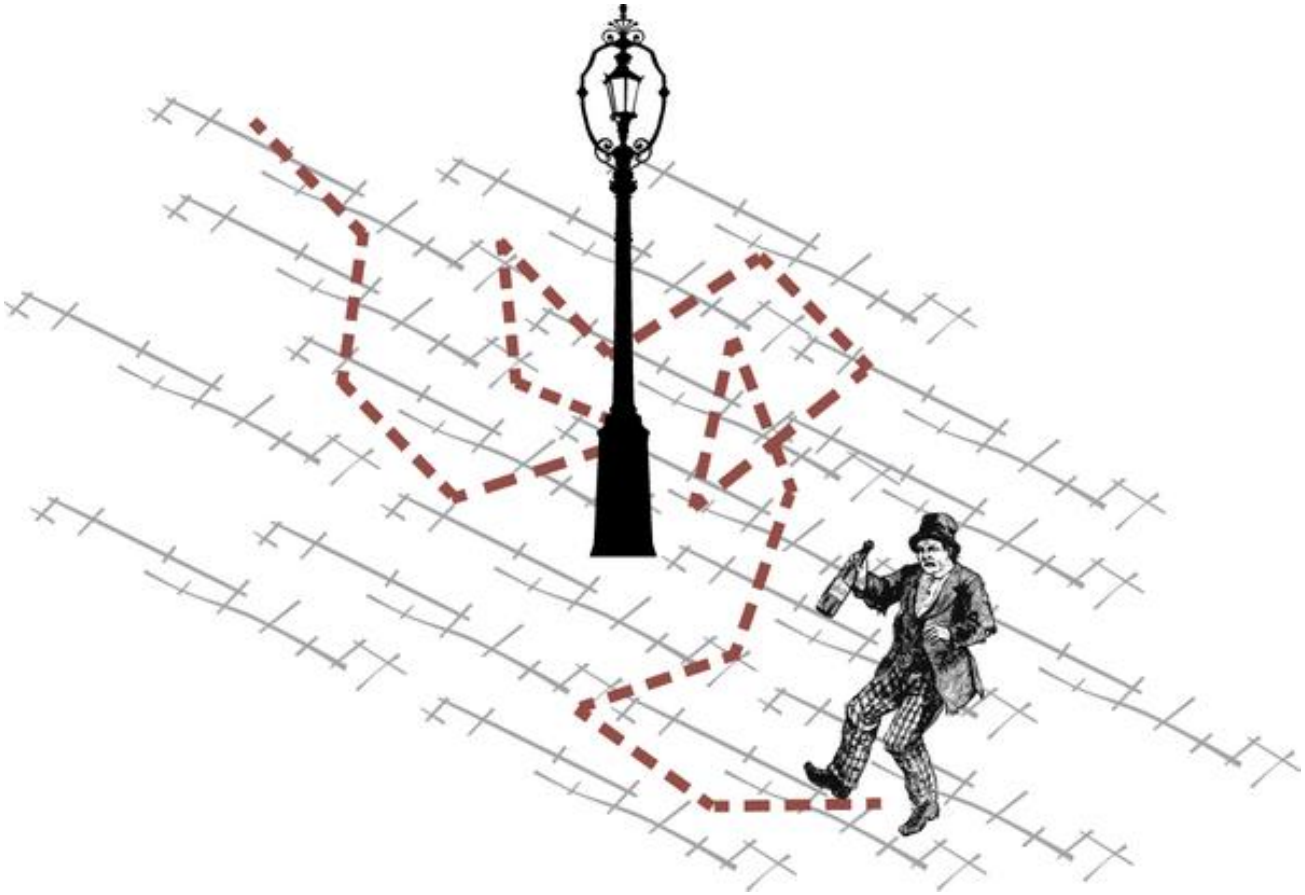
Metallurgia fisica



Esercitazione 5

La diffusione:
il random walk model

Il Random Walk Model



…anche noto come il drunkard's model
Random walk=passeggiata aleatoria

Modello continuo della diffusione

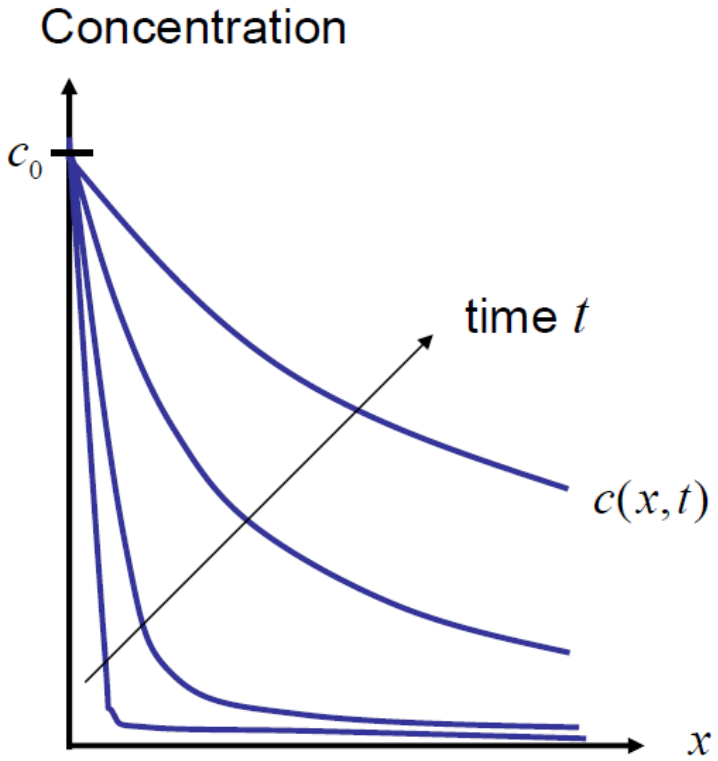
Legge di Fick: $\frac{\partial c}{\partial t} = D\frac{d^2c}{dx^2}$



