of a pool (irregular!) be measured by throwing stones?



- pool with area F_n in a field with *known(!)* area A

MC integration II

- fraction of the *randomly* thrown stones which fall into the pool:

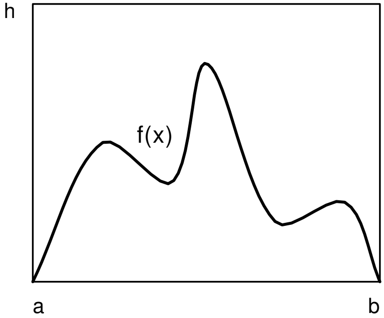
$$\frac{n_\mathrm{p}}{n} = \frac{F_n}{A} \tag{1}$$

(n stones, n_p hit pool)

- determine F_n with help of the *hit-or-miss method*:

$$F_n = A\,\frac{n_\mathrm{p}}{n} \tag{2}$$

MC integration III



- choose rectangle of height h, width $(b-a)$, area $A = h \cdot (b-a)$, such that $f(x)$ within the rectangle
- generate n pairs of random variables x_i, y_i with $a \leq x_i \leq b$ and $0 \leq y_i \leq h$
- fraction n_t of the points, which fulfill $y_i \leq f(x_i)$ gives estimate for area under $f(x)$ (integral)

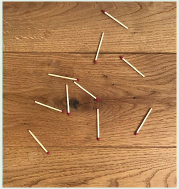
Excursus: Buffon's needle problem – determine π by throwing matches

Buffon's question (1773): What is the probability that a needle or a match of length ℓ will lie across a line between two strips on a floor made of parallel strips, each of same width t?
$\rightarrow x$ is distance from center of the needle to closest line, θ angle between needle and lines ($\theta < \frac{\pi}{2}$), hence the *uniform* probability density functions are

$$p(x) = \begin{cases} \frac{2}{t} & : \quad 0 \leq x \leq \frac{t}{2} \\ 0 & : \quad \text{elsewhere} \end{cases} \qquad p(\theta) = \begin{cases} \frac{2}{\pi} & : \quad 0 \leq \theta \leq \frac{\pi}{2} \\ 0 & : \quad \text{elsewhere} \end{cases}$$

x, θ independent $\rightarrow p(x,\theta) = \frac{4}{t\pi}$ with condition $x \leq \frac{\ell}{2}\sin\theta$. If $\ell \leq t$ (short needle):

$$P(\text{hit}) = \int_{\theta=0}^{\frac{\pi}{2}} \int_{x=0}^{\frac{\ell}{2}\sin\theta} \frac{4}{t\pi} dx d\theta = \frac{2\ell}{t\pi}$$



$\rightarrow$ count hits and misses and then:

$$\pi = \frac{2\ell}{t}\frac{1}{P(\text{hit})} = \frac{2\ell}{t}\frac{n_{\text{hit}} + n_{\text{miss}}}{n_{\text{hit}}}$$

MC integration V

Sample-mean method

- the integral

$$F(x) = \int_a^b f(x)\,dx \tag{3}$$

is given in the interval $[a, b]$ by the mean $\langle f(x)\rangle$ (mean value theorem for integration)

- choose arbitrary x_i (instead of regular intervals) and calculate

$$F_n = (b-a)\langle f(x)\rangle = (b-a)\,\frac{1}{n}\sum_{i=1}^{n} f(x_i) \tag{4}$$

where x_i are uniform random numbers in $[a, b]$

$$\left(\text{cf. rectangle rule}\quad F_n = \sum_{i=1}^{n} f(x_i)\Delta x \quad \text{with fixed } x_i, \Delta x = \frac{b-a}{n}\right) \tag{5}$$

Importance sampling I

Idea: improve MC integration by a better sampling → introduce a positive function $p(x)$ with

$$\int_a^b p(x)dx = 1 \tag{6}$$

and rewrite integral $\int_a^b f(x)dx$ as

$$F = \int_a^b \left[\frac{f(x)}{p(x)}\right] p(x)dx \tag{7}$$

this integral can be evaluated by *sampling according to* $p(x)$:

$$F_n = \frac{1}{n}\sum_{i=1}^{n}\frac{f(x)}{p(x)} \tag{8}$$

Note that for the *uniform case* $p(x) = 1/(b-a)$ → the *sample mean method* is recovered.
Now, try to minimize variance σ^2 of integrand $\frac{f(x)}{p(x)}$ by choosing $p(x) \approx f(x)$, especially for large $f(x)$

