This opinion was confirmed by the scientists from the Botanical Garden in Prague as well.

Requirements of the Movement

Of course, we must deal with the question of why this movement occurs only in aspen leaves and not for example with oak ones. These conditions can be characterized in the whole as morphological. The best way to explain this is a comparison.

Firstly - the aspen leaf-stalk is quite flat, that means it is very easy to be bend (the section of the oak one is round). Next fact is that the joint between the stalk and the blade has only very small area (the plane of the blade is perpendicular to the plane of the stalk) and therefore although it is very firm, it is very flexibile as well. Besides that the aspen stalk is quite long compared to the oak and so less air flow is necessary to make the leaf move. We must consider as well, that the axis of the stalk and the plane of the blade contain an angle of 20 degrees. All mentioned gifts of the aspen leaf are shown on the picture.

There is yet another experiment that was done. We were finding the centre of gravity of the leaf with and without the stalk (see photo). The result showed that in both cases it is placed quite near to the point of joint between the stalk and the blade of the leaf, which makes the leaf even more easy to tremble.

Summary

The problem was posed in a very imprecise way – it wants us to find the causes of a phenomenon which – in reality – does not occur. We showed (and consequently verified by a practical experiment) that the aspen leaf requires a very light air flow. On the other hand it has very good conditions for movement (these are mentioned above). They are quite visible in the photo below and in the real leaf.

