Examples of Physics quantities visualization.

P. Hruska
Department of Physics, Faculty of Electrical Engineering and Communication
Brno University of Technology

Abstract

Paper presents several examples of Physics quantities visualization in different fields of Physics: Kinematics, Wave motion, Quantum Mechanics and Molecular Physics. In those examples, there are the quantities definitions or relations, the visualization is based on. The M–scripts and M–functions, or their relevant parts with basic comments are attached to each example.

Keywords:

Kinematics, Pulses superposition, Uncertainty relations, Brownian motion, Matlab, Physics teaching

Introduction

Visualization of quantities and processes is an integral part of modern Physics teaching. Variety of computer programs have been used by the authors for this purpose. Matlab7, with its powerful graphics appears to be one of the most convenient tool to support this effort. Four selected examples of the quantities visualization are presented: visualization of tangent and normal acceleration of moving particle, superposition of wave pulses, the Heisenberg uncertainty relations and the Brownian motion. We tried to keep the complexity of Matlab functions at an introductory level. Hints, added for more examples compilations, could be motivating for lecturers to work out and use further Matlab visualization scripts and functions.

1. Kinematics. Tangent and normal acceleration vectors of a moving particle

Motions, their classifications and comparisons, are the main kinematic topics. The basic, introductory quantities of particle-like objects, used in kinematics, are the *position vector* $\vec{r}$, instantaneous *velocity* $\vec{v}$ and *instantaneous acceleration* $\vec{a}$. The quantities are vectorial functions of time *t*. Curve $\vec{r} = \vec{r}\,(t)$ is referred to as *trajectory*. Instantaneous velocity and instantaneous acceleration are first and second time derivatives of $\vec{r}$: ('instantaneous' in this connection will be omitted below.)

$$\vec{r} = \left[x(t), y(t), z(t)\right] \quad (1.1)$$

$$\vec{v} = \frac{d\vec{r}}{dt},\ \vec{a} = \frac{d\vec{v}}{dt} \quad (1.2), (1.3)$$

Acceleration can be decomposed into *tangent* and *normal* components ($\vec{a}_t$ and $\vec{a}_n$). The components are related to velocity changes: the change of magnitude of $\vec{v}$ ($\left|\vec{v}\right| = v$) and the change of direction of $\vec{v}$ ($\vec{u}_t = \vec{v}\,/\,v$, $\vec{u}_t$ being the unit tangent vector), resp. Vector $\vec{v}$ is always tangent to the trajectory. It holds

$$\vec{a}_t = \frac{dv}{dt}\vec{u}_t \quad (1.4)$$

$$\vec{a}_n = \vec{a} - \vec{a}_t \quad (1.5)$$

Visualization of vectorial quantities and investigation of their time evolution belong to frequent teaching tasks. Functions such as comet(x,y), comet3(x,y,z), or some other ways of adding arrows to graphs, offered by Matlab, are not fully suitable for the visualization of kinematic vectors. Therefore, we have compiled M-file functions sip(A,B) and sip2(A,B) to visualize the vectorial quantities as arrows. The functions are short and simple, compared to another Matlab function quiver(x,y,u,v), that is rather complex and suitable for very large vectorial fields.

The function sip(A,B) plots a vector B, with its starting point at point A, while [x y] = sip2(A,B) returns data to the arrow handle. M-function sip2(A,B) is given below

```
function [x y]=sip2(A,B);
% Data for 2x5 ARROW matrix

AB=[A;A+B];
x = AB(:,1);y = AB(:,2);
dx = x(2) - x(1) + eps;
dy = y(2) - y(1);
b = norm([dx dy]);
fi = atan(dy/dx);
% appearence
ahx = 0.061*b;awy = ahx/2;
% arrowhead
xt = [-ahx, 0,-ahx]* sign(dx);
yt = [ awy/2, 0, -awy/2 ];
% rotation by fi
roo = [cos(fi) -sin(fi); ...
   sin(fi) cos(fi)]* [xt; yt];
xt = roo(1,:) + x(2);
yt = roo(2,:) + y(2);
x = [xt xt(2) x(1)];
y = [yt yt(2) y(1)]; % plot(x,y)
```

Comments on [x y]=sip2(A,B)
A = [a1 a2], B = [b1 b2] are 2D point/vector coordinates. Magnitude of vector B is denoted by *b*, direction is given by angle *fi,* formed by the vector B and (+*x*) axis . The arrowhead segment is the hypotenuse of rectangular triangle of sides 0.06**b* and 0.03**b*. The coordinates of head segments are initially *xt, yt* – two 3 × 1 matrices. The rotation of the arrow head through angle *fi* is done by

matrix [cos(*fi*) -sin(*fi*); sin(*fi*) cos(*fi*)]
and matrix [*xt; yt*]
multiplication.