$\rightarrow$ slowly varying integrand $f(x)/p(x)$
$\rightarrow$ smaller variance σ^2

Example: Normal distribution

Evaluate integral $F = \int_a^b f(x)dx = \int_0^1 e^{-x^2}dx$ (error function) $\rightarrow F_n = \frac{1}{n}\sum_{i=1}^{n} \frac{e^{-x^2}}{p(x)}$

	$p(x) = 1$	$p(x) = Ae^{-x}$ †
x	$(b-a)*r+a$	$-\log(e^{-a} - \frac{r}{A})$
n	4×10^5	8×10^3
σ	0.0404	0.0031
$\sigma/\sqrt{n}$	6×10^{-5}	3×10^{-5}
total CPU time††	19 ms	0.8 ms
CPU time / trial	50 ns	100 ns

† A from normalization $A = (\exp(-a) - \exp(-b))^{-1}$, ††CPU time on a Intel Core i7-4771 3.5 GHz

$\rightarrow$ the extra time needed per trial for getting x from uniform r is usually overcompensated by the smaller number of necessary trials for same $\sigma/\sqrt{n}$

Similar: Metropolis algorithm (Metropolis, Rosenbluth, Rosenbluth, Teller & Teller 1953) useful for averages of the form

$$\langle f \rangle = \frac{\int p(x) f(x) dx}{\int p(x) dx} \quad \text{e.g.} \quad \langle f \rangle = \frac{\int e^{-\frac{E(x)}{k_\mathrm{B} T}} f(x) dx}{\int e^{-\frac{E(x)}{k_\mathrm{B} T}} dx}, \tag{9}$$

The Metropolis algorithm uses *random walk* (see below) of points $\{x_i\}$ (1D or higher) with asymptotic probability distribution approaching $p(x)$ for $n \gg 1$. Random walk from *transition probability* $T(x_i \to x_j)$, such that

$$p(x_i) T(x_i \to x_j) = p(x_j) T(x_j \to x_i) \qquad \text{(detailed balance)} \tag{10}$$

$$\text{e.g., choose}\, T(x_i \to x_j) = \min\left[1, \frac{p(x_j)}{p(x_i)}\right] \quad \left(\text{where, e.g., } p_j/p_i = \exp\left(-\frac{E_j - E_i}{k_\mathrm{B} T}\right)\right) \tag{11}$$

Metropolis algorithm

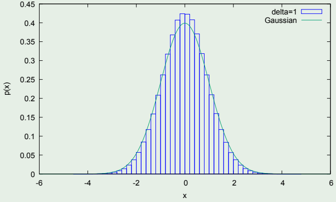
1. choose trial position $x_{\text{trial}} = x_i + \delta_i$ with random $\delta_i \in [-\delta, +\delta]$
2. calculate $w = p(x_{\text{trial}})/p(x_i)$ (might be: $w = \exp\left(-\frac{E(x_{\text{trial}})-E(x_i)}{k_B T}\right)$)
3. if $w \geq 1$, accept and $x_{i+1} = x_{\text{trial}}$ ($\rightarrow \Delta E \leq 0$)
4. if $w < 1$ ($\rightarrow \Delta E > 0$), generate random $r \in [0; 1]$
5. if $r \leq w$, accept and $x_{i+1} = x_{\text{trial}}$ (and compute desired quantities, e.g. $f(x_{i+1})$)
6. if not, $x_{i+1} = x_i$

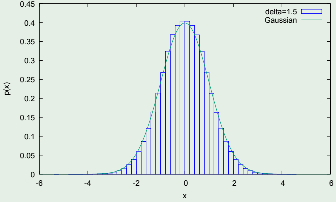
(finally: $\langle f \rangle = \frac{1}{n}\sum_{i=1}^{n} f(x_i)$)
problem: optimum choice of δ;

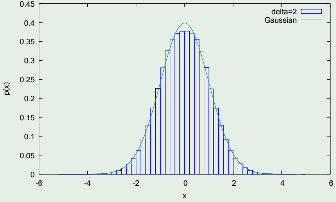
if too large, only small number of accepted trials $\rightarrow$ inefficient sampling
if too small, only slow sampling of $p(x)$.
Hence, rule of thumb: choose δ for which $\frac{1}{3} \ldots \frac{1}{2}$ trials accepted

also: choose x_0 for which $p(x_0)$ is largest $\rightarrow$ faster approach of $\{x_i\}$ to $p(x)$

