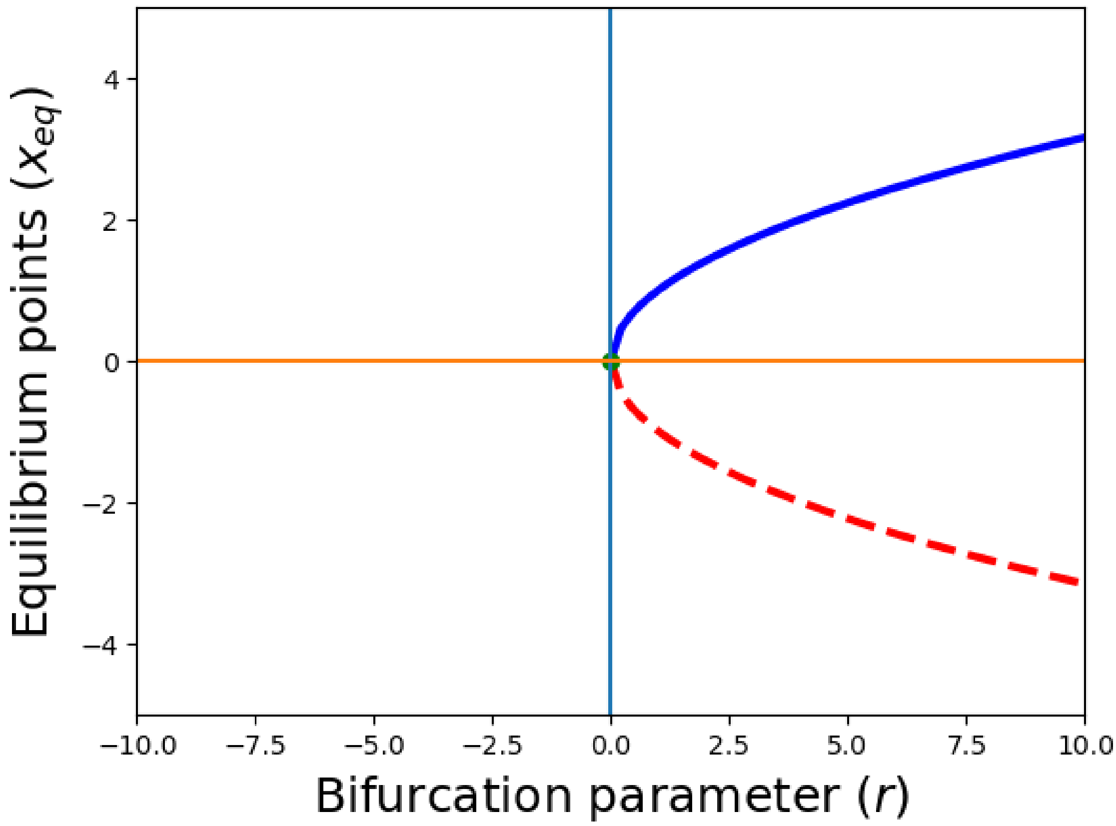
Saddle node bifurcation, analysis

(r<0, r>0, r=0), equation:

$$\dot{x} = r + x^2$$

Entrée [104]:

```
#Saddle node bifurcation, analysis (r<0, r>0, r=0)|| equation==> x_dot=r-x^2=0
from pylab import *
def y1(mu):
    return sqrt(mu)
def y2(mu):
    return -sqrt(mu)
domain = linspace(0, 10)
plot(domain, y1(domain), 'b-', linewidth = 3) #"b" is blue
plot(domain, y2(domain), 'r--', linewidth = 3) # "r--" is red dashed line
plot([0], [0], 'go')
plot([0,0],[-5,5], "-")
plot([-10,10],[0,0], "-")
axis([-10, 10, -5, 5])
xlabel('Bifurcation parameter ($r$)', fontsize = 18)
ylabel('Equilibrium points ($x_{eq}$)', fontsize = 18)
show()


```



We have introduced above the *Saddle-Node bifurcation* without naming it. This bifurcation is associated to the differential equation: $x'(t) = r + \alpha x^2$, where r and α are the control parameters. For $\alpha > 0$ we are speaking about subcritical bifurcation. Lets $\alpha = 1$, the equilibrium points are easy to determine and they are immediately obtained : $x_{eq} = \pm \sqrt{r}$, this means that the equilibrium point exist only for $r < 0$. We can summarize the result in the following table:

Equilibrium point	$r < 0$	$r > 0$
$x_{eq} = \sqrt{r}$	doesn't exist	stable
$x_{eq} = -\sqrt{r}$	doesn't exist	unstable

Homework Go and modify the code to run the transcritical bfurcation and plot the diagram

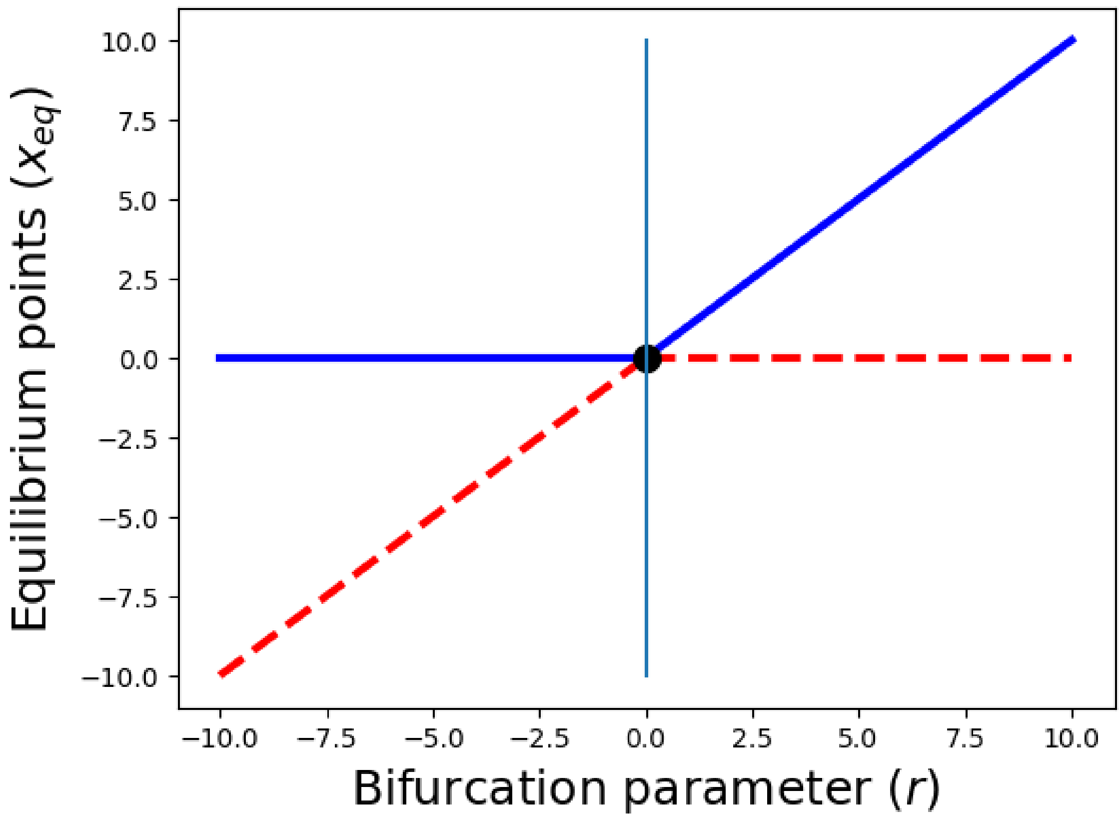
Transcritcal bifurcation, analysis

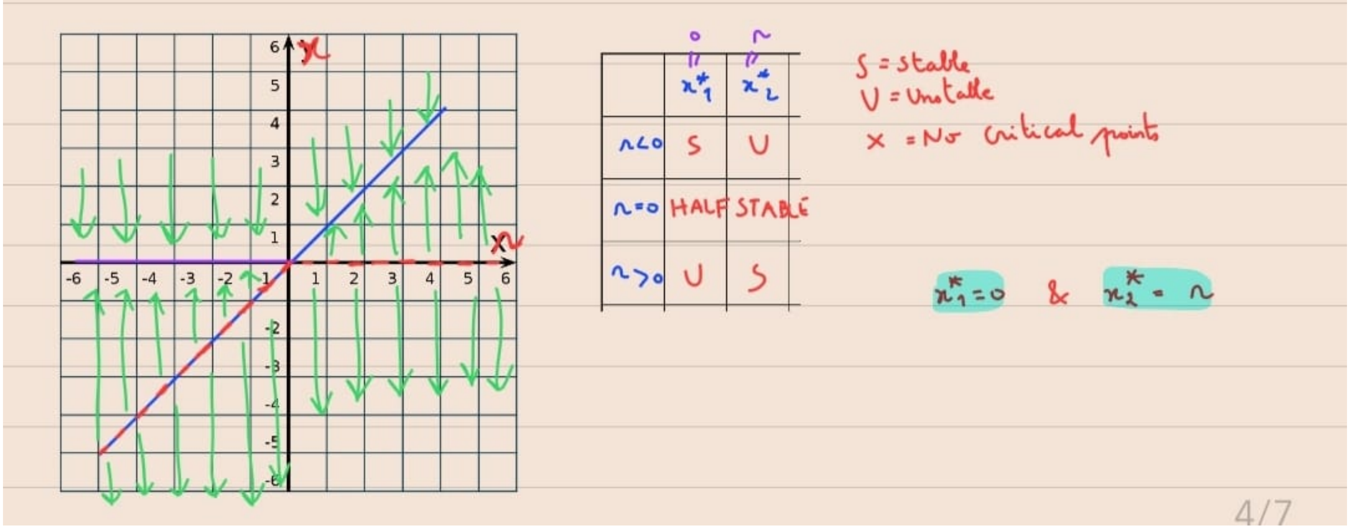
(r<0, r>0, r=0), equation is:

$$\dot{x} = rx - x^2$$

Entrée [103]:

```
#Saddle node bifurcation, analysis (r<0, r>0, r=0)|| equation==> x_dot=r-x^2=0
from pylab import *

def y1(x):
    return x

def y2(x):
    return np.zeros_like(x)


domain1 = linspace(0, 10) # x values
domain2 = linspace(-10, 0) # x values

plot(domain1, y1(domain1), 'b-', linewidth = 3) #"b" is blue
plot(domain2, y1(domain2), 'r--', linewidth = 3) #"r" is red

plot(domain1, y2(domain1), 'r--', linewidth = 3) #"b" is blue
plot(domain2, y2(domain2), 'b-', linewidth = 3) #"r" is red

plt.plot([0], [0], marker='o', markersize=10, color='black')  # You can specify the

plot([0,0],[-10,10], "-")

xlabel('Bifurcation parameter ($r$)', fontsize = 18)
ylabel('Equilibrium points ($x_{eq}$)', fontsize = 18)

