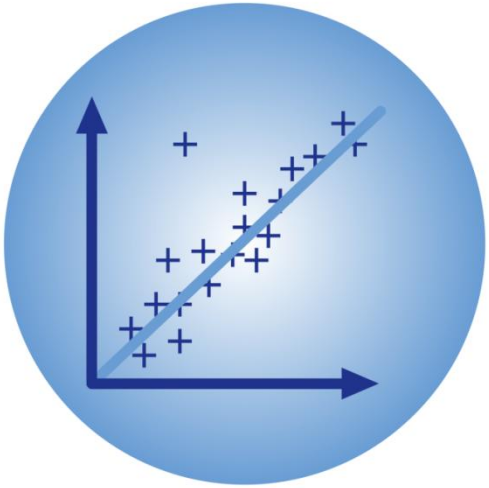


BPhO Computational Challenge

Seminar 09: Random walks

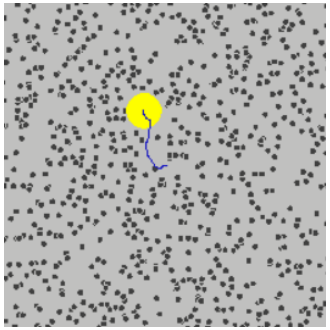
Dr Andrew French.

December 2021.

BPhO British Physics Olympiad

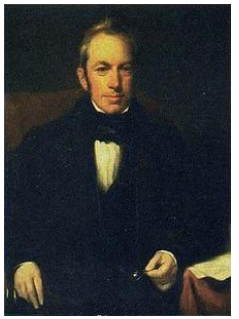
Brownian motion – a *random walk*

Brownian motion, initially observed as the random jittering of pollen grains in a microscope slide, is due to the random jostling of molecular motion. In the base of the pollen grains, it is the smaller (invisible) air molecules which are colliding at random. How far will a given particle move in a specified time, given its motion is random?



Brownian motion simulation

Consider motion in one direction in N steps of fixed length l. The caveat is that each step is either forward or backwards, and the direction is 'chosen' randomly.



Robert Brown (1773-1858)

The total displacement is $x = l\sum_{i=1}^{N} a_i$ where $a_i = -1 \text{or} 1$

A sensible measure of the distance travelled is the *root-mean-square (RMS) displacement:*

$$\sqrt{\langle x^2 \rangle} = l\sqrt{\left\langle \left(\sum_{i=1}^{N} a_i \right)^2 \right\rangle} = l\sqrt{\left\langle \sum_{i=1}^{N} a_i^2 + \sum_{i=1,i\neq j}^{N} \sum_{j=1}^{N} a_i a_j \right\rangle}$$

$$\left\langle \sum_{i=1}^{N} a_i^2 \right\rangle = N \quad \text{and} \quad \left\langle \sum_{i=1,i\neq j}^{N} \sum_{j=1}^{N} a_i a_j \right\rangle = 0$$

Since a is a random choice between -1 and 1

$$\therefore \boxed{\sqrt{\langle x^2 \rangle} = l\sqrt{N}}$$

$$\sqrt{\langle x^2 \rangle} = l\sqrt{\left\langle \left(\sum_{i=1}^{N} a_i \right)^2 \right\rangle} = l\sqrt{\left\langle \sum_{i=1}^{N} a_i^2 + \sum_{i=1, i\neq j}^{N} \sum_{j=1}^{N} a_i a_j \right\rangle}$$

$$x = l\sum_{i=1}^{N} a_i \qquad a_i = -1 \text{ or } 1$$

1D random walk

$$\left\langle \sum_{i=1}^{N} a_i^2 \right\rangle = N \quad \text{and} \quad \left\langle \sum_{i=1, i\neq j}^{N} \sum_{j=1}^{N} a_i a_j \right\rangle = 0$$

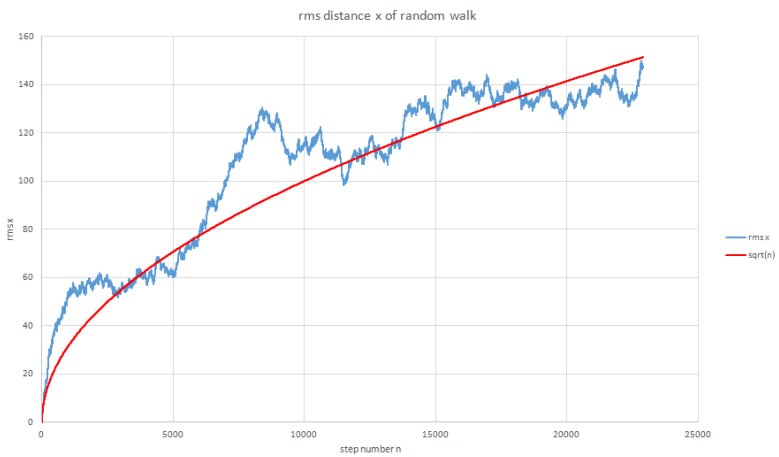
$$\therefore \sqrt{\langle x^2 \rangle} = l\sqrt{N}$$

$\sqrt{\langle v^2 \rangle}$ We might use the RMS speed here

If the average molecular speed is $\langle v \rangle$, the number of steps in t seconds is: $N = \dfrac{\langle v \rangle t}{l}$