Metropolis algorithm for Gaussian standard distribution







$\delta = 1.$
$f_{\text{accept}} = 0.72$
$\langle x \rangle = 0.0007533$
$\langle x^2 \rangle = 0.90306$

$\delta = 1.5$
$f_{\text{accept}} = 0.64$
$\langle x \rangle = -0.0000153$
$\langle x^2 \rangle = 0.935376$

$\delta = 2.$
$f_{\text{accept}} = 0.56$
$\langle x \rangle = 0.000200396$
$\langle x^2 \rangle = 0.988051$

$\langle x \rangle$ and $\langle x^2 \rangle$ computed from

```
xmean = xmean + xtrial ; xxmean = xxmean + xtrial * xtrial ;
...
xmean = xmean / naccept ; xxmean = xxmean / naccept ;
```

Metropolis algorithm IV

Typical applications for Metropolis algorithm: computation of integrals with weight functions $p(x) \sim e^{-x}$, e.g.,

$$\langle x \rangle = \frac{\int_0^\infty x e^{-x} dx}{\int_0^\infty e^{-x} dx} \tag{12}$$

$$\langle A \rangle = \frac{\int A(\vec{X}) e^{-U(\vec{X})/k_\mathrm{B} T} d\vec{X}}{\int e^{-U(\vec{X})/k_\mathrm{B} T} d\vec{X}} \tag{13}$$

where the latter is the average of a physical quantity A in a liquid system with good contact to a thermal bath, fixed number of particles (with $\vec{X} = (\vec{x}_1, \vec{x}_2, \ldots)$ of all particles) and volume $\rightarrow$ canonical ensemble, e.g.,

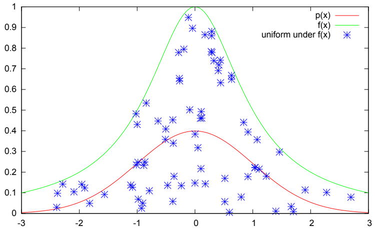
$$\left\langle \frac{m v_{ik}^2}{2} \right\rangle = \frac{1}{2} k_\mathrm{B} T \tag{14}$$

Rejection sampling (acceptance-rejection method)

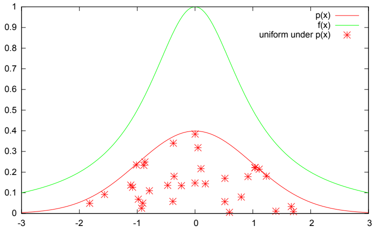
Problem: get random x for any $p(x)$, also if $P(r)^{-1}$ not (easily) computable

Idea:

- area under $p(x)$ in $[x, x+dx]$ is probability of getting x in that range
- if we can choose a random point in *two dimensions* with uniform probability in the area under $p(x)$, then x component of that *point* is distributed according to $p(x)$
- so, on same graph draw an $f(x)$ with $f(x) > p(x) \quad \forall x$
- if we can uniformly distribute points in the area under curve $f(x)$, then all points (x, y) with $y < p(x)$ are uniform under $p(x)$



$\Longrightarrow$

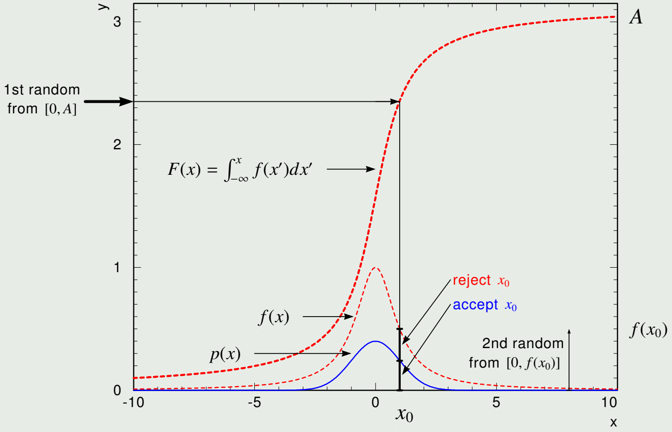


Rejection sampling (acceptance-rejection method) II

Creation of arbitrary probability distributions with help of rejection sampling (especially for compact intervals $[a, b]$):

- let $p(x)$ be the required distribution in $[a, b]$
- choose a $f(x)$ such that $p(x) < f(x)$ in $[a, b]$, e.g., $f(x) = c \cdot \max(p(x)) =$ const. where $c > 1$
- it is $A := \int_a^b f(x)dx$, i.e. A(x) must exist and must be invertible: $A(x) \to x(A)$
- generate *uniform* random number in $[0, A]$ and get the corresponding $x(A)$
- generate 2nd *uniform* random number y in $[0, f(x)]$, so x, y are uniformly distributed on A (area under $f(x)$)
- *accept* this point if $y < p(x)$, otherwise *reject* it