The resultant arrow coordinates [*x y*] are the output parameters of function [x y] = sip2(A,B). Matrix [*x y*] of size 2 × 5 is a unique object, suitable for handle setting. Function sip(A,B) has no output parameters, but its last line is active: plot(x,y)

As an example of visualization, the motion of a particle along a trajectory with the characteristic acceleration vector $\vec{a}$, its tangent and normal components $\vec{a}_t$ and $\vec{a}_n$ are presented, Fig.1. Attached is the M-script hacc1.m, generating the visualization. Since the velocity vector $\vec{v}$ is always tangent to the trajectory, it is used for the vector $\vec{a}_t$ construction.

```
% hacc1
% Visualization of a moving particle  tangent (at)
% and normal (an) acceleration vectors

% -- Part 1 --------------------------------
syms t real;
r=[2*sin(t)-t/5-4, 3*sin(t/1.5)+t/5-1];
ezplot(r,[-0.5,5*pi]) % trajectory
v=diff(r);a=diff(v); % basic definitions
vv=sqrt(v(1)^2+v(2)^2); % velocity magnitude
v0=v/vv;      % unit tangent vector
atv=diff(vv); % definition of at magnitude
at=atv*v0;    % vector at
an=a-at;      % vector an

 % -- Part 2 ---------------------------------
t=0;R=subs(r);A=subs(a);At=subs(at);An=subs(an);
[x y]=sip3([0 0],R);
position=line('EraseMode','xor','Color', ...
'r','XData',x,'YData',y);
[x y]=sip3(R,A);
acc=line('EraseMode','xor','Color', ...
'm','LineWidth',2.5,'XData',x,'YData',y);
```

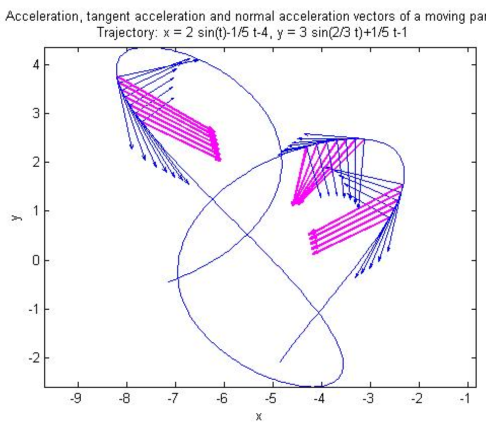


Fig1. A particle moving along a given trajectory. Vectors of acceleration, tangent acceleration and normal acceleration are visualized at selected instants of time.
Trajectory and times in Fig1:
r=[2*sin(*t*)-*t*/5-4, 3*sin(*t*/1.5)+*t*/5-1]; 0<*t*<5*pi
t= [0.9:0.1:1.3 2.4:0.1:3.0 3.3*pi:0.1:3.5*pi]

[x y]=sip3(R,At);
acct=line('EraseMode','xor','Color', ...
'b','LineWidth',2,'XData',x,'YData',y);
[x y]=sip2(R,An);
accn=line('EraseMode','xor','Color', ...
'r','LineWidth',2,'XData',x,'YData',y);
 	
% -- Part 3 ----------------------------------
for t=0.1:0.1:5*pi
R=subs(r);A=subs(a);At=subs(at);An=subs(an);
[x y]=sip2([0 0],R);
set(position,'XData',x,'YData',y);
[x y]=sip2(R,A);
set(acc,'XData',x,'YData',y);
[x y]=sip2(R,At);
set(acct,'XData',x,'YData',y);
[x y]=sip2(R,An);
set(accn,'XData',x,'YData',y);
pause(0.1) % drawnow
end
	Comments on hacc1.m
	
	Kinematic equations (definitions) - part 1 - are treated in terms of symbolic variables - *t, r, v, v0, vv, a, atv, at, an.*
	Function diff is symbolic time derivative.
	Graphic handles - position, acc, acct, accn - part 2 – are created at t = 0. Numeric vector components *R, A, At, An*, obtained by substitutions, serve as input parameters of M -function sip3 for arrow data [x y] evaluation.
	The actual visualization - Part 3 - is a standard EraseMode-xor animation, in which numeric data t are substituted into symbolic variables $\vec{r}$, $\vec{a}$, $\vec{a}_t$, $\vec{a}_n$ and resultant numeric variables *R, A, At, An* are transformed into arrow graphic data, continually supplied to the object handles, erasing old and redrawing new values of the objects – arrows, representing the vectors. The command drawnow should be used, if pause(), slowing down the animation, is omitted.