Hence the RMS random walk displacement in t seconds is predicted to be:

$$\sqrt{\langle x^2 \rangle} = l\sqrt{N} = \sqrt{l\langle v \rangle t}$$



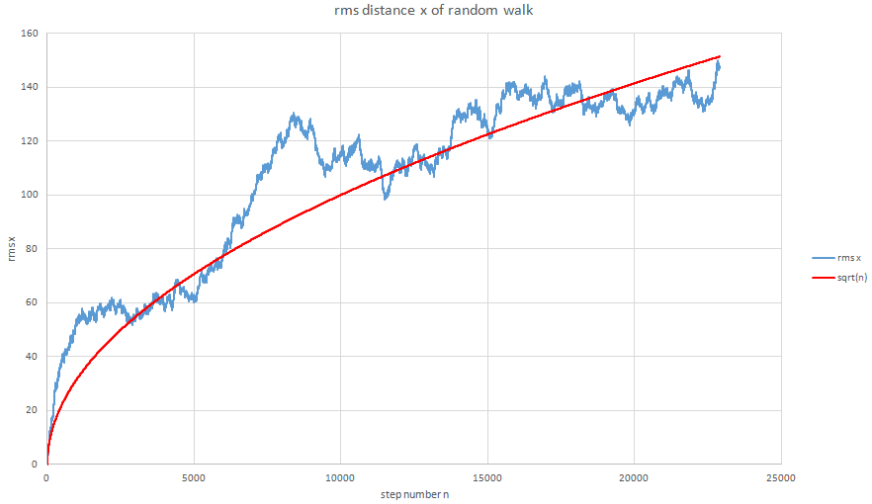
The step size l can be associated with the **mean free path** between molecular collisions. We can define the mean free path to be the average distance travelled by a molecule in time t divided by the number of molecules it will likely collide with in that time.

$$l = \frac{\sqrt{\langle v^2 \rangle}t}{\underbrace{\pi d^2 \sqrt{2}\sqrt{\langle v^2 \rangle}t}\times n}$$

Distance travelled by molecule

'Interaction volume'

number of molecules per unit volume

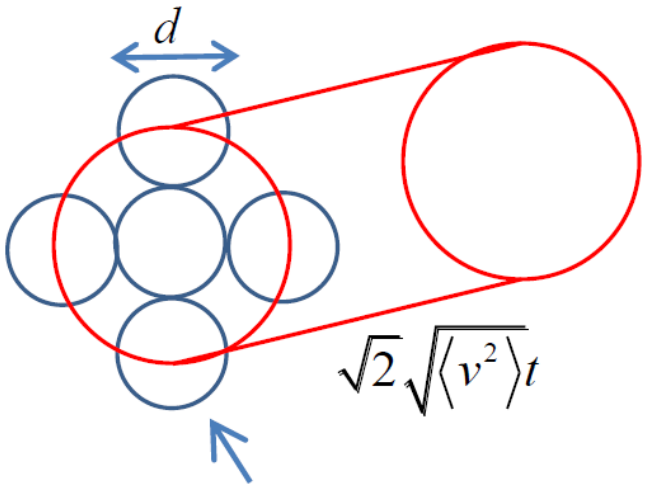


$$l = \frac{1}{\pi\sqrt{2}d^2 n}$$

Mean free path

The interaction volume is root 2 larger because all molecules are in *relative* motion. Hence the length of the 'interaction tube' is proportional to the average *relative* speed

Mean free path



Colliding particles, assumed to be circular with diameter d

velocities of molecules i and j

$$\langle v_{rel} \rangle = \sqrt{\left\langle \left| \mathbf{v}_i - \mathbf{v}_j \right|^2 \right\rangle}$$

$$\langle v_{rel} \rangle = \sqrt{\left\langle \left(v_i^2 + v_j^2 - 2\mathbf{v}_i \cdot \mathbf{v}_j \right) \right\rangle}$$

$$\langle v_{rel} \rangle = \sqrt{2\langle v^2 \rangle - 2\langle \mathbf{v}_i \cdot \mathbf{v}_j \rangle}$$

$$\langle v_{rel} \rangle = \sqrt{2}\sqrt{\langle v^2 \rangle}$$

We can determine the **mean free path** for an ideal gas by using the **Ideal Gas Equation**

Since the number of moles is $\frac{nV}{N_A}$ ← volume

pressure → $pV = \frac{nV}{N_A}RT$ ← Absolute temperature

Molar gas constant

$R = 8.314\,\text{Jmol}^{-1}\text{K}^{-1}$

$\therefore n = \frac{p}{k_B T}$

Boltzmann's constant $k_B = 1.38 \times 10^{-23}\,\text{JK}^{-1}$

Hence $\boxed{l = \frac{k_B T}{\pi\sqrt{2}d^2 p}}$

If we divide this by the particle diameter d we arrive at **Knudsen's number (Kn)**. This dimensionless constant determines whether our statistical mechanics argument is valid, or whether a 'continuum' concept is needed.

The latter model is what is used to describe much of **fluid mechanics** i.e. where we consider the fluid as a continuously varying entity rather than a series of discrete, and randomly moving, molecules colliding.