Example: normal distribution $p(x)$ sampled by $f(x) = (x^2 + 1)^{-1}$



$\frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}$ (blue solid line) sampled with help of the function $\frac{1}{x^2+1}$ (red dashed) whose integral is $\arctan(x)$ (thick dashed red) and hence $F(x)^{-1} = \tan(x)$, see source code on page 19

Rejection sampling (acceptance-rejection method) IV

Requirements:

- $p(x)$ must be computable for every x in the intervall
- $f(x) > p(x) \rightarrow$ always possible, as $\int_{-\infty}^{+\infty} p(x)dx = 1$ (i.e. $A > 1$)
- to get x_0 for a chosen value in $[0, A]$ requires usually: $\int f(x)dx = F$ is analytically invertible, i.e. $F(x)^{-1}$ exists

$\rightarrow$ this is easy for a compact interval $[a, b]$, e.g., choose a $c > 1$ such
$F(x) = c \cdot \max(p(x)) \cdot (x - a) = k(x - a)$
$\rightarrow x = F/k - a$ for randomly chosen F in $[0, A]$, where $A = k \cdot (b - a)$

Rejection sampling (acceptance-rejection method) V

Example: acceptance-rejection for normal distribution (see p. 17)

```
double p(double x){ return exp(-0.5*x*x)/sqrt(2.*M_PI); }
double f(double x){ return 1./(x*x+1.); }
double  inv_int_f(double ax){ return tan(ax - M_PI /2.); }
 ...
for (int i = 0; i < nmax; ++i){
  // get random value between 0 and A:
  ax = A * double(rand())/double(RAND_MAX);
  // obtain the corresponding x value:
  x = inv_int_f(ax);
  // get random y value in interval [0,f(x)]:
  y = f(x) * double(rand())/double(RAND_MAX);
  // test for y =< p(x) for acceptance:
  if ( y <= p(x) ) { cout << x << endl ;}
}
```

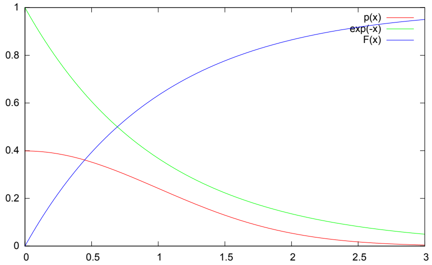
Rejection sampling (acceptance-rejection method) VI

In our example:

- it is $p(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}$ the standard normal distribution; normal distributions with $\sigma \neq 1, \mu \neq 0$ can be obtained by transformation
- the comparison function $f(x) = \frac{1}{x^2+1}$ is always $f(x) > p(x)$, moreover:
  - $F(x) = \int_{-\infty}^{x} f(x')dx' = \arctan(x) - \arctan(-\infty) = \arctan(x) - \left(-\frac{\pi}{2}\right)$
    $\rightarrow F(x) = \arctan(x) + \frac{\pi}{2}$
  - the total area A under $f(x)$ is $\int_{-\infty}^{+\infty} f(x')dx' = \arctan(+\infty) - \arctan(-\infty) = \pi$
  - the inverse $F(x)^{-1}$, which returns x for a given value $F \in [0, A]$ simply $x = \tan\left(F - \frac{\pi}{2}\right)$
  - efficiency of the acceptance is $N_{\text{accepted}}/\mathtt{NMAX} = \int p(x)/\int f(x) = 1/\pi \approx 0.32$, i.e. efficiency can be increased by choosing $f(x) = \frac{1}{2}\frac{1}{x^2+1}$, then $x = \tan\left(2F - \frac{\pi}{2}\right) \rightarrow 63\%$ acceptance

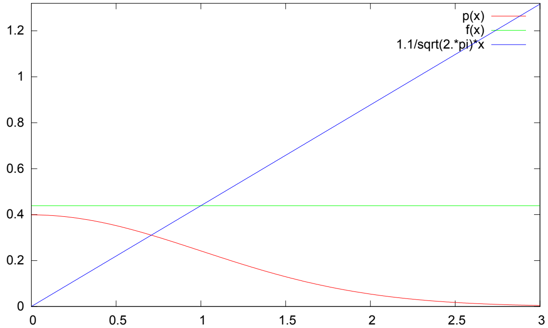
Alternative choice I: $f(x) = \exp(-x)$ only for $x \geq 0$, then

