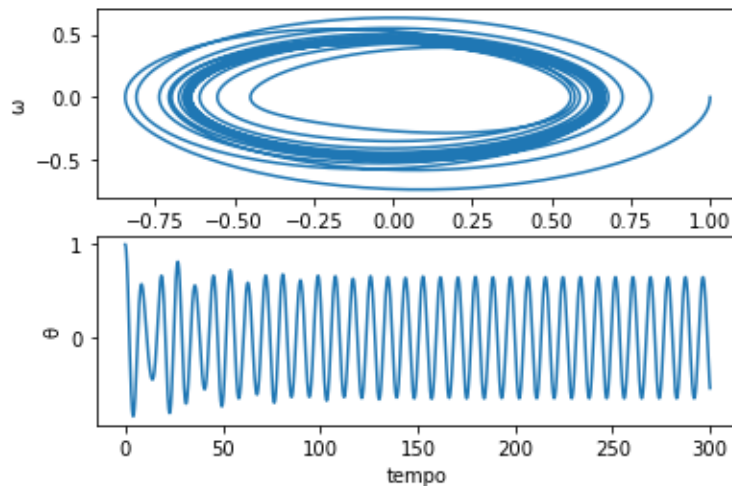


2) Forçamento fraco e tempo transiente.

Após alterar os parâmetros do pêndulo forçado para torná-los adimensionais, temos que Ω é o valor do torque externo e A é o valor da amplitude desse torque.

Inicialmente, utilizamos os valores $b = 0.05$, $\Omega = 0.7$ e $A = 0.3$ para analisar o comportamento do espaço de fase do sistema e o comportamento da variável θ em função do tempo. Vamos usar as condições iniciais $\theta(0) = 1$ e $\omega(0) = 0$ para todos os gráficos do exercício 2 e 3.

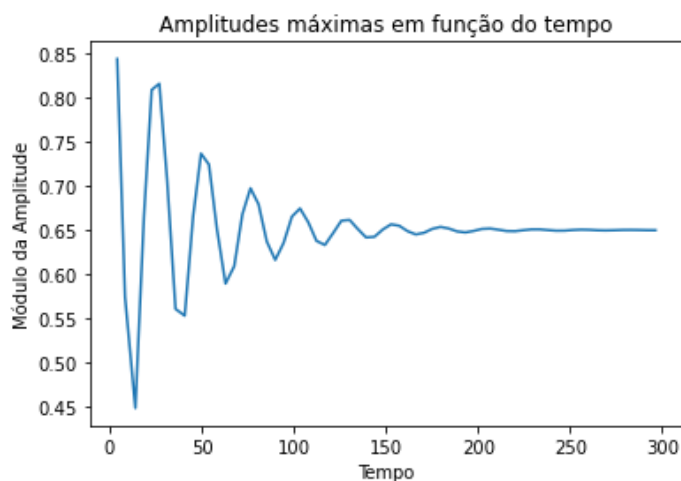
Os códigos utilizados estão no fim deste relatório.



Os gráficos ao lado mostram o espaço de fase (em cima) e $\theta(t)$ (abaixo). Podemos ver que o sistema, após um certo tempo transiente, entra num estado estacionário de libração. Esse estado tem frequência igual a $(\Omega/2\pi)$.

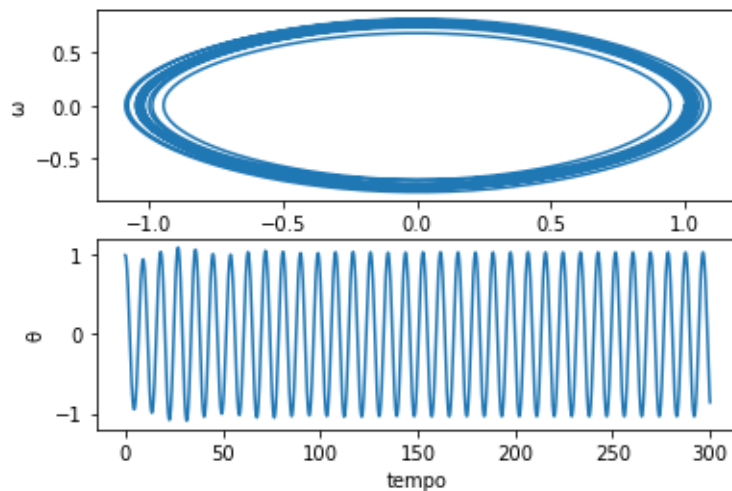
Principalmente no gráfico $\theta(t)$ fica bem visível a presença do tempo transiente e quando passa para o estado estacionário.

Para estimar de forma mais precisa o valor do tempo transiente, foi feito o plot de um gráfico que mostra o valor do módulo da amplitude em função do tempo, dessa forma, podemos ver em que ponto aproximado o módulo da amplitude se estabiliza e o tempo transiente acaba.