$$l = \frac{k_B T}{\pi \sqrt{2} d^2 p}$$

$\mathrm{Kn} \ll 1$	Continuum
$\mathrm{Kn} > 1$	Statistical mechanics

For a typical air molecule on Earth

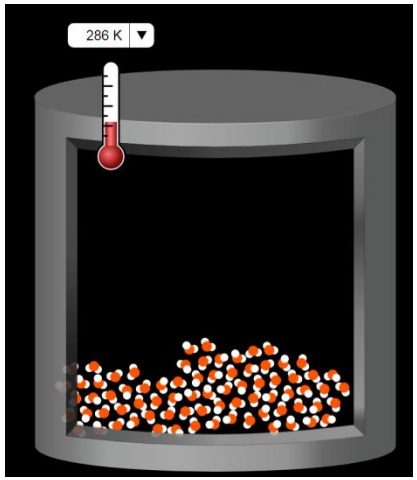
$$d = 0.3\mathrm{nm},\ p = 10^5\,\mathrm{Pa},\ T = 293K$$

$$\therefore l = 1.0 \times 10^{-7}\,\mathrm{m}$$

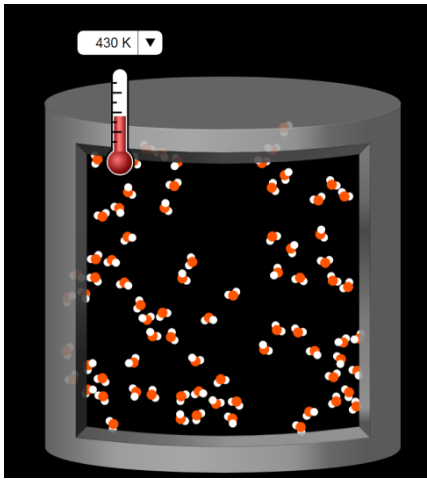
$$\therefore \mathrm{Kn} = \frac{l}{d} = 333$$

So a statistical argument is justified

Fluid



Gas

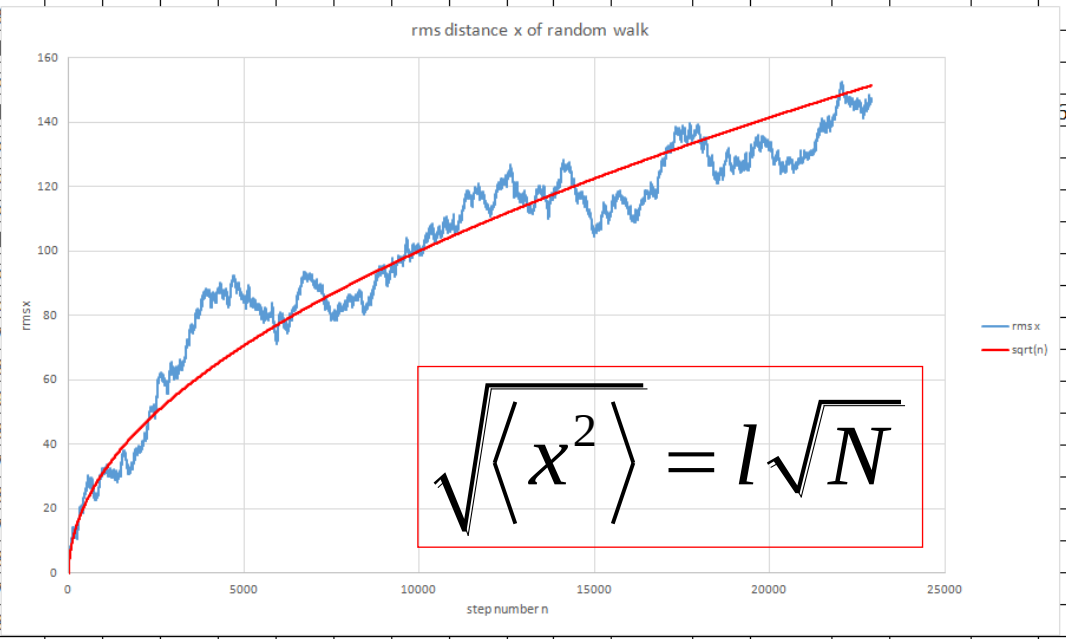


PhET States of Matter

1D RANDOM WALK SIM A. French Nov 2019

STEP SIZE = 1

		1	2	3	4	5	6	7	8	9	10	11	12		1	2	3	4	5	6	7	8	9	10	11	12	
n	**sqrt(n)**	**x**	**x**	**x**	**x**	**x**	**x**	**x**	**x**	**x**	**x**	**x**	**x**		**x^2**	**x^2**	**x^2**	**x^2**	**x^2**	**x^7**	**x^2**	**x^2**	**x^2**	**x^2**	**x^2**	**x^2**	**rms x**
0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	-1	1	1	1	-1	1	1	-1	-1	1	-1		1	1	1	1	1	1	1	1	1	1	1	1	1
2	1.4142	2	-2	2	2	2	0	0	2	-2	0	0	0		4	4	4	4	4	0	0	4	4	0	0	0	1.528
3	1.7321	3	-3	1	3																		1	1	1	1	1.915
4	2	4	-4	2	4																		4	0	0	0	2.38
5	2.2361	5	-5	1	5																		1	1	1	1	2.887
6	2.4495	4	-4	0	4																	5	0	4	0	0	2.708
7	2.6458	3	-5	-1	5																		1	9	1	1	2.887
8	2.8284	4	-4	-2	6																		0	4	4	4	3
9	3	3	-5	-3	5																		1	9	1	1	3.109
10	3.1623	2	-6	-2	4																		4	4	4	4	2.944
11	3.3166	1	-7	-1	5																		1	9	9	9	3.512
12	3.4641	0	-8	0	6																		0	16	16	4	4
13	3.6056	-1	-7	-1	7																		1	25	25	1	4.359
14	3.7417	0	-6	-2	8																		4	16	36	4	4.546
15	3.873	-1	-5	-1	9																		9	25	25	9	4.865
16	4	-2	-4	0	8																		16	36	16	4	4.655
17	4.1231	-3	-3	-1	7																		9	25	25	1	4.655
18	4.2426	-4	-2	-2	8																		16	36	16	0	5.196
19	4.3589	-3	-3	-1	7																		25	25	25	1	5.066
20	4.4721	-2	-2	-2	8																		36	36	36	0	5.715
21	4.5826	-1	-3	-3	7																		25	49	25	1	5.627
22	4.6904	-2	-4	-4	8																		16	64	16	0	5.538
23	4.7958	-1	-5	-3	9	9	-3	11	-3	3	-9	5	-1		1	25	9	81	81	9	121	9	9	81	25	1	6.137
24	4.899	-2	-4	-2	8	8	-4	10	-2	4	-10	4	0		4	16	4	64	64	16	100	4	1				
25	5	-3	-5	-1	7	9	-3	11	-3	3	-9	5	-1		9	25	1	49	81	9	121	9	9				
26	5.099	-2	-4	-2	8	8	-4	10	-4	4	-8	6	-2		4	16	4	64	64	16	100	16	1				
27	5.1962	-3	-5	-1	7	7	-3	11	-3	5	-9	7	-1		9	25	1	49	49	9	121	9	2				
28	5.2915	-2	-6	0	6	6	-2	12	-4	4	-8	6	-2		4	36	0	36	36	4	144	16	1				
29	5.3852	-1	-5	1	7	5	-1	11	-5	3	-9	5	-3		1	25	1	49	25	1	121	25	9				
30	5.4772	-2	-4	0	8	4	0	10	-6	2	-10	4	-2		4	16	0	64	16	0	100	36	4				
31	5.5678	-3	-5	1	9	3	1	11	-5	3	-9	5	-1		9	25	1	81	9	1	121	25	9				