$$c\,(x, t) = c_s - (c_s - c_0)\mathrm{erf}\left(\frac{x}{2\sqrt{Dt}}\right)$$

BC: $c\,(x = 0) = c_0$

IC: $c\,(x > 0,\ t = 0) = 0$



C'è bisogno di conoscere il coefficiente di diffusione per poter risolvere il problema. Come si procede?

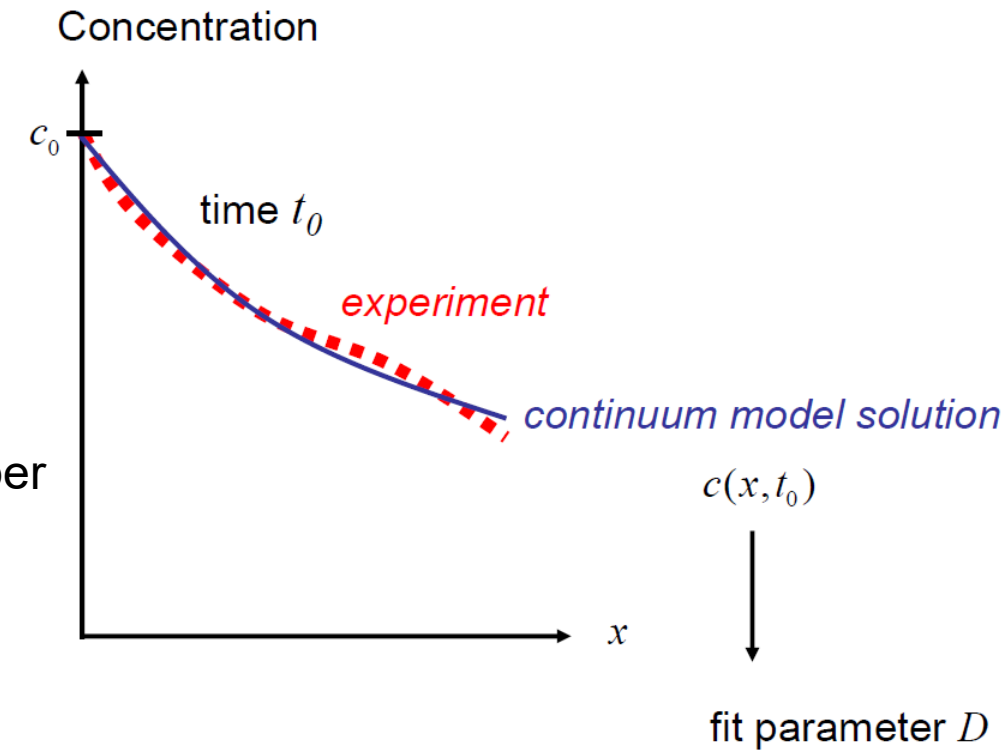
Come ottengo in pratica D?

