Computational Material Science

Lucia Nicola

Department of Industrial Engineering
University of Padova

lucia.nicola@unipd.it

General aim

The aim of this course is to

1. obtain a general overview of the modeling techniques used nowadays in material science and engineering

2. experience how simple simulations are designed

3. learn how to write a short code to perform simulations

4. experience how painful it can be to debug a code

5. learn how to avoid common mistakes through `sanity checks'

6. interpret results of the simulations

Course structure

- Classes on Wednesdays, from 8:30 am to 11:30 am (?)

- Start with a lecture, then assignments in class

- Two presentations (March 27 and May 8) given by **you**: `Modeling in *your preferred research topic*'.

- No books.

Course content

- Intro: what is a model and a simulation

- Modeling the nano-scale: atomistic simulations

- Modeling the micro-scale: dislocation dynamics simulations

- Modeling the macro-scale: the finite element method

Your presentations

1. Present a research topic of your liking. What type of simulation work has been done so far in this research area/topic? Search for a review paper to present on March 27

2. Select from the references in the review paper an article of your liking to present on May 8

The evaluation

- 30% presentation articles and discussion

- 70% evaluation and discussion of computer assignments

Today

1. What is a model, and a simulation?

2. What type of simulations are used in materials science & engineering?

3. Modeling materials at different scales

4. In class assignment: the random walk model

Learning goals of the day:

• be able to distinguish between a model and a simulation
• become aware of the existence of modeling techniques appropriate at different scales
• distinguish between a discrete and a continuum model
• be able to write your first computer program to model a random walker

Modeling and simulation

What is the difference between a model and a simulation?

A model

Physical situation

Model

The model misrepresents distances and directions but clearly captures connectivity

Modeling and simulation

What is the difference between a model and a simulation?

• Model: a mathematical description of a physical phenomenon. It differs from a theory in that it idealises the real behavior to a given accuracy, does not attempt to capture it fully.

• Simulation: procedure of solving the set of equations provided by the model.

A useful model

Essentially, all models are wrong, but some are useful.
G. Box

- Omits or distorts the inessential to accurately capture the relevant physics of the problem

- Predicts behavior under conditions that were not previously studied

Computational Materials Science

Is the employment of modeling and simulation to understand and predict material behavior.

Used in combination with experiments to reach understanding in processes that govern material behavior.

In modern material science it has also become a tool to discover new materials, compounds or alloys, and material phenomena.

It providea qualitative and even quantitative predictions that can be used to design new materials with advanced properties.

Seeks understanding: by simplifying a problem, thought experiments

Predicts behavior: of unstudied phenomena, difficult to achieve or very time/resources consuming

Responsibility of the modeler

VERIFICATION and VALIDATION

Responsibility of the modeler

VERIFICATION: check that the program (code) has no mistakes (bugs), and that it really calculates what is set to calculate

VALIDATION: check that the model gives a good description of the system of interest
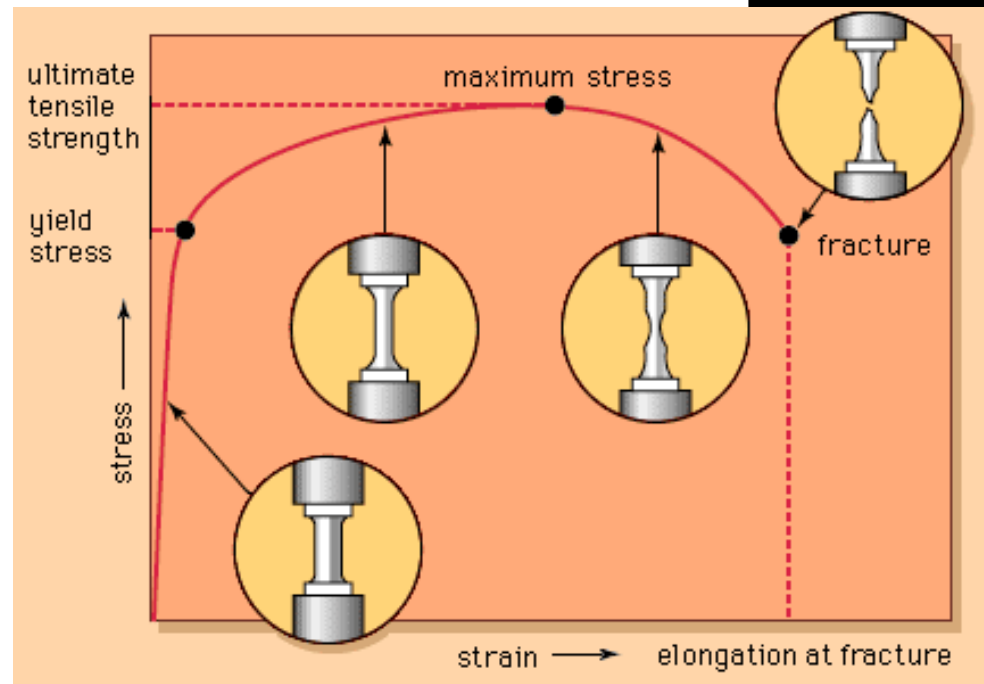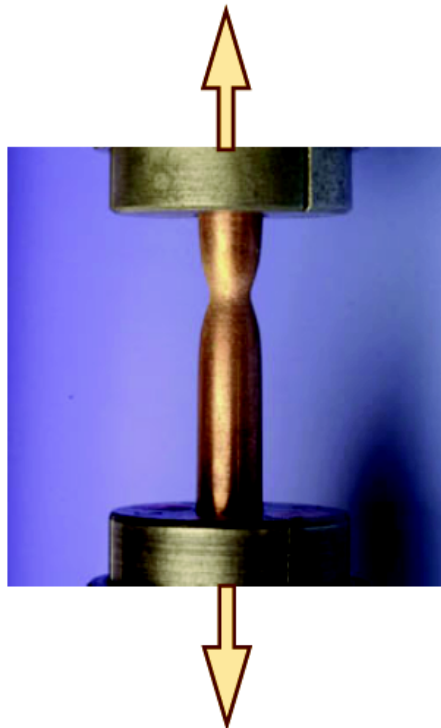
Mechanical behavior of materials

Engineering materials

▶ Engineers normally think of a material like copper as an infinitely divisible continuum medium.

▶ The mechanical properties of such a material are obtained by performing experiments, such as a tensile test:
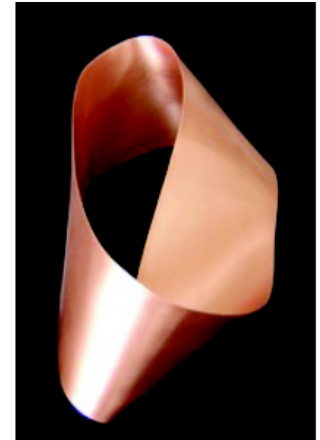
www.fergusonsculpture.com

ultimate tensile strength
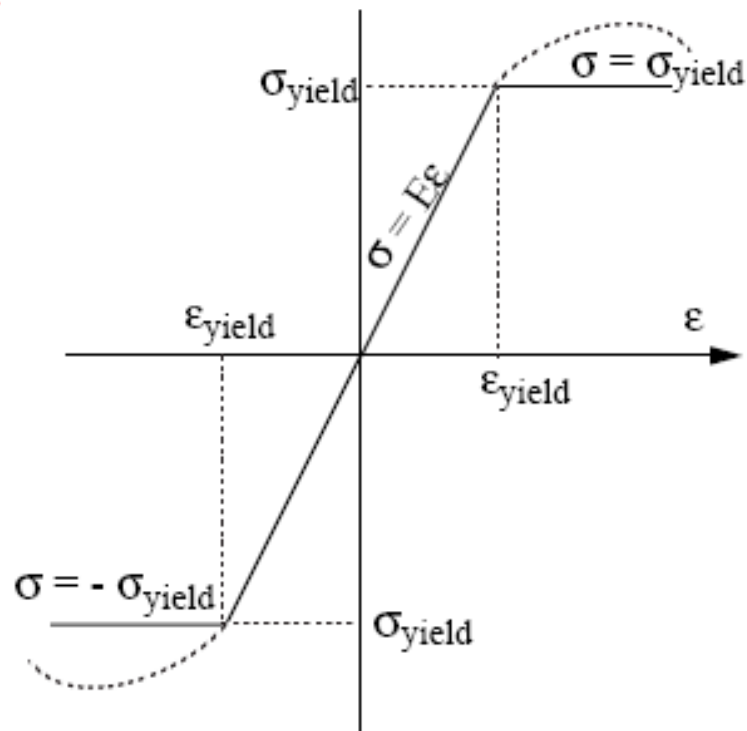
maximum stress

yield stress

fracture

stress

strain →   elongation at fracture

16

brittle vs ductile

Engineering tensile stress, σ

small toughness (ceramics)

large toughness (metals)

very small toughness (unreinforced polymers)

Adapted from Fig. 6.13, *Callister 7e.*
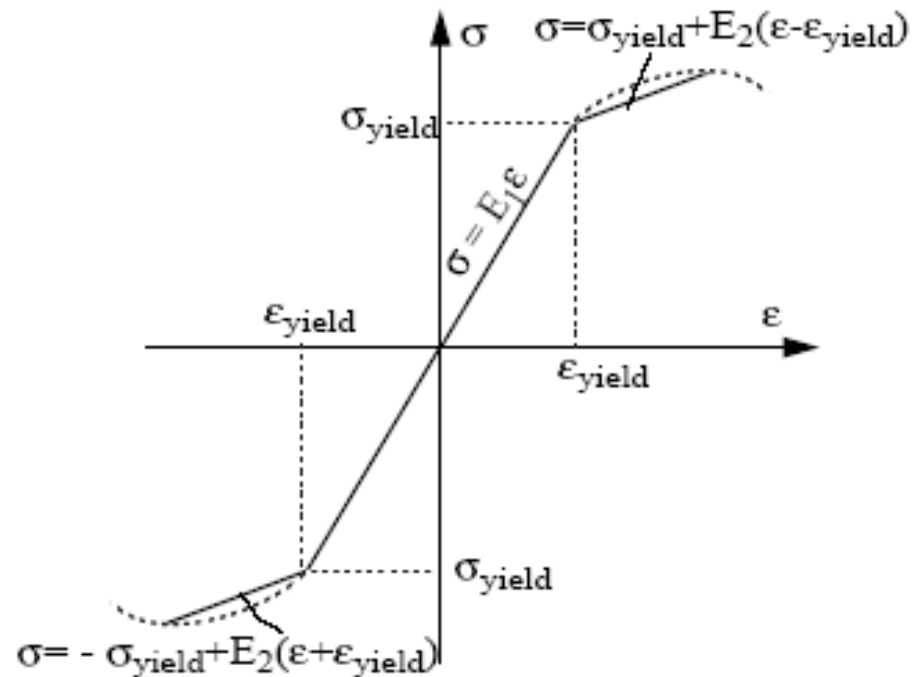
Engineering tensile strain, ε
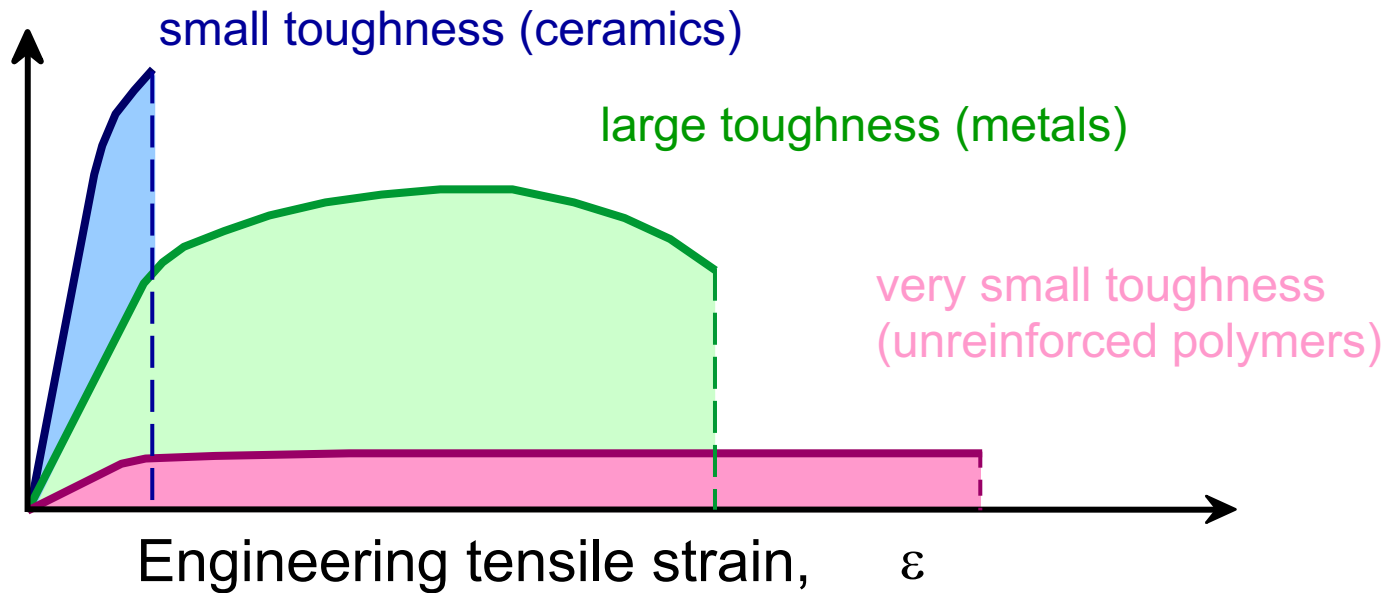
Elastic-perfectly plastic