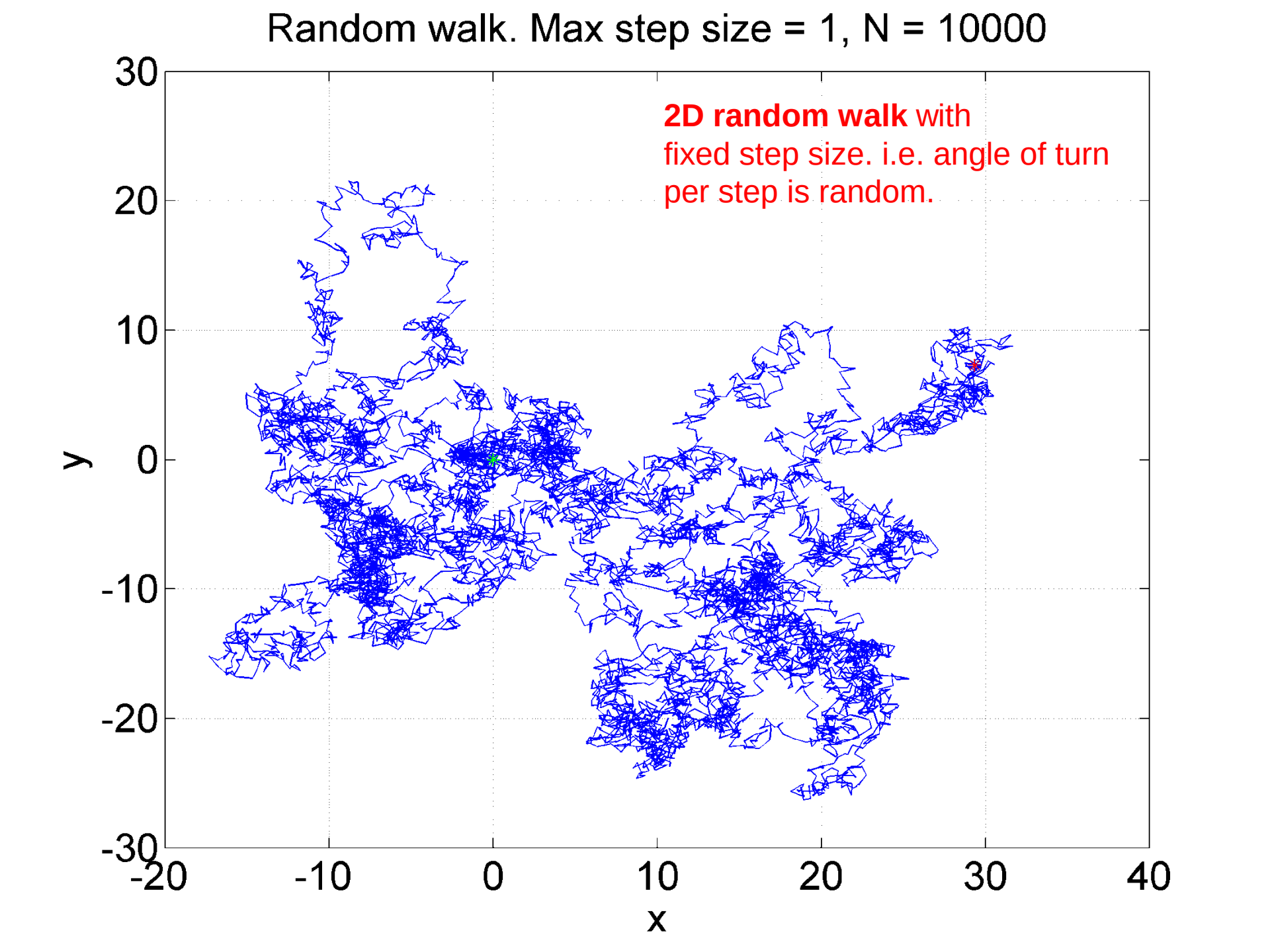


$$x = l\sum_{i=1}^{N} a_i$$

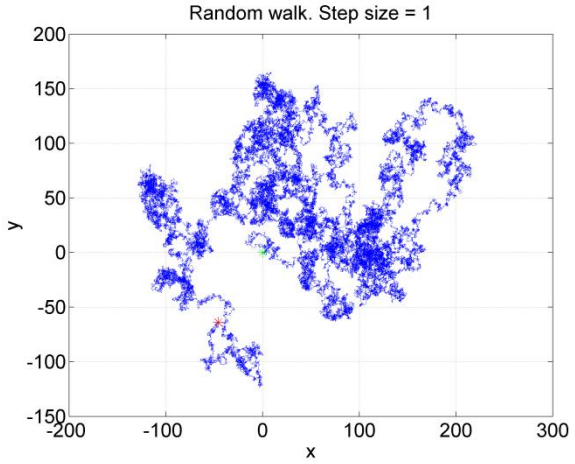
$$a_i = -1 \text{or} 1$$

$$l = 1$$

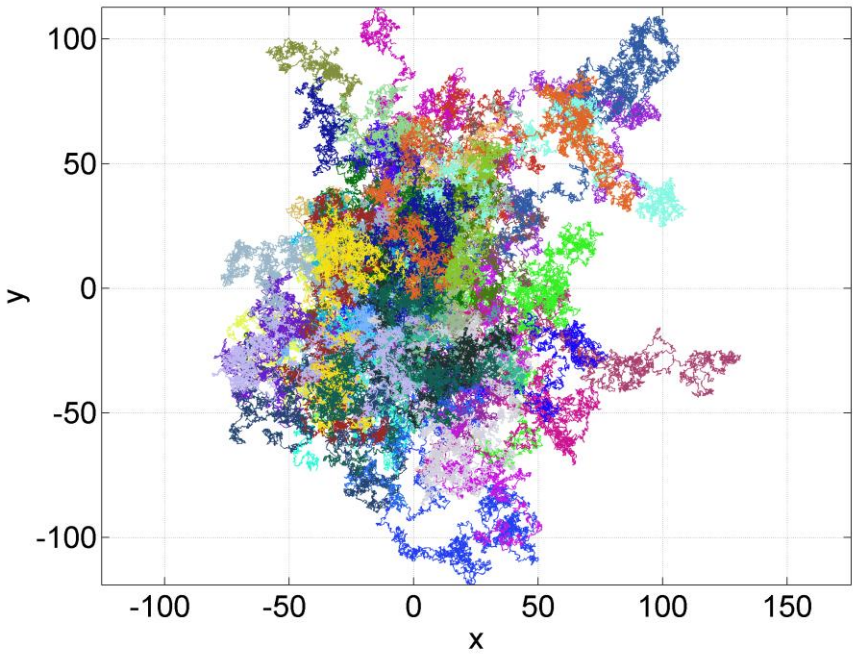
1D random walk simulation in Excel

