2006-474: ENHANCING STUDENT UNDERSTANDING OF MECHANICS USING SIMULATION SOFTWARE

Arnaldo Mazzei, Kettering University
Arnaldo Mazzei is an Associate Professor of Mechanical Engineering at Kettering University. He received his Ph.D. in Mechanical Engineering from the University of Michigan in 1998. He specializes in dynamics and vibrations of mechanical systems and stability of drivetrains with universal joints. His current work relates to modal analysis, stability of drivetrains, finite element analysis and CAE. He is a member of ASME, ASEE and SEM.

Richard Scott, University of Michigan
Richar A. Scott received his Ph.D. in Engineering Science from The California Institute of Technology. He is a Professor of Mechanical Engieering at the University of Michigan, Ann Arbor. He has obtained a teaching award from the College of Engineering and was selected as professor of the semester four times by the local chapter of Pi-Tau-Sigma.

Enhancing student understanding of mechanics using simulation software

Abstract

The wide availability of commercial codes, such as MAPLE®, has made it possible to significantly enhance the teaching of undergraduate courses in mechanics. New problems, usually non-linear, can be introduced which previously could not be treated because of the lack of analytic solutions. By means of numerical solutions to these problems, students can get a feel for finite difference approaches and, perhaps more importantly, their physical understanding can be enhanced and new phenomena explored. The following examples are presented with the underlying equations cast, as much as possible, into non-dimensional form:
(i) a finite difference scheme, (ii) a non-linear pendulum subjected to various initial conditions, showing how the period depends on the amplitude, (iii) a non-linear softening spring showing the existence of instabilities, (iv) the stability of an inverted pendulum restrained by a spiral spring, illustrating the existence of multiple equilibrium states and their stability and (v) a numerical simulation of a sweep test (forced motion of a single-degree-of-freedom system in which the forcing frequency varies with time), showing that if the sweep rate is too fast, no resonances will be observed.

Introduction

The use of software as a teaching aid in undergraduate mechanical engineering courses has been discussed by several authors. A common type used is mathematics software which allows for a wide range of applications from basic to advanced engineering courses. MATLAB®, MATHCAD® and MAPLE® [1] are some common examples.

In a previous paper [2]one of the authors (Mazzei) discussed his experiences on using commercially available simulation software for dynamics teaching and improving learning. This was done using MSC ADAMS® for rigid body dynamics. Feedback from students showed that simulations can help visualize and understanding of mechanical systems dynamic behavior.

There are numerous papers on the use of software in engineering undergraduate courses. For example in reference[3] Gharghouri discusses his experiences on using MAPLE® for teaching a numerical analysis course. An approach to calculate eigenvalues and eigenvectors for a mechanical vibrations course is given and possible advantages of using the software instead of programming languages is discussed. It was concluded that the use of the software greatly enhanced the delivery of the numerical analysis course.

Another example of a positive use of MAPLE® as a teaching aid can be found in reference[4] where Gerber discusses how the software aided in teaching circuits and systems.

In this paper several examples developed utilizing MAPLE® are presented. The goal is to improve student understanding of mechanical vibrations and dynamics by investigating non-linear physical phenomena and resonance with the aid of the software.

MAPLE® Examples

Finite difference scheme

In the examples that follow the solutions are obtained using a MAPLE® finite difference scheme. To give students some sense of the finite difference process a simple example is solved using the Euler method.

Consider the problem of a falling solid sphere resisted by a quadratic drag force: cv^2. The equation of motion is:

$$\frac{dv}{dt} = g - \frac{c}{m}v^2 \equiv f(v) \tag{1}$$

If $v = 0$ at $t = 0$, the exact solution (from integral tables) is:

$$v = \sqrt{\frac{mg}{c}}\tanh\{(\sqrt{\frac{cg}{m}})t\} \tag{2}$$

In the Euler method, the time range of interest is divided into uniform intervals of magnitude h. The solution at time t_{i+1} is approximated by a projection using the slope at t_i, and replacing the derivative by $(v_{i+1} - v_i)/h$ (see FIGURE 1). This leads to:

$$v_{i+1} = v_i + hf(v_i) \tag{3}$$

Consider an aluminum sphere (density $2700Kg/m^3$) of radius $0.1m$ falling in an oil (density $900Kg/m^3$). The coefficient c is obtained from the formula:

$$F_D = \frac{1}{2}c_D A\rho_{oil}v^2 = cv^2 \tag{4}$$

where c_D is taken to be 0.47. A is the projected area and equals πR^2. This gives $c = 6.645$ (Kg/m). Then equation (3) may be written:

$$v_{i+1} = v_i + h(9.81 - 0.5875v_i^{\,2}) \tag{5}$$

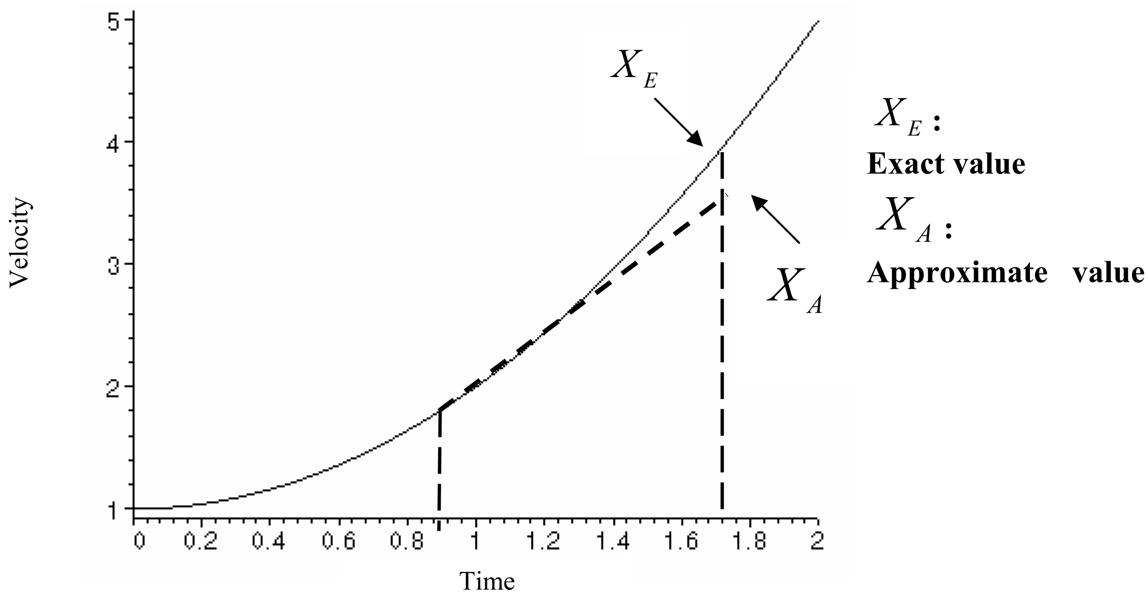

FIGURE 1 – FINITE DIFFERENCE SCHEME

The values of v_{i+1} given by equation (5) for $h = 0.5$ are plotted in FIGURE 2 together with the exact solution. Excellent agreement between the two solutions is seen, the maximum error being of the order of 10%.

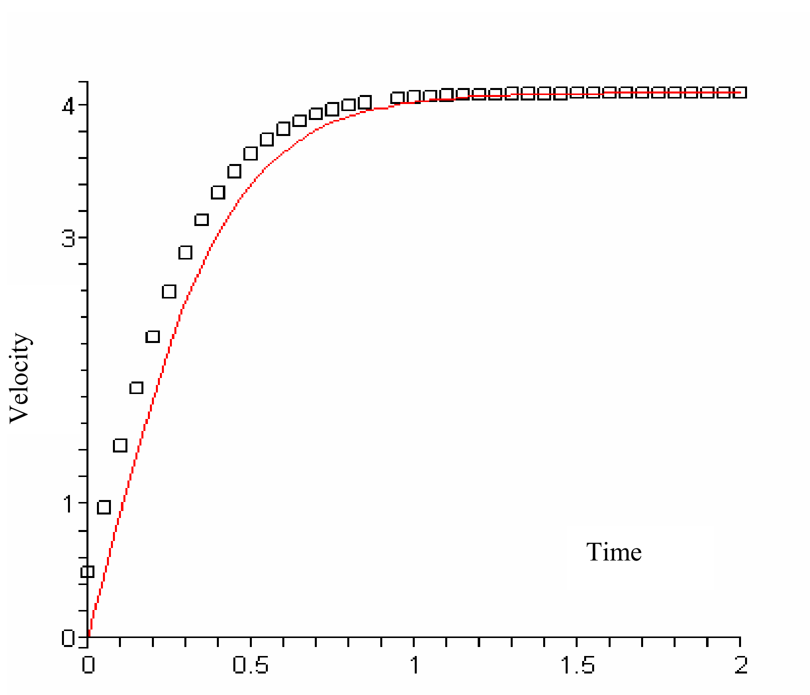

FIGURE 2 – FINITE DIFFERENCE AND EXACT SOLUTION

Non-linear Pendulum

The equation of motion for a simple pendulum is (see FIGURE 3):

$$\frac{d^2\theta}{dt^2}+\frac{g}{l}\sin(\theta)=0$$

(6)

where g is the gravitational constant, l is the length of the pendulum and θ is an angular coordinate. Noting that $\sqrt{g/l}$ has dimensions of frequency, one can introduce the dimensionless time: $\tau=(\sqrt{g/l})t$, so that $d/dt=(d/d\tau)(d\tau/dt)=(\sqrt{g/l})(d/d\tau)$.

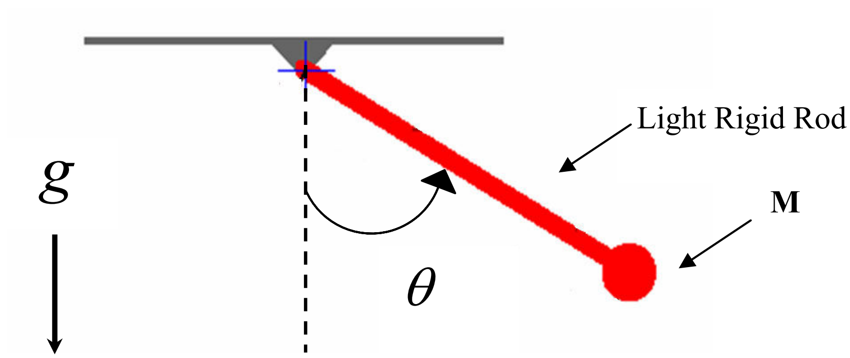

FIGURE 3 – SIMPLE PENDULUM

Then equation (6) becomes:

$$\frac{d^2\theta}{d\tau^2}+\sin(\theta)=0$$

(7)

For small motions $\sin(\theta)\approx\theta$ and the equation is linear. The dimensionless period is, regardless of initial amplitude, $T_L=2\pi$.

For large motions equation (7) is non-linear and is solved numerically for various initial conditions using MAPLE® (initial position is varied, initial velocity is set to zero in all cases; see appendix for code worksheet).

The initial angular displacement is increased in steps of 0.1 rad and the responses are shown in FIGURE 4. For reference purposes the first curve (the smallest initial angle) is highlighted in the plot; this response coincides with the one obtained from a linear approach for the interval studied. As can be seen the period increases as the initial angle grows.

