# DOING PHYSICS WITH A COMPUTER IN HIGH SCHOOLS:

## DESIGNING AND IMPLEMENTING NUMERICAL EXPERIMENTS

**Maria Peressi and Giorgio Pastore**

University of Trieste and CNR-INFM DEMOCRITOS

DEMOCRITOS

DEmocritos MOdeling Center for Research In aTOmistic Simulation

INFM

# Outline

- Introduction (simulations:
role in physics and physics education)
- Our project in High Schools
- An example: the Brownian motion

# Introduction

Computers

**in physics research** and **in physics education**:

nowadays widely used for many purposes.

Here: **focus only on computer simulations**

= virtual experiments in which our representation of the physical reality, though necessarily schematic and simplified, can be tuned and varied at will **("what-if" experiments) => Predictive power**

("The purpose of computing is insight, not numbers", R. Hamming)

# Computer simulations in physics research

The third paradigm of physics:

- experiment
- theory
- simulation

time for them to play a similar role in physics education?

# Computer simulations in physics education

* A specific aspect of computer aided physics teaching

* Already widely used as an interactive and efficient learning tool:

- several powerful simulations software packages and integrated environments
- many Java applets concerning physical phenomena
- ...

BUT: students hardly can get a feeling of "what is a computer simulation" carrying on numerical experiments with a software ready-to-use

AN ALTERNATIVE APPROACH (not new, however…):

- going beyond the use of simulation software towards a **real computational physics laboratory**

- making students aware of **"what's inside"** a simulation code

- **designing and implementing together with students** simple computer experiments to study real-world problems, the focus being on actual problem solving

A number of VALUABLE EXPERIENCES already done along these lines

# Our experience

with 16-18 years students with typical High School math/phys background

OUR CONTRIBUTION: Transfer of **tools** and **methodology** from **computational physics research** to **education**

FOCUS ON:

- problems that students can afford using basic math. and phys. knowledge normally available to them;
- problem solving; algorithms; select basic numerical and graphical tools (ours: "poor-man" simulations...)

NOT ON:

- specific programming languages or techniques
- sophisticated tools

# Our experience:
## organization of the project

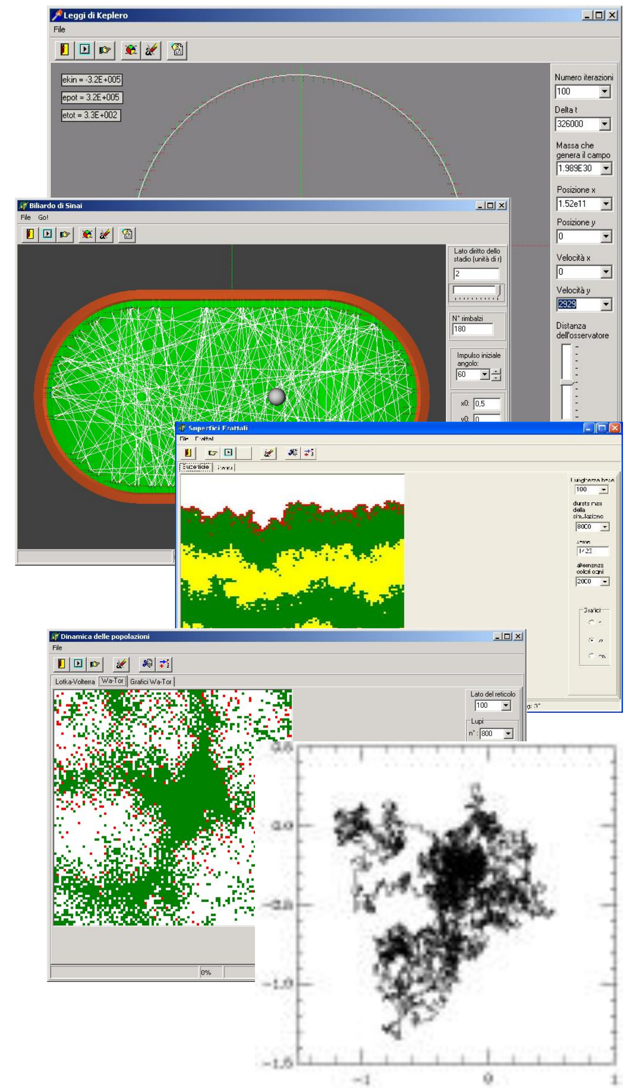
- introductory lecture to classes

- assisted hands-on sessions in computer labs on specific problems (enrolled students; recently also at: “Summer School of Modern Physics for High School Students”, Udine, July 2009)