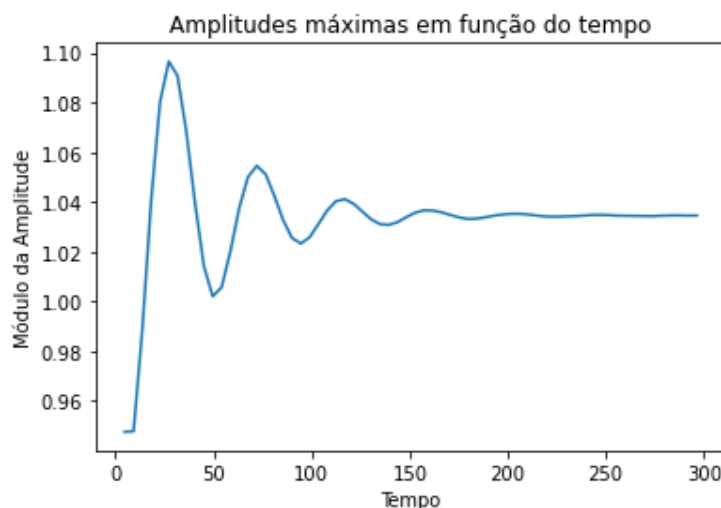
A partir desse gráfico podemos ver que a estabilidade é atingida entre $100 < t < 150$. Porém, sendo mais preciso, dentro desse intervalo, podemos aproximar para um valor mais próximo de $t = 150$, que é onde as flutuações de amplitude se tornam consideravelmente menores.

Repetindo o mesmo processo para $A = 0.4$, com os mesmo valores de b e Ω , temos:



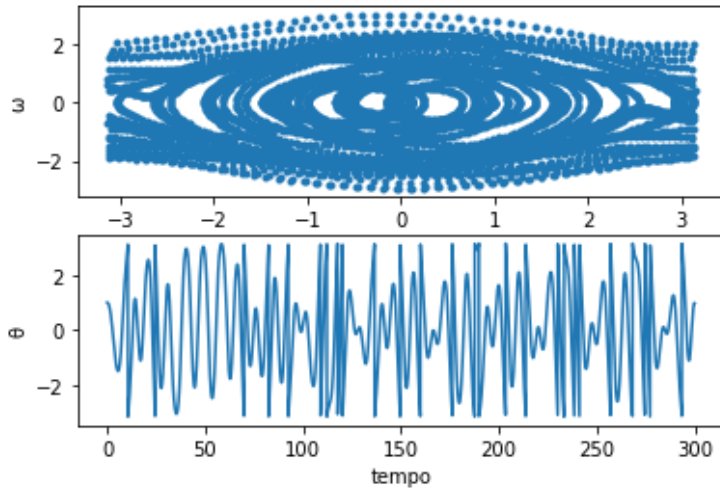
É possível ver que o sistema se comporta da mesma maneira de quando tinha o valor de $A = 0.3$. O sistema ainda atinge um estado estacionário e mostra um tempo transiente, ainda que não fique tão visível no gráfico de $\theta(t)$ quanto ficou no exemplo passado.

Para as condições iniciais usadas, podemos ver que o sistema sai do tempo transiente mais rapidamente que o sistema anterior, e com módulos de amplitudes menos discrepantes. Porém, durante a execução da atividade, foi observado que dependendo das condições iniciais, o $A = 0.4$ pode ou não atingir a estabilidade para o tempo que estamos utilizando, foram testadas mais de uma condição inicial, e a partir delas, o sistema pode demorar mais ou menos para atingir a estabilidade. Porém para as condições iniciais usadas, obtivemos o resultado a seguir para o gráfico de amplitudes.



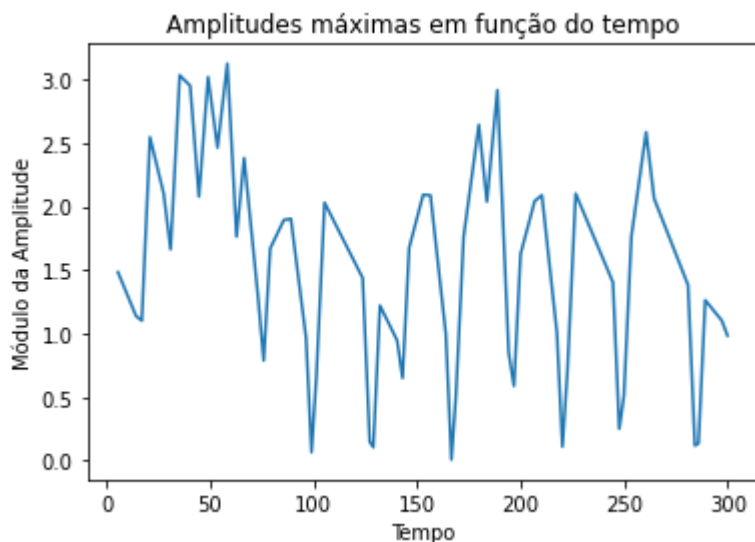
Como é possível visualizar, o gráfico ao lado, apesar de diferente, mostra o mesmo comportamento do gráfico de amplitudes de $A = 0.3$. Dessa forma, podemos estimar o tempo transiente para um intervalo entre 120 e 150, mais especificamente para um valor de aproximadamente $t = 135$.