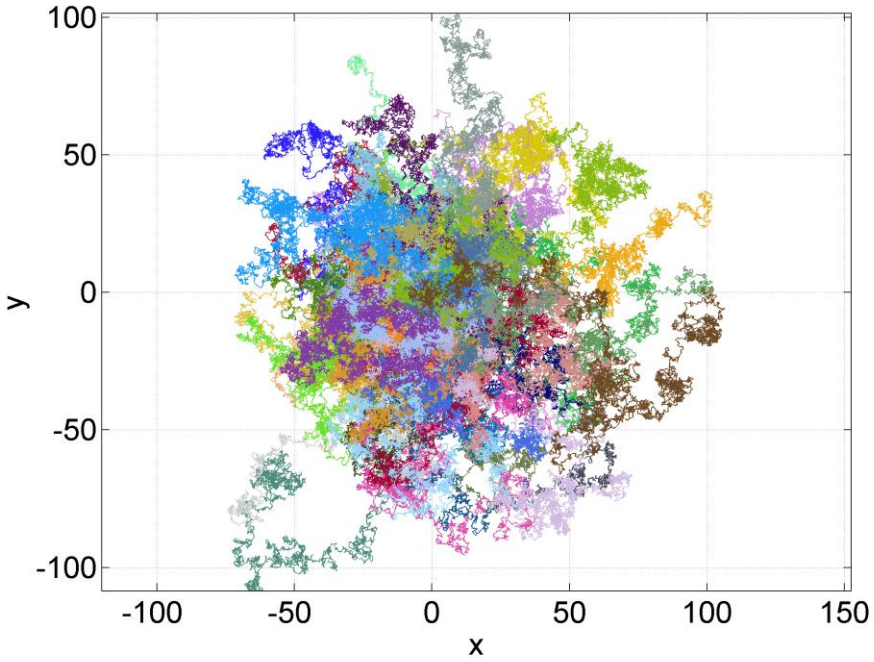

Random walk. Max step size = 1, N = 10000
2D random walk with
fixed step size. i.e. angle of turn
per step is random.
30
20
10
0
-10
-20
-30
y
-20
-10
0
10
20
30
40
x