- the integral $F(x)$ is $\int_0^x = -\exp(-x) + 1$
- the total area $\int_0^\infty \exp(-x)dx = 1 > 0.5 = \int_0^\infty p(x)$
- the inverse is $x = -\log(-x + 1)$
- to obtain also negative $x \rightarrow$ add random sign $\pm$



Rejection sampling (acceptance-rejection method) VIII

Alternative choice II: $f(x) = 1.1 \cdot \max(p(x))$ in the compact interval $[0, 3]$, then



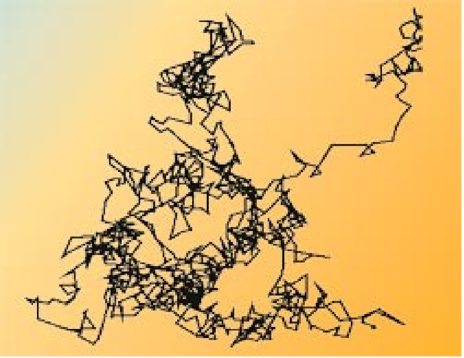
- it is $\max(p(x)) = \frac{1}{\sqrt{2\pi}}$ in $[0, 3]$
  $\rightarrow f(x) = \frac{1.1}{\sqrt{2\pi}}$ in $[0, 3]$
- hence $F(x)^{-1}$ is $x = \frac{F\sqrt{2\pi}}{1.1} - 0.$
- the total area A is $\frac{1.1}{\sqrt{2\pi}} \cdot (3 - 0)$

$\rightarrow$ clear: this choice (const. function) works only for compact intervals, otherwise A is infinite and $F(x)^{-1}$ does not exist

Random walk

Random walk I

Idea: Brownian motion, e.g., dust in water (lab course: determination of diffusion coefficient $D = \frac{\langle x^2 \rangle}{2t}$, with Fick's laws of diffusion: $j = -D\partial_x c$ and $\dot{c} = D\partial_x^2 c$)



frequent collisions between dust particles and water molecules
$\rightarrow$ frequent change of direction
$\rightarrow$ trajectory not predictable even for few collisions
$\rightarrow$ motion of dust particle into any direction with same probability

$\rightarrow$ Random walk

like "drunken sailor": N steps of equal length in arbitrary direction will lead to which distance from start point?

Random walk II

In one dimension:

- let's start at $x = 0$, each step with length ℓ
- for each step: probability p for step to the right and $q = 1 - p$ to the left (independent from previous step)
- displacement after N steps

$$x(N) = \sum_{i=1}^{N} s_i \quad \text{where } s_i = \pm\ell \quad \rightarrow x^2(N) = \left(\sum_{i=1}^{N} s_i\right)^2 \tag{15}$$

- for $p = q = 1/2 \rightarrow$ coin flipping
- for large N: $\langle x(N) \rangle = 0$ expected
- but for $\langle x^2(N) \rangle$? $\rightarrow$ rewrite Eq. (15)

$$x^2(N) = \sum_{i=1}^{N} s_i^2 + \sum_{i \neq j=1}^{N} s_i s_j \tag{16}$$

where (for $i \neq j$) $s_i s_j = \pm\ell^2$ with same probability, so: $\sum_{i \neq j}^{N} s_i s_j = 0$

Random walk III

- because of $s_i^2 = \ell^2 \rightarrow \sum_{i=1}^{N} s_i^2 = N\ell^2$:

$$\langle x^2(N) \rangle = \ell^2 N \tag{17}$$

- especially for constant time intervals of the random walk

$$\langle x^2(t) \rangle = \frac{\ell^2}{\Delta t} N \Delta t \quad \left(= \frac{\ell^2}{\Delta t} t \right) \tag{18}$$

- generally: if $p \neq 1/2$ and p for $+\ell$

$$\langle x(N) \rangle = (p - q)\ell N \tag{19}$$

$\rightarrow$ linear dependence on N

Random walk IV

Example: Diffusion of photons in the Sun

Simplification: constant density n, only elastic Thomson scattering (free e^-) with (frequency independent) cross section $\sigma_{\mathrm{Th}} = 6.652 \times 10^{-25}\,\mathrm{cm}^2$
mean free path length:

$$\ell = \frac{1}{n\sigma_{\mathrm{Th}}} = \left(\frac{\varrho}{m_{\mathrm{H}}}\sigma_{\mathrm{Th}}\right)^{-1} \tag{20}$$

one dimension $\rightarrow$ only $R = R_\odot$, total time $t = N\Delta t$

$$\Rightarrow t = 9 \times 10^{10}\,\mathrm{s} = 2900\,a \ll t_{\mathrm{KH}} (= 3 \times 10^7\,\mathrm{a})$$