Por fim, repetindo o mesmo processo para $A = 0.6$, e mantendo os mesmos valores de b e Ω , temos:



Podemos verificar que para o $A = 0.6$, temos um comportamento completamente diferente dos 2 exemplos anteriores. Para esse sistema atual, os gráficos não mostram sinais de atingirem um movimento estacionário após um tempo transiente. O espaço de fase não converge para uma trajetória que aparentemente se repete e o gráfico de $\theta(t)$ em nenhum momento atinge estabilidade.

Observando agora o gráfico das amplitudes que anteriormente foi usado para definir o tempo transiente, podemos ver que os módulos das amplitudes não param de variar e não entram num movimento de libração. Logo, para esses parâmetros do sistema, não é possível definir tempo transiente nem observar o movimento periódico do pêndulo.



3) Diagrama de Poincaré

Usaremos o diagrama de Poincaré para identificar se o pêndulo tem comportamento periódico ou potencialmente caótico. Para identificar isso, vamos plotar um gráfico de (θ, ω) do espaço de fase, mas diferente do gráfico feito anteriormente, usaremos intervalos específicos entre os pontos plotados, para poder visualizar como se distribuem.

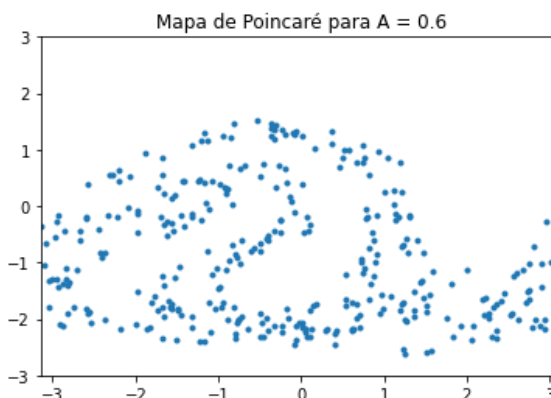
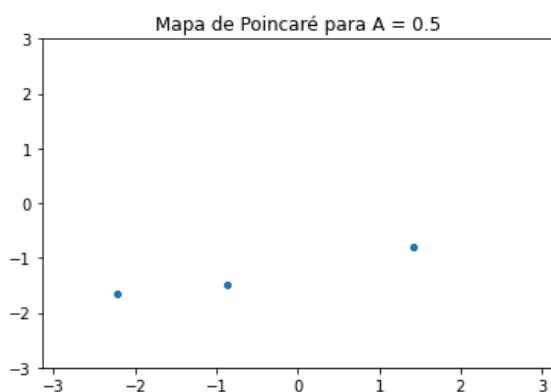
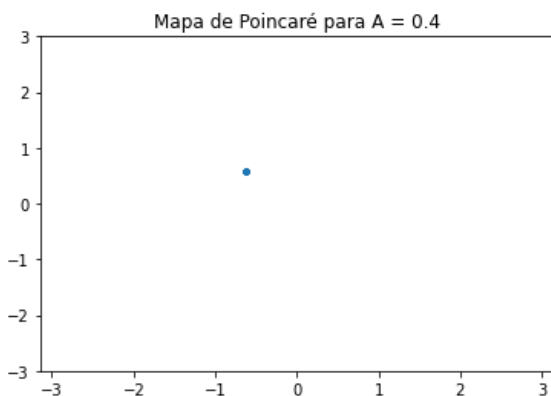
Para determinar esses intervalos, usamos o valor do torque externo, pois para o regime de oscilação forçada, esse valor é proporcional ao período. Dessa forma podemos

definir que $P = 2\pi/\Omega$ e pegar intervalos que sejam múltiplos inteiros de P , garantindo assim que usaremos os valores desejados.