- on demand: additional sessions on programming language for motivated students

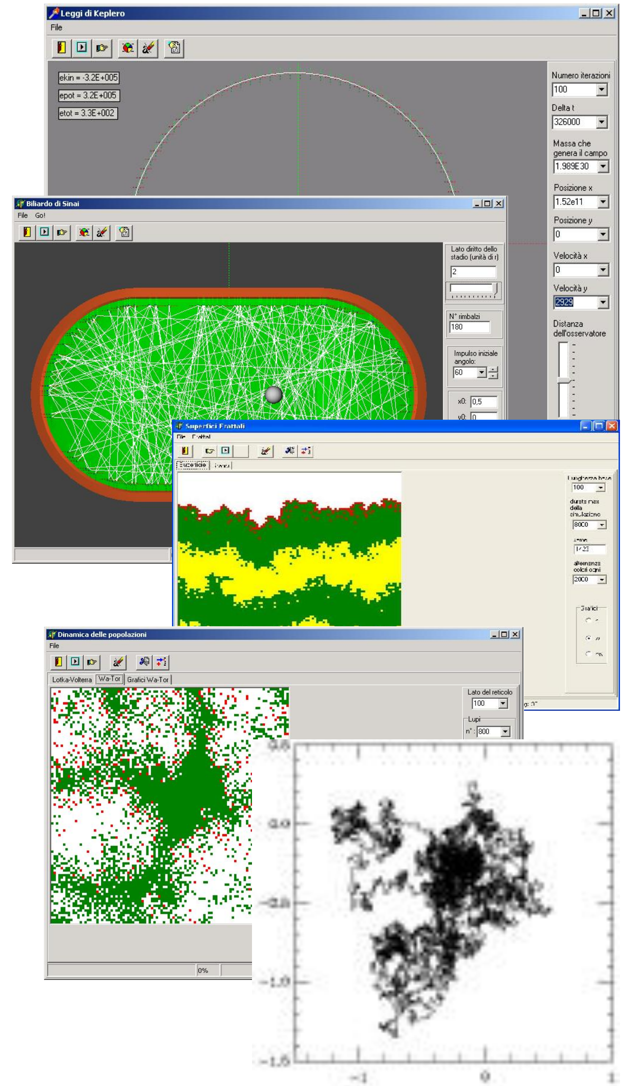
# Our experience:
## some specific problems

- motion of bodies in external forces (in gravitational fields [planets]; in electrostatic and magnetic fields…)
- physics of classic billiards and chaos (from circular billiards to the stadium billiards)
-fractal growth of surfaces (different models)
- prey-predator problem
- ray optics (light refraction in inhomogeneous media)
- Brownian motion



# Our experience:
## some specific problems

- motion of bodies in external forces (in gravitational fields [planets]; in electrostatic and magnetic fields…)
- physics of classic billiards and chaos (from circular billiards to the stadium billiards)
-fractal growth of surfaces (different models)
- prey-predator problem
- ray optics (light refraction in inhomogeneous media)
- Brownian motion



# A case study: the Brownian motion

**prototype** of realistic problems

- **difficult to solve analytically** at high school level (requiring stochastic differential equations or integral calculus...)

but ...

- **easier to solve** numerically !

# The physical system

Einstein, 1905 - explanation of Brownian motion:
- The motion of pollen is due to a stochastic force(*) caused by many collisions with solvent particles much smaller and light (not visible on the chosen observation scale).

(*) can be dealt with statistical methods, without worrying about the details of the dynamics of the small molecules of the solvent

- measurable quantity: mean square displacement $\langle R^2 \rangle$ of the particles
- key relation between diffusion coefficient D and solvent viscosity $\eta$ (through a quite sophisticated mathematical treatment - not for students!) :

$$\langle R^2 \rangle = 2dDt \quad \text{with} \quad D = k_B T/(6\pi\eta P)$$

(t time, d dimensionality of the system, P radius of brownian particles)

# The numerical approach:
## the ingredients

Here: NOT Einstein's, but Langevin's (1906) approach arriving at a Newtonian equation of motion including a *random* force due to the solvent

See: De Grooth BG, Am. J. Phy. 67, 1248 (1999)

Ingredients:

* large Brownian particles - solvent interactions described by: **elastic collisions** between large particle (mass *M*, velocity *V*) and small (solvent) particles (*m, v*);