show()

```





Assignment 27 of September

The equation of motion for the pendulum is found by equating the mass times acceleration of the pendulum bob to the component of the force acting on the bob – its weight – along the direction of motion. The bob moves on the arc of a circle of radius L , and the distance traveled along the tangent to the arc is denoted, $s.L$ is the length of the pendulum.

Question

- 1 - Set the governing equation
- 2 - Write a program to solve the obtained differential equation.
- 3 - Plot the time evolution of the angle θ and represent the corresponding phase space.

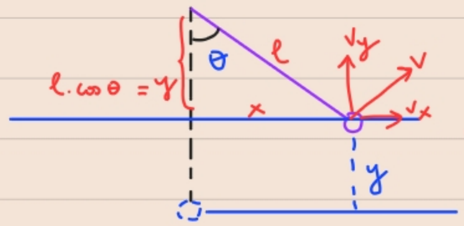
1

EQUATION OF MOTION - LAGRANGIAN(L)

Given : Pendulum w/ mass $= m$

→ Assume : (1) Massless Rod &

(2) Small oscillation



$\sin\theta = \frac{x}{\ell}$

$x = \ell.\sin\theta$ || $h = \ell - \ell\cos\theta$ || $y = \ell - \ell\cos\theta$

$$KE = \frac{1}{2}mv^2 = \frac{1}{2}m(V_x^2 + V_y^2)$$

⇒ $x = \ell.\sin\theta$

$V_x = \frac{dV}{dt} = \dot{x} = \ell_t.\sin\theta + \ell(\sin\theta)'_t$

$\therefore \dot{x} = \ell.\dot{\theta}.\cos\theta$ (y a constant)