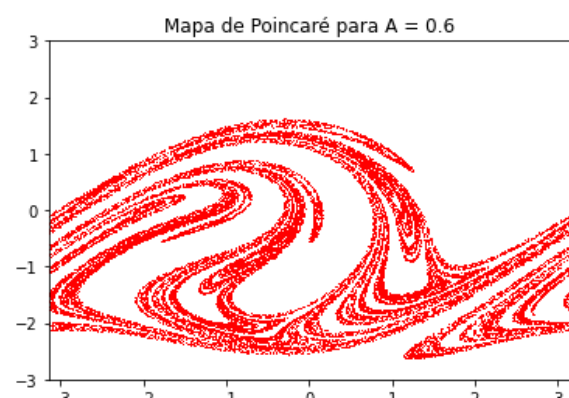
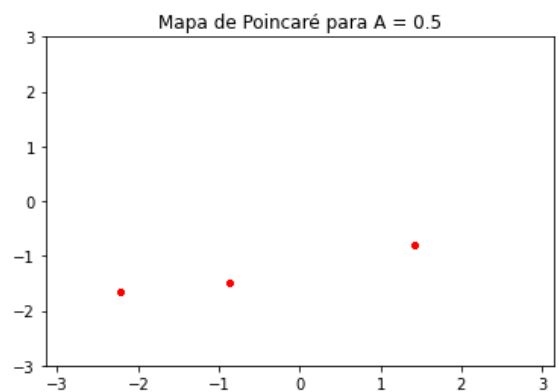
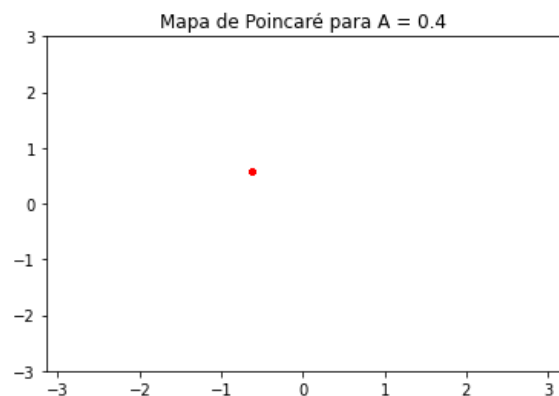
Para o diagrama de Poincaré, se tivermos um movimento periódico, obteremos apenas poucos pontos na tela (exemplo: um, dois, três ou cinco), porém, se o sistema mostrar comportamento caótico, obteremos uma nuvem de pontos de forma peculiar, possivelmente fractal.

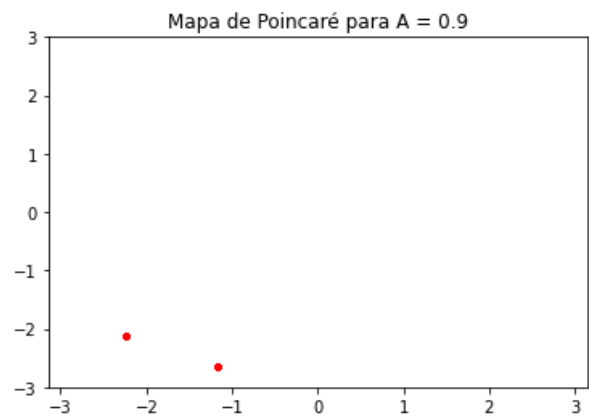
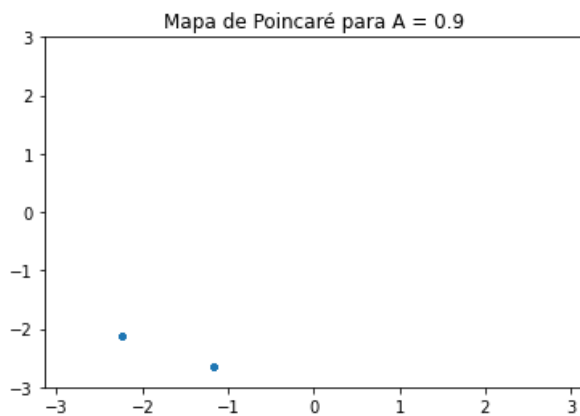
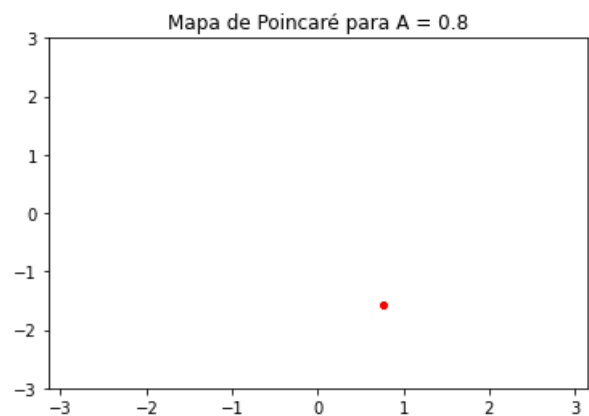
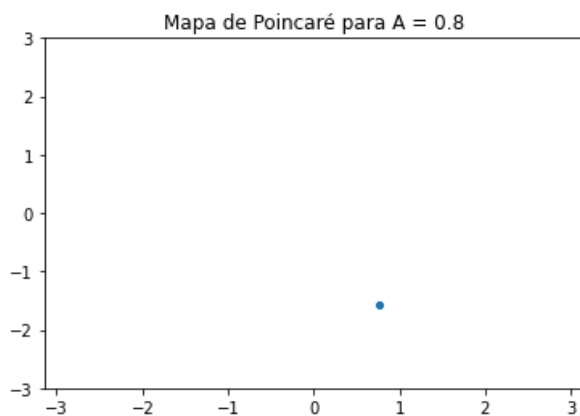
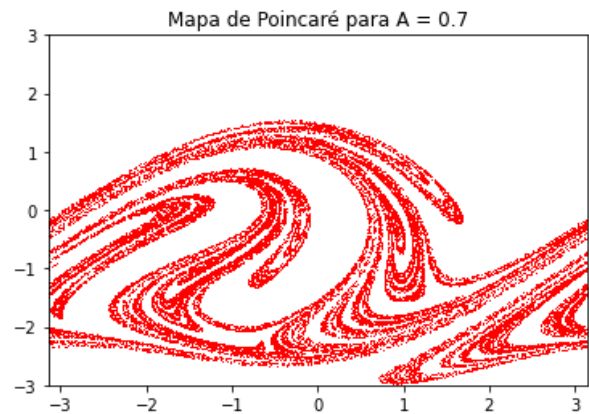
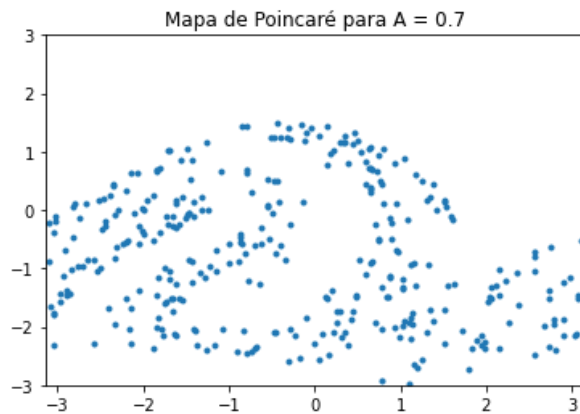
Construiremos o diagrama de Poincaré para as seguintes amplitudes: $A = 0.4, 0.5, 0.6, 0.7, 0.8$ e 0.9 . Para fazer os gráficos, inicialmente foi utilizado um número baixo de períodos para testar o código. Após resultados satisfatórios com baixo número de períodos, foi utilizado um número alto. Essa abordagem foi feita por limitações computacionais. Compilar um código com alto número de períodos demora muito tempo, por isso foi feito inicialmente com um número baixo. Segue os resultados para todas as amplitudes de torque externo.