Approccio top-down

Misuro sperimentalmente la concentrazione e faccio un fit della curva

Trovo D

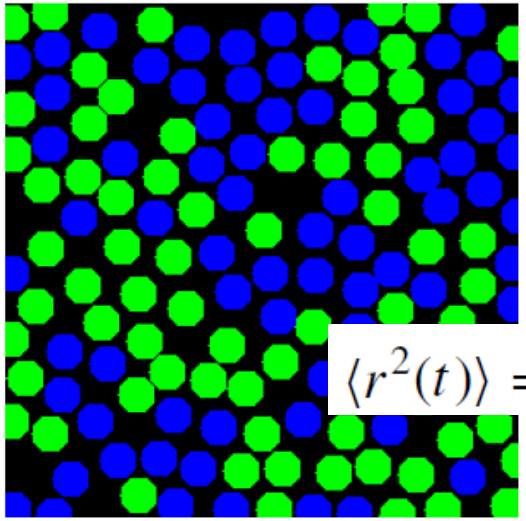
Controllo che sia il valore corretto per un problema differente (diverse condizioni al contorno)



Posso utilizzare il valore trovato per risolvere il problema in caso di situazioni più complesso per cui le misure non possono essere effettuate.

Modello atomistico della diffusione

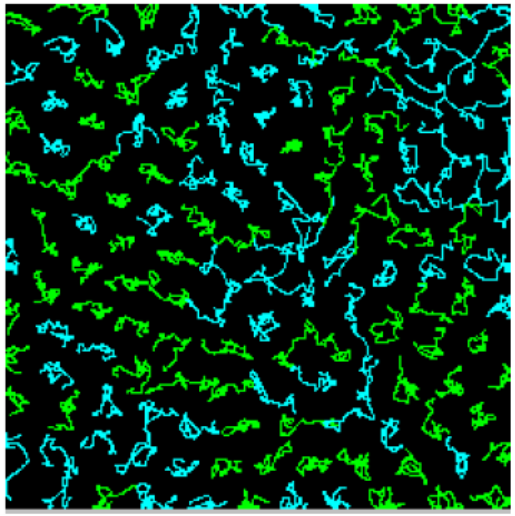
Il coefficiente di diffusione di un materiale può essere calcolato considerando lo spostamento quadratico medio degli atomi che lo compongono per tempi sufficientemente lunghi



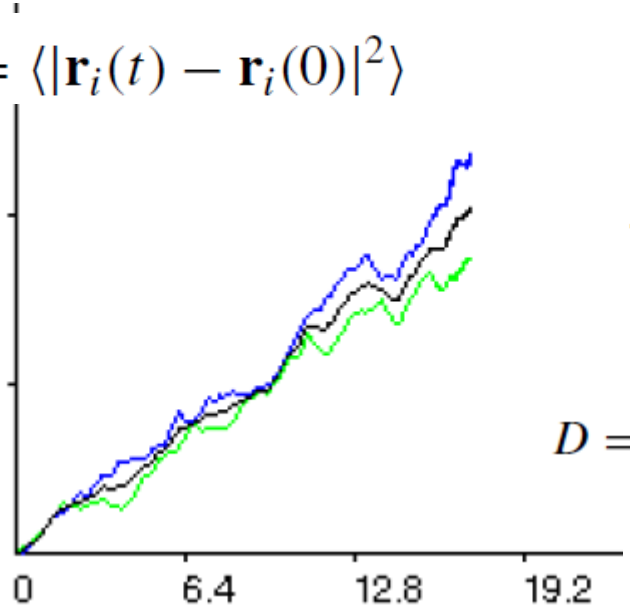
Particles

Mean square displacement

$$\langle r^2(t) \rangle = \langle |\mathbf{r}_i(t) - \mathbf{r}_i(0)|^2 \rangle$$