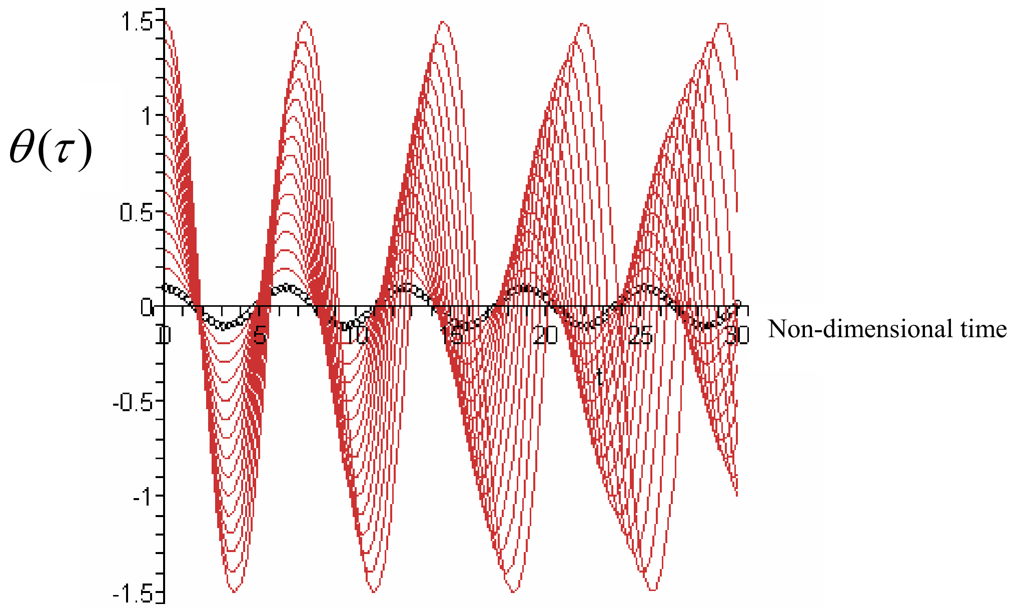

FIGURE 4 – PENDULUM RESPONSES TO DIFFERENT INITIAL CONDITIONS

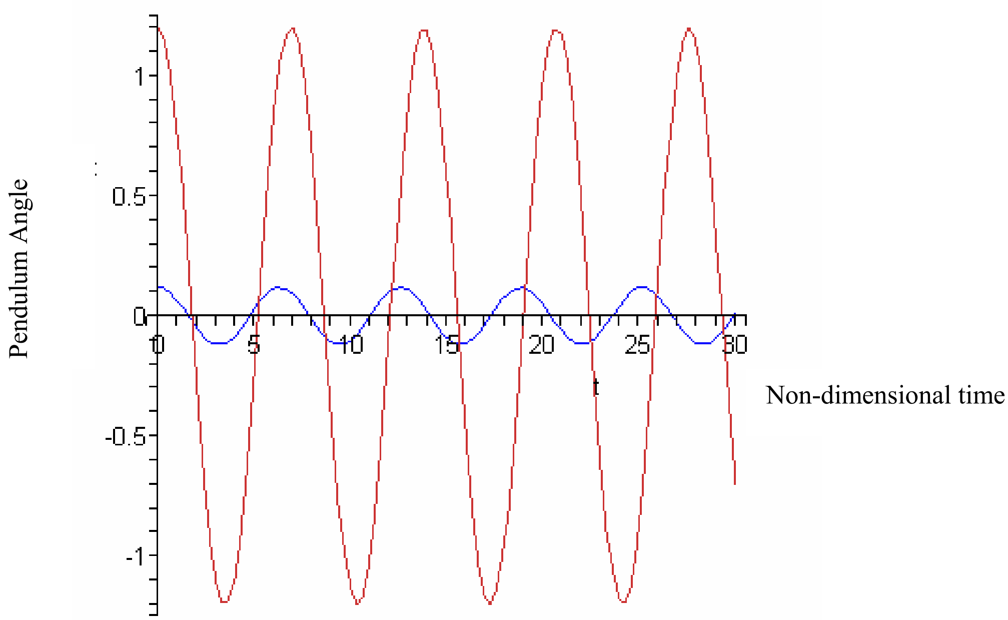

FIGURE 5 – RESPONSES FOR SMALL AND LARGE INITIAL ANGLES

A difference of about 10% between the "linear" period and the non-linear period occurs when the initial angle reaches a value of about 1.2 rad (approximately 70 degrees). When the initial angle reaches a value around 1.55 rad (approximately 89 degrees) the difference is about 17%. FIGURE 5 shows a comparison between the pendulum responses for both a small and large initial angle. For this case the large angle is about ten times larger than the small one. It can be seen in the figure that the period for "large" response is distinctly different from the "small" one showing the dependence of the period on the initial condition. Students should note that for a non-linear system the period is not a system characteristic. It depends on the initial amplitude.

These results can be explained intuitively as follows. In the range $0 \le \theta \le \pi/2$, θ is greater than $\sin(\theta)$. Hence the effective restoring force in the non-linear case is less than that in the linear case. The motion slows down; the period is larger.

Stability of a mass-non-linear-spring oscillator

The next example illustrates the phenomenon of instability. Consider a mass restrained by a non-linear softening spring sliding along a straight line on a smooth horizontal surface. The spring force is given by:

$$F_s = Kx - K_1 x^3, \quad K, K_1 > 0 \tag{8}$$

The non-linear differential equation of motion is:

$$m\ddot{x} + Kx - K_1 x^3 = 0 \tag{9}$$

The question at hand is how does the system behave subjected to various initial conditions?

Introducing the dimensionless time $\tau = (\sqrt{K/m})t$, equation (9) becomes:

$$\frac{d^2 x}{d\tau^2} + x - \frac{K_1}{K} x^3 = 0 \tag{10}$$

For $K_1/K = 0.1$, the dimensionless spring force is shown in FIGURE 6.

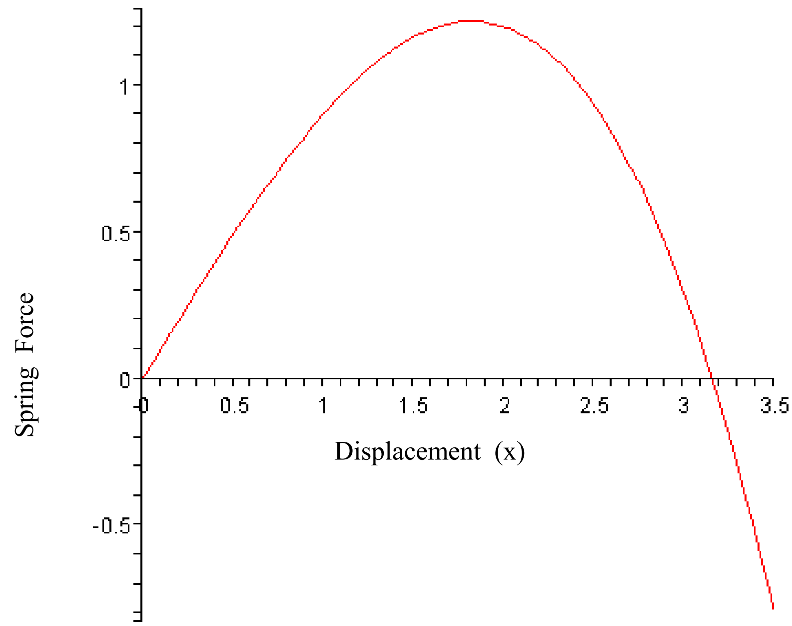

FIGURE 6 – DIMENSIONLESS SPRING FORCE FOR NON-LINEAR OSCILLATOR

First, equation (10) is solved subjected to the initial conditions: $x(0) = 0.01$, $dx(0)/d\tau = 0$. For this small initial displacement, the spring force is positive and the response of the system should be stable. The results shown in FIGURE 7 verify this.