$$\sigma = \begin{cases} \sigma_{yield} & \varepsilon \geq \varepsilon_{yield} \\ E\varepsilon & -\varepsilon_{yield} \leq \varepsilon \leq \varepsilon_{yield} \\ -\sigma_{yield} & \varepsilon \leq -\varepsilon_{yield} \end{cases} \qquad \tau = \begin{cases} \tau_{yield} & \gamma \geq \gamma_{yield} \\ G\gamma & -\gamma_{yield} \leq \gamma \leq \gamma_{yield} \\ -\tau_{yield} & \gamma \leq -\gamma_{yield} \end{cases}$$
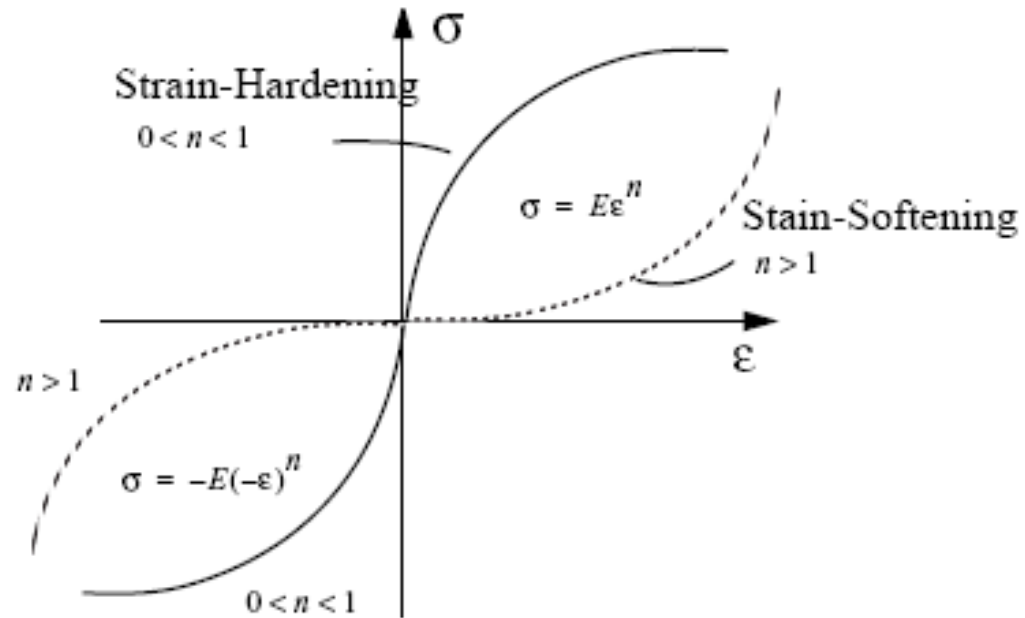
Linear strain hardening

$$\sigma = \begin{cases} \sigma_{yield} + E_2(\varepsilon - \varepsilon_{yield}) & \varepsilon \geq \varepsilon_{yield} \\ E_1\varepsilon & -\varepsilon_{yield} \leq \varepsilon \leq \varepsilon_{yield} \\ -\sigma_{yield} + E_2(\varepsilon + \varepsilon_{yield}) & \varepsilon \geq \varepsilon_{yield} \end{cases}$$
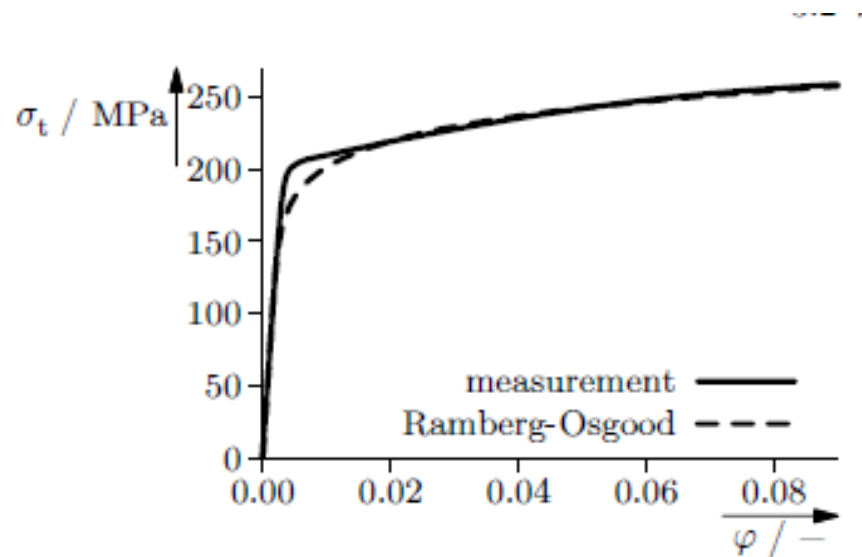
Non-linear strain hardening

Power Law

$$\sigma = \begin{cases} E\varepsilon^n & \varepsilon \geq 0 \\ -E(-\varepsilon)^n & \varepsilon < 0 \end{cases}$$

Ramberg-Osgwood law

$$\varphi = \frac{\sigma_t}{E} + \left(\frac{\sigma_t}{K}\right)^{1/n}$$

$$\sigma_t = K\varphi^n$$

Fig. 3.16. True stress-strain curve of AlMgSi 1 with fitted Ramberg-Osgood equation. The used parameters are $E = 66.2\,\mathrm{GPa}$, $K = 327\,\mathrm{MPa}$, $n = 0.10$

n=from 0.1 to 0.45

Phenomenological models

Q: How do you think the previous models were designed?

Phenomenological models

Q: How do you think the previous models were designed?

A: They are inferred from observation, thus fully phenomenological

Q: Why are they useful?

Phenomenological models

Q: How do you think the previous models were designed?

A: They are inferred from observation, thus fully phenomenological

Q: Why are they useful?

A: They can be used instead of performing additional experiments to predict the behavior of specimens similar to the ones tested experimentally.
They can possibly be extended to different shapes and loading conditions.

Q: What are their pitfalls?

Phenomenological models

Q: How do you think the previous models were designed?

A: They are inferred from observation, thus fully phenomenological

Q: Why are they useful?

A: They can be used instead of performing additional experiments to predict the behavior of specimens similar to the ones tested experimentally.
They can possibly be extended to different shapes and loading conditions.

Q: What are their pitfalls?
A: A slightly different material (composition, crystal structure, phase, shape) can lead to significantly different results
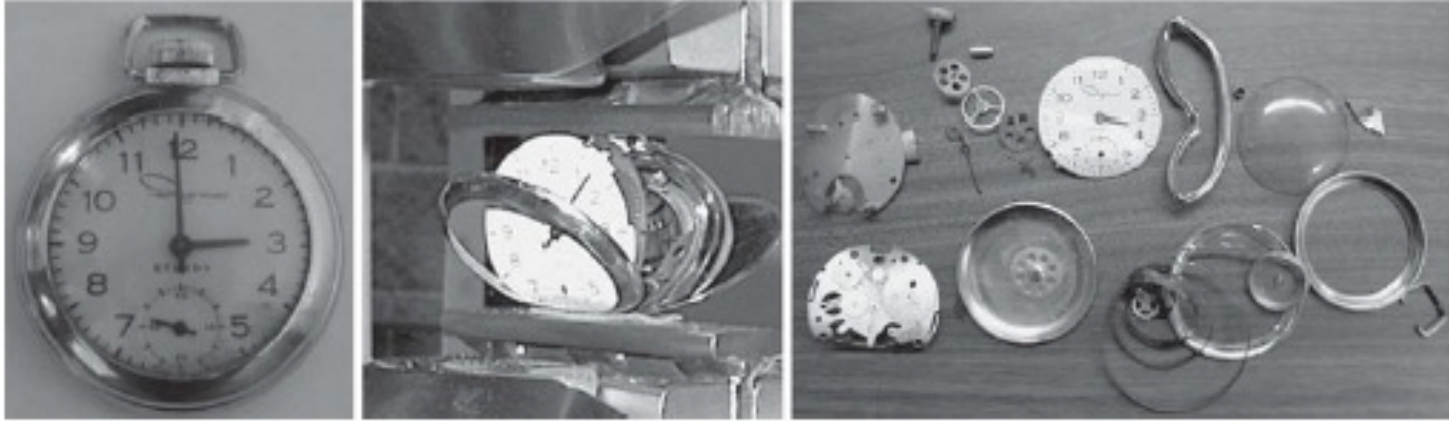
Egon Orowan

The tensile test is very easily and quickly performed, but it is not possible to do much with its results, because one does not know what they really mean. They are the outcome of a number of very complicated physical processes taking place during the extension of the specimen. The extension of a piece of metal is, in a sense, more complicated than the working of a pocket watch, and to hope to derive information about its mechanism from two or three data derived from measurements during the tensile test is perhaps as optimistic as would be an attempt to learn about the working of a pocket watch by determining its compressive strength.
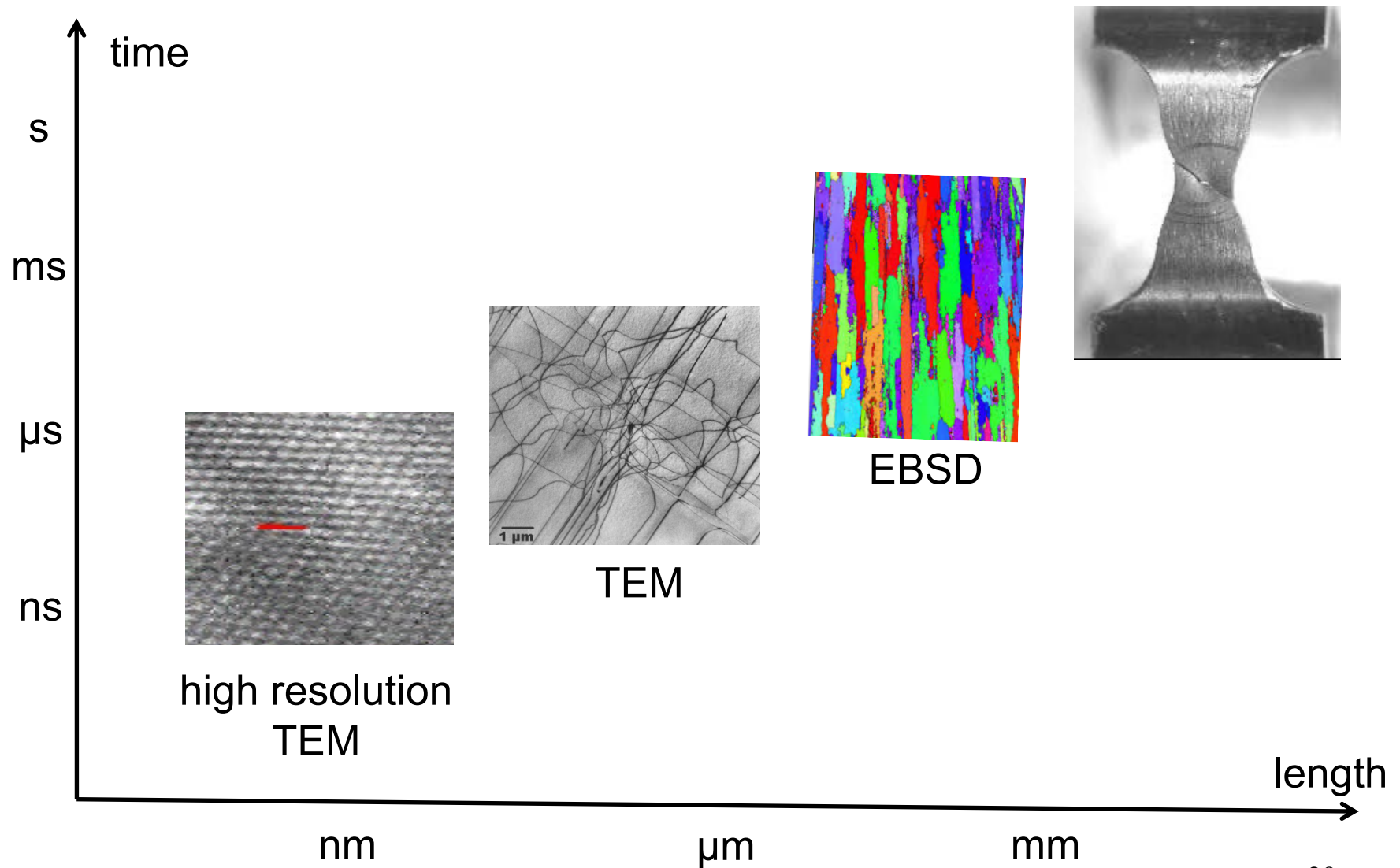