In a very similar way, one can visualize Cartesian components of particle velocity, acceleration, components of force (being given mass of the particle). The procedure can be extended to body translations and rotations.

2. Wave motion. Superposition of wave pulses

The Wave Equation (WE) is a central differential equation describing a distinct phenomenon in Physics – Waves. The waves in elastic media are elastic (acoustic) waves. The displacement of a medium element from its equilibrium position is u(x, t), the phase velocity is c. The WE reads (1D)

$$\frac{d^2u}{dx^2} - \frac{1}{c^2}\frac{d^2u}{d^2t} = 0 \qquad (2.1)$$

Three kinds of waves are known in Physics: Electromagnetic waves ($\vec{E}$ and $\vec{B}$ are varying electric and magnetic fields), probability or deBroglie's waves (symbol ψ) and elastic waves, mentioned above. Linearity is an important feature of the WE. It implies that a linear combination of arbitrary solutions of the WE is also a solution. It is well known *Principle of Superposition.* Let us mention, that waves

$$u = f(x \pm ct) \qquad (2.2)$$

f being arbitrary function, are solutions of (2.1). Take notice of the independent variables x, t mutual positions in the function f argument.

Two pulses *y1, y2* are used in the coming example superpos2e.m. They travel along (x) axis at opposite velocities ($\pm$ c). When they encounter each other in space, they superpose, without mutual affecting. They can be treated as one wave: *y3=y1+y2.*

```
% superpos2e.m
% Superposition of two pulses,
% travelling in opposite directions

% - Part 1 --------------------------------------
t=-20:.3:20;
y1=exp(-(t-15+1/10).^2);
y2=1/3*exp(-(t+15-1/10).^2/9);
y3=y1+y2;
figure1=figure;
% -- subplot1:----------------------------------
axes1=axes('Position',...
  [0.13 .7683 0.775 0.1567],'Parent',figure1);
set(axes1,'XLim',[-16 16],'YLim',[-0.1 1.1]);
plot1=plot(t,y1,'Parent',axes1);grid;
set(plot1,'EraseMode','xor','Color',...
   [0 0 1],'LineW',1.5);
% -- subplot2:----------------------------------
axes2=axes('Position',...
   [0.13 .5492 0.775 0.1567],'Parent',figure1);
set(axes2,'XLim',[-16 16],'YLim',[-0.1 1.1]);
plot2=plot(t,y2,'Parent',axes2);grid;
set(plot2,'EraseMode','xor',...
   'Color',[1 0 0],'LineW',1.5);
% -- subplot3:----------------------------------
axes3=axes('Position',...
   [0.13 .111 0.775 0.36],'Parent',figure1);
set(axes3,'XLim',[-20,20],'YLim',[-0.1 2.1]);
plot3=plot(t,y3,'Parent',axes3);grid;
set(plot3,'EraseMode','xor',...
   'Color','m','LineW',2.5);
pause

% -- Part 2 ----------------------------------------
for k=1:300
  y1=exp(-(t-15+k/10).^2);
set(axes1,'XLim',[-20 20],'YLim',[-0.1 1.1]);
set(plot1,'XData',t,'YData',y1);
  y2=1/3*exp(-(t+15-k/10).^2/9);
set(axes2,'XLim',[-20 20],'YLim',[-0.1 1.1]);
set(plot2,'XData',t,'YData',y2);
  y3=y1+y2;
set(axes3,'XLim',[-20 20],'YLim',[-0.1 2.1]);
set(plot3,'XData',t,'YData',y3);
drawnow          % pause(0.1)
end
```