Trajectories



$$D = \frac{1}{6t}\langle R^2 \rangle$$

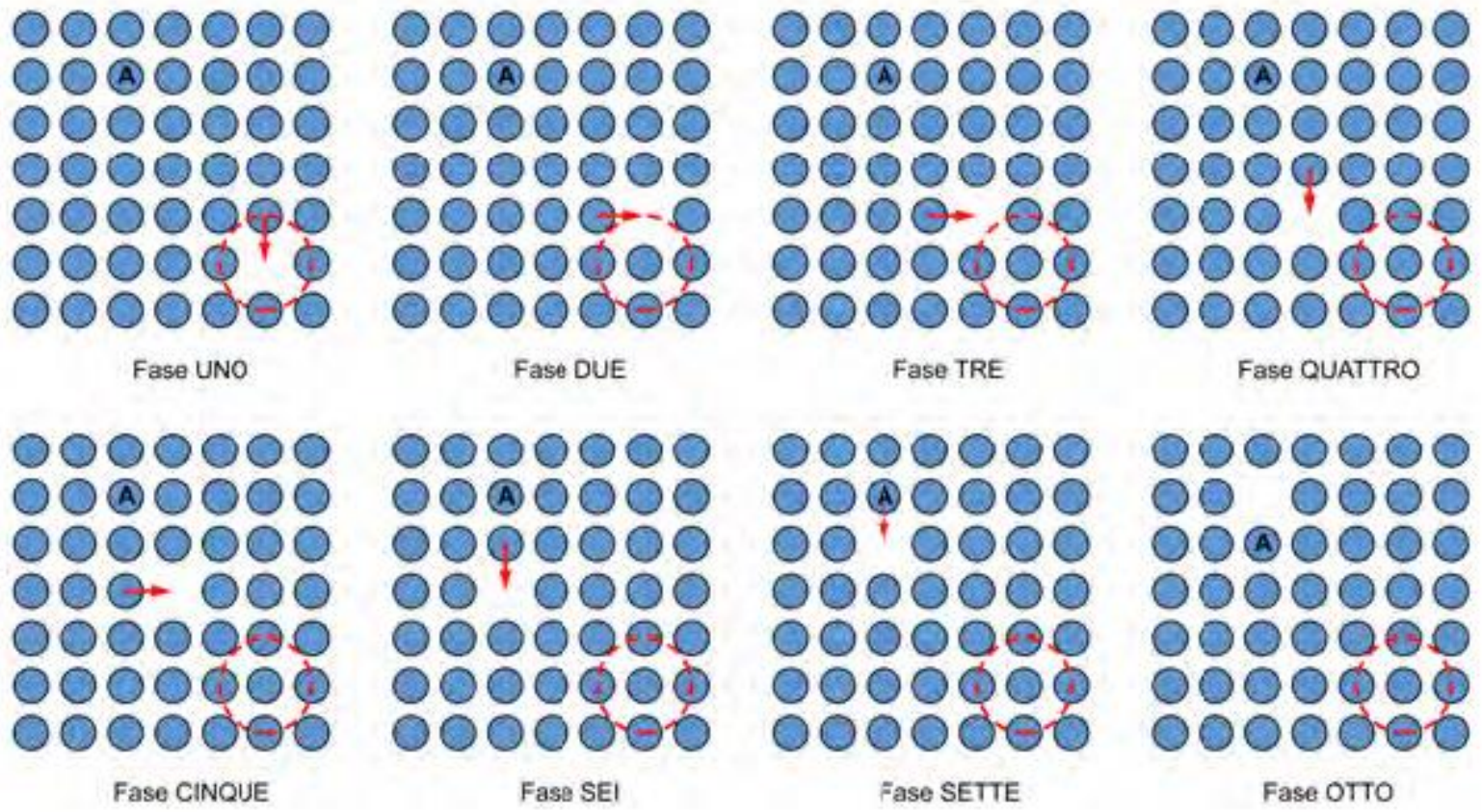
Autodiffusione in un solido cristallino

In un solido cristallino la diffusione avviene solamente se si verificano due condizioni:

1. Presenza di vacanze

2. Temperatura sufficientemente elevata

L’autodiffusione



Consegna:

Scrivete un programma per calcolare lo spostamento diffusivo di una singola lacuna sulla superficie di un cristallo assumendo che:

1) Il cristallo abbia struttura cristallina cubica semplice
2) Le vibrazioni atomiche possano essere ignorate (il tempo necessario perché un atomo riesca a saltare in una lacuna è molto più lungo)
3) Sia presente un'unica lacuna
4) La lacuna ad ogni passo può andare ad occupare solo la posizione degli atomi che hanno distanza minima da essa.

Calcolate la distanza quadratica media.

Input, output?