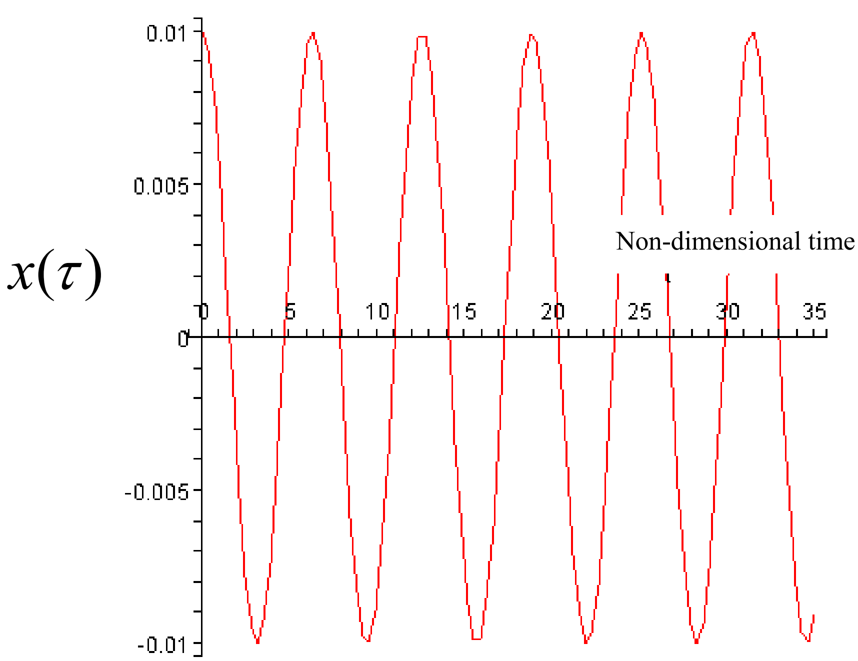

FIGURE 7 – RESPONSE FOR SMALL INITIAL DISPLACEMENT

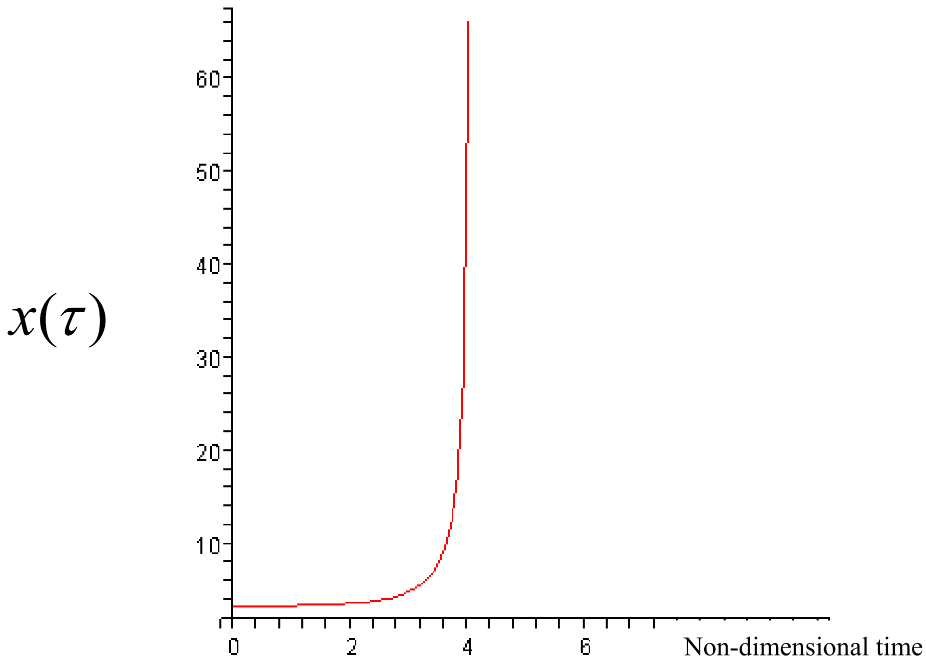

FIGURE 8 – RESPONSE FOR LARGE INITIAL DISPLACEMENT

Consider now the case where the initial displacement is large, for instance, $x(0) = 3.20$ (and $dx(0)/d\tau = 0$). Now the spring force is negative and one would anticipate unstable response, as indeed seen in FIGURE 8. This further illustrates that for non-linear systems, the response can change radically for different initial conditions.

Stability of an inverted pendulum

In the next example, concepts of stability and instability will be further explored and highlighted.

Consider an inverted rod, pinned at its lower end acted upon by gravity and a torsional spring which always exerts a restoring torque $G\theta$ (G - spring stiffness), regardless of how large θ is (FIGURE 9).

The equation of motion is:

$$J\ddot{\theta} - mg\frac{l}{2}\sin(\theta) + G\theta = 0 \tag{11}$$

Where J is the mass moment of inertia of the rod about the pin and m and l are the mass and length of the rod, respectively.