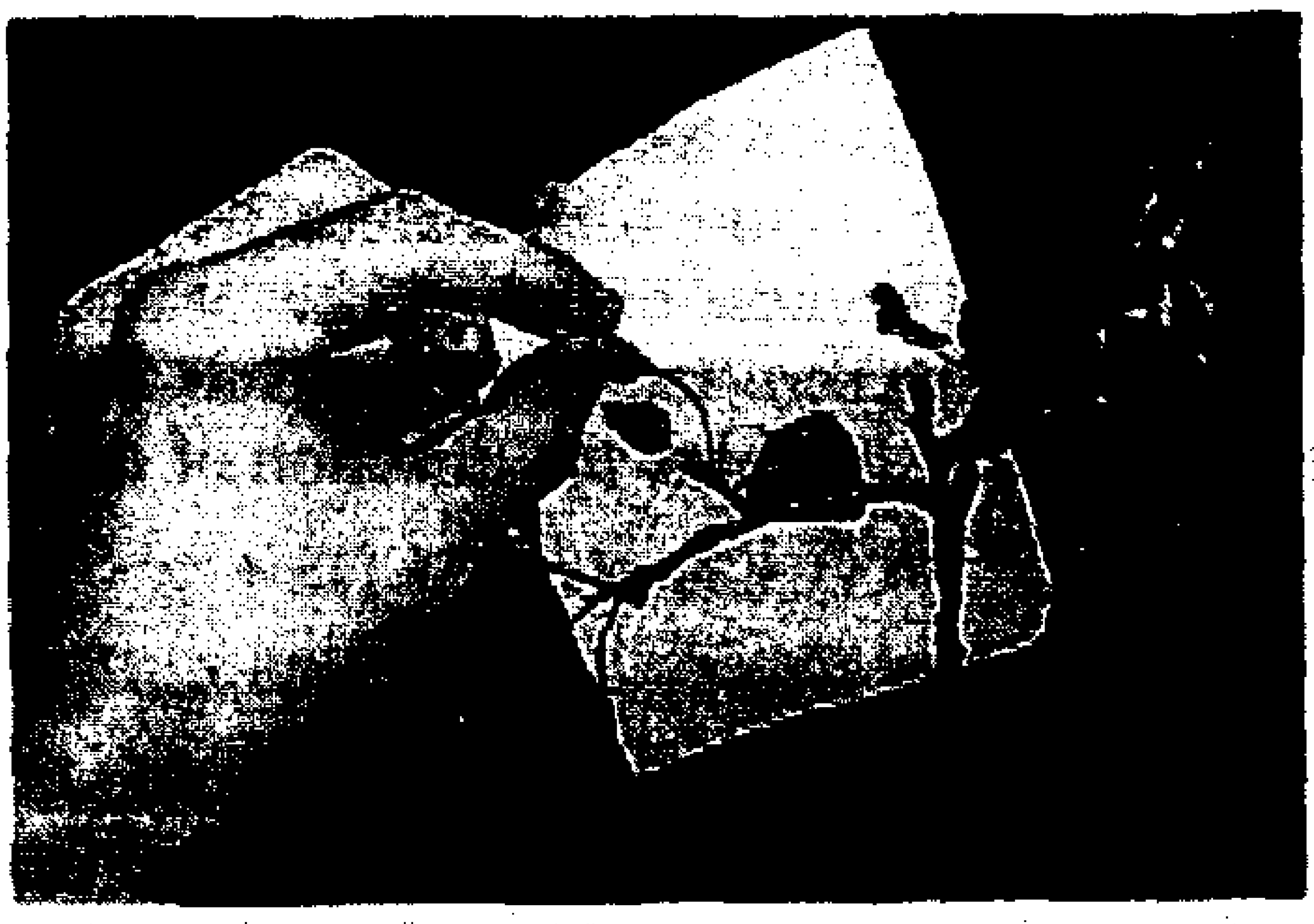

Fig. 14 Detailed photo of an aspen leaf

Problem No.8 - Superball

INTRODUCTION

This task seems to be one of the most interesting. We solved it theoretically – we produced two physical models of the situation.

The ball has mass $m = 8.23$ g, diameter $R = 2.6$ cm and we dropped it from the height $h_0 = 5$ cm.

SOLUTION

We were interested in the question of when we reach the most jumps, i.e. when the base is as hard as possible .

The ball rebounds from the hard base and jumps to the height h_1, falls, jumps again to the height h_2 etc.

The ratio between the rebound velocity and the impact velocity is called the *coefficient of restitution*, the symbol used for it is r (according to the literature).

$$r = \frac{v_1}{v_2} = \frac{\sqrt{2gh_1}}{\sqrt{2gh_2}} = \frac{\sqrt{h_1}}{\sqrt{h_2}}$$

where h_1 is the height of the fall and h_2 is the height of the jump

This coefficient shows the extent of energy loss by impact. All energy lost during the action is covered here.

We didn t consider in any of our calculations, resp. the coefficient of restitution does involve following phenomena: air friction, elastic waves extending in the ball during the jump and deformation caused by them, the irregular shape of the ball – it hasn't the proper shape of a sphere.

The height h_{n+1} is obviously smaller than h_n. Then the amplitude of jumps becomes smaller. The question is when it stops jumping. A certain amount of energy won't be sufficient for the rebound of the ball. How can we find this boundary?

The ball is deformed with every fall. After a short time, the ball returns to its original shape, resilience, the energy of elasticity changes back to kinetic energy and the ball rebounds. We can mathematically determine the boundary of maximum deformation from the energy, that the ball has when its center of gravity is at the lowest point. We consider the loss of energy during the deformation is as big as the loss during the extention, what is an approximation. The conclusion: the energy at the moment of maximum deformation is $(E_{n+1} + E_n)/2$.

The ball stops jumping as soon as the energy is just enough to lift the center of gravity y higher, the energy is equal to m.g.y. Then jumping turns into vibration.

We would like to explain, why we can ignore the air friction effect. We calculated the ratio between the energy losses caused by air friction and by the coefficient of restitution within first jump and because the decreasing power is proportional to v^2, it means, these losses are the highest ones.

The calculation:

c	$= 0.6$	coeff. of ball's aerodyn. resistance
ρ	$= 1.2759\ kg.m^{-3}$......	density of air (overastimated)
m	$= 0.00823\ kg$	ball's mass
r	$= 0.026\ m$	ball's perimeter
h_0	$= 0.05\ m$	
h_1	$= 0.0405\ m$	
S	$= \pi r^2$	effective cross-section
g	$= 9.81\ m.s^{-1}$	
$v(h)$	$= \sqrt{2gh}$	impact velocity

$F(v) = \frac{1}{2}cS\rho v^2$ .... decereasing power caused by friction

$L = \int F(x)\,dx$ .... energy losses

$$Lair = \int_0^{h_0} \frac{1}{2}cS\rho v^2(h)\,dh + \int_0^{h_1} \frac{1}{2}cS\rho v^2(h)\,dh$$

$$Lrest = mg(h_0 - h_1)$$

$$ratio = \frac{Lair}{Lrest}$$

Lair .... losses caused by air friction

Lrest .... losses caused by coeff. of restitution

$ratio = 1.076\ \%$

We see, this ratio is so small (remembering overastmation of c), that we can neglect air friction effects completely.