Egon Orowan

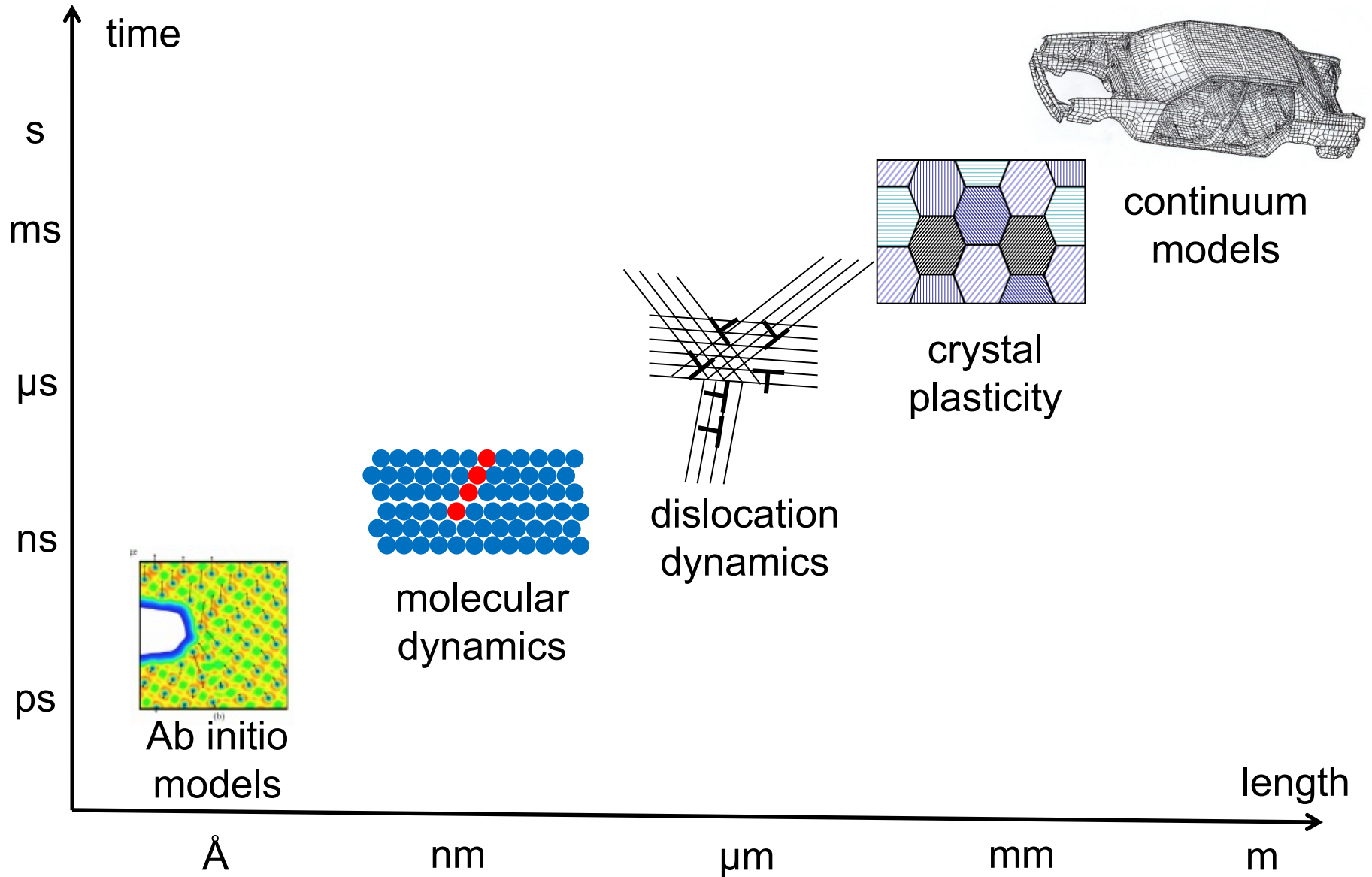Born 1902 in Budapest
Died 1989 Cambridge

Orowan's pocket watch
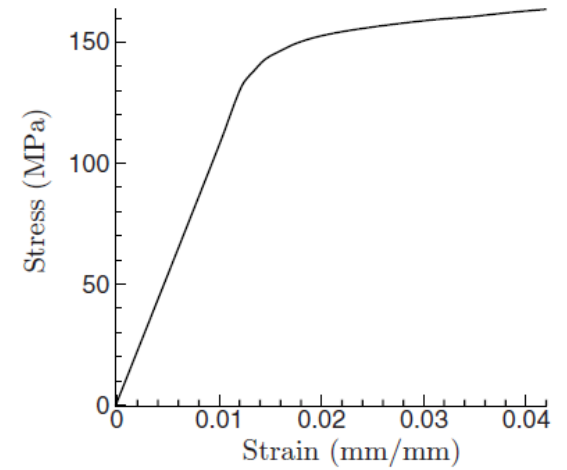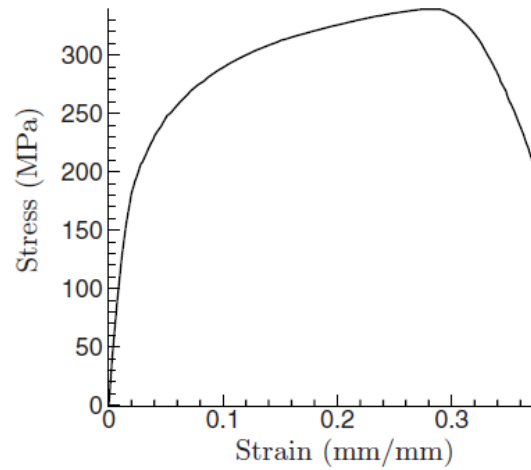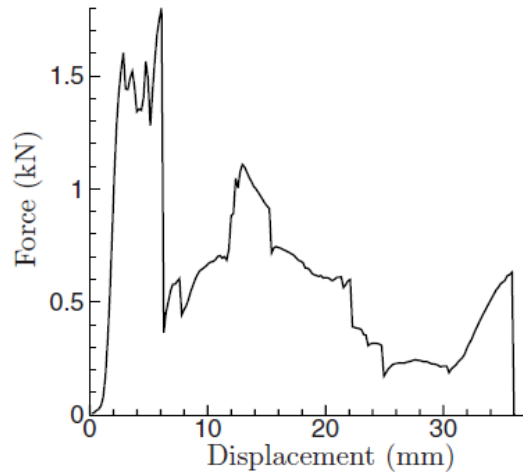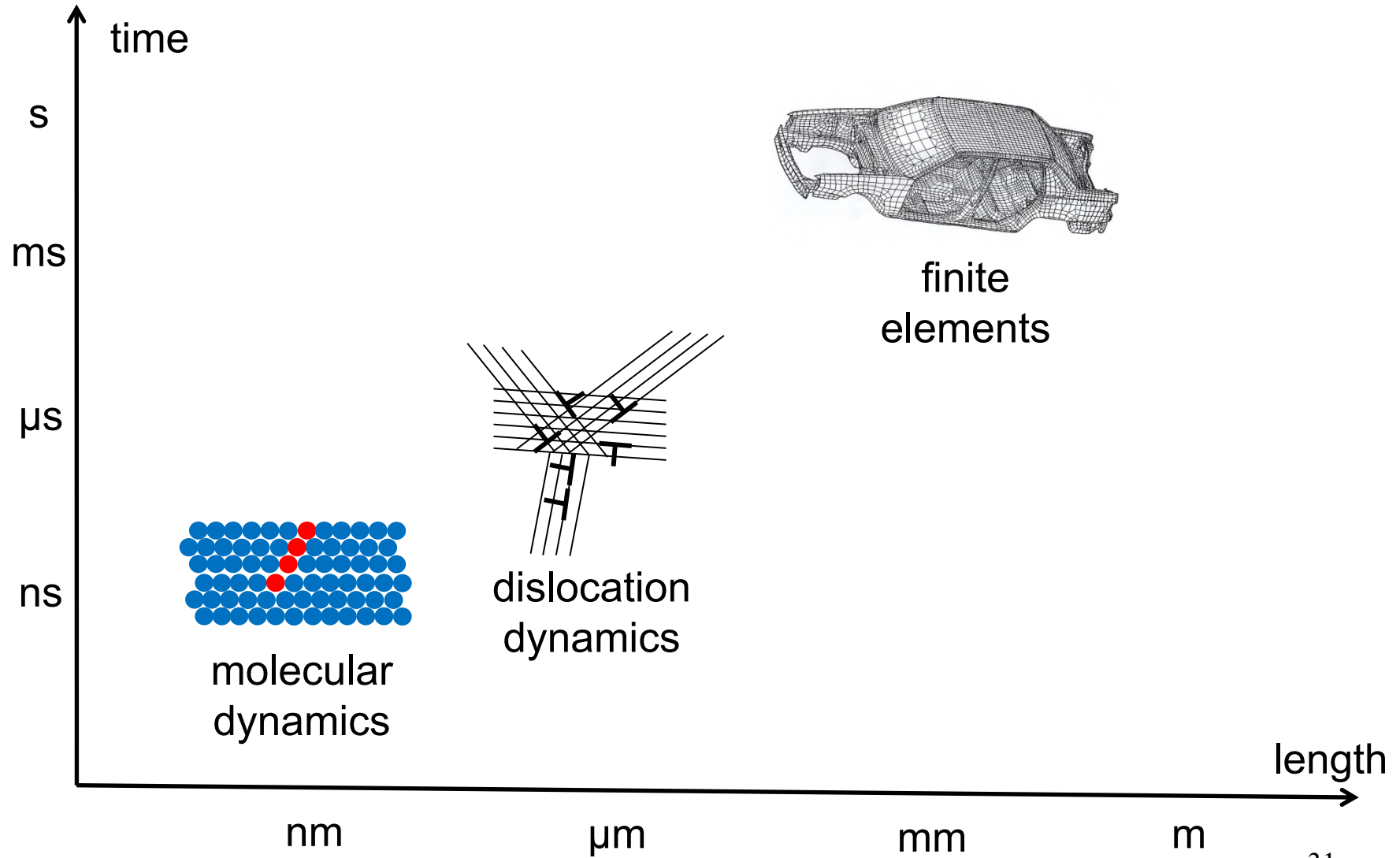
Experiments and models
at different scale

time

s

ms

μs

ns

high resolution
TEM

TEM

EBSD

length

nm

μm

mm

29

models at different scales

time

s

ms

μs

ns

ps

continuum
models

crystal
plasticity

dislocation
dynamics

molecular
dynamics

Ab initio
models

length

Å   nm   μm   mm   m

time

s

ms

μs

ns

molecular
dynamics

dislocation
dynamics

finite
elements

length

nm   μm   mm   m

Quantum DFT — Atomistic MD — Coarse Grained DPD — Continuum FEM

pm — nm — μm — mm

Length Scale
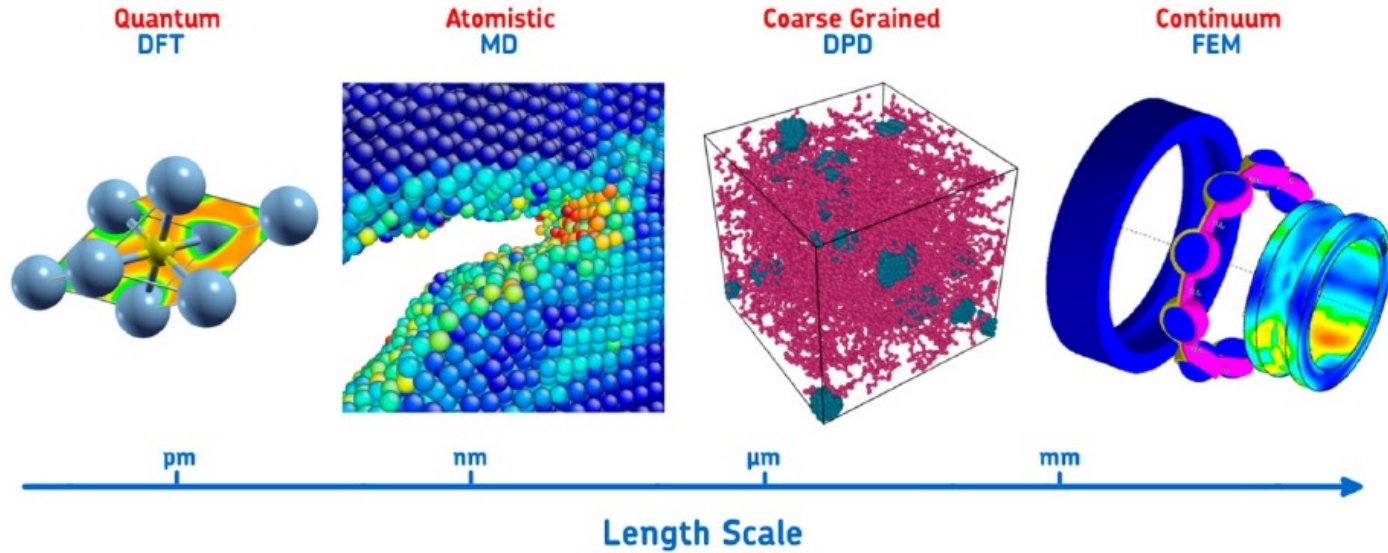
BEARING RESEARCH: GOING TO THE ATOMIC SCALE

TECHNOLOGY This article provides insight into the four main lines of research through modeling at SKF: bearing steel, fatigue mechanisms, polymer design and tribology.

AUTHOR:
SEBASTIÁN ECHEVERRI RESTREPO,
RESEARCHER, SKF ENGINEERING & RESEARCH CENTRE (ERC), SKF B.V., NIEUWEGEIN, THE NETHERLANDS

Tuesday 29 March 2016

Ab initio simulations

Density functional theory is a quantum mechanics modeling technique to study materials accounting for their electronic structure.

It is the only method that does not rely on fitting parameters.

The positions of the electrons are obviously not calculated exactly, instead one uses functionals of the electron density.

Allows the design of new materials without empirical parameters

Applications:

Bi on Pt (111)

• Effect of dopants in phase transformations
• Electrocatalyst surfaces, as Pt (111)
• Mechanical properties of new compounds (E, G)
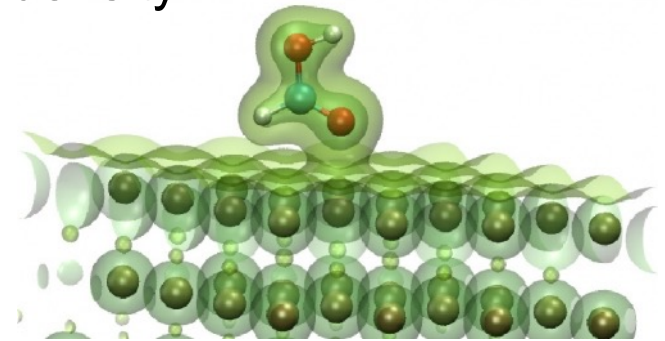
Why DFT?