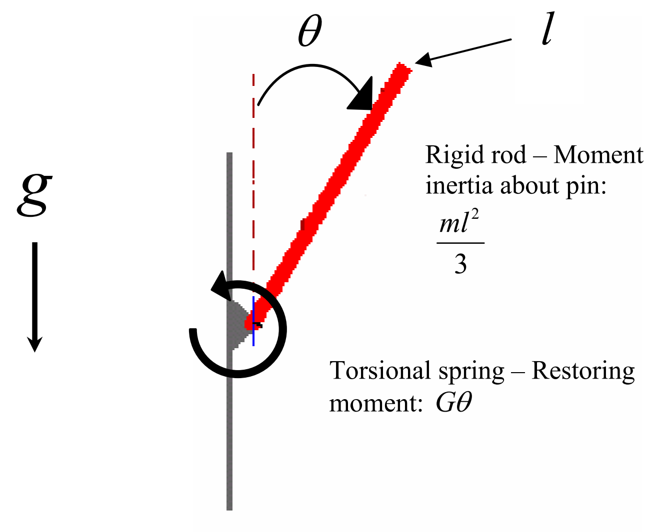

FIGURE 9 – INVERTED PENDULUM

The equilibrium states for this system are given by setting $\ddot{\theta} = 0$:

$$mg\frac{l}{2}\sin(\theta_E) = G\theta_E$$

(12)

which has solutions (i) $\theta_E = 0$ and (ii) $\sin(\theta_E) = (2G/mgl)\theta_E$. The constant G has dimensions of torque and one can set $G = nmgl$, where n is an arbitrary number. Thus $\sin(\theta_E) = 2n\theta_E$, which for $n = 0.49$ has a root $\theta_E = 0.35 rad$ $(\approx 20^\circ)$.

An interesting question is whether the states given by solutions (i) and (ii) are stable.

One can address this by a perturbation approach, i.e., set $\theta = \theta_E + \varepsilon$, $\varepsilon << 1$; then equation (11) gives:

$$J\ddot{\varepsilon} - mg\frac{l}{2}\sin(\theta_E + \varepsilon) + G(\theta_E + \varepsilon) = 0$$

Using a trigonometric relation and noting that $\sin(\varepsilon) \approx \varepsilon$ and $\cos(\varepsilon) \approx 1$, one obtains:

$$J\ddot{\varepsilon} - mg\frac{l}{2}\sin(\theta_E) - mg\frac{l}{2}\cos(\theta_E)\varepsilon + G(\theta_E + \varepsilon) = 0$$

Using solution (ii), one finds:

$$J\ddot{\varepsilon} + (G - mg\frac{l}{2}\cos(\theta_E))\varepsilon = 0$$

(13)

Setting $G = 0.49mgl$, equation (13) gives:

$$J\ddot{\varepsilon} + mgl(0.49 - \frac{1}{2}\cos(\theta_E))\varepsilon = 0$$

(14)

Case $\theta_E = 0$

Equation (14) becomes:

$$J\ddot{\varepsilon} - mgl(0.01)\varepsilon = 0$$

This has solutions of the form $\exp(\pm\sqrt{(0.01mgl/J)}t)$ which predicts growth with time; $\theta_E = 0$ is unstable. What ultimately happens can not be predicted by equation (14), since the linearization process breaks down. One must address this issue by solving the non-linear equation (11),which has to be done, usually, numerically.

Case $\theta_E = 0.35rad$

Equation (14) becomes:

$$J\ddot{\varepsilon} + mgl(0.0199)\varepsilon = 0$$

This has solutions of trigonometric form and the state is stable (bounded).

Introducing the non-dimensional time $\tau = (\sqrt{g/l})t$, setting $G = 0.49mgl$ and $J = \frac{ml^2}{3}$, equation (11) becomes:

$$\frac{d^2\theta}{d\tau^2} - \frac{3}{2}\sin(\theta) + 1.47\theta = 0$$

(15)

A plot of the numerically obtained response for θ as a function of the non-dimensional time is given in FIGURE 10. The initial conditions were taken to be $\theta(0) = 0.01$ (which is in the vicinity of the unstable state) and $d\theta(0)/d\tau = 0$. It is seen that the motion initially grows but finally reaches a bounded oscillatory state about the equilibrium point (solution (ii)). The oscillations are not symmetric about the equilibrium state, which can be explained by noting that for angles smaller than $0.35rad$ the "restoring force" term in equation (15) is weaker than for large values of the angle, leading, consequently, to larger amplitudes (see FIGURE 11 below).

A plot of the "restoring force" is given in FIGURE 11. Note that the term is negative for angles smaller than $0.35rad$, which could cause unstable response of the system described by equation (15). However when the angle grows to larger values the term becomes positive leading to bounded oscillations.