Número de períodos = 1.000:



Número de períodos = 100.000:





Foi testado utilizar o número de períodos = 1.000.000, mas não foi possível rodar o programa para alguns valores de A nessa configuração, por limitações da máquina, por isso foi usado 100.000.

Analisando os diagramas de Poincaré é possível ver claramente quais valores de A tem comportamento periódico e quais têm comportamento caótico. Os valores $A = 0.4, 0.5, 0.8$ e 0.9 apresentam comportamento periódico, mas os gráficos desses valores diferem na quantidade de pontos que aparecem no diagrama. Essa diferença se dá por conta da diferença de periodicidade do sistema para cada valor de A .

Para os valores de $A = 0.4$ e 0.8 , temos apenas um ponto no gráfico, ou seja, o período do gráfico é igual ao período utilizado no diagrama de Poincaré. Esses 2 sistemas apresentam período simples com valor $(2\pi/\Omega)$.

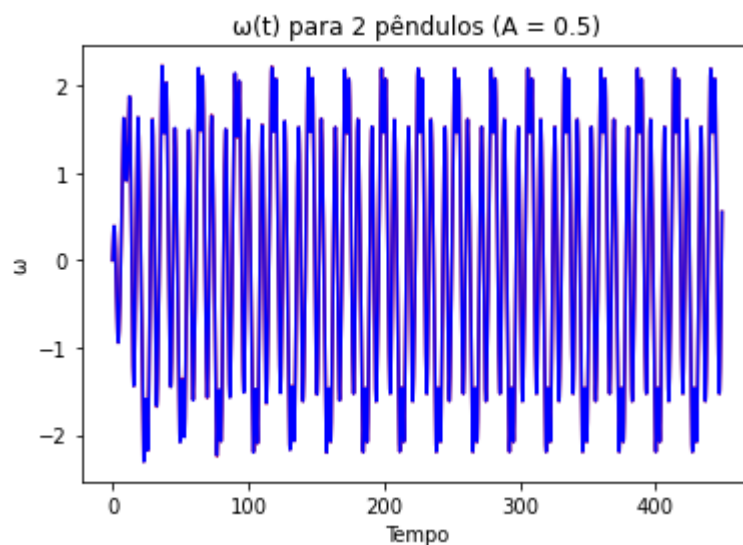
Já os sistemas com valores $A = 0.5$ e 0.9 apresentam três e dois pontos, respectivamente. Isso significa que o primeiro sistema apresenta dobramento de período, nesse caso período-3, o espaço de fase passa três vezes pelo diagrama de Poincaré antes de voltar ao ponto inicial. Assim, o período desse sistema é $3*(2\pi/\Omega)$. Já o comportamento do sistema de $A = 0.9$, é similar, porém com período-2, logo, $P = 2*(2\pi/\Omega)$.