1. Experiments

In the experimental part we measured the coefficient of restitution first.

We took photos of the first two jumps with the camera that with an open shutter. The coefficient of restitution can be determined from the proportion of the heights we have photographed.

The conclusion of our experiments is: the coefficient of restitution is approx. $r_0 = 0.85$ (for the impact velocity 1 m . s^{-1}).

Then we tried to measure the number of jumps by digitalisation of the microphone signal. The microphone was placed under the china cup. The cup is very suitable for this purpose because it has big mass and solidity. The used cap had a concave bottom, so the ball didn't run away.

At first, we tried to digitalise the signal with the help of the sound card "Sound Blaster" that enables 8 bit sampling. We used Windows (Sound Secorder, Pocket Utility). Due to the low sensibility of earphone we had to use preamp. But we didn't gain satisfactory results. We were forced to use ISES (Inteligent School Experimental System). Using ISES we attained interesting results but we couldn't count all the jumps because of noise in the last part of jumping.

Although, we were able to measure the time of jumping, it was 4.1 ÷ 4.6 s, as it's visible from our experiments.

2. Theoretical models

a) model 1 – the mass point on the spring

The first approximation we used was based on following assumptions:

I) The coefficient of restitution is constant (for each ball), independent on the impact energy. In that case we can consider that the energy r^2E_n is left from the starting energy.

II) The force for the compression of the ball is directly proportional to this compression y. The ball is placed by the spring with certain rigidity. Its length equals the perimeter of the ball.

Consequences:

The height: $h_{n+1} = r^2 h_n \Rightarrow h_n = h_0 r^{2n}$

The time: $T_n = 2\sqrt{2h_n/g} \Rightarrow T_n = 2\sqrt{2h_0/g}\, r^n$, h_n and T_n are max. height and time of n-th jump.

The assumption II implies: $F = -ky$, where k is the ball's rigidity. The results of our experimental measurment is: $k = 3 \div 4$ kN . m^{-1}.

Max. depression we can determine from $E = \frac{1}{2}ky^2$.

$y = \sqrt{2E/k}$, where

$E = mgh_n(1 + r^2)/2$ ... ball s energy in the time of max. depression.

If the rebound energy is sufficient just for lifting the ball y higher, jumping turns into the vibrations.

$$
\begin{aligned}
E_{n+1} &= E_p \\
mgh_n r^2 &= mgy \\
h_n r &= mgh_n(1 + r)/k \\
h_n r^4 &= mg(1 + r^2)/k \\
h_n &= \frac{mg(1+r^2)}{kr^4} \\
\ln(h_n) &= \ln(h_0 r^{2n}) \\
n &= \frac{\ln(h_n) - \ln(h_0)}{2\ln(r)}.
\end{aligned}
$$

It is the minimal height of the jump. After the fall from this height the ball stops jumping and begins to vibrate. But we are not interested in this height, but we would like to know the number

The rigidity of the ball was determined experimentally. We pressed the ball by height and measured the depression. The rigidity was (3500 ± 500) N . m^{-1}.

We will get for our values: $h_n \doteq 52$ m, $n \doteq 60$ jumps.

This model looks very nice and accurate, but it doesn't work. We found, that the experimentally found time of jumping was not equal to the time determinated by our model. The main mistake was assumption that the coefficient of restitution is constant with the time. Its real dependence on the impact velocity is showed at the graph. The coefficient of restitution closes to 1 with lowering impact velocity.

It is not true that the force needed for y pressing the ball is directly proportional to y. Hertz s task says that the force grows with $h^{3/2}$.

These facts forced us to make a more complicated and proper model.

b) model 2

Model 2 is based upon following assumptions.