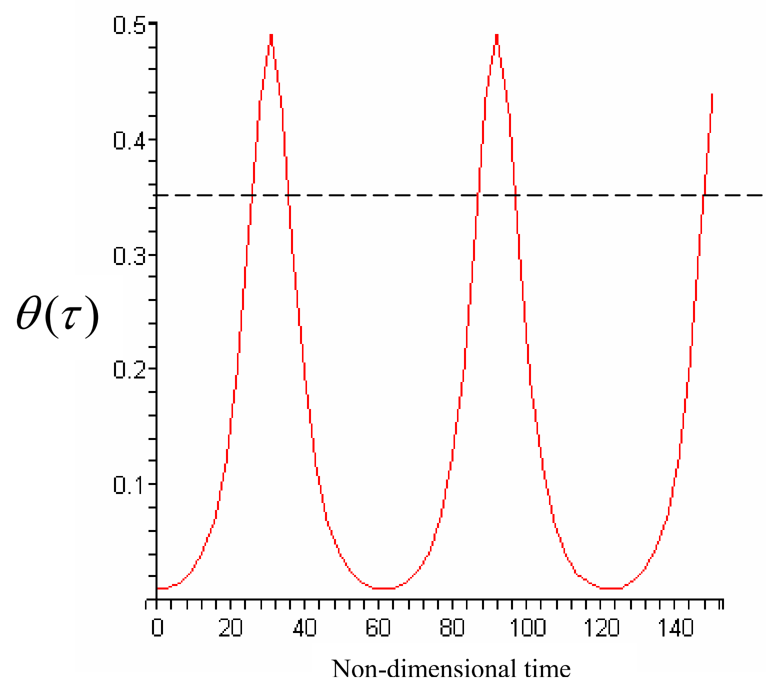

FIGURE 10 – INVERTED PENDULUM RESPONSE

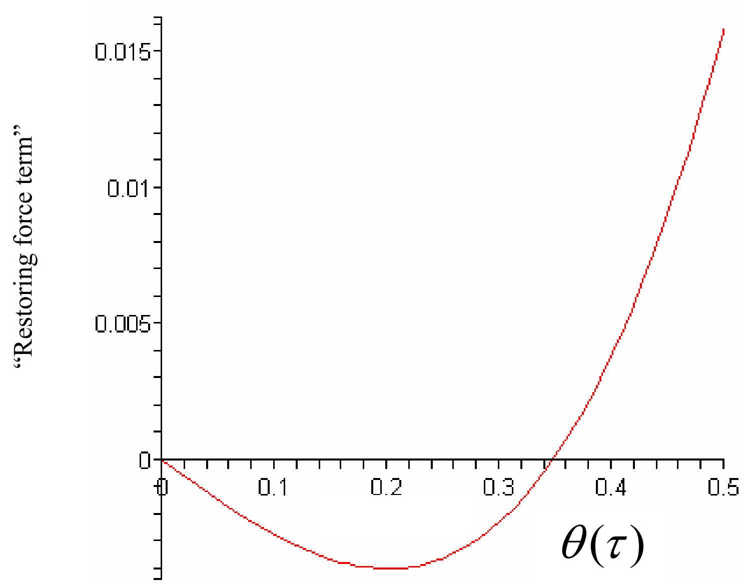

FIGURE 11 – RESTORING FORCE TERM FOR INVERTED PENDULUM

Forced oscillator with time-dependent frequency excitation

In determining resonant frequencies in practice frequently sweep tests are used, that is tests in which the applied forcing function has a frequency which continuously changes with time (see Lalanne, reference [5]). This shortens the duration of the test, but if the sweep rate is too high resonances may be missed. Intuitively this may be seen on looking at the forced motion solution exactly at resonance, that is, the solution to:

$$\frac{d^2x}{dt^2} + \omega^2 x = \frac{F_0}{m}\sin(\omega t)$$

MAPLE® gives the following solution:

$$x(t) = C_1 \sin(\omega t) + C_2 \cos(\omega t) - \frac{1}{2}\frac{\cos(\omega t)t}{\omega}$$

Note the term that grows linearly with time (t), at least for the undamped case, showing that resonance requires time to build up. For moderate damping one could anticipate similar problems.

To illustrate the fact that resonances may be missed, the following model problem is studied.

Consider a single mass-spring-damper system with an excitation that has a time-dependent frequency. The non-dimensional equation of motion for the system is given below.

$$\frac{d^2x}{d\tau^2} + c_1\frac{dx}{d\tau} + x = \sin(\nu(\tau)\tau) \tag{16}$$

Here $\tau = \omega t$ is a non-dimensional time, where ω is the undamped natural frequency, $c_1 = c/(m\omega)$ ($c =$ damping coefficient, $m =$ system mass) and x is a non-dimensional displacement (physical displacement divided by F_0/k, F_0 = force amplitude, k =spring stiffness). The issue pursued is how much time the frequency of the excitation must remain at the resonant value for resonance to be observed.

It is seen from equation (16) that when $\nu(\tau) = 1$ the system is being forced at its undamped natural frequency and large, but bounded, response is anticipated. This can be observed in FIGURE 12, which shows the response of the system (zero initial conditions) obtained using MAPLE®. The following numerical values were used: $m = 10Kg$, $k = 250N/m$, $\omega = 5rad/s$ and $c = 10Ns/m$ (corresponding to a damping ratio of approximately 10%). Note that in equation (16) the amplitude of the non-dimensional forcing function is 1 and it would cause a non-dimensional deflection of 1 if applied statically (hereafter referred to as the static deflection).

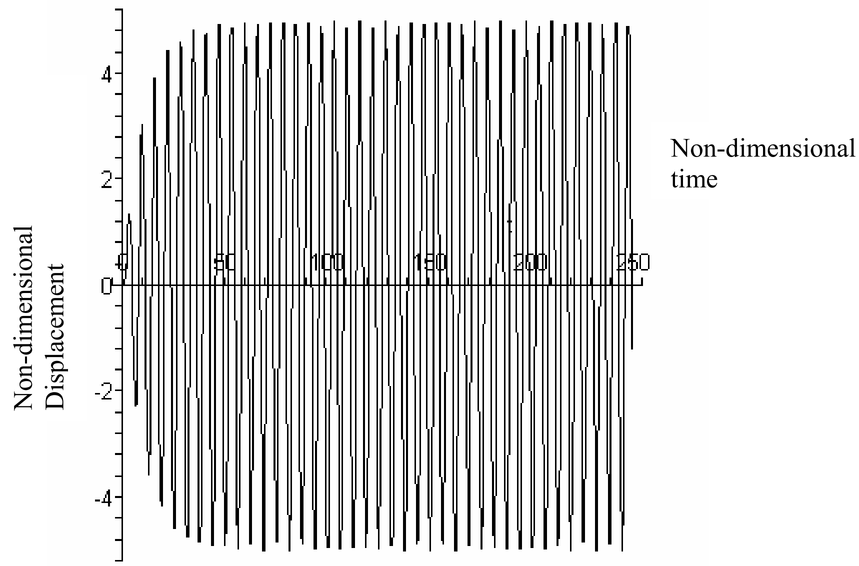

FIGURE 12 – RESPONSE AT RESONANCE CONDITION

Now, a non-constant profile for the excitation frequency is considered. The excitation is taken to follow a trapezoidal profile; starting from zero it reaches the resonant value, remains at this value for a certain amount of time and then returns to zero. A typical profile is shown in FIGURE 13 where the time spent at resonance (hereafter called "dwell time") is approximately 1/3 of the period of undamped oscillations (2π).

Shown in FIGURE 14 is the response for a dwell time of approximately 6 times the period of undamped oscillations. The peak amplitude is seen to be about four times the static deflection and one would say resonance has occurred (as verified by the peak amplitude seen in FIGURE 12).