Charting the complete elastic properties of inorganic crystalline compounds

Over 1000 inorganic compounds

Maarten de Jong[1,*], Wei Chen[2,*], Thomas Angsten[1], Anubhav Jain[2], Randy Notestine[3], Anthony Gamst[3], Marcel Sluiter[4], Chaitanya Krishna Ande[5], Sybrand van der Zwaag[6], Jose J. Plata[7], Cormac Toher[7], Stefano Curtarolo[8], Gerbrand Ceder[9], Kristin A. Persson[2] & Mark Asta[1]
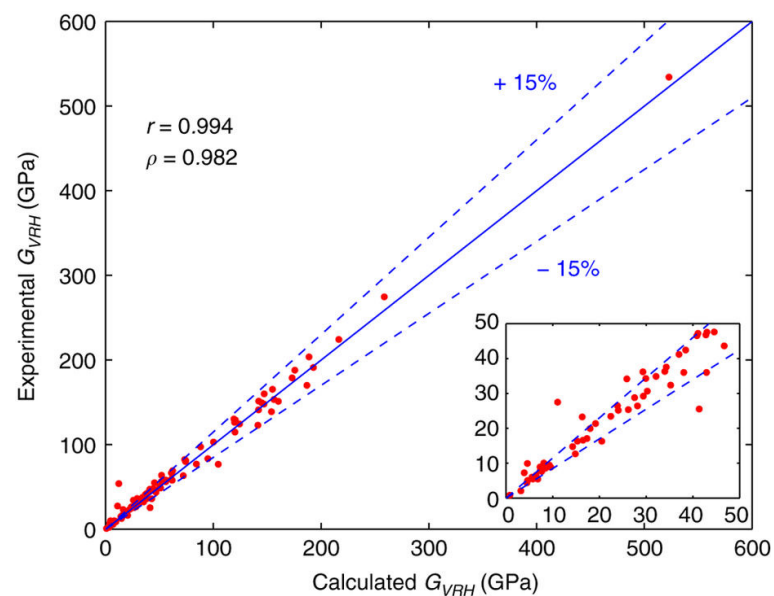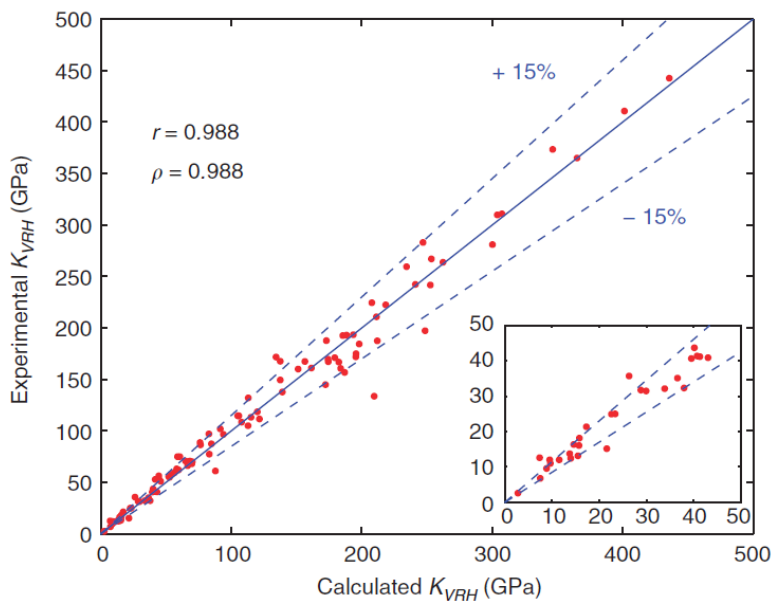
Pros and cons of DFT

PROS:

- Very accurate description of the atoms and their interactions
- Can predict novel compounds (with predefined requisites)
- Provides accurate data to be used as input at larger scales

CONS:

- Can handle only a 1000 atoms
- Is highly time and resources consuming: simulations take months and require supercomputers
- Acquiring the skills to perform the simulations and analyzing results requires significant training
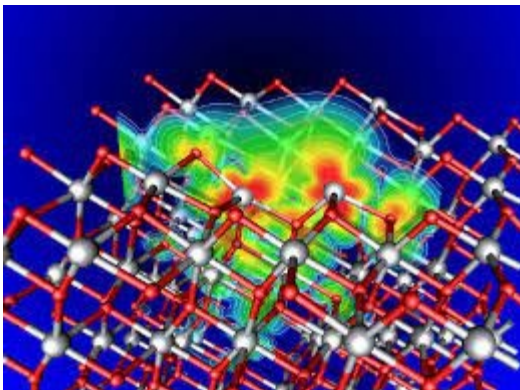
The most diffused software for
DFT simulations is called **VASP**:
the Vienna Ab-initio Simulation Package

Molecular dynamics

Molecular dynamics is a modeling technique that applies Newtonian mechanics to atoms and molecules, which move in response to the forces that they exert on each other and that they experience from the outside world.
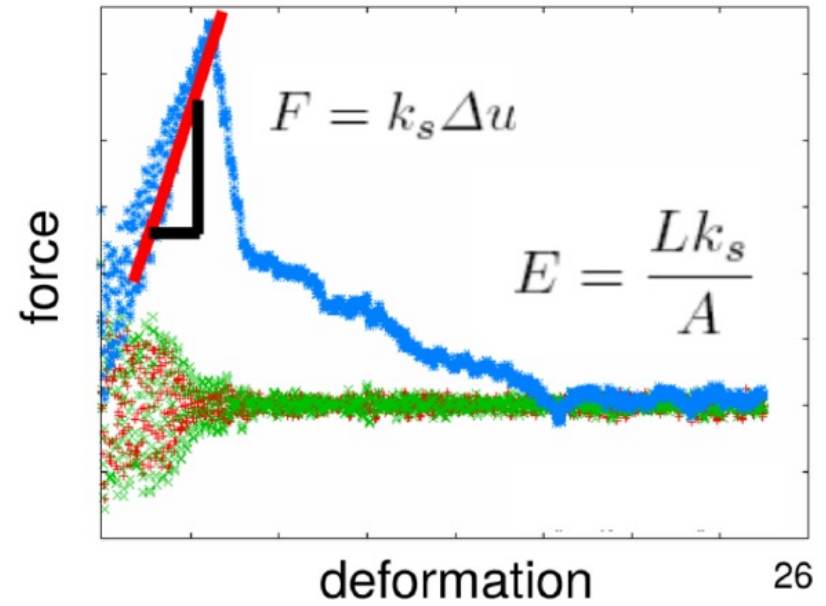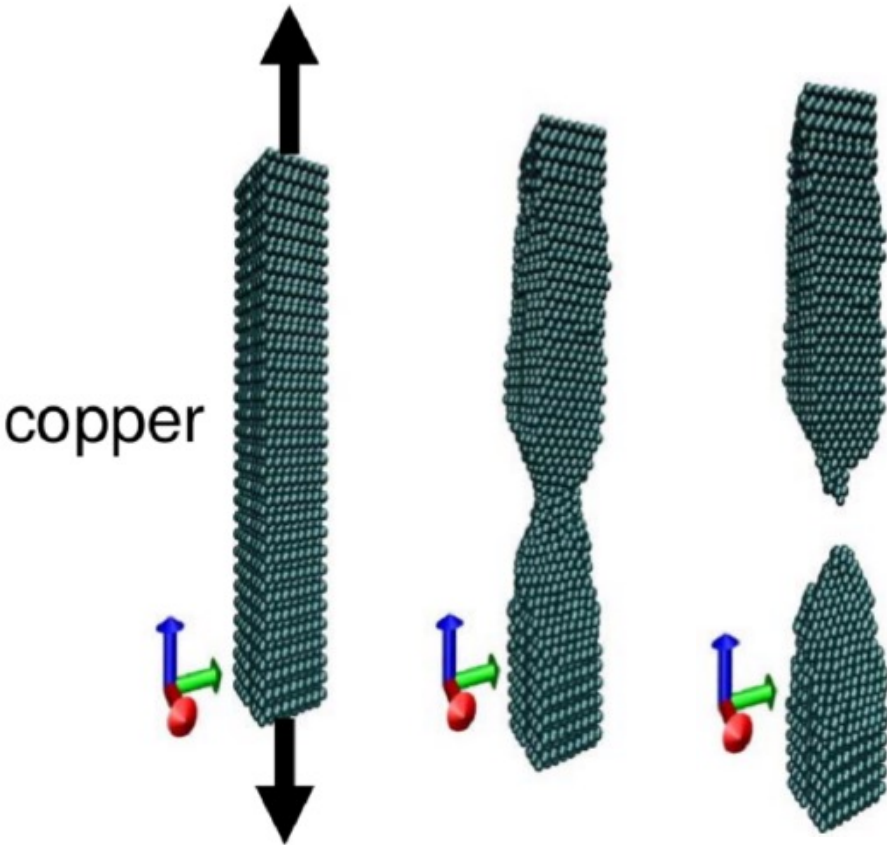
Dates back to 1957

Applications include:

- Determination of the mechanical response of nanoscale materials
- Defects and their interactions
- Deposition
- Oxidation and simple reactions
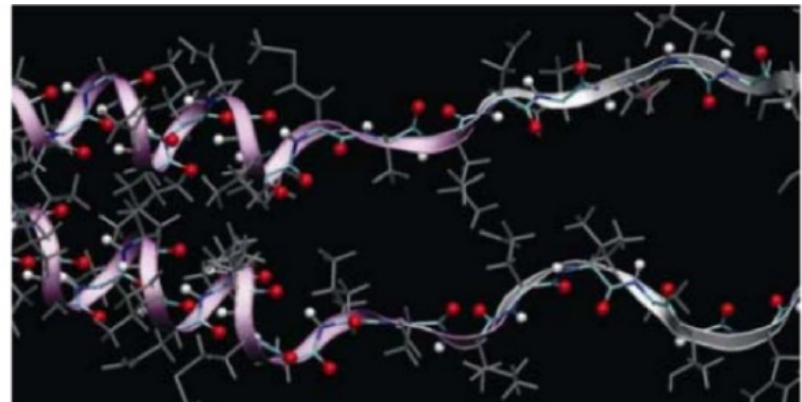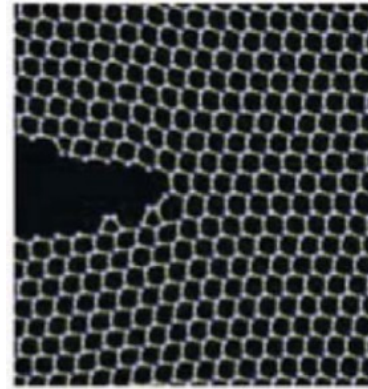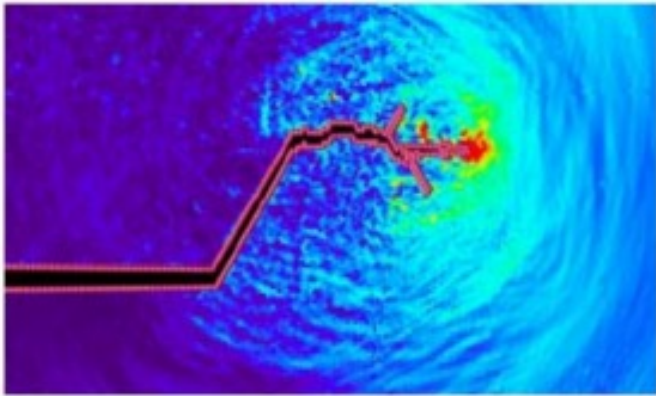- Extreme conditions (0K, radiation)

Graphene Layer

Rough Substrate
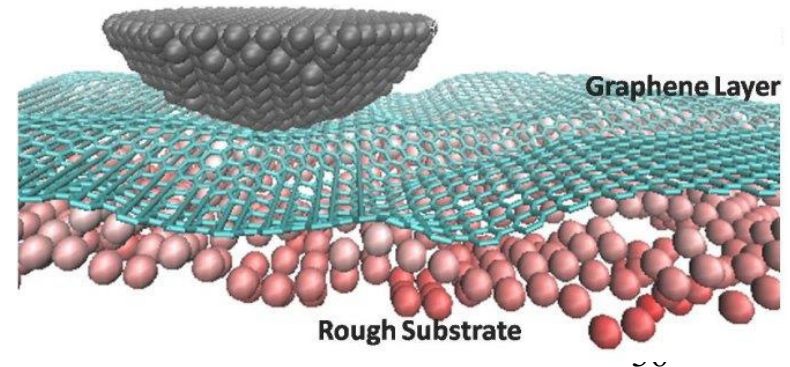
Determination of E from MD

$$F = k_s \Delta u$$

$$E = \frac{L k_s}{A}$$

copper

force

deformation

26

37

Applications of atomistic simulations

Atomistic models also enable us to predict failure, fracture, adhesion, diffusion constants, phase diagrams, protein unfolding…

Pros and cons of MD

PROS:

• Good accuracy in the description of the atoms and many of the interactions between atoms
• Appropriate to study properties of nano-scale materials
• Can model some simple reactions like oxidation (work-in-progress)

CONS:

• Interatomic potentials are not available for all compounds
• Can handle small systems (about one cubic micrometer of copper) and short time scales
• Is highly time and resources consuming: it is customary to run on a computer cluster, better with GPU card
• Acquiring the skills to perform meaningful simulations and analyzing results requires training

The most diffused software for MD simulations is **LAMMPS**: Large-scale Atomic/Molecular Massively Parallel Simulator