I) The coefficient of restitution is not constant, but in our interval of impact velocities ∈ < 0, 1 m . s^{-1} > could be approximated with a line segment between points [0, 1] and $[1, r_{h_0}]$, where r_{h_0} is a coefficient of restitution by impact of the height h_0. We follow the graph showing the dependence of coefficient of restitution on the impact velocity.

II) Compressing of the ball is described as compression of the sphere, i.e. force grows with $h^{3/2}$ according to Hertz s theory.

The mathematical expression of the assumption I is:

$r = (r_{h_0} - 1)v + 1$, where $\underline{v}$ is the impact velocity.

Note: We substitute $a = r_{h_0} - 1$.

The dependence between the jumps height and the previous one is:

$h_{n+1} = h_n r_n^2$, substitution:

$$h_{n+1} = h_n(a\sqrt{2gh_n} + 1)^2 \qquad (1)$$

The next height:

$$h_{n+2} = h_n(a\sqrt{2gh_n} + 1)^2[a\sqrt{2gh_n}(a\sqrt{2gh_n} + 1)]^2$$

The length of the equation grows with each successive height. It is not possible to derive a general relation from the recurent relation (1), as was possible in model a. We tried to substitute the line segment with some other curve as a part of parabola or exponentional curve and so on. All the possibilities lead to useless recurent relations.

Therefore we had to use numerical methods and a computer. We wrote a program which calculates the jumpes for both models numerically. It determines the ball's deformation corresponding to the coefficient of restitution, the duration of the jump, the duration of the deformation and overall time of jumping. Let's calculate the deformation y. We assume that the energy in the time of the maximal deformation is equaled to a mean of energies before and after impact because the ball's energy is changing into heat fluently.

$$F = -ky^{3/2},\ W = \int F\,dy,\ W = 2/5\,y^{5/2}$$

$$y = \left[\frac{5W}{2k}\right]^{\frac{2}{5}} = \left[\frac{5mg}{4k}(h_n + h_{n+1})\right]^{\frac{2}{5}}$$

In this case, we had to determine the constant k more accurately than in the first case. Using the Hertz's theory again:

$y = F^{2/3}[D^2(1/R + 1/R')]^{1/3}$, where R is first's and R' the second's ball perimeter (Hetz's theory is about two ball's impact, in our case $1/R'$ equals 0).

$$D = \frac{3}{4}\left(\frac{1-\sigma^2}{E} + \frac{1-\sigma'^2}{E'}\right)$$

where σ and σ', resp. E and E' are the Poisson's ratios resp. moduluses of elasticity of materials from which the balls are made. In our case, the primed values are used for the pad and normal ones for the ball. Neccessary values are shown in following table:

Material	σ	E[MPa]
Ball	0.48	1.1
China Pad	0.23	58,000

We can write: $y = F^{2/3}[D^2/R]^{1/3}$,

$$F = h^{3/2}\sqrt{R/D},$$

$$F = h^{3/2}k,$$

$$k = \sqrt{R/D}.$$

Value of k expressed numerically: $k = 217000\ \mathrm{N.m^{-3/2}}$

How to determine the time of ball's deformation? We used the Hertz's theory again.

$$t_d = 2\int_0^y \frac{dy}{\sqrt{v^2 - ky^{5/2}/m}} = 2\left(\frac{m^2}{k^2 v}\right)^{1/5}\int_0^1 \frac{dx}{\sqrt{1-x^{2/5}}}$$

where m is the ball's mass, v is the impact velocity, k is a constant equaled to $4.\sqrt{R/5D}$, y is the ball's compression.

This formula was numerically integrated and then used in our program.
When we solved this problem by computer, we get these results:

$n = 143$ jumps

$t = 4.608$ s

SUMMARY

We've done several experiments. We determined the ball's coefficient of restitution when the impact velocity was $1\ \mathrm{m.s^{-1}}$, we tried direct measuring of ball's jumps, but the last ones got lost in noise.

We found out, the coefficient of restitution was 0.85 and the ball jumps for 4.1/4.6 s.