⇒ $y = \ell - \ell\cos\theta$

$V_y = \frac{dy}{dt} = \dot{y} = \ell_t - \ell_t.\cos\theta - \ell(\cos\theta)'_t$

$\dot{y} = \ell.\dot{\theta}\sin\theta$ (y a constant)

$$\therefore KE = \frac{1}{2}m\left[(\ell.\dot{\theta}.\cos\theta)^2 + (\ell.\dot{\theta}.\sin\theta)^2\right] = \frac{1}{2}m\left[\ell^2.\dot{\theta}^2.\cos^2\theta + \ell^2.\dot{\theta}^2.\sin^2\theta\right]$$

$$= \frac{1}{2}m\left[\ell^2.\dot{\theta}^2(\cos^2\theta + \sin^2\theta)\right]$$

Kinetic energy $KE = \frac{1}{2}m.\ell^2.\theta^2$

$$P.E = mgh = mg.\ell.(1 - \cos\theta)$$

$$L = KE - PE = \frac{1}{2}m\ell^2\dot{\theta}^2 - \frac{1}{2}mg\ell(1 - \cos\theta)$$

$$L = \frac{1}{2}m\ell^2\dot{\theta}^2 - \frac{1}{2}mg\ell + \frac{1}{2}mg\ell\cos\theta$$

$\frac{dL}{d\theta} = -mg\ell\sin\theta$ || $\frac{dL}{d\dot{\theta}} = mg\ell^2\dot{\theta}$ || $\frac{d}{dt}\left(\frac{dL}{d\dot{\theta}}\right) = m\ell^2\ddot{\theta}$

Replace in $\frac{d}{dt}\left(\frac{dL}{d\dot{\theta}}\right) - \frac{dL}{d\theta} = 0$

$\Rightarrow ml^2\ddot{\theta} - (-mgl \sin\theta) = 0$

$\Rightarrow \cancel{m}l^{\cancel{2}}\ddot{\theta} + \cancel{m}g\cancel{l} \sin\theta = 0$ (divide by m then l)

$\Rightarrow \ddot{\theta}l + g\sin\theta = 0$

Since θ is small i.e $\theta < \frac{\pi}{12}$ $\sin\frac{\pi}{12} = 0{,}258 \approx \frac{\pi}{12} = 0{,}2618$

$\therefore \sin\frac{\pi}{12} \approx \frac{\pi}{12} \Rightarrow \sin\theta \approx \theta$

$\Rightarrow \ddot{\theta}l + g\theta = 0$

$\Rightarrow \ddot{\theta} + \frac{g}{l}\theta = 0$ General 2nd order DE describing Motion for a pendulum.

⚠ if $w^2 = \frac{g}{l}$ $\therefore w = \frac{g}{l}$

$\therefore \theta = A\cdot\sin(w\cdot t)$

with A = amplitude of osci llation

θ = function of time (t)

$w = \frac{g}{l}$ = angular frequency of the oscillation

⚠ $w = 2\pi\cdot f$ & $f = \frac{1}{2\pi}w$

$\therefore f = \frac{1}{2\pi}\sqrt{\frac{g}{L}}$

The governing equation is as follows:

Newton's second law states that **force acting on an object = mass object*object acceleration**

$$F = m * a$$

m=mass of the bob

g=acceleration due to gravity, so second derivative of distance w.r.t time

s=the distance traveled along the tangent to the arc

θ=angle the pendulum makes with the vertical

therefore:

$$m * \frac{d^2s}{t^2} = -m * g * sin(\theta)$$

simplifying "m" we get

$$\frac{d^2s}{t^2} = -g * sin(\theta)$$

finally if θ<15 degrees or π/12

then sin(θ)=0.256 close to π/12=0.2618 and sin(θ) ≈ θ

$$\frac{d^2s}{t^2} = -g * \theta$$

Since s=Lθ, we can replace by θ=s/L and write this as:

$$\frac{d^2\theta}{t^2} = -\frac{g}{L} * \theta$$

which is the governing equation with:

$$w = \frac{g}{L}$$

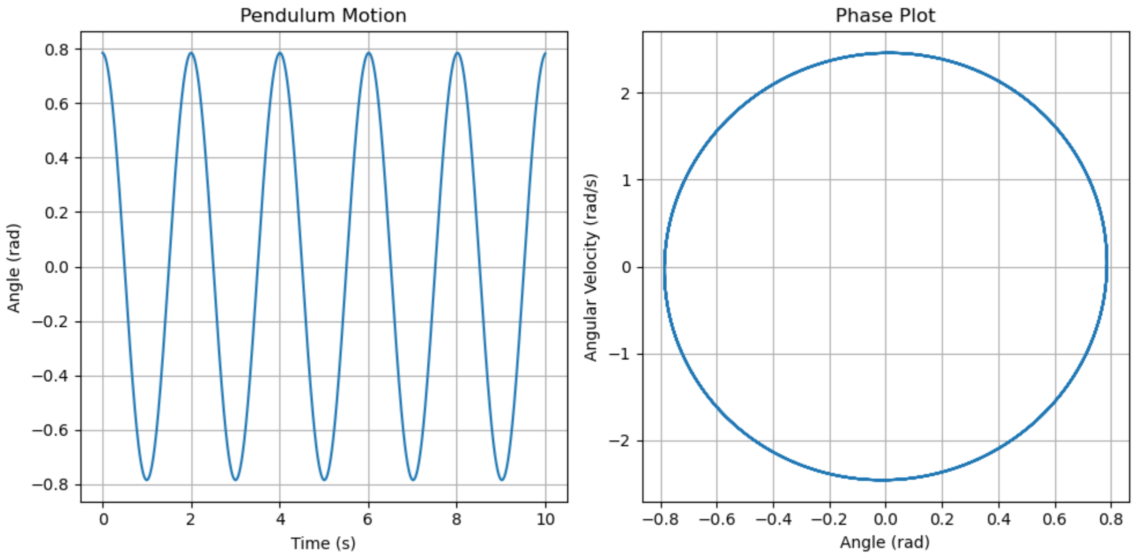
and

$$T = 2\pi * \sqrt{\frac{L}{g}}$$

2

Entrée [113]:

```python
import math
import matplotlib.pyplot as plt

# Pendulum parameters and initial conditions
L = 1.0  # length of pendulum (m)
g = 9.81  # acceleration due to gravity (m/s^2)
theta = 0.785398  # 45 degrees initial displacement angle (radians)
omega = 0.0  # initial angular velocity (rad/s)
t = 0.0  # initial time (s)
dt = 0.01  # time step for Euler's method (s)

# Lists to store time, angle and angular velocity values
t_values = [t]
theta_values = [theta]
omega_values = [omega]

# Euler's method
for _ in range(1000):
    omega = omega - (g/L)*theta*dt
    theta = theta + omega*dt
    t = t + dt
    t_values.append(t)
    theta_values.append(theta)
    omega_values.append(omega)

# Plot the angle as a function of time
plt.figure(figsize=(10, 5))
plt.subplot(1, 2, 1)
plt.plot(t_values, theta_values)
plt.xlabel('Time (s)')
plt.ylabel('Angle (rad)')
plt.title('Pendulum Motion')
plt.grid(True)

# Phase plot
plt.subplot(1, 2, 2)
plt.plot(theta_values, omega_values)
plt.xlabel('Angle (rad)')
plt.ylabel('Angular Velocity (rad/s)')
plt.title('Phase Plot')
plt.grid(True)


plt.tight_layout()
plt.show()

```



Entrée [109]:

```
t_values
```

Out[109]:

```
[0.0,
 0.01,
 0.02,
 0.03,
 0.04,
 0.05,
 0.060000000000000005,
 0.07,
 0.08,
 0.09,
 0.09999999999999999,
 0.10999999999999999,
 0.11999999999999998,
 0.12999999999999998,
 0.13999999999999999,
 0.15,
 0.16,
 0.17,
```

Entrée [108]:

```
theta_values
```

Out[108]:

```
[0.785398,
 0.784627524562,
 0.7830873295224047,
 0.7807789258125478,
 0.7777045779764689,
 0.773867301949395,
 0.7692708620991088,
 0.7639197675331033,
 0.757819267675148,
 0.7509753471156032,
 0.7433947197405381,
 0.7350848221454075,
 0.7260538063397523,
 0.7163105317500778,
 0.7058645565287565,
 0.6947261281774804,
 0.6829061734944621,
 0.6704162878552459,
```