```
function doublependulum1
%simulation of a double pendulum - chaotic motion
%Hugh Hunt, Nov 2005
%www.hughhunt.com

global M1 M2 I1 I2 B1 B2 AA AL G AX
%primary pendulum:
      M1=5;                          %mass
   I1=M1*0.1^2;          %moment of inertia about G1
   B1=0.1*sqrt(2);       %distance to G1
   AA=0.25;                    %distance to A, the pivot of secondary pendulum
   AL=3*pi/4;                  %angle between OG1 and OA
%secondary pendulum:
      M2=2;                          %mass
      I2=0;                          %moment of inertia about G2
   B2=1.8;                           %distance to G2
G=9.81;                              %g

%equilibrium angles theta1 and theta2:
t1=-atan(M2*AA*sin(AL)/(M1*B1+M2*AA*cos(AL)));
t2=0;
t1=t1+pi;t2=pi;          %initially inverted

t1d=0;t2d=0;                    %initial angular velocities
t1d=0;t2d=0.1;                  %given initial kick
T=60;                                %duration of animation
%this is for 'perfect' parametric excitation
%   M1=1;I1=0;B1=sqrt(2);AA=1;AL=3*pi/4;M2=1;I2=0;B2=12-0.02;G=12;t1=-
pi/4;t2=0;t1d=0;t2d=0.1;T=200;

figure(3)
mx=max([B1 B2+AA]);
AX=[-mx mx -mx mx];
plotbox(t1,t2)
%If you want trace animation then uncomment the plotbox line in FDOUBLEPEND

z0=[t1 t2 t1d t2d]';
options=odeset('vectorized','on','reltol',1e-3,'abstol',1e-6);
[t,z]=ode23tb(@fdoublepend1,[0 T],z0,options);

t1=z(:,1);t2=z(:,2);t1d=z(:,3);t2d=z(:,4);

if T<=60 animate,end   %don't do animations if
%save animsave ti t1i t2i M1 M2 I1 I2 B1 B2 AA AL

figure(1)
subplot(211)
plot(t,z(:,1:2),'linewidth',2);
xlabel('time (s)')
ylabel('rotation \theta_1 & \theta_2 (rad)')
subplot(212)
b=t1+AL-t2;
V=-M1*G*B1*cos(t1) - M2*G*(B2*cos(t2) + AA*cos(AL+t1));
T=0.5*((I1+M1*B1^2)*t1d.^2 + I2*t2d.^2 + M2*(B2*t2d.*sin(b)).^2 + M2*(AA*t1d +
B2*t2d.*cos(b)).^2);
plot(t,[V-V(1) T T+V-V(1)],'linewidth',2)
xlabel('time (s)')
```

```
ylabel('PE , KE , PE+KE (J)')

%animate
figure(2)
global AX
   dt=0.1;   %this value gives about-real-time on a toshiba satellite 4090XCDT
      ti=(0:dt:max(t));
      t1i=interp1(t,t1,ti);
   t2i=interp1(t,t2,ti);

tic
for jj=1:length(ti)
   plotbox1(t1i(jj),t2i(jj))
   text(0,-B1-B2,[num2str(ti(jj),'%4.2f') ' s'],'horizontalalign','center')
   %if jj==1 pause(1),end
   % MOV(jj)=getframe(gcf);
end

%movie2avi(MOV,['doublependulum.avi'],'fps',8,'compression','Indeo5','quality',6
5)
    print -f1 -djpeg doublependulum
disp([toc max(ti)])

function zdot=fdoublepend1(t,z)
global M1 M2 I1 I2 B1 B2 AA AL G AX
%nb this code has been vectorized - but no gain in speed
%was achieved because it is accessed in vector form infrequently

t1=z(1,:);t2=z(2,:);t1d=z(3,:);t2d=z(4,:);
b=t1+AL-t2;
C1=-(M2*B2*AA*sin(b).*t2d.^2 + M1*G*B1*sin(t1) + M2*G*AA*sin(AL+t1));
C2=-(-M2*AA*B2*sin(b).*t1d.^2  + M2*G*B2*sin(t2));
A11=I1+M1*B1^2+M2*AA^2;
A12=M2*AA*B2*cos(b);
A21=M2*AA*B2*cos(b);
A22=I2+M2*B2^2;
det=A11.*A22-A12.*A21;
t1dd=(A22.*C1-A12.*C2)./det;
t2dd=(A11.*C2-A21.*C1)./det;

zdot=[t1d;t2d;t1dd;t2dd];

%plotbox(t1(1),t2(1))   %uncomment this line for trace animations


function plotbox1(t1,t2)
global B1 B2 AA AL AX
line1=[0 B1 0 -B1 0 AA*sin(AL)] + ...
   i*[0 -B1 -2*B1 -B1 0 -AA*cos(AL)];
line2=[AA*sin(AL)+[0 B2*sin(t2-t1)]] + ...
   i*[-AA*cos(AL)+[0 -B2*cos(t2-t1)]];
plot(line1*exp(i*t1),'b','linewidth',5);
hold on
plot(line2*exp(i*t1),'r','linewidth',5);
plot([0 AA*sin(t1+AL)],[0 -
AA*cos(t1+AL)],'ko','markersize',8,'markerfacecolor','k')
hold off
```

```
axis(AX)
axis('square')
axis('off')
drawnow
```