* **momentum and energy conservation** at each collision

$$MV+mv = MV'+mv'$$

$$MV^2/2+mv^2/2 = MV'^2/2+mv'^2/2$$

# The numerical approach:
## the equation of motion

After **reasonable assumptions** *(many collisions (i) in a time interval $\Delta t$, where $V_i$ are the same…, m<<M…, …)* =>

arrive at a simple expression for $M\Delta V/\Delta t = M(V' - V)/\Delta t$ :

$$Ma = F_s - \gamma V(t)$$

$F_s$ : stochastic force, i.e. the cumulative effect, in the time interval, of many collisions with smaller particles

$-\gamma V(t)$ : drag force, opposite to $V(t)$ ($\gamma > 0$); $\gamma$ can be expressed (using Stokes' formula for a sphere of radius P) as: $\gamma = 6\pi\eta P$.

(both forces have the same origin, in the collisions with the smaller particles)

# The numerical approach:
## discretization of the equation of motion

$$Ma = F_s - \gamma V(t)$$

Rewritten as: $M\Delta V/\Delta t = \Delta V_s / \Delta t - \gamma\, V(t)$

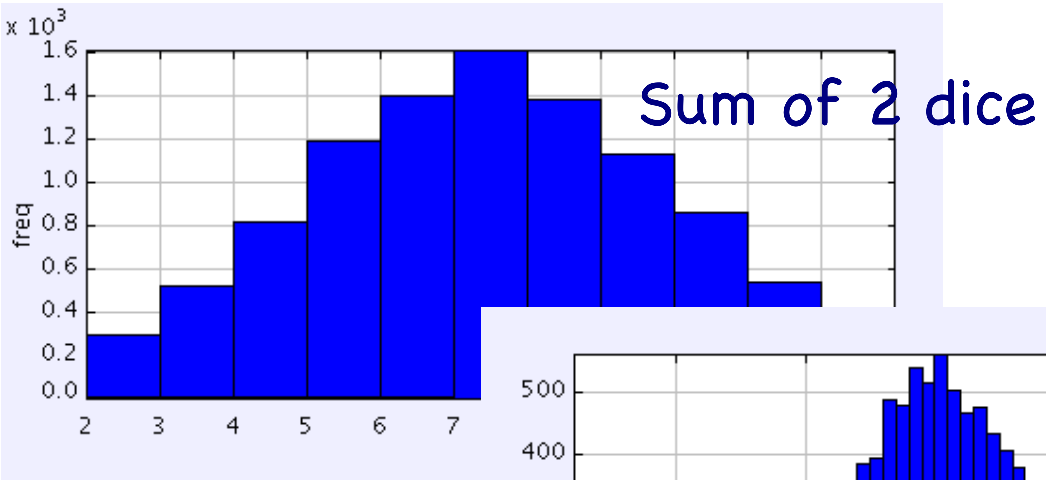
=> $V_{q+1} = V_q + \Delta V_s - \gamma(\Delta t/M)V_q$

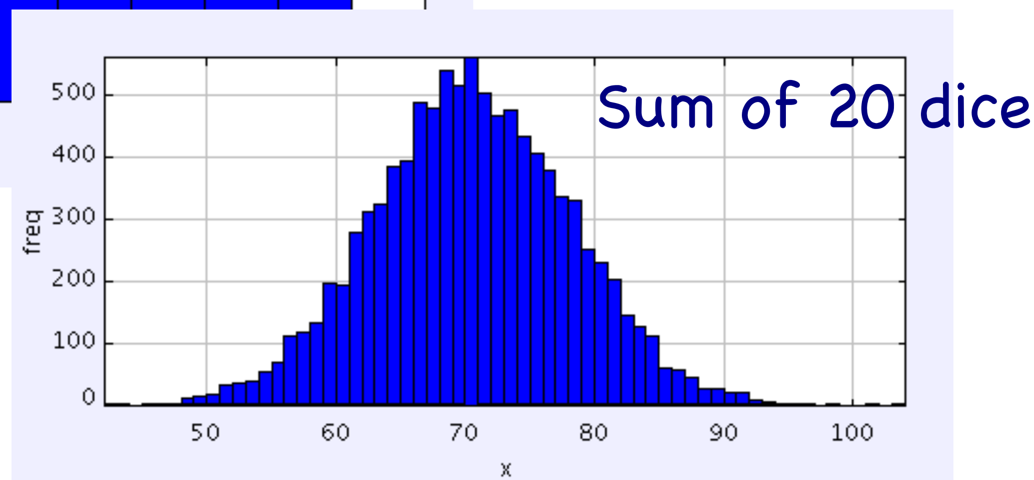
with:

$\Delta V_s = 2mv/M = (\ldots) = 1/M\; v/|v|\; \sqrt{(2\gamma k_B T/n)}$;

At each collision $v/|v|$ is −1 or +1 => after N collisions ???