Dislocation dynamics

Dislocation dynamics is a modeling technique that treats plasticity as the collective motion of dislocations. The dislocations are treated as line singularities in on otherwise continuum medium

macro   micro
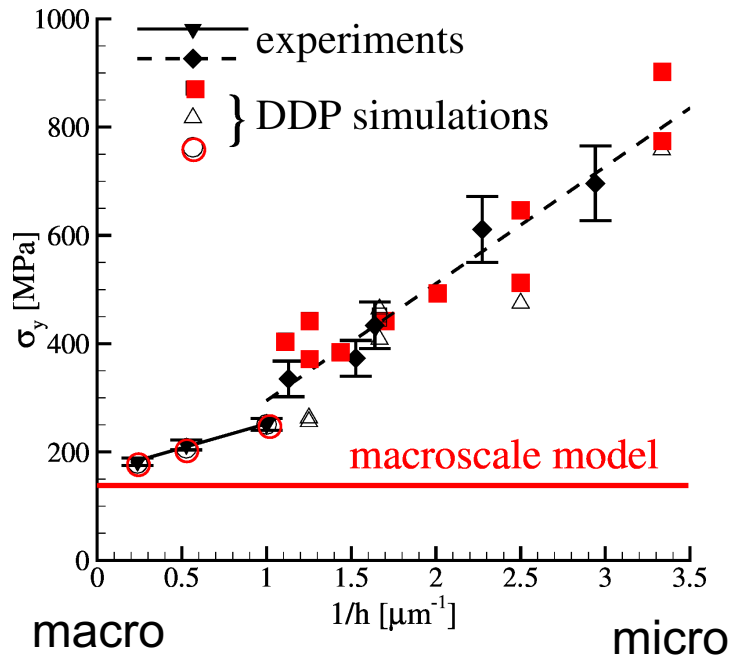
The method is used to study plastic deformation in micro-scale **metal** objects, where the discrete nature of dislocations is relevant.
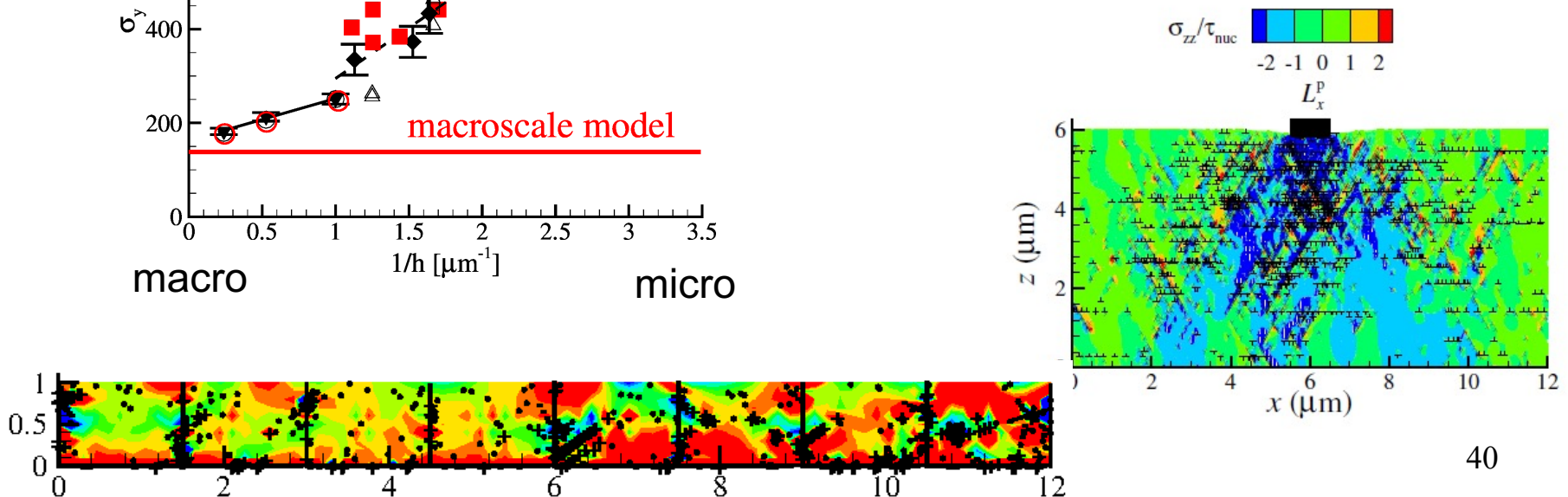
Pros and cons of DD

PROS:

- Good accuracy in the calculation of plastic deformation in single and polycrystals
- Appropriate for metals

CONS:

- Can handle a size of about 100 cubic micrometers
- Is time and resources consuming, although it can be run on a single workstation or PC
- Acquiring the skills to perform the simulations and analyzing results requires training

The most diffused software for DD simulations is PARADIS

Crystal plasticity

Crystal plasticity is a modeling technique that accounts for plastic slip on crystallographic slip systems. Can tackle single crystals as well as polycrystals. It is routinely used to predict metal forming.

7.8°, 30% reduction

15°, 30% reduction

30.7°, 30% reduction
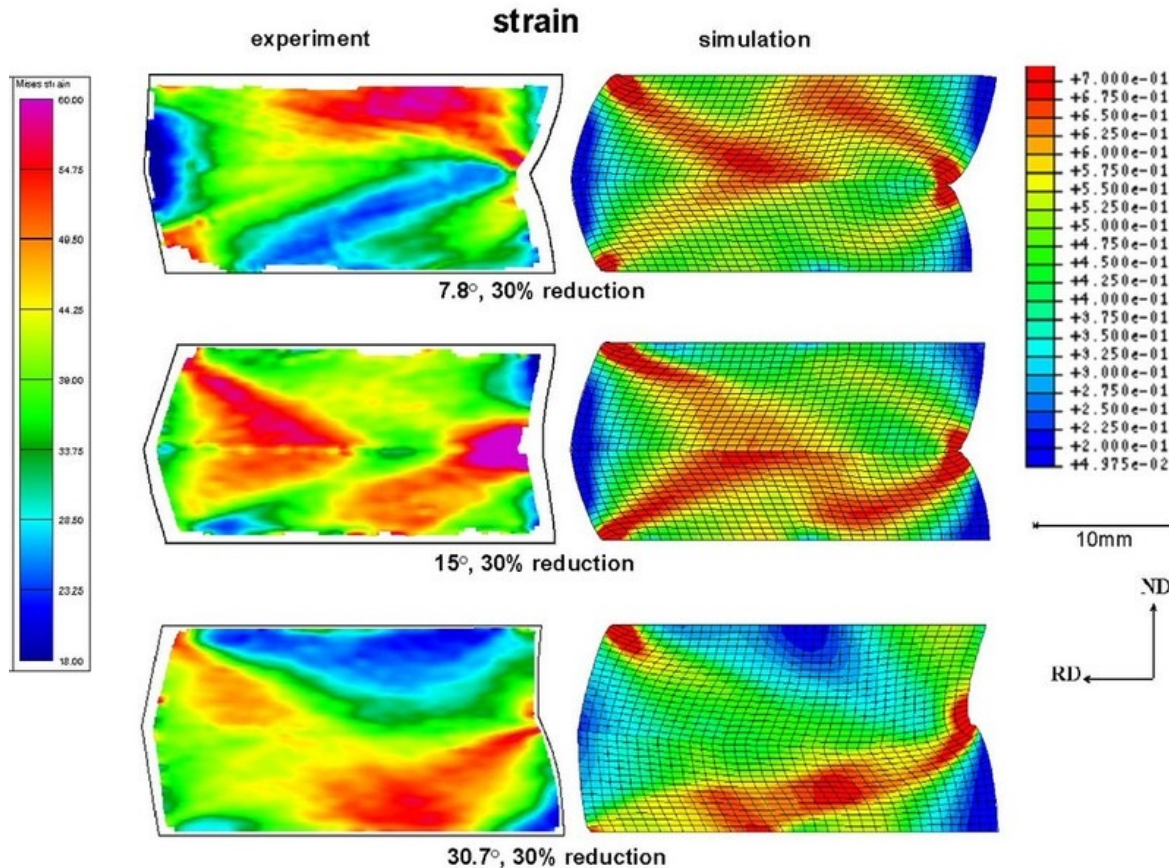
Pros and cons of CP

PROS:

- Can capture grain orientation and therefore is good for modeling the mechanical behavior of anisotropic single crystals or textured policrystalline materials
- Computationally efficient
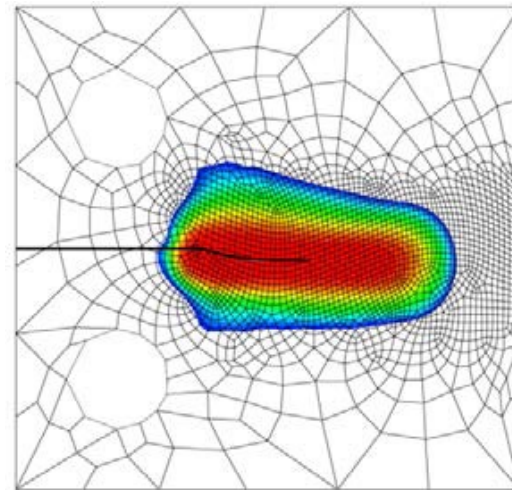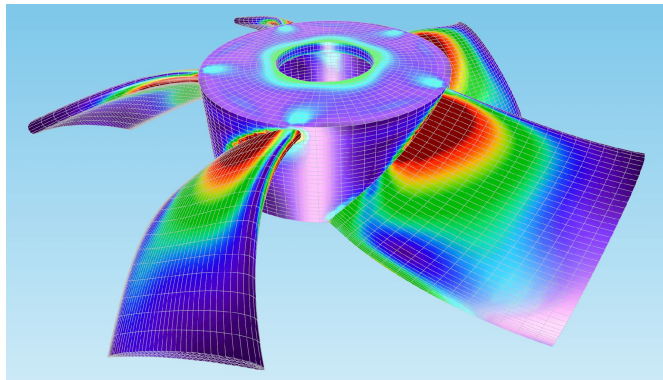- It is used in large companies

CONS:

- Much information used as input is phenomenological

The most diffused commercial package for CP simulations is ABAQUS with some dedicate umat subroutines

The finite element method

The finite element method is a numerical technique used in the field of continuum mechanics to solve boundary value problems on almost any shaped region.

In continuum mechanics it is mostly used to calculate displacements stresses and strains in macro-scale bodies subject to mechanical and/or thermal loads.

Application of the finite element method

ω ϵ_{xx} ϵ_{xx}^{p} lab
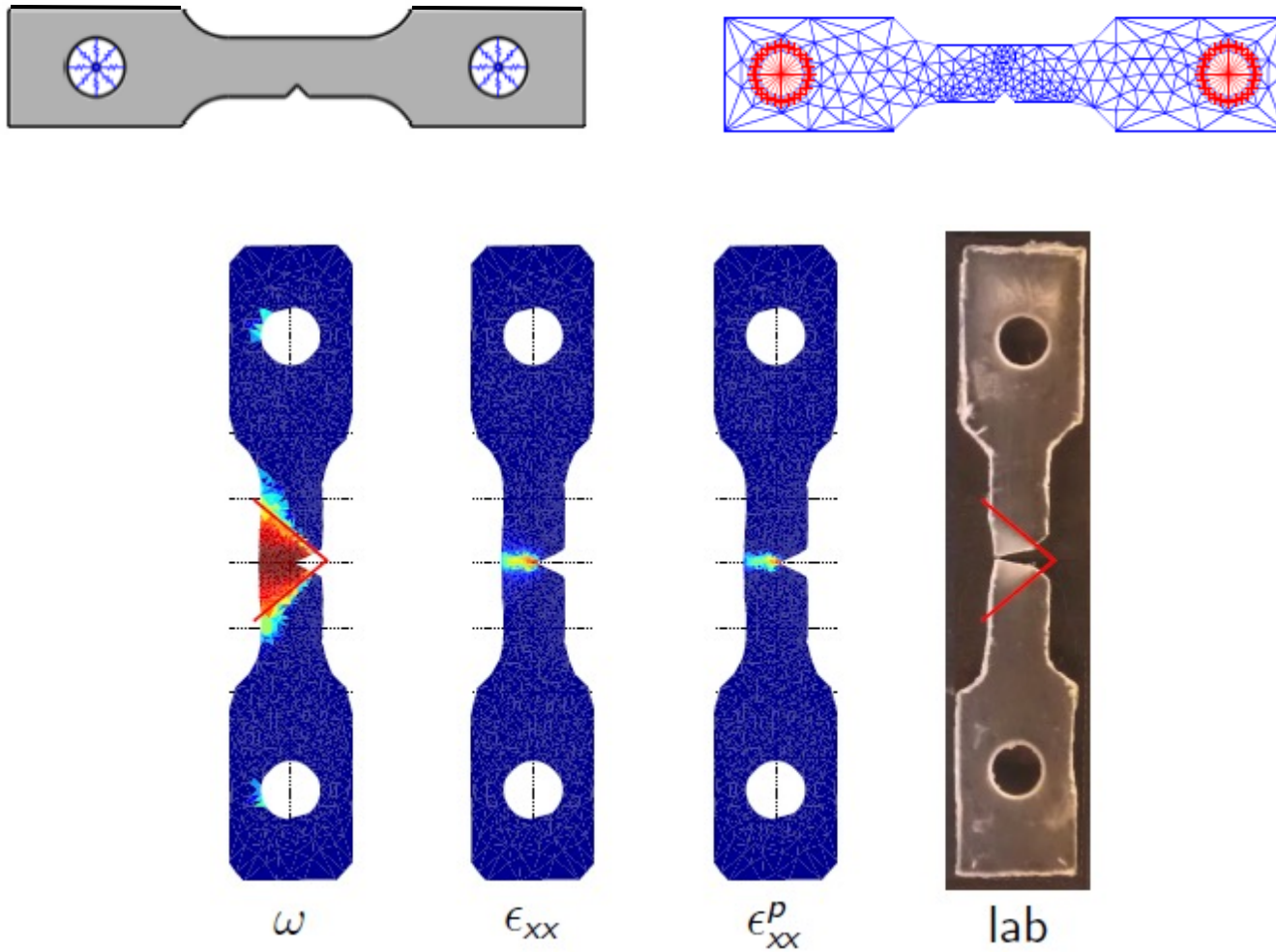
Pros and cons of FEM