Input, output, function

1) Identificare input e output

OUTPUT: coordinate ***x,y*** della vacanza ad ogni passo e spostamento al quadrato ***rs2***

INPUT: numero di passi ***nt***

2) Definire una funzione in matlab che colleghi input ed output:

function [out1, out2, …] = function_name (in1, in2,…)

```
function[rs2,x,y]= rwn(nt)
```

Programma

Scrivere un programma, chiamato rwn.m, per:

1) Tracciare la traiettoria di una singola vacanza sul reticolo quadrato che si muove di **nt** passi

2) Per ogni passo calcolare a che distanza si trova rispetto all'origine, ed elevare la distanza al quadrato

Passi principali

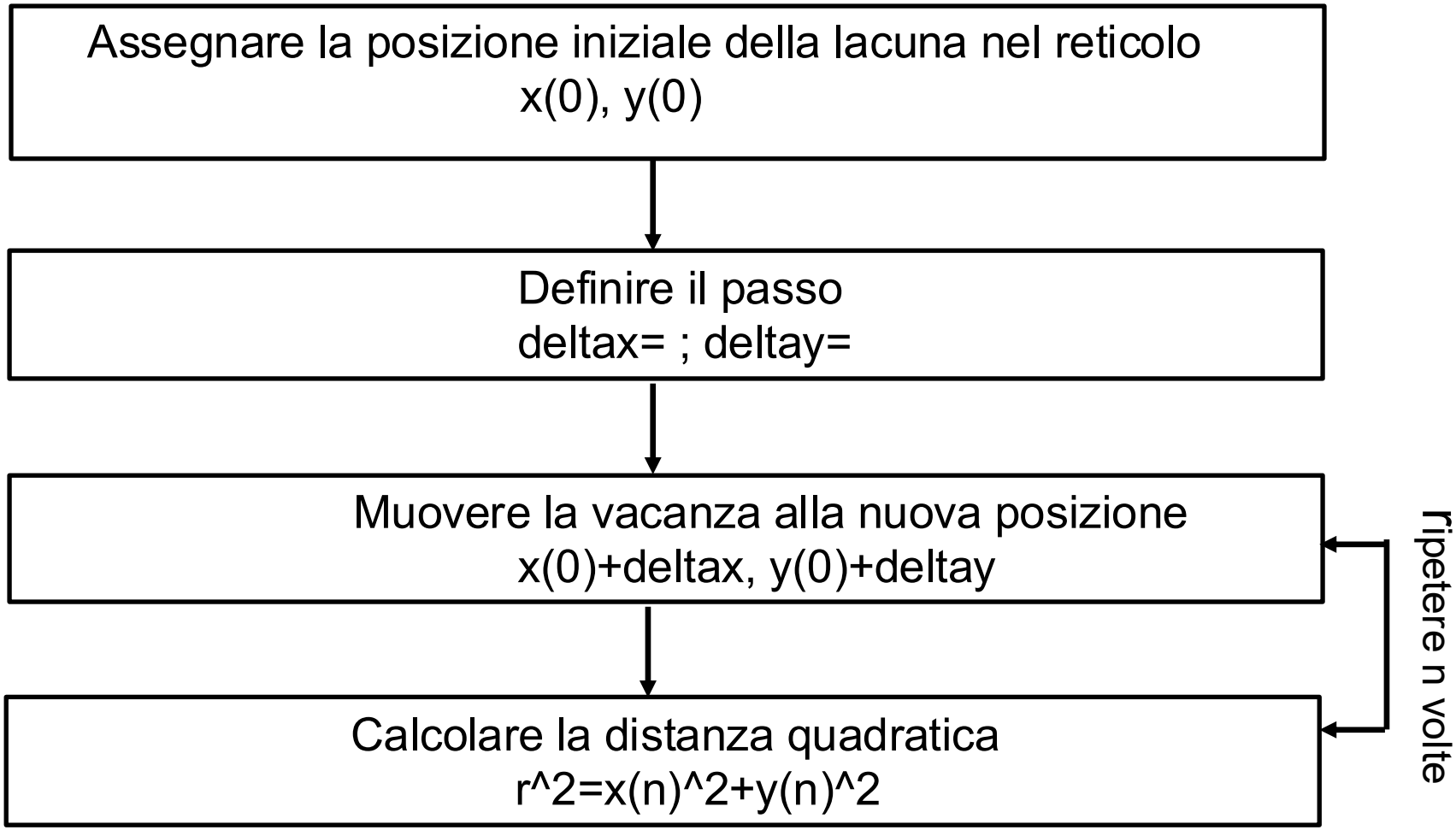
1. Assegnare la posizione iniziale della lacuna nel reticolo