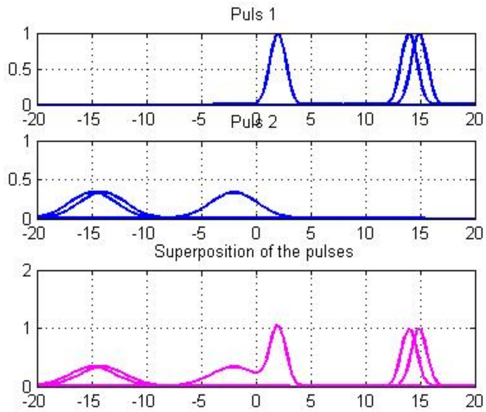


Fig2. Superposition of two pulses, traveling in opposite directions at three instants of time.
Pulses:
y1=exp(-(*t*-15+*k*/10).^2);
y2=1/3*exp(-(*t*+15-*k*/10).^2/9); *k* are parameters in seconds.

Comments on superpos2e.m

The shape of the pulses is chosen to be bell-like functions *y1* and *y2* in Part 1 for x = ±14.9 m (phase velocity = 1m/s). The shape of the pulses can be easily modified. The pulses move along horizontal axis. Ticks are in meters.

The figure is divided into 3 subplots with handles: axes1, axes2 axes3.
The plot handles are:
plot1, plot2, plot3
The superposition, as an implication of linearity of the Wave Equation, is indicated by adding up the two pulses. For the actual visualization, the EraseMode-xor animation is applied using the loop. Function data are supplied into subplots by setting the handles property 'YData'.
The speed of animation can be controlled by inserting command pause(p), *p* in seconds. The first pause before the loop (next to Part 2) can be omitted on repeated script application.

Using a similar approach, we can construct a series of wave visualizations, such as standing waves, superposition of plane waves of near frequency and near wavelength in order to visualize group and phase velocities, to mention few examples.

3. Quantum Mechanics (QM). The Heisenberg relations of uncertainty.

Uncertainty of the particle position Δx and the corresponding uncertainty of momentum Δp are related by

$$\Delta x * \Delta p \ \geq \ 2\pi\hbar\,, \qquad (3.1)$$

where $\hbar$ =1.054 572 6 ×10^{-34} Js is the Planck constant. The expression on the right hand side of (1.3) depends on the definition of uncertainty. Inequality (1.3) is the Heisenberg relation of uncertainty for *x*-coordinate and p_x-component of a particle, a basic relation of Quantum mechanics. It states, that simultaneous unlimited accuracies of a particle position along (*x*) axis and *x*-component of momentum are fundamentally impossible. There are analogous relations for (y, p_y), (z, p_z) and (*E, t*).

The reason of validity (3.1) is not in the measuring instruments, it rises from the way we describe the particle position and momentum. Full information on a particle state is carried by its wave function ψ(*x*) (de Broglie wave). The probability (probability density) of finding the particle is $|\psi|^2$. The wave function of a free particle of momentum $p = \hbar\ k$ *is* a monochromatic wave

$$\psi(x) = \exp(j\ p/\hbar * x) \tag{3.2}$$

It is the asymptotic case of a wave function with accurate momentum and infinite inaccuracy of position.

The example is based on a property of the wave packet.

Superposition of waves of varying wavelength – or momentum $p= \hbar k = \dfrac{2\pi}{\lambda}\hbar$ results in finite accuracy of position at the expense of momentum accuracy, as shown below. Almost arbitrary superposition can be easily done in MATLAB.

In this example, several wave packets were constructed by superposition of plane harmonic waves exp(j*kx*). The superposition was performed by integration throughout a chosen interval of wavenumbers *k*, denoted by Δ*k*.

$$u(x) = \int_{k_0-\Delta k/2}^{k_0+\Delta k/2} \exp(jkx)dk \tag{3.2}$$

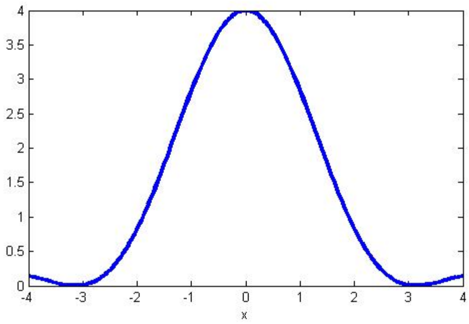
Since some physical quantities, like intensity, energy and (here) probability of finding the particle, are related to the amplitude squared, we introduced another wave packet *w*=|*u*| ^ 2. To determine its width, equal to that of *u*, we chose the separation Δ*x* of the ends of the first maximum. The product of the two uncertainties Δ*x,* Δ*k* is independent of either uncertainty, being equal to 2π. Up to this point of the example, no mention on QM was said. Having multiplied the latter relation by the Planck constant $\hbar$, one obtains equality of (3.1) .

