Lab-II

Devansh Shukla
I18PH021

1 Aim

Write and execute octave programs for simulating motion of a simple pendulum.

2 Theory

3 Program

3.1 Simple pendulum

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% SimplePendulum
%
% Program to solve/simulate simple pendulum
% Author: Devansh Shukla I18PH021
% 9th March, 2022
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

pkg load symbolic

% set symbolic vars
syms theta0 omega t t0

theta = theta0 * cos(omega * (t - t0))


% initial parameters
m = input("Enter mass of bob ");
l = input("Enter the length of string ");
t0 = input("Enter t0 ");
tf = input("Enter tf ");

% initial parameters
% assuming the initial theta is pi/4
theta0 = pi/4;
g = 9.81; % m/s/s gravitational acceleration
omega = sqrt(g/l);

idx = 1;
x = y = theta = dtheta = vx = vy = [];

% loop for numerically computing theta, dtheta, positions and velocities
for t=t0:0.1:tf
    theta(idx) = theta0 * cos(omega * (t - t0));
    dtheta(idx) = -omega * theta0 * sin(omega * (t-t0));
    x(idx) = l * sin(theta(idx));
    y(idx) = -l * cos(theta(idx));
    vx(idx) = l * dtheta(idx) * cos(theta(idx));
    vy(idx) = l * dtheta(idx) * sin(theta(idx));
    idx = idx + 1;
endfor

% for computing energy
v2 = vx.*vx + vy.*vy;
kinetic_energy = 0.5 * m .* v2;
potential_energy = m*g .* (l .+ y);

% plotting the trajectory
figure()
hold on
grid on
set(gcf, 'PaperSize', [6, 3]);
set(gca,'XMinorTick','on','YMinorTick','on');
plot(x, y, "linewidth", 2);
title("Trajectory");
xlabel("X[m]");
ylabel("Y[m]");
ylim([-l-0.5, 0])
set(gcf, 'renderer', 'painters');
legend boxoff
print -dpng pendulum_traj.png
hold off

% plotting the positions
figure()
hold on
grid on
plot(x, "linewidth", 2);
plot(y, "linewidth", 2);
set(gcf, 'PaperSize', [6, 3]);
set(gca,'XMinorTick','on','YMinorTick','on');
title("Position[m]");
```

```
xlabel("Time(s) [1 unit = 0.1s]");
ylabel("Displacement(m)");
legend("X", "Y");
set(gcf, 'renderer', 'painters');
legend boxoff
print -dpng pendulum_pos.png
hold off