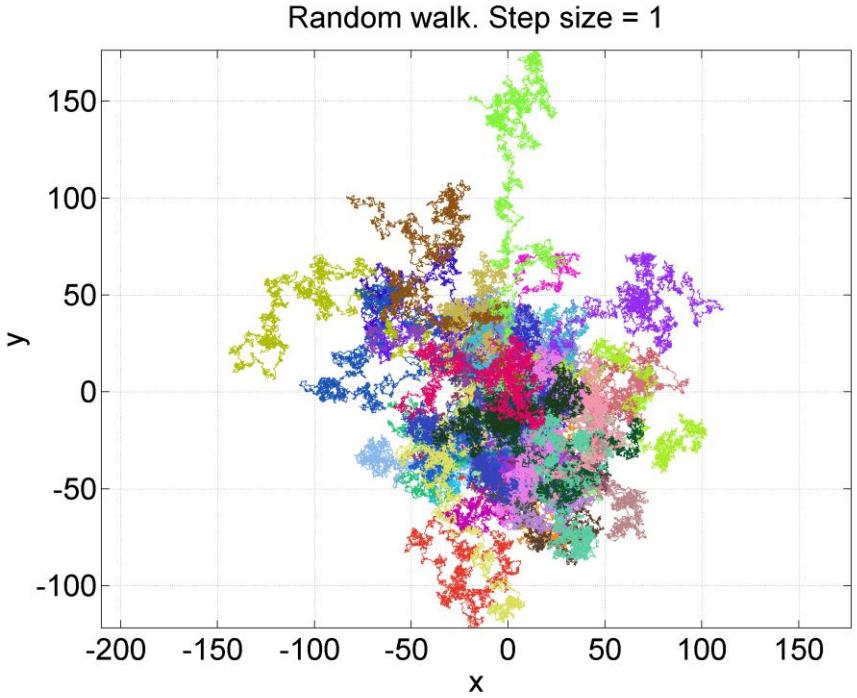
2D random walk
MATLAB
simulation

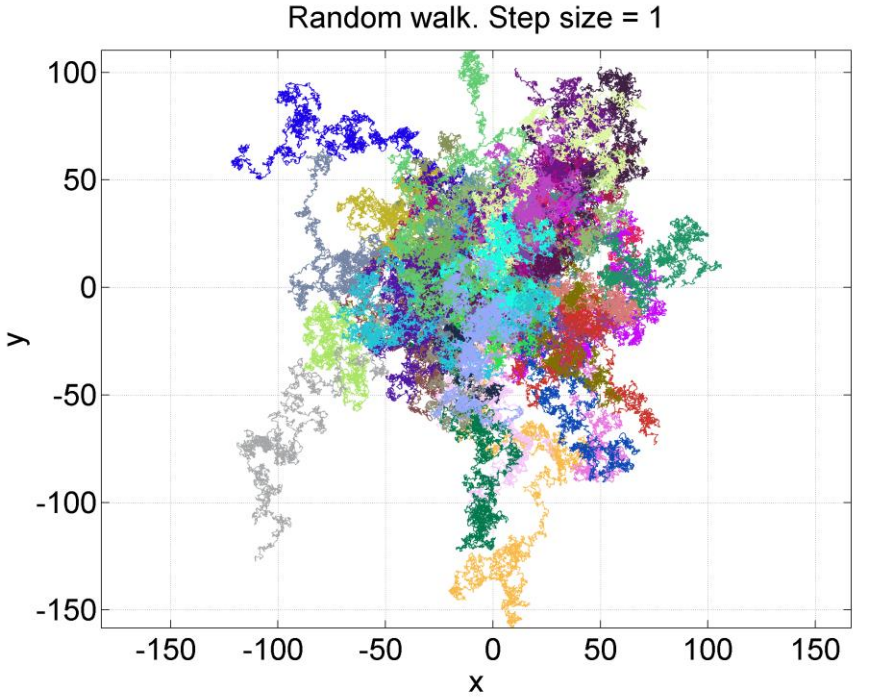



```
% A visual representation of a random walk.
% Step sizes are fixed, but directions are random.

%Number of steps
N = 1e6;

%Fixed step size
s = 1;

%Initilize x,y position vectors, starting from the origin.
x = zeros(1,N); y = zeros(1,N);

%Determine random walk
for n=2:N
    theta = 2*pi*rand;
    x(n) = x(n-1) + s*cos(theta); y(n) = y(n-1) + s*sin(theta);
end

%Plot random walk
plot(x,y,'b-'); hold on;
plot( x(1),y(1),'g*' ); plot( x(end),y(end),'r*' );
xlabel('x'); ylabel('y'); title( ['Random walk. Step size = ',num2str(s)] );
grid on;

%Print a PNG file of the random walk
print( gcf, 'random walk.png','-dpng','-r300' );

%End of code
```