$$\Delta x * \hbar\ \Delta\kappa = \Delta x * \Delta p = \Delta x * \Delta p = 2\pi\hbar$$

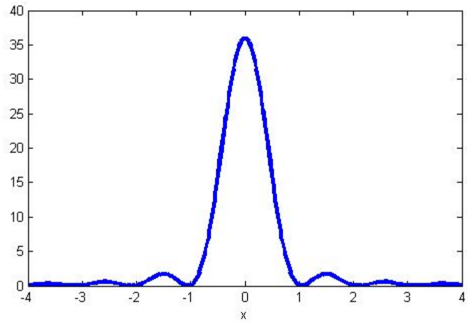
Thus, we have entered QM. The meaning of Δ*x* is a space region, where probability density of finding the particle in question is very high, compared to its neighborhood, where it is negligible, while Δ*p* is the interval of momentum, that was utilized in the wave packet *u or w* creation. Therefore one has to accept that the particle under analysis should have more values of momentum –i.e. all the values within Δ*p.*

Four probability curves were generated, each by integrating a monochromatic wave ψ = exp(j*kx*) within a specified interval of wavelengths. Corresponding intervals Δ*p* are: $2\hbar$, $6\hbar$, $14\hbar$, and $80\hbar$. Brief comparison of the curves width yields for each curve a relation $\Delta x * \Delta p = 2\pi\hbar$. The curves are displayed in Fig.3.

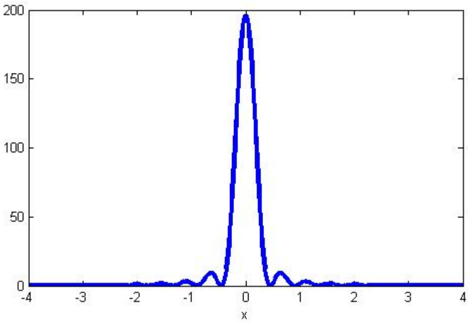
Thus, it clarifies in a natural manner the essence of the Heisenberg relations.



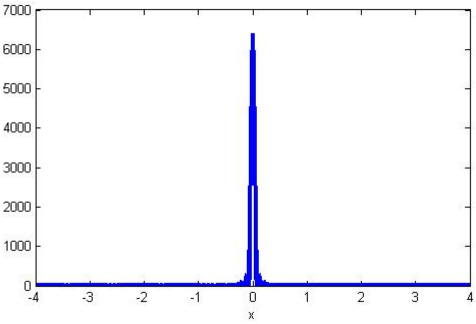
$\Delta p = 2\hbar$, $\Delta x = 3.1415933$



$\Delta p = 6\hbar$, $\Delta x = 1.0471944$



$\Delta p = 14\hbar$, $\Delta x = 0.44879785$



$\Delta p = 80\hbar$, $\Delta x = 0.078555641$

Fig. 3. Wave packets created by superposition of monochromatic waves. Relation between the packets width and the interval Δp is the basis of the Heisenberg relations of uncertainty.

```
% h1exp
% packet generation by superposition
% of plane waves from range Δk=(-k, k), and
% packet width Δx evaluation. Product  Δx*Δk =2*pi,
% i.e. is  INDEPENDENT of both Δk, Δx
digits(10)
syms k x real
psi=exp(j*k*x);
aa=1.2;me=10;
disp(['  Δx' '             Δk' '           Δx*Δk']);
for m=1:me

% -- Part 1 ---------------------------
u=int(psi,k,100-aa^m,100+aa^m);
w=abs(u)^2;

% -- Part 2 ---------------------------
h{m}=@(x) eval(w);% wave packet handles
d(m)=fminbnd(h{m},0,(pi+.5)/aa^m);

% -- Part 3 ---------------------------
figure(m),fplot(h{m},[-4,4]);%graphing
disp([vpa(d(m)) vpa(2*aa^m) vpa(2*d(m)*aa^m)])
end;
```