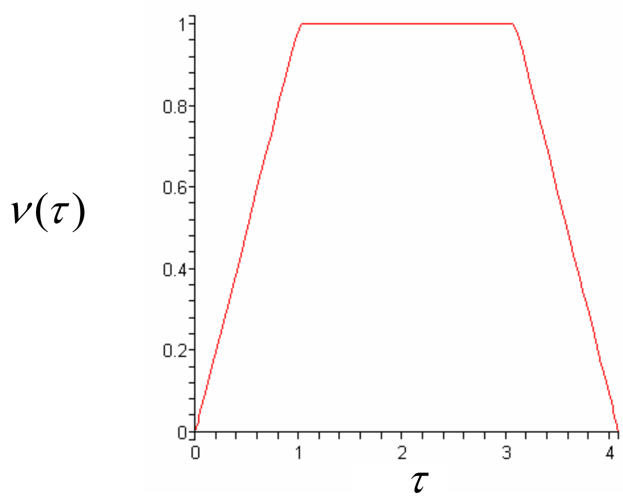

FIGURE 13 – EXCITATION TRAPEZOIDAL PROFILE

The response to an excitation for a dwell time of about 1/3 of the period is shown in FIGURE 15. It is seen that the peak amplitude is close to the static deflection and the result would not be interpreted as resonance. The sweep rate (dwell time) can not be too fast!

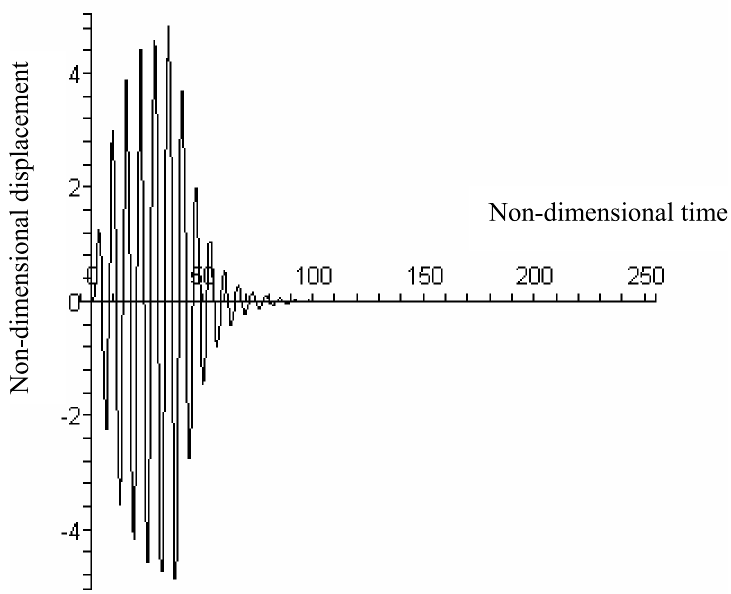

FIGURE 14 – RESPONSE TO TRAPEZOIDAL EXCITATION PROFILE I

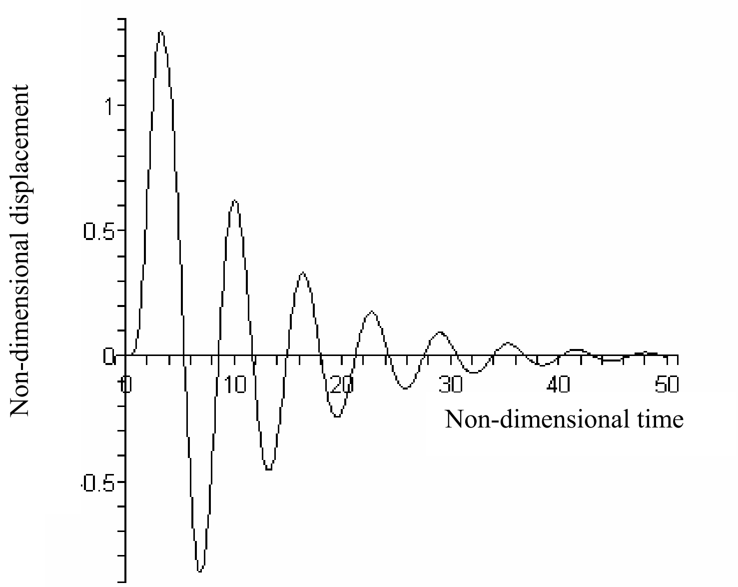

FIGURE 15 – RESPONSE TO TRAPEZOIDAL EXCITATION PROFILE II

Conclusions

In many institutions the required introductory courses now expose students to mathematics software such as MATLAB®, MAPLE®, etc. Advantage can be taken of this in beginning mechanics courses to explore some hitherto intractable problems which shed light on interesting and important dynamical phenomena. The examples presented were: (i) a comparison between some non-linear systems and their linearized counterparts; (ii) an exploration of the effect of time-dependent frequency on resonances; (iii) an illustration of the existence of multiple equilibrium states and their stability.

References

[1] www.maplesoft.com

[2] A. Mazzei, "Integrating simulation software into an undergraduate dynamics course: a web-based approach," *Proceedings of the 2003 American Society for Engineering Education Annual Conference & Exposition, Nashville - TN*, 2003.

[3] P. Gharghouri, "Integrating a computer algebra software into engineering curriculum: problem and benefits," *Proceedings of the 1998 American Society for Engineering Education Annual Conference & Exposition, Seattle - WA*, 1998.

[4] E. L. Gerber, "MAPLE for circuits and systems," *Proceedings of the 1998 American Society for Engineering Education Annual Conference & Exposition, Seattle - WA*, 1998.

[5] C. Lalanne, *Mechanical Vibration and Shock - Sinusoidal Vibration*: Hermes Penton Science Ltd, 2002.

Appendix

1)

```
> #finite_difference
> restart:with(linalg):with(plots):
> h:=.05;
> v(0):=0.0;
> for i from 0 to 40 do;
> v(1+i):=v(i)+h*(9.81-.5875*v(i)^2);c[i+1]:=[i*h,v(i+1)];
> end do;
> ve:=4.0861*tanh(2.4008*t);
> fig01:=plot(ve,t=0..2):
> display(fig01);
> subs(t=10,ve);
> simplify(%);
> subs(t=2,ve);
> simplify(%);
> fig02:=plot([c[1],c[2],c[3],c[4],c[5],c[6],c[7],c[8],c[9],c[10],c[11],
> c[12],c[13],c[14],c[15],c[16],c[17],c[18],c[20],c[21],c[22],c[23],c[24
> ],c[25],c[26],c[27],c[28],c[29],c[30],c[31],c[32],c[33],c[34],c[35],c[
> 36],c[37],c[38],c[39],c[40],c[41]],style=point,symbol=box,color=black)
> :
> display(%);
> plots[display]({fig01,fig02});
```