PROS:

• Can solve a boundary value problem on domains with practically any shape
• Computationally very efficient
• Very widely used, in small and large companies, there are implementations of a large variety of material behaviors
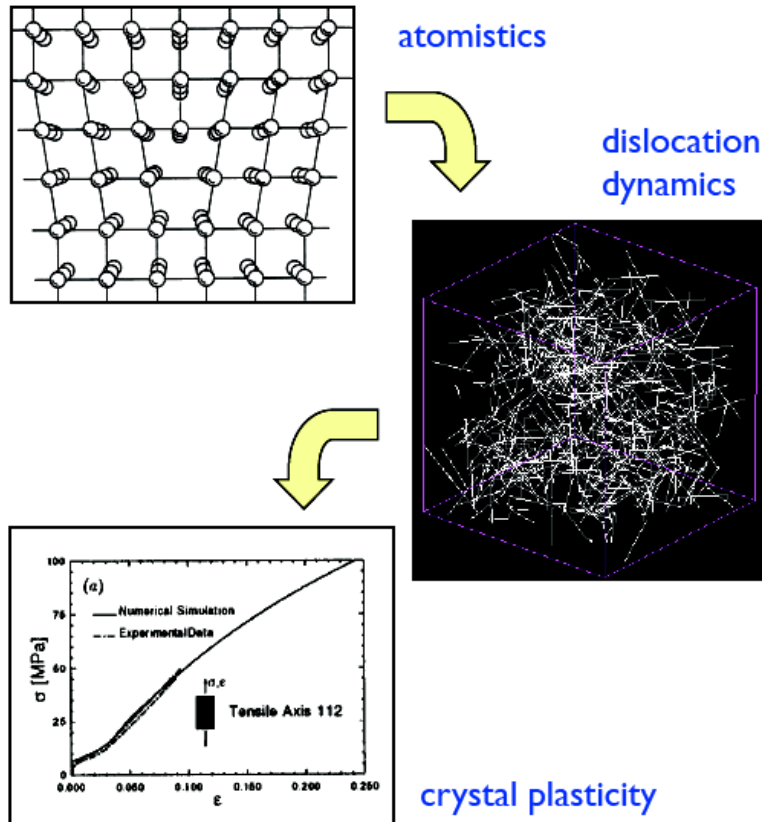
CONS:

• Material behavior is described phenomenologically (enters through elastic constants, yield strength and hardening laws)
• Interactions between interfaces (different bodies, cracks) are at best educated guesses

The most flexible and reliable commercial packages are ABAQUS and MSC-Marc, many use ANSYS
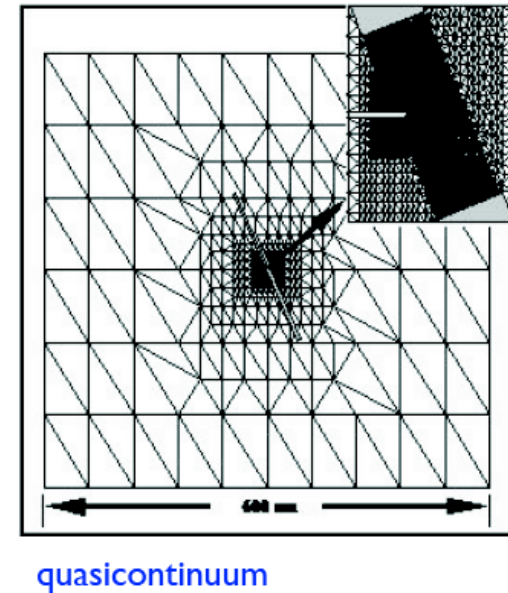
Multiscale modeling

Complex materials problems where the governing phenomena span different scales of materials behavior requiring multiscale approaches
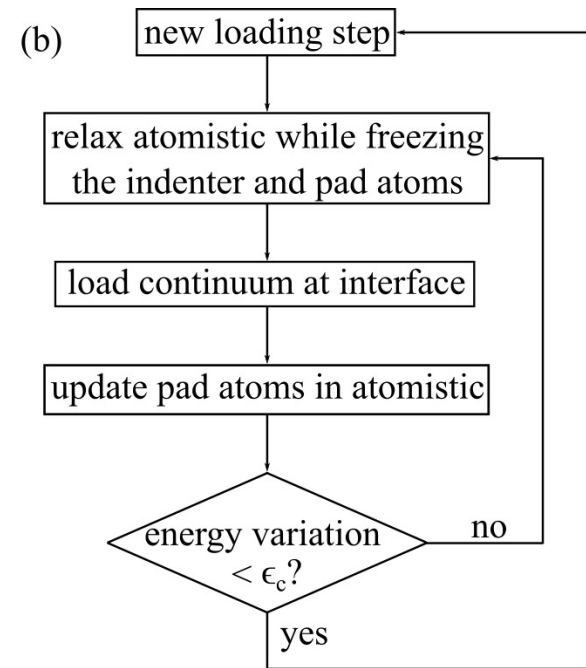
(a) indenter

H_A

atomistic domain
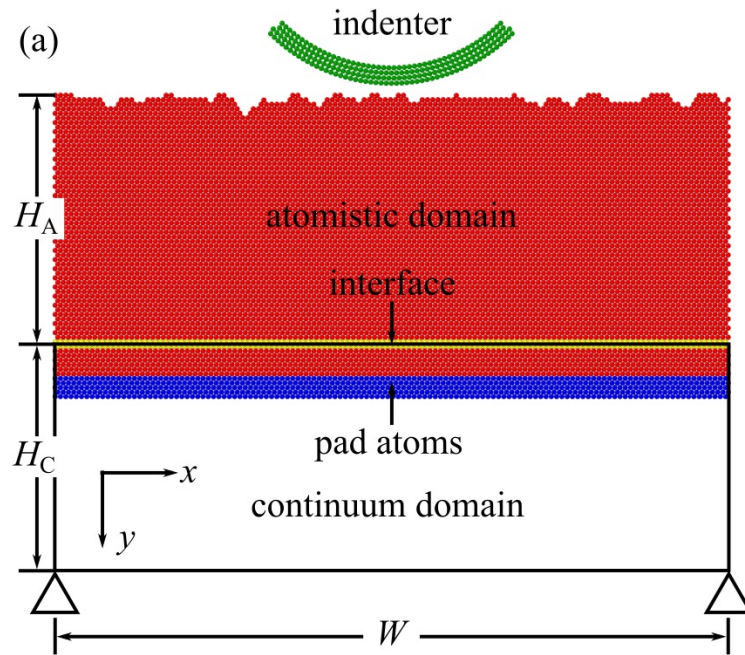
interface

pad atoms

H_C

x

y

continuum domain

W

(b) new loading step

relax atomistic while freezing the indenter and pad atoms

load continuum at interface

update pad atoms in atomistic

energy variation $< \epsilon_c$?

no

yes

Continuum models

Continuum description of the solid

Constitutive law: relationship that predicts the deformation to an applied load

The constitutive law can be determined through experiments or even guessed intuitively, it is in any case phenomenological.

The phenomenological approach is based on fitting
and can therefore not be used to predict new behavior or design new materials.

Materials modeling aims at developing constitutive laws, descriptive of material behavior, informed by observations at various scales.

Discrete models

One or more of the ingredients is discrete

Density functional theory: atoms + a functional that describes electrons

Molecular dynamics: atoms

Dislocation dynamics: slip planes, dislocations, dislocation sources, obstacles

Crystal plasticity: slip systems, crystals

What is the appropriate model?

- Deformation of a turbine blade during operation

- The core energy of a dislocation

- Dislocation nucleation

- Nanoindentation of a Cu specimen

- Surface roughening during scratching

Different approach to modeling

1) Select a modeling technique and run (perform) simulations --- similar to performing experiments with an existing set-up

2) Develop own model or modify existing model and run simulations ---similar to building own set-up and performing experiments

Health warning: in both cases it is very wise to know what you are doing ;-)
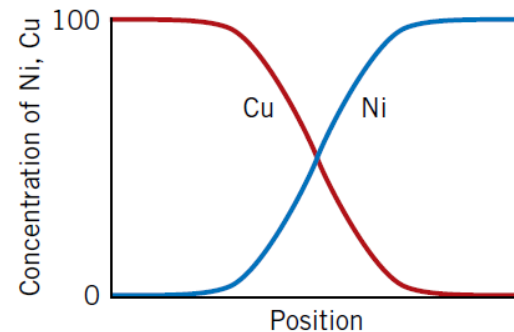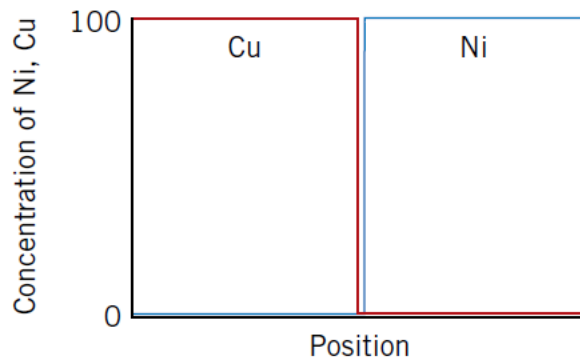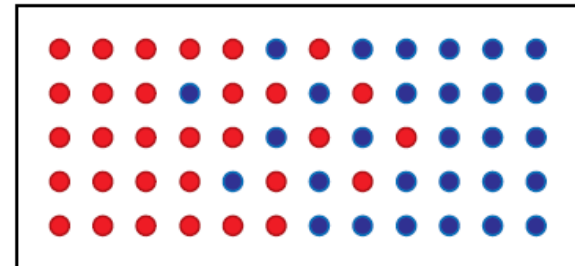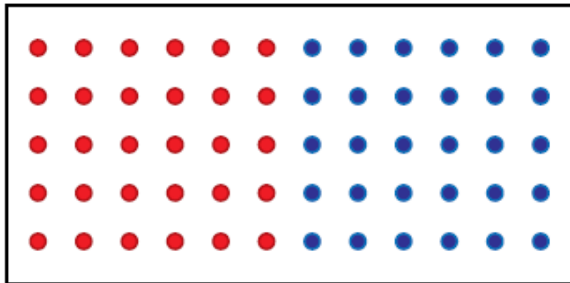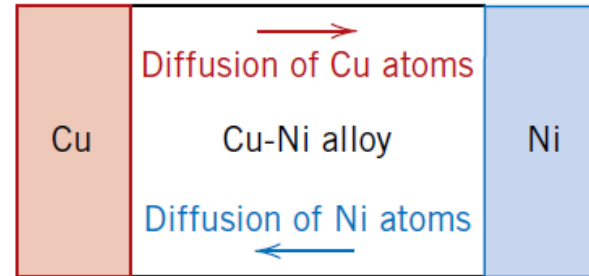
The Random Walk Model
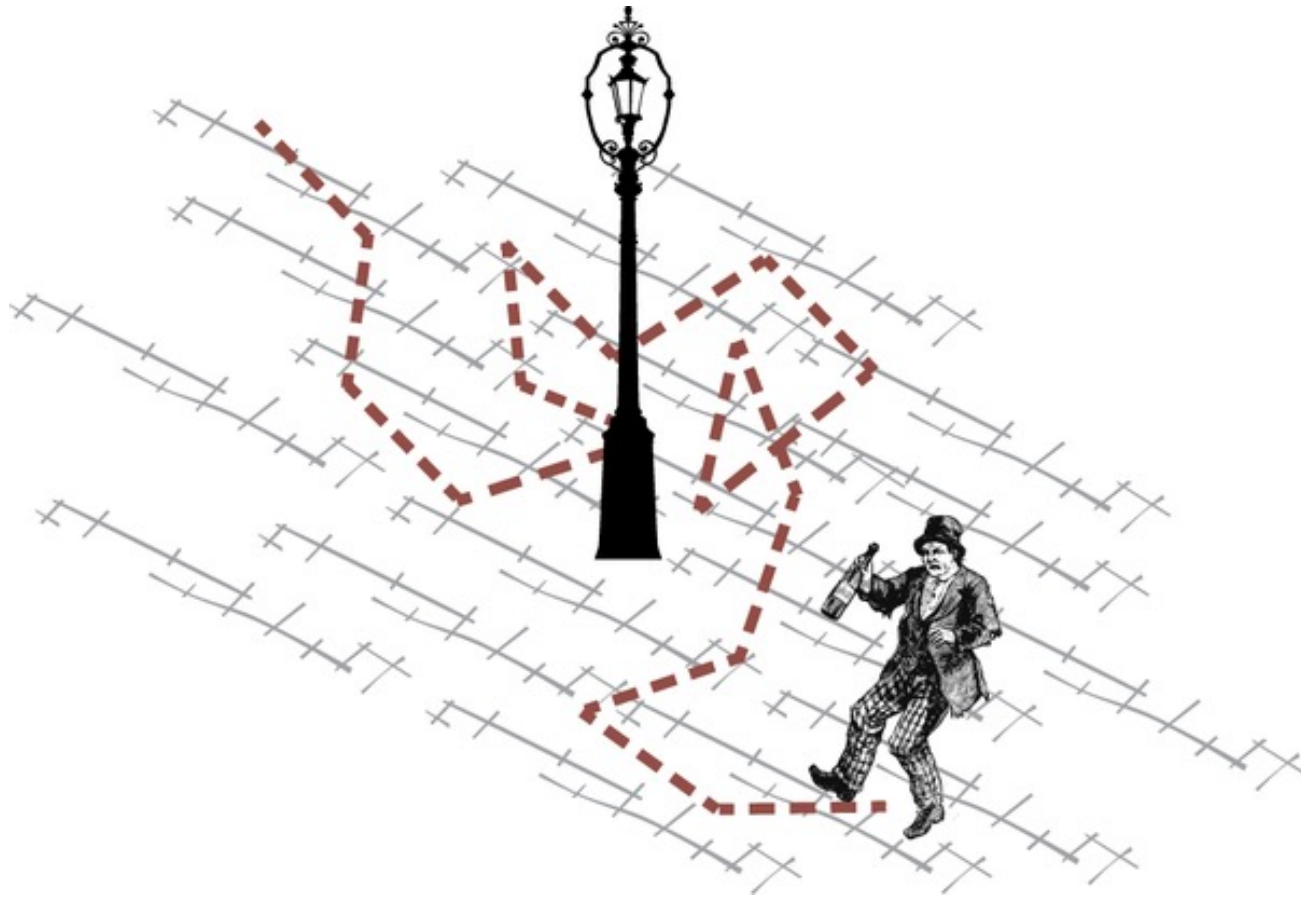
…otherwise known as the drunkard's model

The random walk model

Can you think of a process in material science that involves `almost' random motion?