The theoretical part of solution consists of two physical models of real situation. The first one is quite simple and the approximations are too rough. The second one uses the Hertz's theory and can not be solved analytically, we have to use numerical methods and computer. In this model we also include the time of contact (about 12% of overall time). Using this model, the theory was cofirmed by the experiment. This model says, the ball rebounds about 140 times.

The difference between the experiments and our second model are small and they could be explained by inaccuracies that arose while we were measuring the coefficient of restitution and by used approximations.

```
program HOPIK;
uses Printer, Graph, Crt, GDFSoft1, Strings;
const
  h0      = 0.05;
  rm      = 0.013;
  rh0     = 0.85;
  g       = 9.81;
  alfa    = 1 - 0.119;
  a       = rh0 - 1;
  m       = 8.23e-3;
```

```
  k        = 2.17e5;
  kp       = 3500;
  MaxN     = 300;
  Px       = 10;
  Py       = 10;
  ScreenX = 639;
  ScreenY = 479;
  Barvy : array[1..9] of byte = (white, lightred, lightgreen,
lightblue, yellow, lightgray, green, blue, black);

var
  h, hp, r : double;
  y, yp    : double;
  tt, t    : double;
  delta    : double;
  n        : word;
  ch       : char;

procedure Inic;
begin
  TextMode (259);
  h := h0;
  hp := h0;
  r := rh0;
  y := 0;
  yp := 0;
  tt := sqrt (2 * h0 / g);
  writeln (lst, 'Model B - coef. of restitut. grows from
value', rh0:3:1, ', H0 = ', h0:5:3, ' m');
  writeln (lst, 'K = ', k:6:0, ' N.m^2/3');
  writeln (lst);
  writeln (lst);
  writeln (lst);
  writeln;
  n := 1
end;

procedure Pis;
var
  s : array[1..6] of string;
begin
  str2 (H, 3, s[1]);
  str2 (Hp, 3, s[2]);
  str (y * 1000:6:4, s[3]);
  str2 (delta, 3, s[4]);
  str2 (t, 4, s[5]);
  str (tt:5:3, s[6]);
  writeln (lst, 'N:', n:3, ', r:', r * 100:3:1, ', H:', s[1],
(', Hp:', s[2],)', y:', s[3], ', t:', s[5], ', tt:', s[6], ',
delta: ', s[4]);
  CekejNaKlav
end;
```

```
procedure Kresli;
var
  x, k : word;
begin
  x := round (N / MaxN * ScreenX) + Px;
  PutPixel (x, round (ScreenY * (1 - R)), Barvy[3]);
  PutPixel (x, round (ScreenY * (1 - y / Rm) - py),
Barvy[5]);
  PutPixel (x, round (ScreenY * (1 - yp / Rm) - py),
Barvy[6]);
  PutPixel (x, round (ScreenY * (1 - Hp / H0) - py),
Barvy[4]);
  PutPixel (x, round (ScreenY * (1 - H / H0) - py), Barvy[2])
end;

procedure InicG;
var
  k : word;
begin
  InicGrafiky ('c:\tp\bgi');
  for k := 0 to ScreenX do
    PutPixel (k, ScreenY - Px, Barvy[1]);
  for k := 0 to ScreenY do
    PutPixel (Px, k, Barvy[1]);
  Kresli
end;

begin
  repeat
    write ('Kreslit [k] / Psat [p]? ');
    readln (ch)
  until ch in ['k', 'K', 'p', 'P'];
  Inic;
  if ch in ['k', 'K'] then InicG;
{  for n := 1 to MaxN do
  begin}
  repeat
    hp := h0 * exp (n * ln (alfa));
    tt := tt + t;
    r := a * sqrt (2 * g * h);
    h := h * sqrt (r);
    t := 2 * sqrt (2 * h / g);
    y := exp (2 / 5 * ln (5 / 4 * m * g / k * h * (1 + sqrt
(a * sqrt (2 * g * h) + 1))));
    yp := sqrt (2 * m * g * hp / kp);
    delta := y - h * sqr (a * sqrt (2 * g * h) + 1);
    if ch in ['k', 'K'] then Kresli
                         else Pis;
    inc (n);
  until delta > 0;
{  end;}

  CekejNaKlav
end.
```