Random walk. Step size = 1
100
50
0
-50
-100
y
-100
-50
0
50
100
150
x

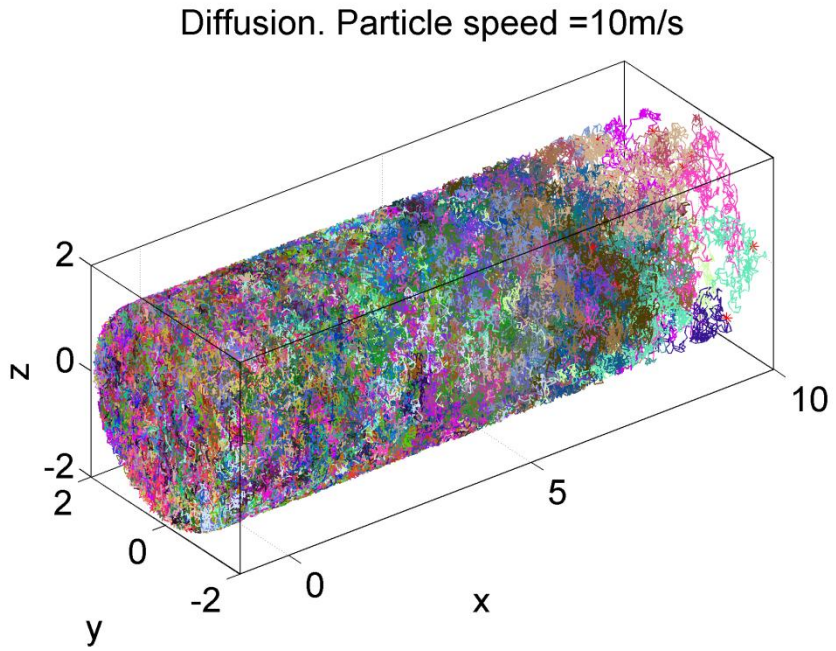

Random walk. Step size = 1
100
50
0
-50
-100
y
-100
-50
0
50
100
150
x

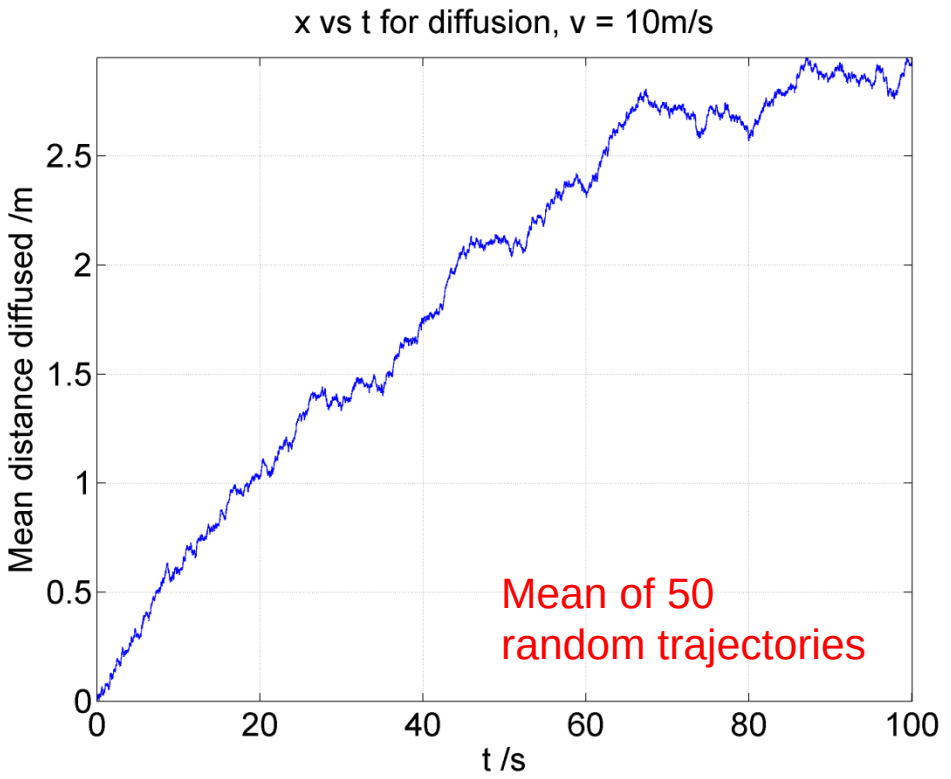

Random walk. Step size = 1
150
100
50
0
-50
-100
y
-200
-150
-100
-50
0
50
100
150
x

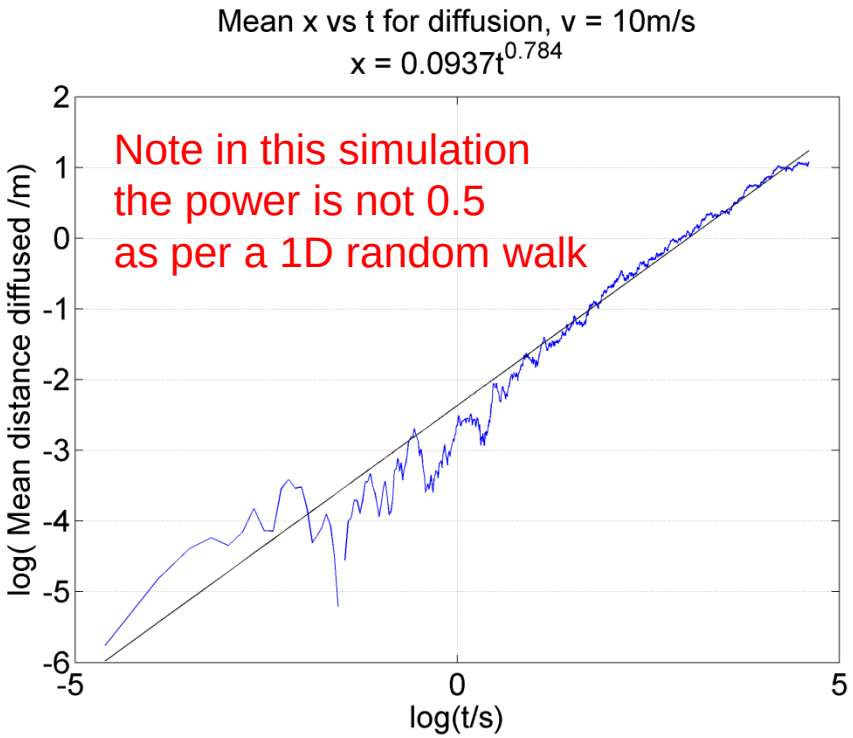

Random walk. Step size = 1
100
50
0
-50
-100
-150
y
-150
-100
-50
0
50
100
150
x

MATLAB implementation of multiple random walks (in a loop)

```
function random_walks

P = 42;  %Numbers of random walks
N = 5000;  %Number of steps
s = 1;  %Fixed step size
fsize = 18; %Graph fontsize

%Initialize axes and then plot random walks
axes('nextplot','add','fontsize',fsize);
for n=1:P
    [x,y] = randomwalk(N,s);
    RGB = rand(1,3); plot(x,y,'-','color',RGB);
end
xlabel('x'); ylabel('y'); title( ['Random walk. Step size = ',num2str(s)] );
grid on; axis equal; box on;

%Print a PNG file of the random walk
filename = ['random walks ',strrep(datestr(now),':','-'),'.png'];
print( gcf, filename,'-dpng','-r300' );
close(gcf);

%%

%Random walk generator
function [x,y] = randomwalk(N,s)
x = zeros(1,N); y = zeros(1,N);
for n=2:N
    theta = 2*pi*rand;
    x(n) = x(n-1) + s*cos(theta); y(n) = y(n-1) + s*sin(theta);
end
```