# The copper-nickel diffusion couple

Diffusion

The phenomenon of material transport by atomic motion from high to low concentration.

For atoms to migrate two conditions must be met:
1) presence of a vacancy
2) atoms must have sufficient (vibrational) energy to break the bonds and cause lattice distortion while moving
The fraction of atoms that can migrate increases with temperature
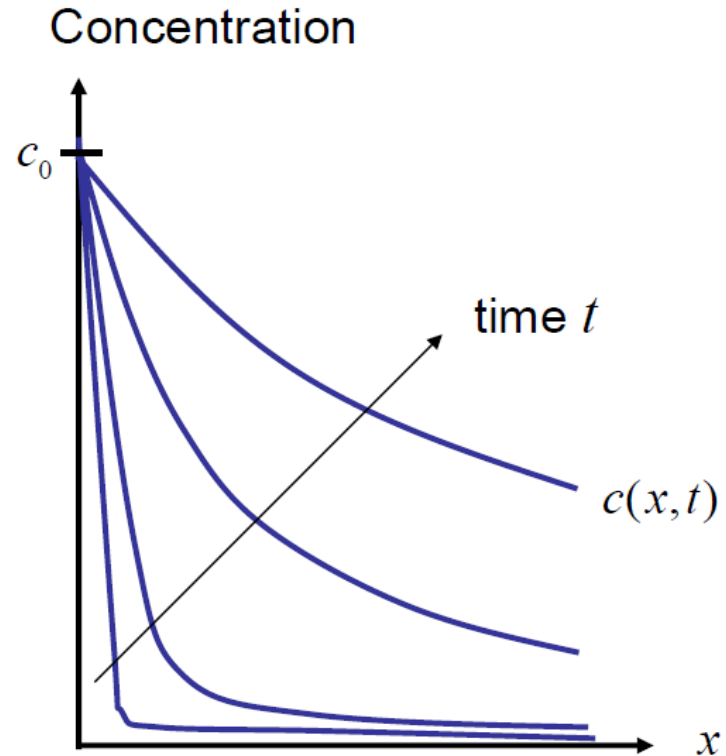
With time
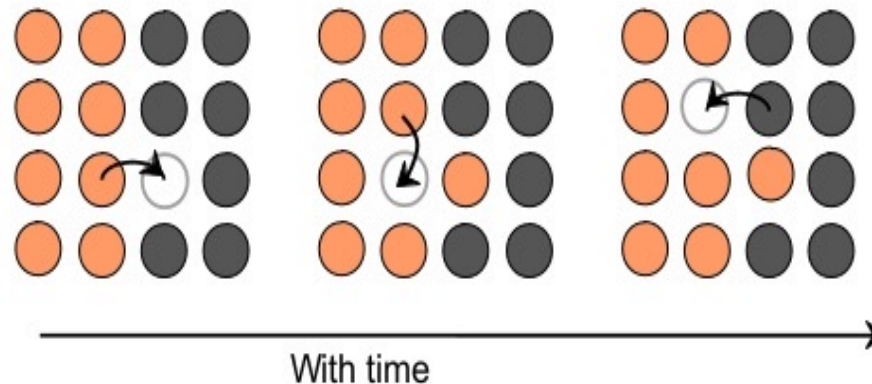
Continuum model of diffusion

Fick's law: $\dfrac{\partial c}{\partial t} = D \dfrac{d^2 c}{dx^2}$

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

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

Concentration

time t

$c(x, t)$

Need diffusion coefficient to solve for distribution!

How do we get D?

Top down approach

Lab experiment

Measure concentration
distribution and fit curves

Use to solve more
complex geometries
or situations for which
no lab measurements
are possible

Concentration

c_0

time t_0

experiment

continuum model solution

$c(x, t_0)$

x
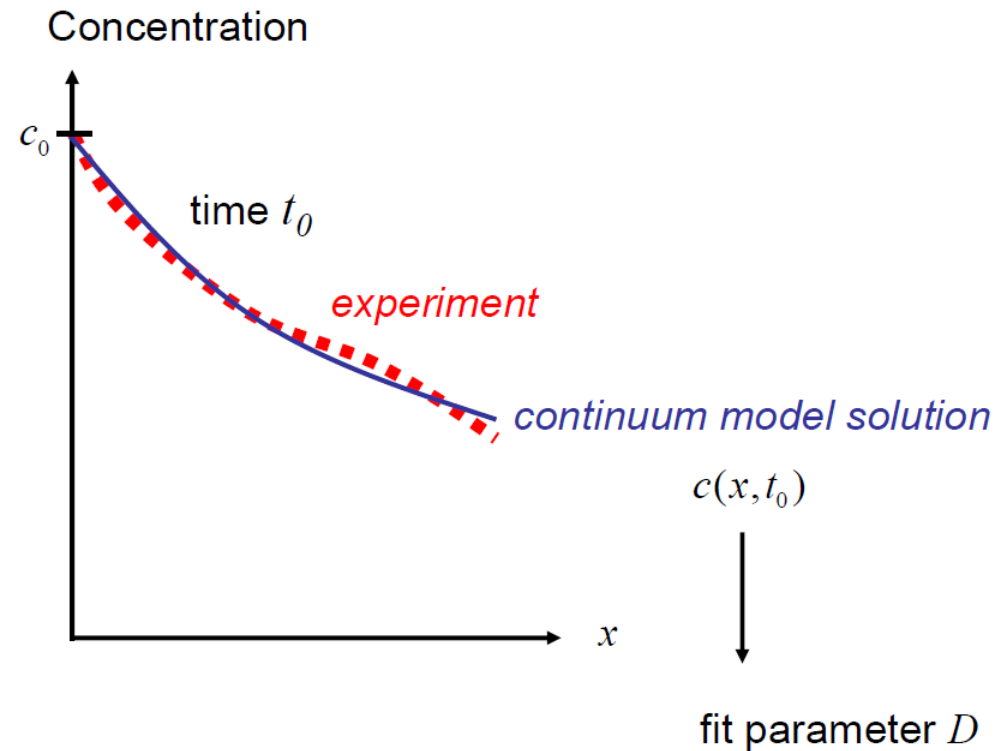
fit parameter D

Continuum model

- **Continuum model requires parameter that describes microscopic processes inside the material**

- Typically need experimental measurements to calibrate
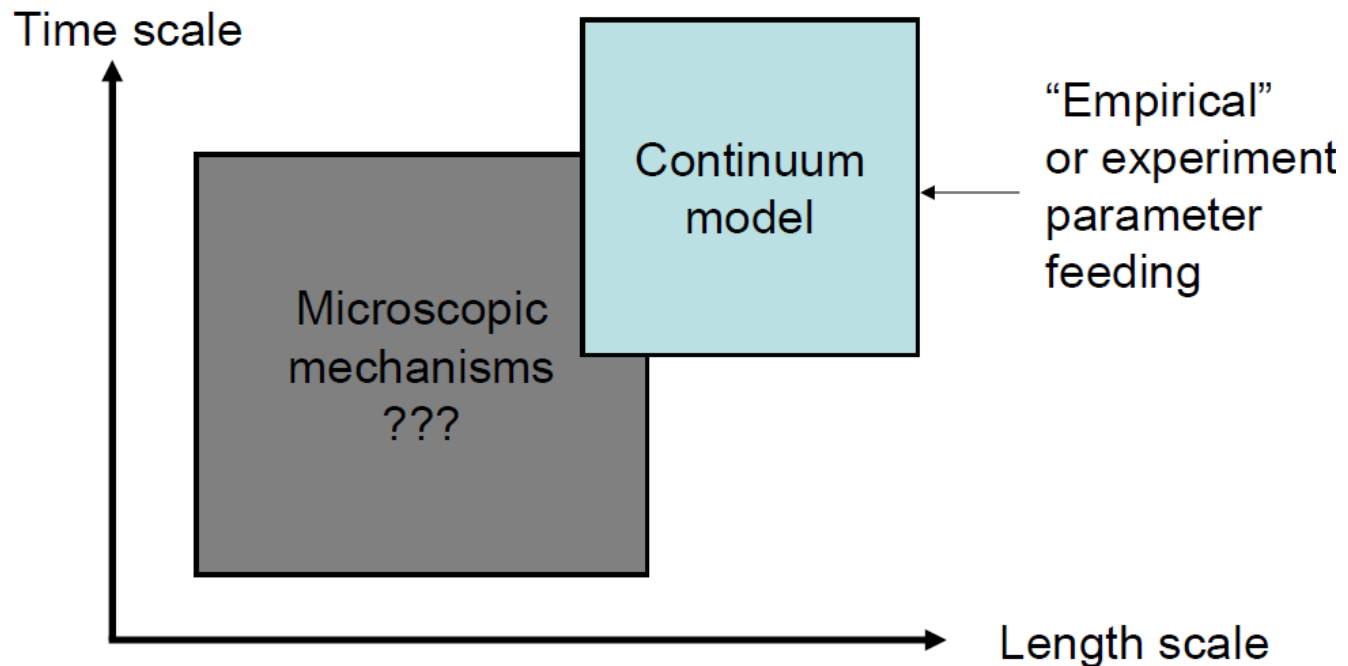
Atomistic model of diffusion

Directly calculate the diffusion coefficient by tracking the trajectories of the atoms through MD
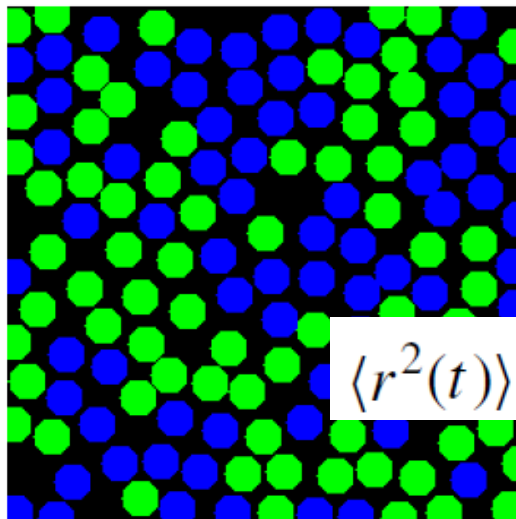
Particles

Mean square displacement

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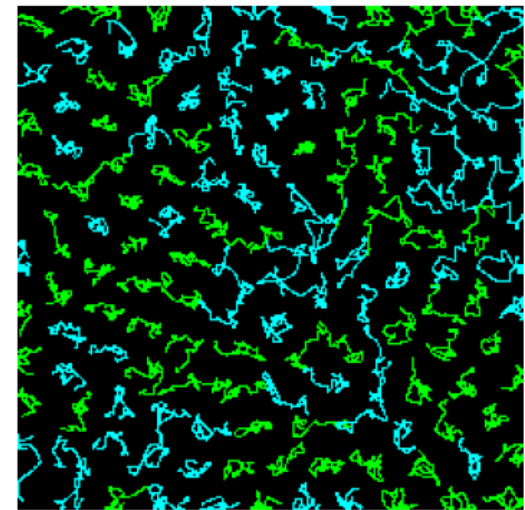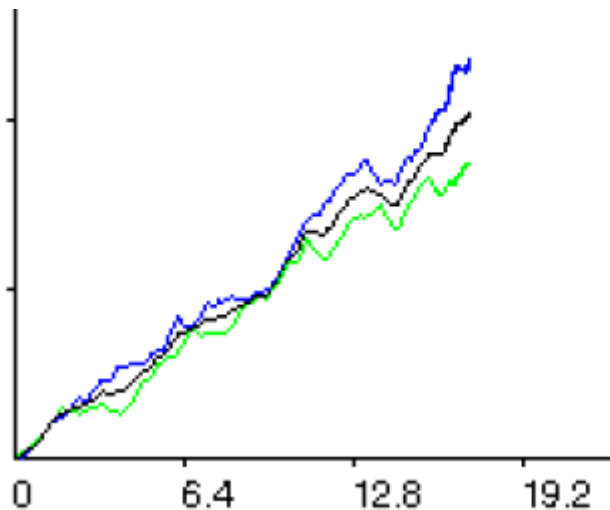
Trajectories

time

Link between continuum and discrete models

The diffusion coefficient relates to the ability of the particle to move a `distance' Δx^2 from left to right over a time Δt

$$D = p\frac{\Delta x^2}{\Delta t}$$
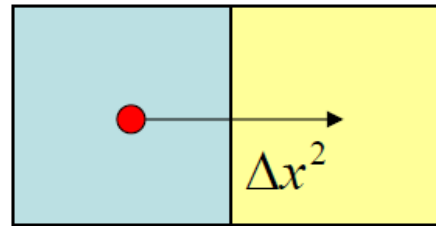
If there is no directionality, so that p=1/2

$$D = p\frac{\Delta x^2}{\Delta t} \longrightarrow D = \frac{1}{2}\frac{\Delta r^2}{\Delta t}$$