2. Muovere la vacanza alla nuova posizione (memorizzare la sua posizione)

3. Calcolare la distanza dall'origine ed elevarla al quadrato

4. Ripetere

Diagramma di flusso



Posizione di partenza

```
function[rs2,x,y]= rwn(nt)
```

```
x(1)=0; % start particle at the origin
y(1)=0; % start particle at the origin
rs2(1)=0; % square displacement is 0 for first step
```

Muoviamo la vacanza

1) Verso quale posizione si muove la vacanza? Definire I salti sul reticolo (quadrato): destra, alto, sinistra, in basso

2) Come si muove? Selezionate i passi in maniera casuale

.

Muoviamo la vacanza

1) Verso quale posizione si muove la vacanza? Definire I salti sul reticolo (quadrato): destra, alto, sinistra, in basso

Per ogni passo ci sono 4 possibilità

(delx(1),dely(1)) = (1,0) = right
(delx(2),dely(2)) = (0,1) = up
(delx(3),dely(3)) = (-1,0) = left
(delx(4),dely(4)) = (0,-1) = down

```
delx = [1  0  -1  0];
dely = [0  1  0  -1];
```

Selezionare i salti

(delx(1),dely(1)) = (1,0) = right
(delx(2),dely(2)) = (0,1) = up
(delx(3),dely(3)) = (-1,0) = left
(delx(4),dely(4)) = (0,-1) = down

```
delx = [1  0  -1  0];
dely = [0  1  0  -1];
```

2) Creare una lista di numeri casuali da 1 a 4, con ***nt*** *valori*

Ingrediente fondamentale: un generatore di numeri casuali

In MATLAB rand(nt,1) dà nt numeri casuali da zero a 1 escluso.

Simple random walk code: rwn.m

```
% random walk on a square lattice
% input the number of jumps nt
% output is the square displacement rs2 and the list of positions x and y
function[rs2,x,y]= rwn(nt)
x(1)=0; % start particle at the origin
y(1)=0;
rs2(1)=0;

nt=100;
fd=ceil(4.*rand(nt,1));% list with nt entries, each a random integer from 1 to 4

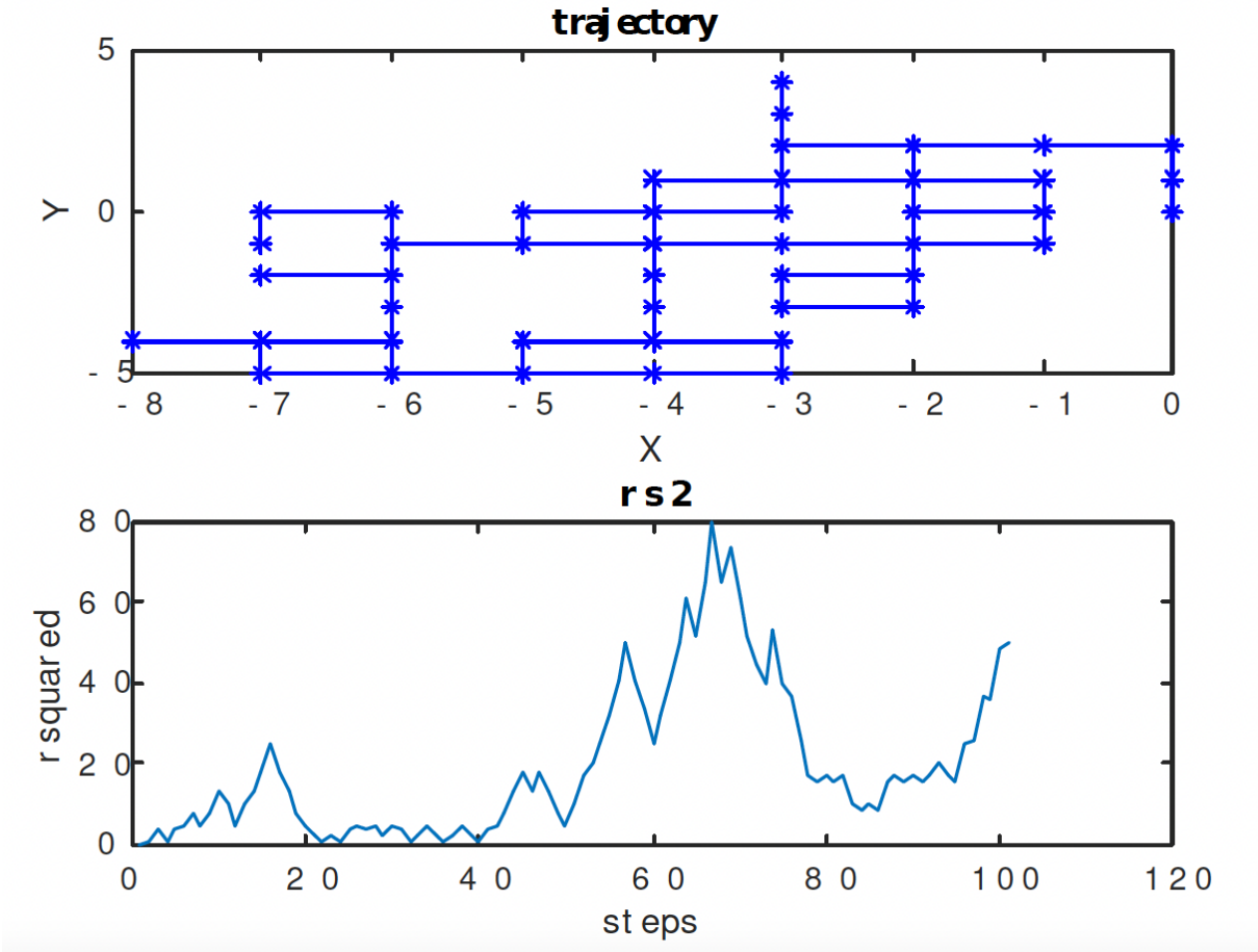
delx = [1 0 -1 0]; % these two lines define the jumps on the square lattice
dely = [0 1 0 -1]; % right, up, left, down

for j=1:nt  % sum over nt jumps
x(j+1)=x(j)+delx(fd(j));  % x position at j+1 jump
y(j+1)=y(j)+dely(fd(j));  % y position at j+1 jump
rs2(j+1)=x(j+1)^2 + y(j+1)^2; % square displacement position at j+1 jump
end
```