Results of h1exp.m evaluation:

```
 Δx              Δk               Δx*Δk
[ 2.617986140,  2.400000000,  6.283166736]
[ 2.181655117,  2.880000000,  6.283166736]
[ 1.818045931,  3.456000000,  6.283166736]
[ 1.515038275,  4.147200000,  6.283166736]
[ 1.262531896,  4.976640000,  6.283166736]
[ 1.052109914,  5.971968000,  6.283166736]
[ .8767582613,  7.166361600,  6.283166736]
[ .7306318844,  8.599633920,  6.283166736]
[ .6088599037,  10.31956070,  6.283166736]
[ .5073832530,  12.38347284,  6.283166736]
```

$\Delta p = \hbar * \Delta k$

Replacement of Δk by Δp yields equality in (3.1), i.e. the Heisenberg relation

Comments on h1exp.m

Integration (Part 1) of plane harmonic waves within the interval Δk =100-aa^m,100+aa^m represents a wave superposition, creating a wave packets u and w=u^2 of equal width. The packet width d determination (Part 2) is performed by function fminbnd(), which requests the handle of function w as its parameter. The handle is created a line earlier and used also for plotting (Part 3). Variable Precision Arithmetic function, vpa enables to display results to 10 (digits(10)) significant figures.

4. Molecular Physics. Brownian motion

A very small macroscopic particle immersed in a liquid or a gas exhibits a random type of motion, called Brownian motion. It reveals very clearly the statistical fluctuations in a system in thermal equilibrium. A variety of important situations are basically similar: random motion of the mirror mounted on the suspension fiber of a sensitive galvanometer, or the fluctuating current in an electric resistor. Surprisingly, the source of damping in the motion, is also the source of fluctuations [2, 3, 4]. Particles of pollen dust (one to two micrometers long) immersed in water at temperature *T* have their mean square energy 3/2k*T*, the same as the individual molecules of water have. Thus, the pollen particles move at mean-square speed v_p

$$v_p = v_w \sqrt{\frac{m_w}{m_p}} \tag{4.1}$$

where v_w , m_w , m_p are mean-square speed of the molecule of water, its mass and the mass of the pollen particle, resp. In the theory of Brownian motion one applies numerous statistical quantities, that can be visualized in graphs, offered by M-function brownian5(N), where N is a number of particles. This function offers graph of the distance *r* of a particle from its initial position and its mean square displacement as functions of time at the end of motion simulation. Other statistical quantities of interest can be included into the M-function and visualized.

The idea of Brownian motion visualization is an animation of a set of N particles, located at $r = [x , y]$, see help, animation[1]. Their locations vary as *r*=*r*+*s**randn(r) . Fixed time period *s* multiplied by random velocity yields a particle displacement. It corresponds the case of taking photos at a fixed rate *s* (seconds) of particles observed by microscope.

<table>
<tr>
<td>

```
function brownian5(N)
% Simulation of Brownian motion of N particles
% Distance traveled by the 'red' particle
% Parameters of the 'red' particle  are recorded
% M-function sip3.m to be on path
fg1=figure(1);

% -- Part 1 -----------------------------------------------.
axes1 = axes('Position',..
   [0.13 0.11 0.60862 0.815],'Parent',fg1);
s=0.02; % particle position sampling time
x=rand(n,1)-.5;y=rand(n,1)-.5;
h1=plot(x,y,'.'); x1=[0];y1=[0];
hold on
hh=plot(x1,y1,'ro');
hold off
title('Brownian motion');axis([-2 2 -2 2])
xlabel('x'),ylabel('y')
rectangle('Position', [-.5 -.5 1 1])
set(h1,'EraseMode','xor','MarkerSize',12)
set(hh,'EraseMode','xor','MarkerSize',4)
hs=line('EraseMode','xor','Color', ...
'r','LineWidth',0.6,'XData',x1,'YData',y1);

.% -- Part 2-----------------------------------------------
axes2 = axes('Position',
 [0.8508 0.11 0.04419 0.815],'Parent',fg1);
xs=1:2;ys=ones(1,2)*sqrt(x1^2+y1^2);
h2=area(xs,ys);title('|R|');
set(h2,'FaceColor','flat');
axis([1 2 -0.001 1.21]);
```

</td>
<td>

Comments on Brownian5(N)
figure(1) is divided in two subplots with handles axes1 and axes2

Positions of N particles are set up in Part 1 by rand(N,1) function within a square of side 1 and centre at origin. Handle of particles positions is: h1. The handle of selected particle, located initially at origin is hh, the handle of its pointer is hs.