Allowing students discover themselves (e.g. with dice-rolling experiments) that the result is a **gaussian random variable** $w_q$ centered in 0, s.d.=$\sqrt{(N/2)}$ => (…) =>


Sum of 2 dice
x 10^3
freq
1.6
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
2
3
4
5
6
7


Sum of 20 dice
freq
500
400
300
200
100
0
50
60
70
80
90
100
x

# The numerical approach:
## discretized equations for positions and velocities

$$V_{q+1} = V_q - (\gamma/M)V_q\Delta t + w_q(\sqrt{(2\gamma k_B T\Delta t)})/M$$

$$X_{q+1} = X_q + V_{q+1}\Delta t$$

- the hearth of our numerical approach
- can be easily implemented for iterative execution

NOTE for students: we are NOT imposing any specific time dependence behavior: it will come out as an "**experimental**" result of the simulation

# The numerical approach:
## Input parameters - I

$$V_{q+1} = V_q\,[1 - (\gamma/M)\Delta t] + w_q(\sqrt{(2\gamma k_B T\Delta t)})/M$$

Discuss with students the parameters of the simulation:

- physical parameters of the system: **T** and $\gamma$ (through $\eta$ and P: $\gamma=6\pi\eta P$)

# The numerical approach:

## Input parameters - II

$$V_{q+1} = V_q [1 - (\gamma/M)\Delta t] + w_q(\sqrt{(2\gamma k_B T\Delta t)})/M$$

- time step $\Delta t$ : cannot be fixed a priori!

Some suggestions from physical and rough numerical considerations

[$(\gamma/M)\Delta t < 1$ to reproduce the situation of $T \approx 0$ (damped motion)

$\Delta t$ too small: too long numerical simulations necessary...

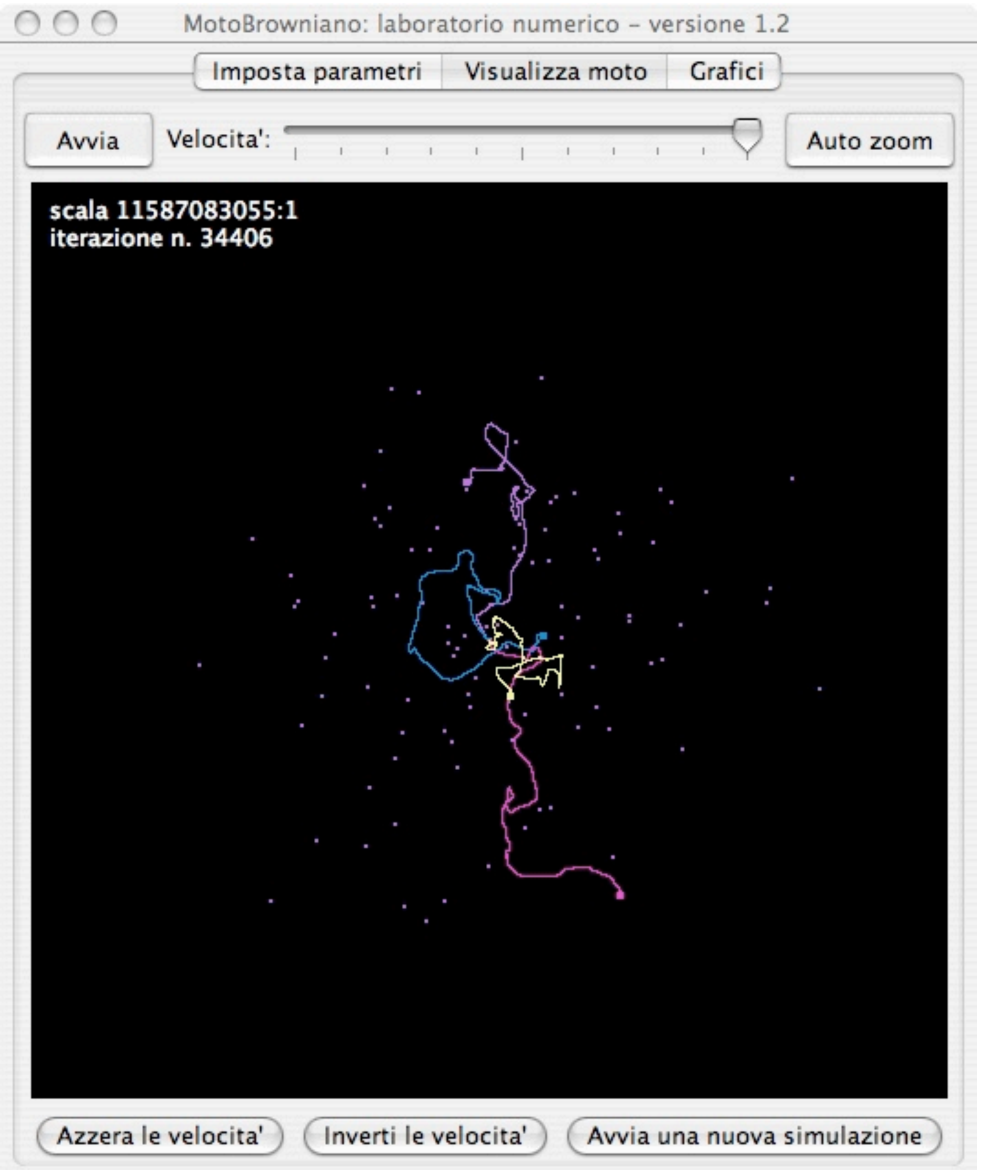
$\Delta t$ too large: serious numerical uncertainties...]