Plot

```
subplot(2,1,1)
plot(x,y,'-b*')
title('trajectory')
xlabel('X')
ylabel('Y')
subplot(2,1,2)
plot(rs2)
xlabel('steps')
ylabel('rsquared')
title('rs2')
```

Traiettoria per singola vacanza



Lanciare la simulazione

Una simulazione in cui la lacuna si muove di nt=100 passi e una in cui su muove di nt=500 passi.

Si plotti:

1) La traiettoria
2) Il quadrato della distanza in funzione del numero di salti

Lanciare tre simulazioni con nt=500 e confrontare traiettorie e distanze al quadrato.

Esercitazione

Per studiare la diffusione sulla superficie del cristallo cubico semplice dobbiamo considerare il moto di un numero molto elevato di lacune, in modo da poter considerare una media rappresentativa del fenomeno.

Scrivere una funzione arw.m che calcoli la distanza quadratica media tra *nd* lacune e plottarla in funzione del numero di passi.

Si testi il programma per un numero piccolo di vacanze.

Come fate a stabilire se il numero di vacanze che avete considerato è sufficiente?

Simulazioni

Chi ha tempo:

1. confronti la diffusione sulla superficie con quella all'interno del solido.
2. confronti la diffusione casuale con quella che si ottiene se si ha una direzione preferenziale. Come varia la diffusione al variare del 'coefficiente di direzionalità'?
3. consideri l'interazione tra vacanze

arw.m

```
% averages over many random walkers
% input:  nt = number of jumps, nd = number of trials
% output:  rwa = mean square displacement,

function [rwa] = arw2(nt,nd)

nt=100;
nd=50;
for j = 1:nt %initialize
    rwa(j)=0;
end

  for j = 1:nd % loop over trials
    [rs2,x,y]=rwn(nt);  % call random walker code
    for k = 1:nt % increment R^2 at each time step (jump)
        rwa(k) = rwa(k) + rs2(k); % increment R^2 at each time step (jump)
    end
  end
        for j=1:nt % divide by number of trials to get average
            rwa(j) = rwa(j)/nd;
        end
```