INDIAN INSTITUTE OF TECHNOLOGY, PALAKKAD

ME 3040 MECHANICAL VIBRATIONS : COURSE PROJECT

“SPHERICAL PENDULUM”

A Spherical pendulum is similar to the simple pendulum, but moves in 3 dimensional space.
System has 2 degrees of freedom.

PROJECT REPORT

SUBMITTED BY – ANIRUDH [131801005]

SUBMITTED TO – Dr. ANOOP AKKORATH

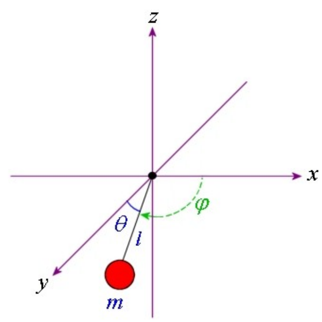
DATE – 03/06/2021

CONTENTS –

Introduction

A Spherical pendulum is similar to the simple pendulum, but moves in 3 dimensional space. System has 2 degrees of freedom. An Inverted Spherical Pendulum which is obtained when we invert spherical pendulum is considered to be very important for studying various applications. The interest in studying spherical inverted pendulum is due to its mathematical model which is considered as simplified model of rocket propelled body or building such that it can be used for control of position of a rocket, for the control of oscillation of buildings, or simply for study of new control techniques.

Deriving equations of motion



pendulum in 3D

from simple trignometry

we get, $x = l \sin(\theta) \cos(\varphi)$.

$y = l \sin(\theta) \sin(\varphi)$

$z = l \cos(\theta)$.

Equations of motion are derived using lagrangian approach.

Lagrangian (L) = T – U.

T = total kinetic energy of system.

U = total potential energy of system.

Deriving Equations :

$$L = T - U$$

$$T = \frac{m}{2}(\dot{x}^2 + \dot{y}^2 + \dot{z}^2)$$

$$U = -mgz$$

$$\dot{x} = l\dot{\theta}\cos(\theta)\cos(\varphi) - l\dot{\varphi}\sin(\theta)\sin(\varphi)$$

$$\dot{y} = l\dot{\theta}\cos(\theta)\sin(\varphi) + l\dot{\varphi}\sin(\theta)\cos(\varphi)$$

$$\dot{z} = -l\dot{\theta}\sin(\theta)$$

$$\Rightarrow \dot{x}^2 = l^2\dot{\theta}^2\cos^2(\theta)\cos^2(\varphi) - 2l^2\dot{\theta}\dot{\varphi}\sin(\theta)\cos(\theta)\sin(\varphi)\cos(\varphi) + l^2\dot{\varphi}^2\sin^2(\theta)\sin^2(\varphi) \quad -1$$

$$\dot{y}^2 = l^2\dot{\theta}^2\cos^2(\theta)\sin^2(\varphi) + 2l^2\dot{\theta}\dot{\varphi}\sin(\theta)\cos(\theta)\sin(\varphi)\cos(\varphi) + l^2\dot{\varphi}^2\sin^2(\theta)\cos^2(\varphi) \quad -2$$

$$\dot{z}^2 = l^2\dot{\theta}^2\sin^2(\theta) \quad -3$$

$$\therefore T = \frac{m}{2}\{1 + 2 + 3\}$$

After simplification of 1 + 2 + 3 , we get

$$T = \frac{m}{2}(l^2\dot{\theta}^2 + l^2\dot{\varphi}^2\sin^2(\theta))$$

$$\therefore U = -mgz = -mgl\cos(\theta)$$

$$\Rightarrow L = T - U = \frac{m}{2}(l^2\dot{\theta}^2 + l^2\dot{\varphi}^2\sin^2(\theta)) + mgl\cos(\theta)$$

$$\Rightarrow \boxed{L = \frac{m}{2}(l^2\dot{\theta}^2 + l^2\dot{\varphi}^2\sin^2(\theta)) + mgl\cos(\theta)} \quad -4$$

Now $\because$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}}\right) - \frac{\partial L}{\partial \theta} = 0$$

where

$$\frac{\partial L}{\partial \theta} = \dot{\varphi}^2 m l^2 \sin(\theta)\cos(\theta) - mgl\sin(\theta)$$

$$\frac{\partial L}{\partial \dot{\theta}} = ml^2\dot{\theta}$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}}\right) = ml^2\ddot{\theta}$$

we get

$$ml^2\ddot{\theta} - \dot{\varphi}^2 ml^2\sin(\theta)\cos(\theta) + mgl\sin(\theta) = 0$$

And rearranging for $\ddot{\theta}$ gives,

$$\boxed{\ddot{\theta} = \frac{l\dot{\varphi}^2\sin(\theta)\cos(\theta) - g\sin(\theta)}{l}} \quad (5)$$

To find $\ddot{\varphi}$, we use

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\varphi}}\right) - \frac{\partial L}{\partial \varphi} = 0$$

where

$$\frac{\partial L}{\partial \varphi} = 0$$

$$\frac{\partial L}{\partial \dot{\varphi}} = \dot{\varphi} ml^2\sin^2(\theta)$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\varphi}}\right) = \ddot{\varphi} ml^2\sin^2(\theta) + 2\dot{\varphi} ml^2\dot{\theta}\sin(\theta)\cos(\theta)$$