Por fim, os sistemas com $A = 0.6$ e 0.7 apresentam a nuvem com formato peculiar anteriormente mencionada e tem uma aparência que se assemelha a um fractal. Então podemos dizer que esses 2 sistemas apresentam comportamento potencialmente caótico.

4) Sensibilidade a condições iniciais

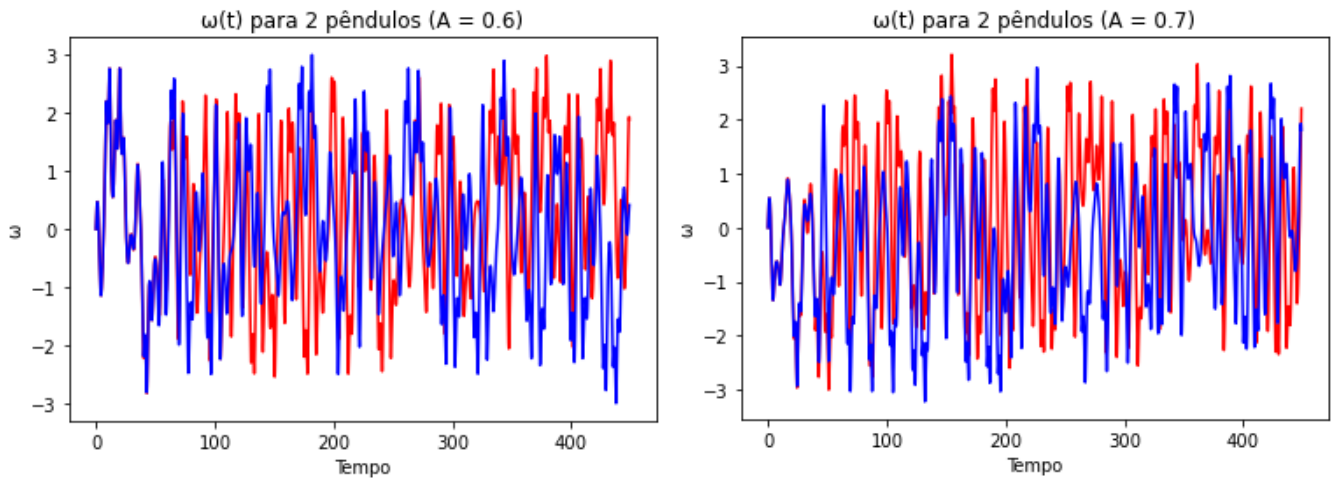
Simulando 2 pêndulos idênticos e usando 2 condições iniciais muito próximas, vamos observar o comportamento do $\omega(t)$ para os 2 sistemas. Usaremos os valores $\omega_1(0) = 0$ e $\theta_1(0) = 0$ para as condições iniciais do primeiro pêndulo e $\omega_2(0) = 0$ e $\theta_2(0) = 0.001$ para o segundo.

Utilizando primeiro um valor não caótico de A :



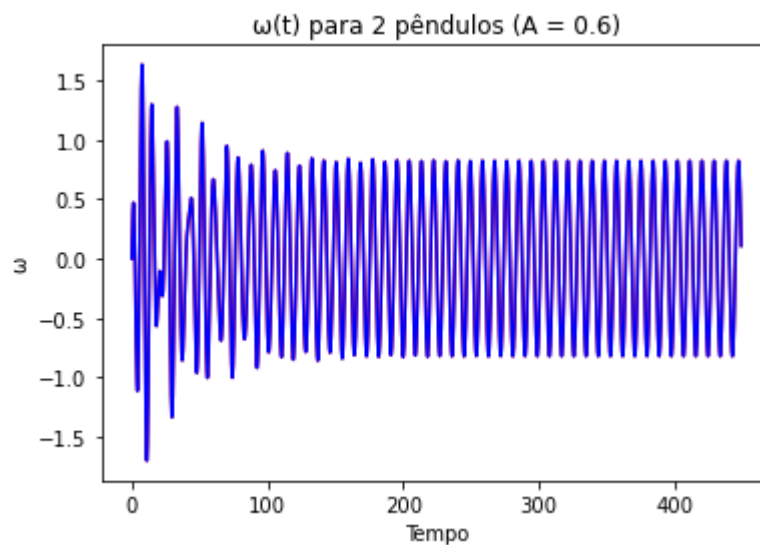
As soluções andam juntas, temos o pêndulo 1 vermelho e o pêndulo 2 azul. Então é possível ver que as soluções se sobrepõem.