Importance of the random walk model

many processes can be described by differential equation similar to diffusion equation (e.g., heat equation, Schrödinger equation with imaginary time)

$$\frac{\partial p(x,t)}{\partial t} = D\frac{\partial^2 p(x,t)}{\partial x^2} \tag{21}$$

with diffusion coefficient D and probability $p(x,t)dx$ to find particle at time t in $[x, dx]$
in 3 dimensions: $\partial^2/\partial x^2 \equiv \nabla^2$
Moments: mean value of a function $f(x)$

$$\langle f(x,t)\rangle = \int_{-\infty}^{+\infty} f(x,t)\, p(x,t)dx \tag{22}$$

$$\Rightarrow \quad \langle x(t)\rangle = \int_{-\infty}^{+\infty} x\, p(x,t)dx \tag{23}$$

Random walk VI

Compute integral in Eq. (23) $\rightarrow$ multiply Eq. (21) by x and integrate over x

$$\int_{-\infty}^{+\infty} x \frac{\partial p(x,t)}{\partial t} dx = D \int_{-\infty}^{+\infty} x \frac{\partial^2 p(x,t)}{\partial x^2} dx \tag{24}$$

left hand side

$$\int_{-\infty}^{+\infty} x \frac{\partial p(x,t)}{\partial t} dx = \frac{\partial}{\partial t} \int_{-\infty}^{+\infty} x\, p(x,t) dx = \frac{\partial}{\partial t} \langle x \rangle \tag{25}$$

right hand side via integration by parts ($\int g\, f\, dx = g\, F| - \int g' F\, dx$), note that $p(x = \pm\infty, t) = 0$, as well as all spatial derivatives ($\partial_x p(x = \pm\infty, t) = 0$):

$$D \int_{-\infty}^{+\infty} x \frac{\partial^2 p(x,t)}{\partial x^2} dx = D x\, \frac{\partial p(x,t)}{\partial x}\bigg|_{x=-\infty}^{x=+\infty} - D \int_{-\infty}^{+\infty} 1 \cdot \frac{\partial p(x,t)}{\partial x} dx \tag{26}$$

$$= 0 \qquad - D\, p(x,t)|_{x=-\infty}^{x=+\infty} = 0 \tag{27}$$

$$\Rightarrow \frac{\partial}{\partial t} \langle x \rangle = 0 \tag{28}$$

I.e. $\langle x \rangle \equiv$ const. for all t. For $x(t=0) = 0 \rightarrow \langle x \rangle = 0$ for all t.

Random walk VII

Analogously for $\langle x^2(t)\rangle$: integration by parts twice

$$\frac{\partial}{\partial t}\langle x^2(t)\rangle = 0 + 0 + 2D\int_{-\infty}^{+\infty} p(x,t)dx = 2D \tag{29}$$

$$\rightarrow \langle x^2(t)\rangle = 2D\,t \tag{30}$$

compare with Eq. (18) $\langle x^2(t)\rangle = \frac{\ell^2}{\Delta t}N\Delta t = \frac{\ell^2}{\Delta t}t$
$\rightarrow$ random walk and diffusion equation have same time dependence (linear)
(with $2D = \frac{\ell^2}{\Delta t}$)

Random numbers

Pseudorandom numbers I

for scientific purposes

- fast method to generate huge number of "random numbers"
- sequence should be reproducible

$\rightarrow$ use deterministic algorithm to generate *pseudorandom* numbers

Linear congruential method

start with a *seed* x_0, use one-dimensional map

$$x_n = (a\, x_{n-1} + c) \mod m \tag{31}$$

- with integers: a (multiplier), c (increment), m (modulus)
- m largest possible integer from Eq. (31) $\rightarrow$ maximum possible period is m $\rightarrow$ obtain $r \in [0, 1)$ by x_n/m
- real period depends on a, c, m, e.g.,
  $a = 3,\ c = 4,\ m = 32,\ x_0 = 1 \rightarrow 1, 7, 25, 15, 17, 23, 9, 31, 1, 7, 25, \ldots$ $\rightarrow$ period is 8 not 32

Other sources of random numbers I

Better randomness can be obtained from physical processes:

- nuclear decay (*real* randomness!), e.g, → measure Δt (difficult to implement)
- image noise, thermal noise (Johnson-Nyquist noise), e.g., → darkened USB camera (simple), special expansion cards with a diode
- "activity noise" in Unix:

  ```
  /dev/random
  /dev/urandom
  ```

  → random *bit* patterns from input/output streams (entropy pool) of the computer
  `/dev/random` blocks, if entropy pool is exhausted (since Linux 2.6: 4096 bit, cf. `/proc/sys/kernel/random/poolsize`)
  `urandom` uses pseudorandom numbers seeded with "real" random numbers

For readout of Unix random devices need to interpret random bits(!) as numbers

Reading from `urandom`

E.g., by using `fstream` and `union`

```
ifstream fin("/dev/urandom/") ;
union {unsigned int num ;
       char buf[sizeof(unsigned int)]; } u ;
fin.read(u.buf, sizeof(u.buf)) ;
cout << u.num ;
```

→ `fstream` reads only `char`, `buf` and `num` are at the same address → read bits in as `char` output as `unsigned int`

Tests for random numbers I

quality check for uniformly distributed random numbers

- *equal distribution:* random numbers should be fair
- *entropy:* bits of information per byte of a sequence of random numbers (same as equal distribution)
- *serial tests:* for n-tuple repetitions (often only for $n = 2$, $n = 3$)
- *run test:* for monotonically increasing/decreasing sequences, also for length of stay for a distinct interval
- and more . . .

Be careful!

There is no necessary or sufficient test for the randomness of a finite sequence of numbers.

$\rightarrow$ can only check if it is "apparently" random

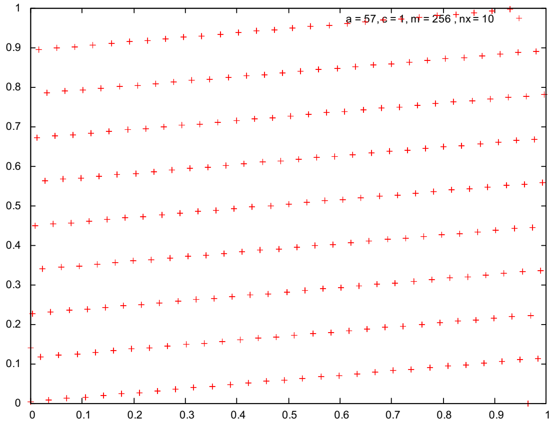
$\rightarrow$ testing for "clumping" of numbers

Test for doublets

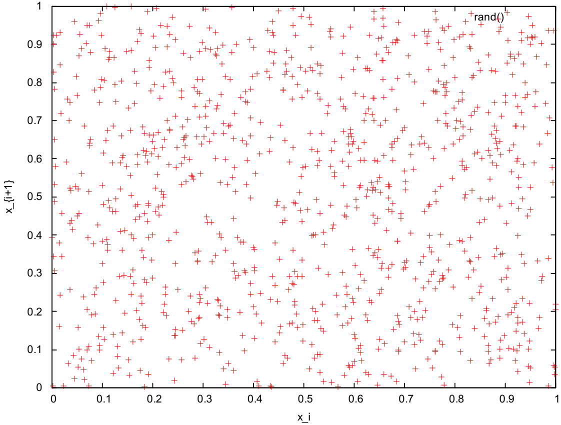
- define a square lattice $L \times L$ and fill each cell at random:
- array $n(x, y)$ with discrete coordinates
- choose random $1 \leq x_i, y_i \leq L$ where x_i, y_i consecutive numbers of random number sequence
- fill cell $n(x_i, y_i)$ (e.g. set boolean to true)
- repeat procedure $t \cdot L^2$ times, t is MC time step
- $\rightarrow$ similar to nuclear decay, therefore expected:
  fraction of empty cells $\propto \exp(-t)$

Simple correlation test

- just plot x_{i+1} over x_i $\rightarrow$ look for suspicious patterns



correlation plot for linear congruential method with bad parameters



same plot but for C++ `rand()` function

Confidence level I

Testing for randomness (also: numbers or detections)
$\rightarrow$ χ^2 test

- let y_i the number of events in bin i and E_i the expectation value
- e.g., $N = 10^4$ random numbers, $M = 100$ bins $\rightarrow E_i = 100$ (numbers/bin)
- the χ^2 value (with y_i measured number of random numbers in bin i):

$$\chi^2 = \sum_{i=1}^{M} \frac{(y_i - E_i)^2}{E_i} \tag{32}$$

  measures the conformity of the measured and the expected distribution

- the individual terms in Eq. (32) should be ≤ 1, so for M terms $\chi^2 \leq M \rightarrow$ *reduced* χ^2 by deviding by $M \rightarrow$ "minimum" red. $\chi^2 = 1$
- e.g., 5 independent runs (each $n = 10\,000$) yield $\chi^2 \approx 92, 124, 85, 91, 99 \rightarrow$ as expected for equal distribution,
  in general: χ^2 should be small (but $\chi^2 = 0$ is suspicious, e.g., here: N-periodicity in random numbers?)