% plotting the energy
figure()
hold on
grid on
set(gcf, 'PaperSize', [6, 3]);
set(gca,'XMinorTick','on','YMinorTick','on');
plot(kinetic_energy, "linewidth", 2);
plot(potential_energy, "linewidth", 2);
title("Energy");
xlabel("Time(s) [1 unit = 0.1s]");
ylabel("Energy(J)");
legend("KE", "PE");
set(gcf, 'renderer', 'painters');
legend boxoff
print -dpng pendulum_energy.png
hold off
```

4 Results

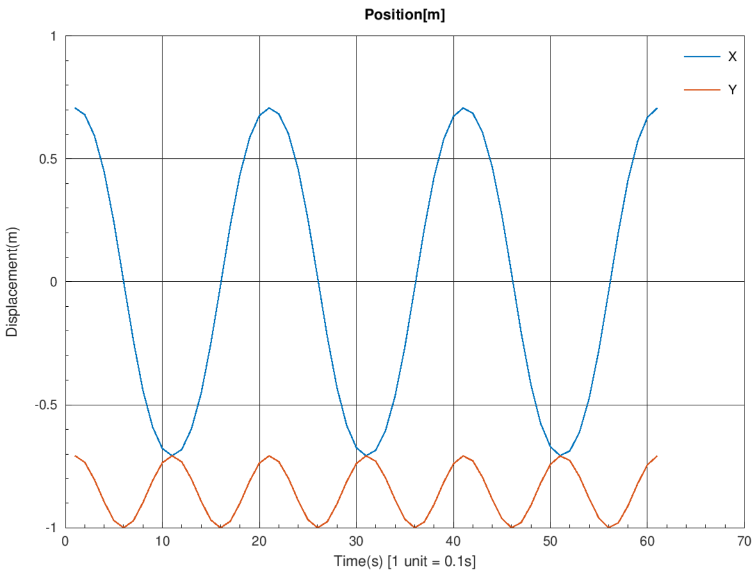
4.1 Terminal output

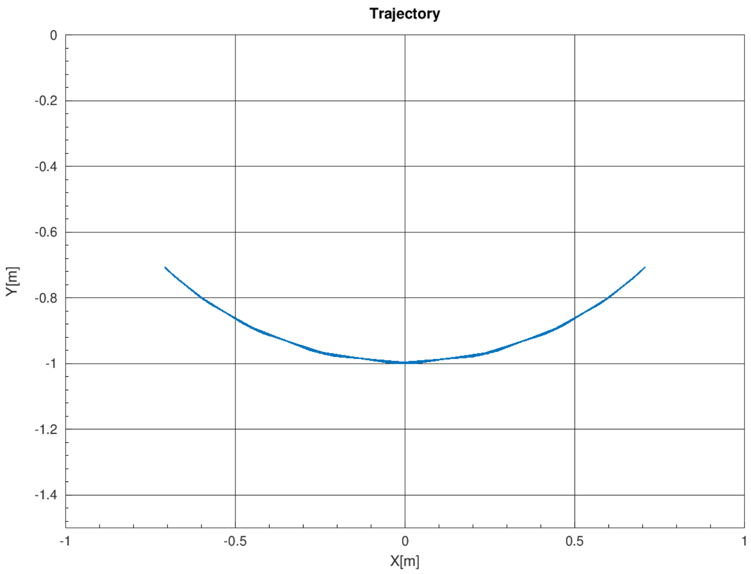
```
(escape) devansh@ds:~/GitHub/Vault/OctaveLab/Programs/outputs$ octave ../SimplePendulum.m
Symbolic pkg v2.9.0: Python communication link active, SymPy v1.5.1.
theta = (sym) θ₀·cos(ω·(t - t₀))
Enter mass of bob 1
Enter the length of string 1
Enter t0 0
Enter tf 5
```

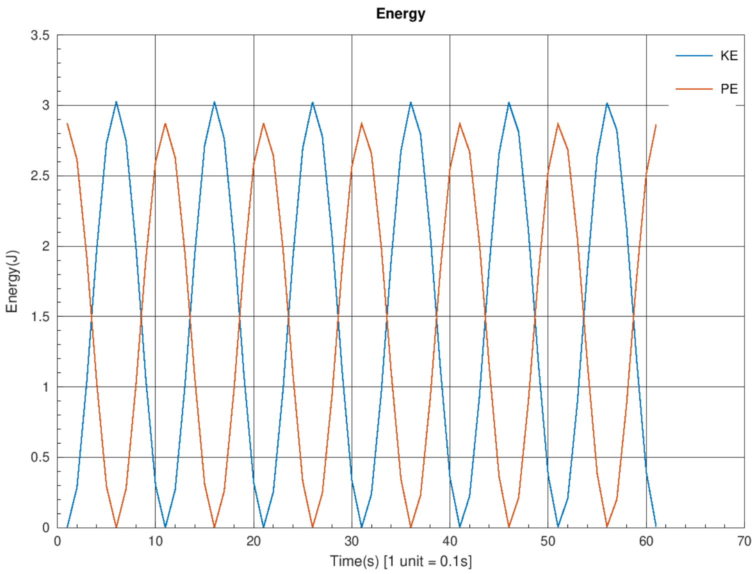
4.2 Plots

Initial parameters

- Length $l = 1\ m$
- Mass of bob, $m = 1\ kg$







5 Remarks

The programs can be used to trace and simulate the motion of any simple pendulum by defining the required parameters.