Observando agora casos com A caótico:



Podemos ver que nenhum dos casos caóticos se assemelha ao resultado não caótico. Então é possível observar o quão sensíveis são as condições iniciais para um sistema caótico. Duas condições muito próximas deram resultados muito diferente com o tempo.

Por fim, alterando $\sin(\theta)$ por θ , retornamos com a linearidade do sistema, e obtemos esse resultado:



Ou seja, mesmo com um $A = 0.6$ que gera um sistema caótico para o sistema não-linear, ao retornar com a linearidade do sistema o comportamento potencialmente caótico some e passamos a enxergar as 2 soluções andando juntas e se sobrepondo novamente.

5) Conclusão

Nessa atividade foi possível observar bem o comportamento de sistemas não-lineares e como eles são sensíveis a pequenas diferenças de parâmetros e de condições iniciais. Mesmo para o sistema se comportando de forma periódica, diferentes condições iniciais fazem ele demorar mais ou menos tempo para atingir o estado estacionário. Além disso, uma pequena mudança de parâmetros pode alterar o comportamento do sistema de forma drástica e o levar a se comportar caoticamente.

Para os casos em que o sistema atinge potencialmente o caos, foi observado também que as condições iniciais não só interferem no sistema como mudam completamente o resultado atingido. Ao retornar com a linearidade do sistema, as condições iniciais e os parâmetros param de resultar em sistemas completamente diferentes com o passar do tempo.

A maior dificuldade encontrada foi relacionada às limitações da máquina para poder rodar o código com número muito alto de períodos, tendo que diminuir esse número para poder obter os gráficos desejados.

6) Códigos

i) Código utilizado para o quesito 2:

```
"""
```

```
Created on Fri Jul 22 09:14:23 2022
```

```
@author: Geraldo Siqueira
```

```
"""
```

```
import numpy as np
import matplotlib.pyplot as plt
```

```
b = 0.05
```

```
O = 0.7
```

```
A = 0.6
```

```
#((x+np.pi)%2*np.pi)-np.pi
```

```
def Fx(x,w,t):
    return w
```

```
def Fw(x,w,t):
    return -b*w - np.sin(x) + A*np.cos(O*t)
```

```
#Runge-Kutta de quarta ordem:
```

```
def rk4 (x, w, t, dt):
    for n in range(1,Nt):
        k1x = Fx(x[n-1], w[n-1], t[n-1])*dt
        k1w = Fw(x[n-1], w[n-1], t[n-1])*dt
```



```

k2x = Fx(x[n-1] + 0.5*k1x, w[n-1] + 0.5*k1w, t[n-1] + 0.5*dt)*dt
k2w = Fw(x[n-1] + 0.5*k1x, w[n-1] + 0.5*k1w, t[n-1] + 0.5*dt)*dt

```

```

k3x = Fx(x[n-1] + 0.5*k2x, w[n-1] + 0.5*k2w, t[n-1] + 0.5*dt)*dt
k3w = Fw(x[n-1] + 0.5*k2x, w[n-1] + 0.5*k2w, t[n-1] + 0.5*dt)*dt

```

```

k4x = Fx(x[n-1] + k3x, w[n-1] + k3w, t[n-1] + dt)*dt
k4w = Fw(x[n-1] + k3x, w[n-1] + k3w, t[n-1] + dt)*dt

```

```

x[n] = x[n-1] + (k1x + 2*k2x + 2*k3x + k4x)/6
#Para deixar theta entre -pi e pi:
x[n] = ((x[n]+np.pi)%(2*np.pi))-np.pi
w[n] = w[n-1] + (k1w + 2*k2w + 2*k3w + k4w)/6
return x, w

```

```

dt = 0.05
t = np.arange(0,300, dt)
Nt = t.size
x = np.zeros(Nt)
w = np.zeros(Nt)
Amp = np.array([])
T = np.array([])

```

```

x[0] = 1
w[0] = 0
rk4(x, w, t, dt)

```

#plotando os gráficos de θ x ω e de θ x tempo

```

fig, ax = plt.subplots(2)
ax[0].plot(x, w, '.')
ax[0].set_xlabel("θ")
ax[0].set_ylabel("ω")
ax[1].plot(t, x)
ax[1].set_xlabel("tempo")
ax[1].set_ylabel("θ")
'''

```

#plotando os gráficos de θ x ω e de θ x tempo parte estável

```

fig, ax = plt.subplots(2)
ax[0].plot(x[Nt//2:Nt-1], w[Nt//2:Nt-1], '.')
ax[0].set_xlabel("θ")
ax[0].set_ylabel("ω")
ax[1].plot(t[Nt//2:Nt-1], x[Nt//2:Nt-1], '.')
ax[1].set_xlabel("tempo")
ax[1].set_ylabel("θ")
'''