The handles are set to EraseMode-xor mode for animation. Instantaneous distance of the selected particle from the origin is indicated by the bar height in subplot 2, Part 2.

Animation starts in Part 3, where the technique try-catch-end is used. On the figure(1) closing, the program jumps into catch part – Part 4, with creating figure(2) and ending up.

During the Brownian motion visualization the particles new positions are continuously generated: (x, y), and (x1, y1), resp. The coordinates of generated displacements are of the form fixed time tau × randn velocity. Function sip2, described above, is used to visualize the selected particle. The values of its instantaneous

</td>
</tr>
</table>

```
set(axes2,'XTickLabel','')
m=1;Rs=0; tau=0.2;

while 1 % -- Part 3-----------------------------------------
   try
x=x+s*randn(n,1); y=y+s*randn(n,1);
x1=x1+s*randn(1,1);y1=y1+s*randn(1,1);
[x2 y2]=sip2([-0.5 -0.5],[0.5+x1 0.5+y1]);
set(h1,'XData',x,'YData',y);
set(hh,'XData',x1,'YData',y1);
set(hs,'XData',x2,'YData',y2);
R=x1^2+y1^2; Rs=Rs+R;
r(m)=sqrt(R); rs(m)=sqrt(Rs);
ys=ones(1,2)*r(m);
set(h2,'Ydata',ys);
m=m+1;pause(tau);

   catch % -- Part 4 ----------------------------------------
      figure(2)
      plotyy(r,'-r.',rs,'-b.'),xlabel('time')
      title(['Particle distance r (red) '...
         'and mean squre distance rs (blue)'])
   break
  end
end
```

displacement from the origin is recorded as

r (m) = sqrt(R),

R= x1^2+y1^2 (4.2)

where $x1$, $y1$, are instantaneous x and y components of the selected particle, and its mean-square displacement as

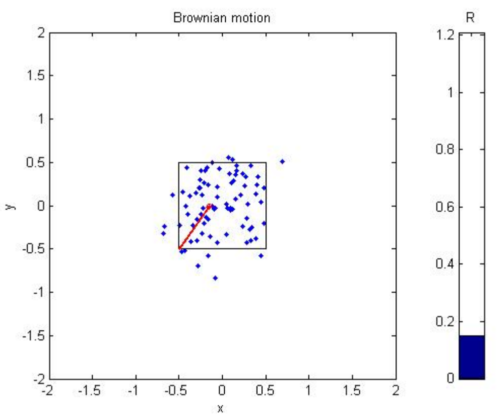
rs(m) = sqrt(R1+R2+…+Rf)

where the sum in the sqrt argument is a cumulative sum of values R as in (4.2).

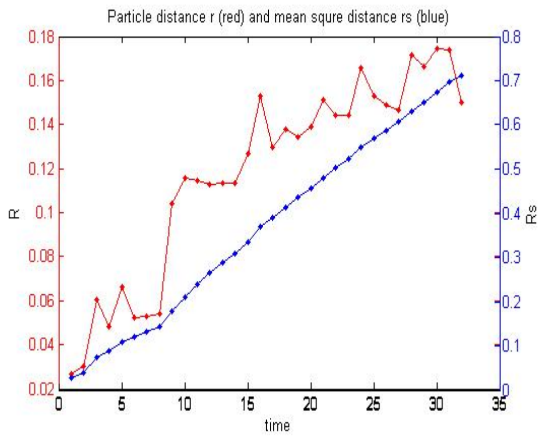
Time dependence of r, rs is plotted in figure(2), using two-axis command

plotyy

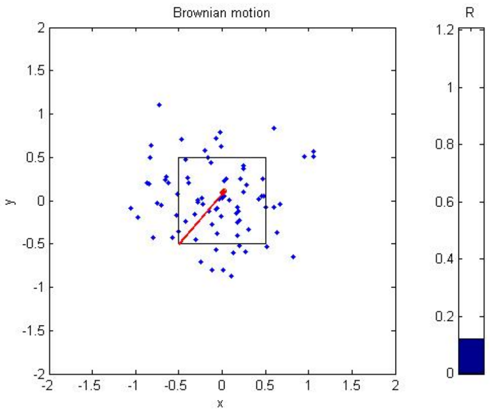
The Brownian motion is shown in the main graph, while the actual distance of a selected particle from its initial position (origin) is visualized by a bar height in the right-hand side graph. At any time the visualization can be interrupted and time evolution of the quantities recorded during the animation be shown. An arrow is supplied to point to the selected particle.