For higher dimensions:

$$D = \frac{1}{2}\frac{1}{d}\frac{\Delta r^2}{\Delta t}$$

3D, large t:

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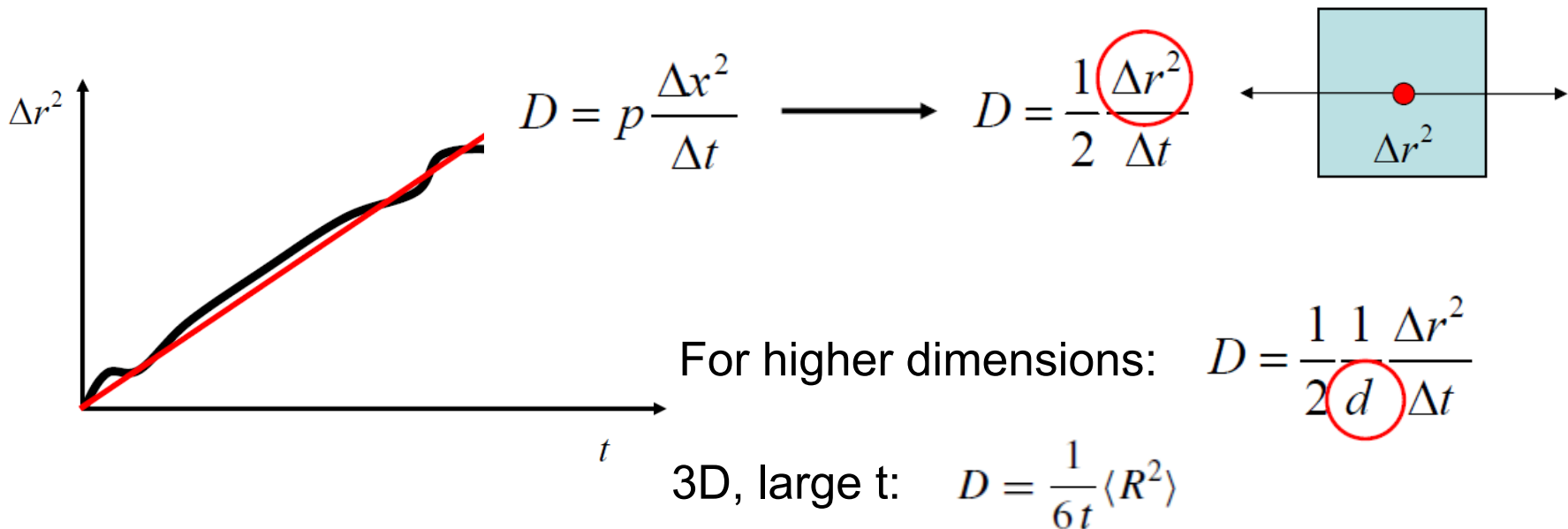
Take home messages

- A model is a (mathematical) simplification of reality

- In engineering, models are mathematical

- Only very simple models can be solved analytically, but TG there are computers that solve numerically lengthy complicated math for us

- There are different simulation techniques to address different times and length scales

- The modeling techniques can be either discrete or based on the continuum description of materials and phenomena

- Diffusion can be modeled as a random walk

In class assignment 1

Let's model diffusion in a simpler way than using atomistic simulations:

Write a computer programme to model diffusion in the bulk of a simple cubic crystal.

Making assumptions

Let's simplify things as much as possible (not more;-)). One can always complicate things later on.

What do we know about diffusion?

All atoms in the crystal vibrate at a given temperature, at 0K there is no diffusion.

We know that an atom will hop into a vacancy if it manages to overcome a certain energy barrier while vibrating.

Diffusion
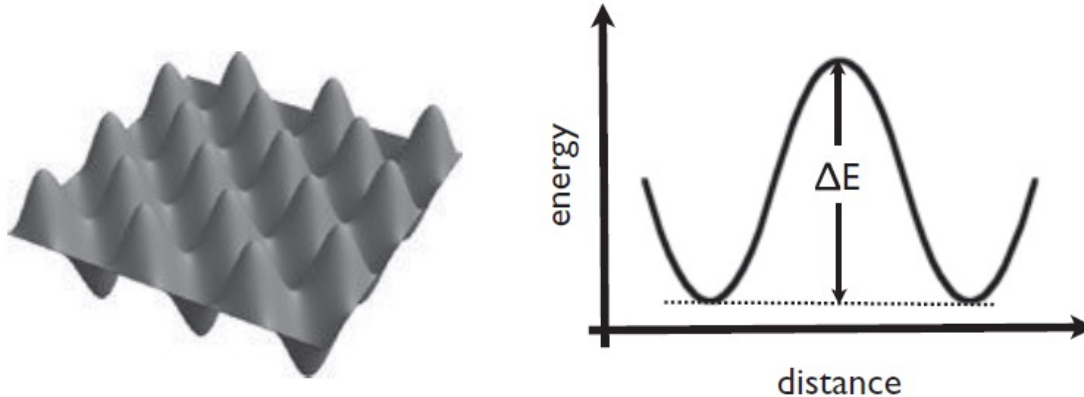
$$D \sim e^{\Delta E / k_B T}$$

1) The time required for a jump to occur is significantly longer than the time required for a vibration of the atom

2) The energy between a site and its nearest neighbor is much lower than between a site and its next-nearest neighbor

Model ignores atomic-level detail and focuses only on the jumps

distance traveled:
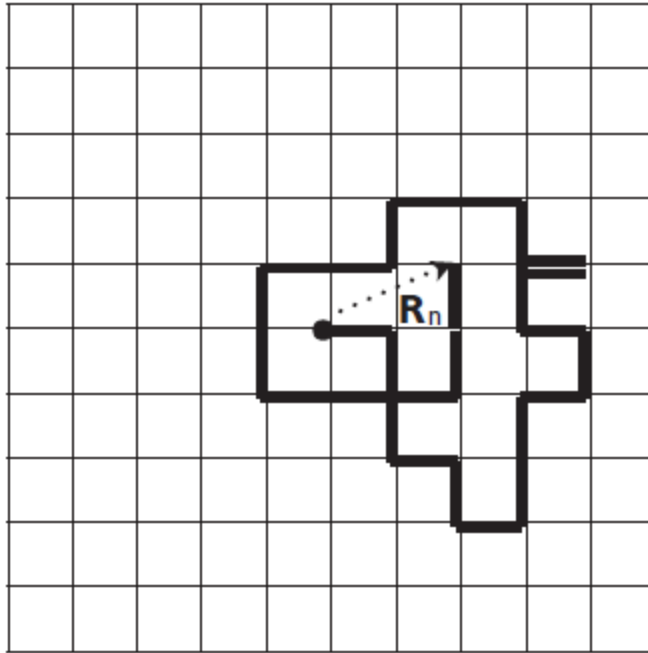
$$R_n = \sqrt{\mathbf{R}_n \cdot \mathbf{R}_n}$$

Nearest neighbor distance is *a*
Moving 27 steps *n=27*

The code

Part 1

Write a small program in MATLAB, called rwn.m, to:

1) track the single trajectory of the random walk of a vacancy on a simple square lattice.

2) calculate the square displacement of the trajectory.

Start with the simpler 2D problem (diffusion on the surface) and then extend it to a 3D problem (diffusion in the bulk).

For students who attended MF→ Consider an FCC crystal instead of a simple cubic lattice. Next consider the motion of an interstitial carbon atom in the FCC crystal.

Main steps

1. Assign an initial position to the vacancy on the square lattice

2. Move the vacancy to a new position (store location)

3. Calculate the traveled distance and repeat until final step

Input, output, function

1) Identify output and input

OUTPUT: *x,y* coordinates of the vacancy at each step, square displacement *rs2*

INPUT: number of jumps *nt*

2) Define a function in matlab which links input and output:

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

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

Remember to initialize variables!

Input, output, function

1) Identify output and input

OUTPUT: *x,y* coordinates of the vacancy at each step,
square displacement *rs2*

INPUT: number of jumps *nt*

2) Define a function in matlab which links input and output:

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

3) Initialization of variables

```
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
```

Move vacancy to new position

HINTS:

1) Where does it move? Define the jumps on the (square) lattice: right, up, left, down

2) How does it move? Select the jumps randomly

Jumps

1) Define the jumps on the (square) lattice: right, up, left, down

For each jump there are four possibilities

(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];
```

Select jumps

1) Define the jumps on the (square) lattice: right, up, left, down

For each jump there are four possibilities

(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) Create a list of random numbers ranging from 1 to 4,with **nt** entries

A needed ingredient: a random number generator
In MATLAB rand(nt,1) gives nt random numbers from zero to 1 excluded.

Simple random walk code in 2D: rwn.m

```matlab
% 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
```

Perform simulations

Run the code for nt=100 and for nt=500 and plot:

1) the trajectory
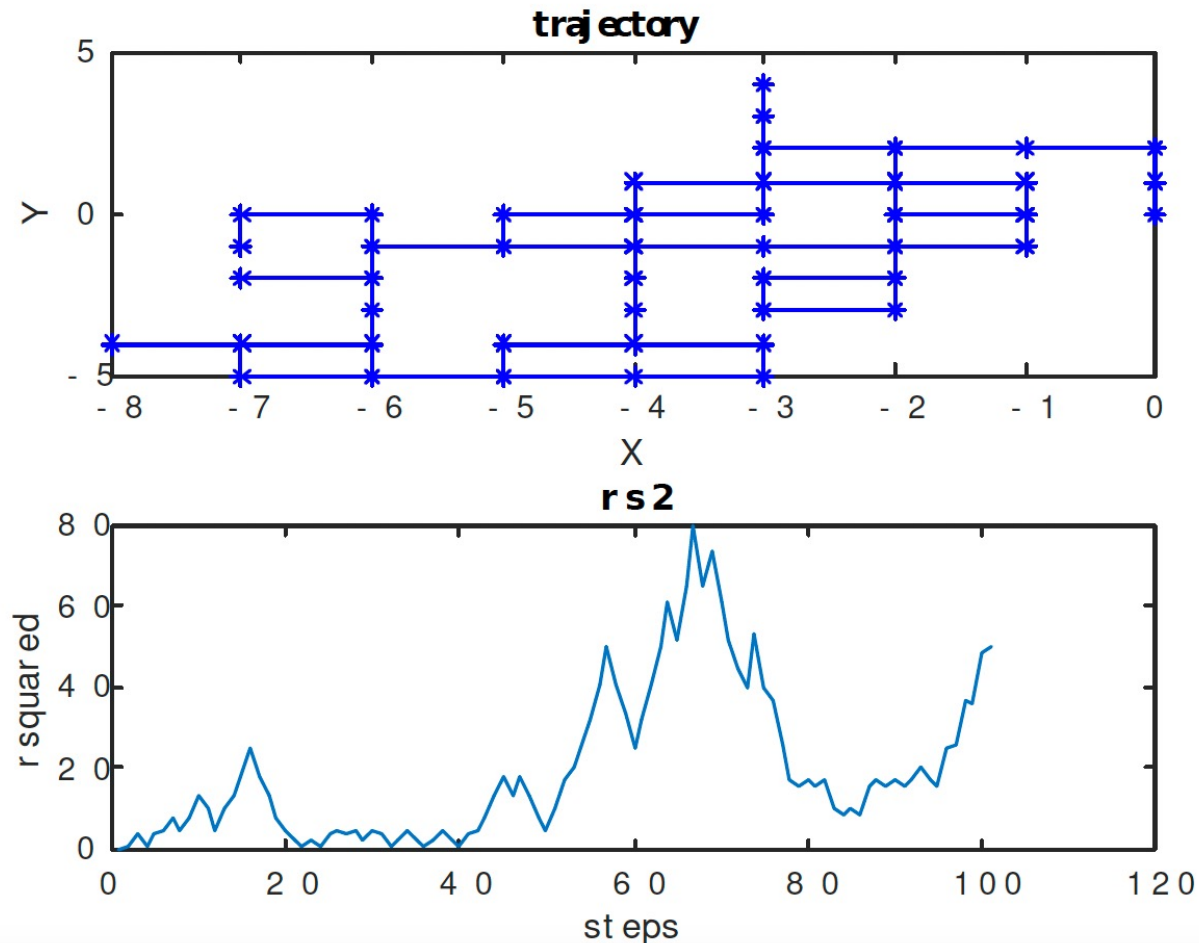2) the square displacement as a function of the steps.

Perform simulations

Run the code for nt=100 and for nt=500 and plot:

1) the trajectory
2) the square displacement as a function of the steps

```
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')
```

In class assignment 2

Write a function, called arw.m, that averages over many random walkers. Compute the mean square displacement and plot it as a function of the number of steps.

First plan your code, write down input and output.

Calling a function

rwn.m contains the function:
function [rs2,x,y] = rwn (nt)

If we want to call it, we can write:
[rs2,x,y]=rwn(nt)

arw.m

```matlab
% 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
```

arw.m

```matlab
figure
subplot(2,1,1)
plot(rwa)
hold on
xlabel('steps')
ylabel('mean square displacement')
title('rwa')
subplot(2,1,2)
plot(sig)
xlabel('steps')
ylabel('standard deviation')
title('sig')
```