So $\ddot{\varphi} ml^2\sin^2(\theta) + 2\dot{\varphi} ml^2\dot{\theta}\sin(\theta)\cos\theta = 0$

And rearranging for $\ddot{\varphi}$ gives, $\boxed{\ddot{\varphi} = \frac{-2\dot{\varphi}\dot{\theta}\cos(\theta)}{\sin(\theta)}} \quad (6)$

$$\ddot{\theta} = \frac{l\,\dot{\varphi}^2 \sin(\theta)\cos(\theta) - g\sin(\theta)}{l}$$

$$\ddot{\varphi} = \frac{-2\,\dot{\varphi}\,\dot{\theta}\cos(\theta)}{\sin(\theta)}$$

Now we can use Matlab to solve these equations.

Matlab code

After deriving equations we now can use Matlab to solve equations.

```
%% Spherical pendulum
% Simulation and animation of a spherical pendulum.
%
%%
clear ;
close all ;
clc

%% Scenario

% Parameters
l = 3;                          % Length           [m]

% Initial conditions
theta0  = pi/6;                 % Polar angle      [rad]
beta0   = 0;                    % Azimuth          [rad]
dtheta0 = 0;                    % d/dt(theta)      [rad/s]
dbeta0  = 3;                    % d/dt(phi)        [rad/s]

x0 = [theta0 beta0 dtheta0 dbeta0]';

% Parameters
tf     = 30;                            % Final time        [s]
fR     = 30;                            % Frame rate        [fps]
dt     = 1/fR;                          % Time resolution   [s]
time   = linspace(0,tf,tf*fR);          % Time              [s]

%% Simulation
```

```
[TOUT,XOUT] = ode45(@(t,x) pendulum(t,x,l),time,x0);

% Retrieving states
alpha   = XOUT(:,1);
beta    = XOUT(:,2);
dalpha  = XOUT(:,3);
dbeta   = XOUT(:,4);

% Coordinates
rx = l*sin(alpha).*cos(beta);
ry = l*sin(alpha).*sin(beta);
rz = -l*cos(alpha);

%% Animation

% Min-max axis
min_x = min(rx)-l/2;
max_x = max(rx)+l/2;
min_y = min(ry)-l/2;
max_y = max(ry)+l/2;
min_z = min(rz)-l/2;
max_z = 0;

figure
set(gcf,'Position',[270   140   640     360  ])

% Create and open video writer object
v = VideoWriter('spherical_pendulum.avi');
v.Quality = 100;
open(v);

hold on ; grid on ; axis equal
set(gca,'CameraPosition',[42.0101   30.8293   16.2256])
```

```
set(gca,'XLim',[min_x max_x])
set(gca,'YLim',[min_y max_y])
set(gca,'ZLim',[min_z max_z])

for i = 1:length(rx)

    cla
    % Vertical line
    plot3([0 0],[0 0],[0 -l],'k--')
    % Point fix
    p = plot3(0,0,0,'Marker','*','Color','k','MarkerSize',10);
    % Pendulum trajectory
    plot3(rx(1:i),ry(1:i),rz(1:i),'b')
    % Pendulum rod
    plot3([0 rx(i)],[0 ry(i)],[0 rz(i)],'r')
    % Pendulum sphere
    plot3(rx(i),ry(i),rz(i),'Marker','o','Color','k','MarkerFaceColor','r','MarkerSize',10);
    % Projections
    plot3(min_x*ones(1,i),ry(1:i),rz(1:i),'g')
    plot3(rx(1:i),min_y*ones(1,i),rz(1:i),'g')
    plot3(rx(1:i),ry(1:i),min_z*ones(1,i),'g')

    frame = getframe(gcf);
    writeVideo(v,frame);

end

close(v);

function dx = pendulum(~, x, l)

    % Parameters
    g = 9.81;                   % Gravity           [m/s2]
```

```
% States
x1 = x(1);
% x2 = x(2);
x3 = x(3);
x4 = x(4);

% State equations
dx1 = x3;
dx2 = x4;
dx3 = (l*x4^2*sin(x1)*cos(x1) – g*sin(x1))/l;
dx4 = -2*x3*x4/tan(x1);
%% note dx3 and dx4 are equations which were derived manually.

dx = [dx1 dx2 dx3 dx4]';

end
```

Key points learnt from Matlab code

- ode45 is solver used for solving differential equation, MATLAB's standard solver for ordinary differential equations (ODEs) is the function ode45. This function implements a Runge-Kutta method with a variable time step for efficient computation.
- @ is a function handle used for indirect calling of function pendulum. pendulum(t, x, l) is user defined function, time is time-span from 0 to 30 seconds divided into 900 equal parts.
- Ode45 will give solution at each instant of time. These values are recorded as column vectors in [TOUT, XOUT]. XOUT has four columns alpha, beta, dalpha and dbeta.
- Set gcf is used for handling graphics of current figure. VideoWriter() function is used for creating objects to write video files.
- Set gca is used to set current axis, **cla** is command for claearing axes, **plot3** is function used to plot in 3-D coordinate space.

References

- http://farside.ph.utexas.edu/teaching/336k/Newtonhtml/node82.html
- https://www.youtube.com/

End