Confidence

- need a quantitative measure that shows normal distribution of the "error" $(y_i - E_i)$ (in particular, we test the hypothesis of uniform distribution) $\rightarrow$ chi-squared distribution

$$p(x,\nu) = \frac{1}{2^{\nu/2}\,\Gamma(\nu/2)} x^{(\nu-2)/2}\, e^{-x/2} \tag{33}$$

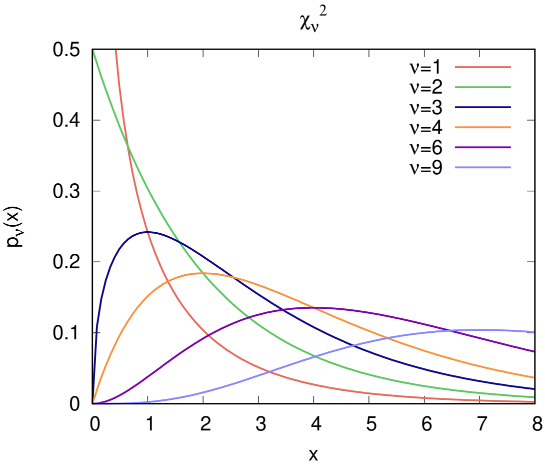
$$\text{where } \Gamma(z) = \int_0^\infty t^{z-1} e^{-t} dt \text{ and } \Gamma(z+1) = z! \tag{34}$$

$\rightarrow$ cumulated χ^2 distribution $P(x,\nu)$:

$$P(x,\nu) = \frac{1}{2^{\nu/2}\,\Gamma(\nu/2)} \int_0^x t^{(\nu-2)/2}\, e^{-t/2} dt \tag{35}$$

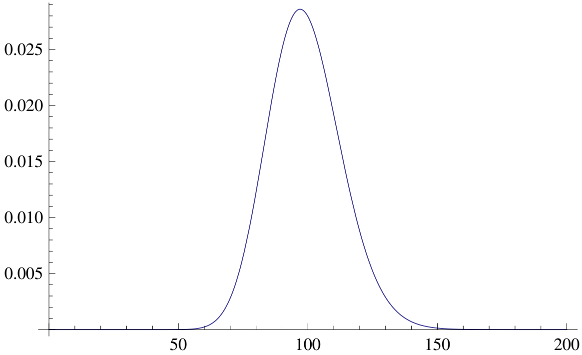
with ν degrees of freedom, here: $\nu = M - 1 = 99$, because of constraint $\sum_{i=1}^{M} E_i = N$

- chi-square distribution



chi-square PDF for different degrees of freedom

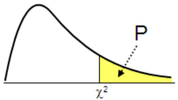
ν



for $\nu > 30$ is $\sqrt{2x} - \sqrt{2\nu - 1}$ approximately normally distributed, for $\nu > 100$ is x approximately normally distributed with $E = \nu$ and and $\sigma = \sqrt{2\nu}$

- function $Q(x,\nu) = 1 - P(x,\nu)$

  $\rightarrow$ probability that $\chi^2 > x$



- we want to check: How likely to get a χ^2 of, e.g., 124 (our largest measured χ^2)?
  $\rightarrow$ solve $Q(x,\nu) = q$ (probability $\chi^2 > x$ for given x,ν) for x, or look it up in tables for $\nu = M - 1 = 99$ (e.g., `https://www.medcalc.org/manual/chi-square-table.php`)

x	138.9	134.6	123.2	110.6	98
q	0.005	0.01	0.05	0.2	0.5

- for our case: 1 out of 5 runs (20%) had $y_2 = 124$, but $Q(x,\nu)$ implies for $x = 123$ only 5%, i.e., 1 out of 20 runs with $\chi^2 \geq 123$
- therefore: confidence level $< 95\%$, rather 80% (because of $q = 0.2$ for $x = 111$)
- try to increase confidence level: more runs $\rightarrow$ if still only 1 out 20 with $\chi^2 > 123$
  $\rightarrow$ confidence level at 95%