Encouraging students' work:

choice of $\Delta t$ is analogous of an instrument calibration !!!

suggestion: start from small $\Delta t$ *s.t.* $\gamma\Delta t/M \ll 1$, increase $\Delta t$ until important changes in the diffusion coefficient are observed.

# Running the code…




$k_B T = 4 \cdot 10^{-21}$J, $M = 1.4 \cdot 10^{-10}$kg,

$\gamma \approx 8 \cdot 10^{-7}$Ns/m

*Snapshot of a numerical simulation of the Brownian motion in 2D of many large particles. The trajectories of four of them are shown*

# Discovering the results

**Students can prove by numerical experiments:**

(i) the linear behavior of the mean square displacement $\langle R^2 \rangle$ with time:

$$\langle R^2 \rangle = 2dD\, t$$

(i) the validity of the Einstein relation between the slope of this line and the solvent parameters (temperature and drag coefficient):

$$\langle R^2 \rangle = (2d\, k_B T / \gamma)\, t$$

# Open questions:

## which programming language?

A few considerations -

(i) the situation among teachers and among students:

- In Italian High Schools: information technology skills: very heterogeneous situation!
- In some classes: a limited know-how of rather "old" languages (Pascal in P.N.I. classes)
- Among teachers: some difficulties - no time, no funds, limited skills

(ii) Our whishes: easy, Open Source, possibly multi-platform

=> First experiences of our project: in Delphi/Pascal (free educational version) - Present experiences: Java

# The numerical approach:
## a possible implementation (Java)

```
...
double posPrec[]=new double[NDIM];
for (int ip=0;ip<nPart;ip++) {
    for (int jc=0;jc<NDIM;jc++) {
posPrec[jc]=pos[ip][jc];
vel[ip][jc] =
    vel[ip][jc] * ( 1 - gamma*dt/massa ) +
    random.nextGaussian()*Math.sqrt(gamma*kT*dt)/massa;
pos[ip][jc] += vel[ip][jc] * dt;
     }
```

# Conclusion

Simulation proposed to students of different High Schools.

Students: able to follow all steps of the analysis, including the implementation of the algorithm.

Two hands-on sessions (about three hours each one): a reasonable choice in order to introduce the physical problem and the specific algorithm, to become a bit familiar with random numbers and finally to understand and make individual numerical experiments with a provided working code.

Writing the code from scratch would be equally possible, but this would require previous training on computer programming.

**FOCUS on: problem solving, algorithms, calibration of the instrument, "experimental" approach**

A version of the Java software developed by the Authors and implementing the algorithm discussed here can be found in:
http://www.democritos.it/edu/

General Web site of the educational projects in Physics of UniTS:
http://www.laureescientifiche.units.it/
(in Italian)

Thanks to: for logistic support

*Thank you for your attention !*