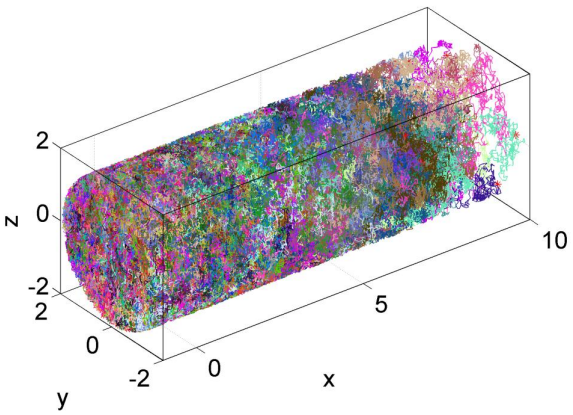


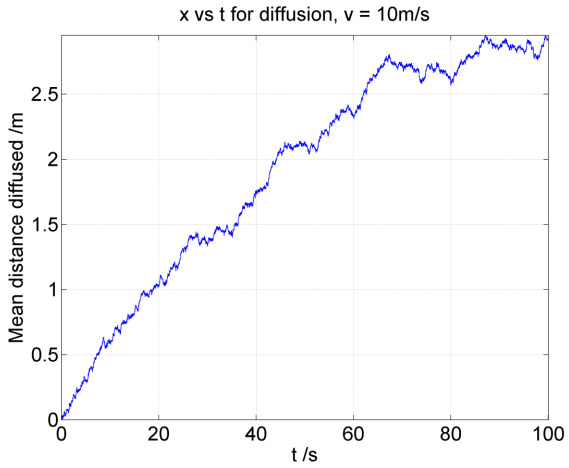


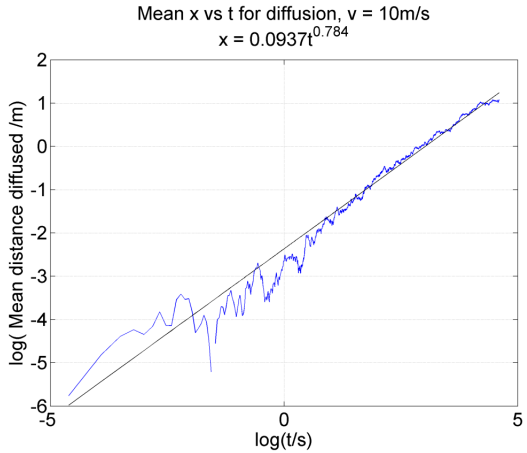


We can compute a 3D **diffusion** model *efficiently* by using a random walk.

The random walk gets around the need to keep track of thousands of particles and their collisions.

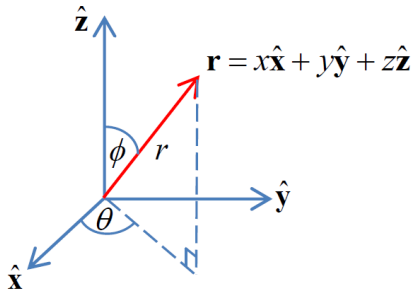




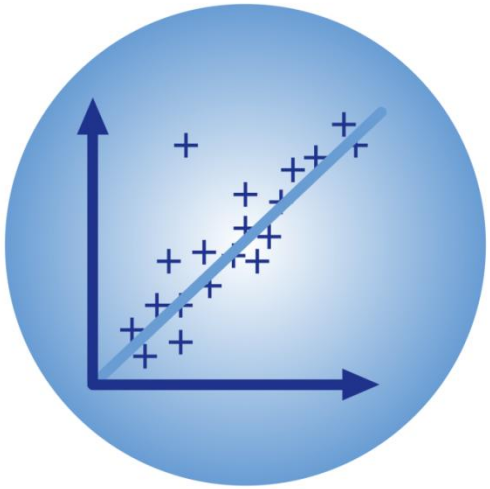



```
%Determine trajectory
function [x,y,z,t] = diffusion_trajectory(a,r,s,dt,tmax)

%Initialize x,y,z,t coordinates
x = 0; y = 0; z = 0; t = 0;

%Compute the random walk!
n=1;
while t(end) < tmax

    %Choose random direction angles
    azi = 2*pi*rand; elev = -0.5*pi + pi*rand;

    %Define magnitude of displacement during timestep
    d = s*dt;

    %Compute next x,y position
    dx = d*cos(elev)*cos(azi);
    dy = d*cos(elev)*sin(azi);
    dz = d*sin(elev);

    %Check if particle has hit a side wall
    if ( ( y(end)+dy )^2 + ( z(end) + dz )^2 ) > r^2
        dz = -dz; dy = -dy;
    end

    %Check if particle has hit back wall
    if ( x(end) + dx ) < a
        dx = -dx;
    end

    %Update x,y,z trajectory
    x(n+1) = x(n) + dx; y(n+1) = y(n) + dy; z(n+1) = z(n) + dz;
    t(n+1) = t(n) + dt; n = n+1;
end
```

Random angles
in azimuth and elevation



Assume elastic
collisions with walls



BPhO Computational Challenge

- Suggested homework
- Q&A

BPhO
British Physics Olympiad