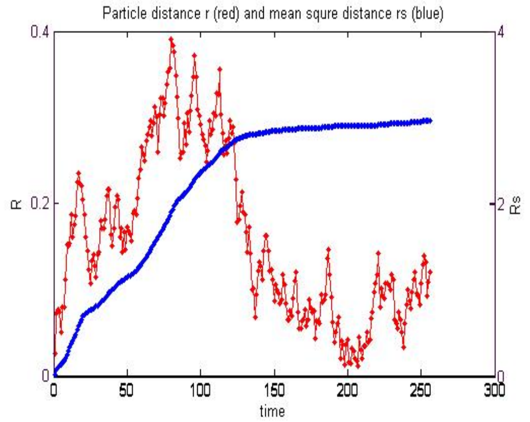


a) Particle distribution after 32 time units. Bar measures the distance of selected particle from origin.



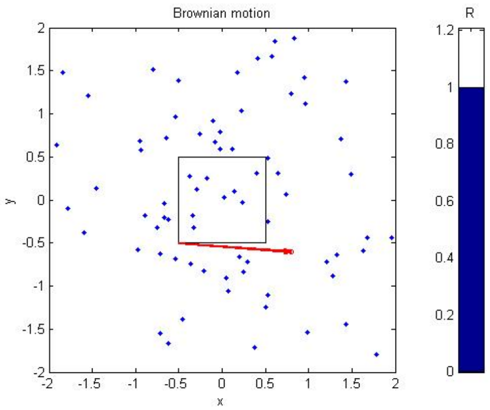
b) Time evolution of instantaneous distance from origin and mean-square distance through 32 time units.

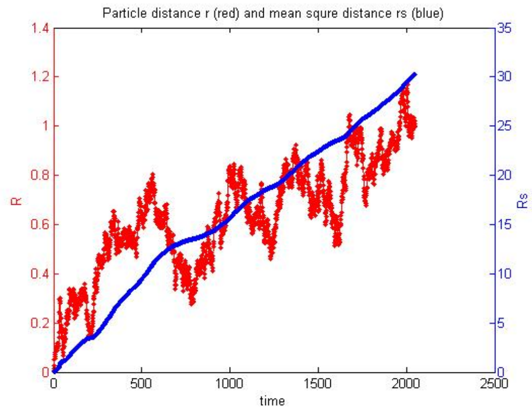




a) Particle distribution after 256 time units. Bar measures distance of selected particle from origin

b) Instantaneous distance from origin and mean-square distance through 256 time units.





a) Particle distribution after 2048 time units. Bar measures distance of selected particle from origin

b) Instantaneous distance from origin and mean-square distance through 2048 time units.

Fig. 4. Brownian motion visualization. Time evolution of statistical quantities of the selected particle

Conclusions

Examples of Physical quantities visualization by static graphs and animations using Matlab 7 are presented. M-functions and M-scripts. with comments are supplied. It is believed, they could facilitate certain parts of Physics teaching.

References

[1] M. Alfonso, E. J. Finn: Fundamentals University Physics, *Addison-Wesley*, Amsterdam 1973
[2] R. P. Feynman, R. B. Leighton, M. Sands: Feynmanovy přednášky z fyziky, *Fragment*, Havlíčkův Brod 2000
[3] Ch. A. Whitney: Random Processes in Physical Systems, *Wiley Interscience*, New York 1990
[4] F. Reif: Fundamentals of Statistical and Thermal Physics, *McGraw-Hill*, New York, Tokyo 1965
[5] P. Hruska, L. Grmela, P. Koktavy: MAPLE in General Physics e-grading, *Proc. New Trends in Physics, NTF, ed. M. Strunc, P. Dobis,* Brno 2004, ISBN 80-7355-024-5.
[6] P. Hruška, L. Grmela, P. Koktavý: Teaching General Physics with MATLAB, *Proc. Int. Conf. on Phys. Teaching in Engineering Education PTEE,* Brno 2005, ISBN 2-914771-28-2

Contact:

Doc. RNDr. Pavel Hruška, CSc.
Ústav fyziky, FEKT VUT v Brne,
Technická 8, 616 00 Brno
Tel. +420 541 143 258, email: hruskap@feec.vutbr.cz.