2)

```
> #non_linear_pendulum
> restart:
> evalf(convert(1.55,degrees));
> with(DEtools):with(plots):
> eq1:=diff(x(t),t,t)+sin(x(t))=0;
> ic1:=x(0)=.1,D(x)(0)=0;
```

```
> ic2:=x(0)=1.55,D(x)(0)=0;
> sol1:=dsolve({eq1,ic1},{x(t)},type=numeric,method=rkf45,output=listpro
> cedure);
> odeplot(sol1,[t,x(t)],0..10);
> plot1:=odeplot(sol1,[t,x(t)],0..30,color=blue,numpoints=100):
> yproc1:=> rhs(sol1[2]);
> t1:=fsolve(yproc1(x)=0, x=0..3);
> t2:=fsolve(yproc1(x)=0, x=3..6);
> t3:=fsolve(yproc1(x)=0, x=6..9);
> period1:=(t3-t1);
> sol2:=dsolve({eq1,ic2},{x(t)},>
type=numeric,method=rkf45,output=listprocedure);
> odeplot(sol2,[t,x(t)],0..10);
> plot2:=odeplot(sol2,[t,x(t)],0..30,color=orange,numpoints=100):
> yproc2:=rhs(sol2[2]);
> t1:=fsolve(yproc2(x)=0, x=0..3);
> t2:=fsolve(yproc2(x)=0, x=5..6);
> t3:=fsolve(yproc2(x)=0, x=8..10);
> period2:=(t3-t1);
> display([plot1,plot2]);
> difference:=(period2*100)/period1;
```

3)

```
> #non_linear_oscillator
> with(DEtools):
> with(plots):
> f:=x-.1*x^3;
> plot(f,x=0..3.5);
> eq:=diff(x(t),t,t)+x(t)-.1*x(t)^3=0;
> ic1:=x(0)=.01,D(x)(0)=0;
> ic2:=x(0)=2,D(x)(0)=0;
> ic3:=x(0)=3.2,D(x)(0)=0;
> sol1:=dsolve({eq,ic1},{x(t)},type=numeric);
> odeplot(sol1,0..9);
> sol2:=dsolve({eq,ic2},{x(t)},type=numeric);
> odeplot(sol2,0..35);
> sol3:=dsolve({eq,ic3},{x(t)},type=numeric);
> odeplot(sol3,0..9);
```

4)

```
> #inverted_pendulum
> with(plots):
> t1:=-sin(x)+.98*x;
> plot(t1,x=0..1);
> fsolve(t1,x=0..1);
> with(DEtools):
> eq:=diff(x(t),t,t)-sin(x)/2+.98*x(t)=0;
> ic1:=y(0)=.01,D(y)(0)=0;
> ic2:=x(0);
> t2:=sin(x)-.98*x;
> plot(t2,x=0..2);
> fsolve(t2,x=0..1);
> .347*180/3.14159;
> eq1:=diff(y(t),t,t)-3*sin(y(t))/2+1.47*y(t)=0;
> sol5:=dsolve({eq1,ic1},{y(t)},type=numeric);
> odeplot(sol5,0..200,numpoints=1000);
```

5)

```
> #sweep_test
> restart:
> with(linalg):with(plots):with(DEtools):
> eq10:=(diff(x(t), `$`(t, 2)))+c1*(diff(x(t), t))+x(t)-sin(nu(t)*t);
> k1:=(omega^2)/(omega0^2);
> c1:=(c0)/(m*omega0);
> #a1:=a01/(m*omega0^2);
> m:=10;k:=250;c0:=10;a01:=25;
> omega:=sqrt(k/m);
> eq10;
> eq20:=eq10;
> #
> omega0:=omega;
> #
> eq30:=(algsubs(sin(nu(t)*t)=0,eq10));
> eq40:=algsubs(diff(x(t),t$2)=A^2,eq30);eq40:=algsubs(diff(x(t),t)=A,eq
> 40);eq40:=algsubs(x(t)=1,eq40);
> zeta:=coeff(eq40,A,1)/(2*sqrt(coeff(eq40,A,0)/coeff(eq40,A,2)));
> zeta1:=c0/(m*2*omega);
> period:=evalf(sqrt(coeff(eq40,A,2)/coeff(eq40,A,0))*(2*Pi));
> damp_period:=evalf(period*sqrt(1-zeta^2));
> sol000:=dsolve({eq30,x> (0)=0.0873,D(x)(0)=0},{x(t)},type=numeric,
> method=gear,output=procedurelist);
> odeplot(sol000,[t,x(t)],0..omega0,numpoints=500,color=black,labels=["t
> ime","disp"]);
> odeplot(sol000,[t,x(t)],0..100*omega0,numpoints=500,color=black,labels
> =["time","disp"]);
> odeplot(sol000,[t,x(t)],0..10*omega0,numpoints=500,color=black,labels=
> ["time","disp"]);
> eq10;
> nu:=1;
> sol001:=dsolve({eq10,x(0)=0.0,D(x)(0)=0},{x(t)}, type=numeric,
> method=gear,output=procedurelist);
> #
> odeplot(sol001,[t,x(t)],0..50*omega0,numpoints=500,color=black);
> #
> nu:='nu';
> duration:=6.28/3;
> ######################################################################
> for i from 1 to 1 do
> nu(t):='nu(t)';
> eq20:
> a:=1;b:=a+duration;c:=a+b;
> nu(t):=1/a*t*Heaviside(t)*Heaviside(a-t)+Heaviside(t-a)*Heaviside(b-t)
> -1/a*(t-c)*Heaviside(t-b)*Heaviside(c-t);
> plot(nu(t),t=0..c);
> eq20;
> sol002:=dsolve({eq20,x(0)=0.0,D(x)(0)=0},{x(t)}, type=numeric,
> method=gear,output=procedurelist):
> odeplot(sol002,[t,x(t)],0..10*omega0,numpoints=500,color=black,labels=
> ["time","disp"]);odeplot(sol002,[t,x(t)],0..50*omega0,numpoints=500,co
> lor=black,labels=["time","disp"]);
> duration:=duration+1.884;nu(t):='nu(t)';
> end do;
> ######################################################################
```