```

#Para estimar o tempo transiente

```

for i in range(1,Nt):
    if w[i]*w[i-1]<0:
        Amp = np.append(Amp, np.abs((x[i]+x[i-1])/2))
        T = np.append(T, t[i])

'''
plt.plot(T, Amp)
plt.title("Amplitudes máximas em função do tempo")
plt.xlabel("Tempo")
plt.ylabel("Módulo da Amplitude")
'''

```

ii) Código utilizado para o quesito 3:

```

''''
Created on Fri Jul 22 13:46:39 2022

@author: Geraldo Siqueira
''''

import numpy as np
import matplotlib.pyplot as plt
from numpy import pi
from numba import jit

b = 0.05
O = 0.7
A = 0.7
#Período => P = 2pi/Omega
P = 2*np.pi/O

@jit
def Fx(x,w,t):
    return w

@jit
def Fw(x,w,t):
    return -b*w - np.sin(x) + A*np.cos(O*t)

#Runge-Kutta de quarta ordem:
@jit
def rk4 (x, w, t, dt):
    for n in range(1,Nt):
        k1x = Fx(x[n-1], w[n-1], t[n-1])*dt
        k1w = Fw(x[n-1], w[n-1], t[n-1])*dt

```

```

k2x = Fx(x[n-1] + 0.5*k1x, w[n-1] + 0.5*k1w, t[n-1] + 0.5*dt)*dt
k2w = Fw(x[n-1] + 0.5*k1x, w[n-1] + 0.5*k1w, t[n-1] + 0.5*dt)*dt

k3x = Fx(x[n-1] + 0.5*k2x, w[n-1] + 0.5*k2w, t[n-1] + 0.5*dt)*dt
k3w = Fw(x[n-1] + 0.5*k2x, w[n-1] + 0.5*k2w, t[n-1] + 0.5*dt)*dt

k4x = Fx(x[n-1] + k3x, w[n-1] + k3w, t[n-1] + dt)*dt
k4w = Fw(x[n-1] + k3x, w[n-1] + k3w, t[n-1] + dt)*dt

x[n] = x[n-1] + (k1x + 2*k2x + 2*k3x + k4x)/6
#Para deixar theta entre -pi e pi:
x[n] = ((x[n]+np.pi)%(2*np.pi))-np.pi
w[n] = w[n-1] + (k1w + 2*k2w + 2*k3w + k4w)/6
return x, w

N_Periodos = 1000000
NP=100
dt = P/NP
t = np.arange(0,N_Periodos*P, dt)
Nt = t.size
x = np.zeros(Nt)
w = np.zeros(Nt)

x[0] = 1
w[0] = 0
rk4(x, w, t, dt)

#Para plotar o mapa de Poincaré
plt.title("Mapa de Poincaré para A = 0.9")
plt.plot(x[2*Nt//3:Nt-1:NP], w[2*Nt//3:Nt-1:NP], 'r,')
plt.xlim([-pi, pi])
plt.ylim([-3, 3])

```

iii) Código utilizado para o quesito 4:

Created on Fri Jul 29 08:44:41 2022

@author: Geraldo Siqueira

```

import numpy as np
import matplotlib.pyplot as plt
from numba import jit

```

```

b = 0.05
O = 0.7
A = 0.6
#Período => P = 2pi/Omega
P = 2*np.pi/O

```

```

@jit
def Fx(x,w,t):
    return w

```

```

@jit
def Fw(x,w,t):
    return -b*w - np.sin(x) + A*np.cos(O*t)

```

#Runge-Kutta de quarta ordem:

```

@jit
def rk4 (x, w, t, dt):
    for n in range(1,Nt):
        k1x = Fx(x[n-1], w[n-1], t[n-1])*dt
        k1w = Fw(x[n-1], w[n-1], t[n-1])*dt

        k2x = Fx(x[n-1] + 0.5*k1x, w[n-1] + 0.5*k1w, t[n-1] + 0.5*dt)*dt
        k2w = Fw(x[n-1] + 0.5*k1x, w[n-1] + 0.5*k1w, t[n-1] + 0.5*dt)*dt

        k3x = Fx(x[n-1] + 0.5*k2x, w[n-1] + 0.5*k2w, t[n-1] + 0.5*dt)*dt
        k3w = Fw(x[n-1] + 0.5*k2x, w[n-1] + 0.5*k2w, t[n-1] + 0.5*dt)*dt

        k4x = Fx(x[n-1] + k3x, w[n-1] + k3w, t[n-1] + dt)*dt
        k4w = Fw(x[n-1] + k3x, w[n-1] + k3w, t[n-1] + dt)*dt

        x[n] = x[n-1] + (k1x + 2*k2x + 2*k3x + k4x)/6
        #Para deixar theta entre -pi e pi:
        x[n] = ((x[n]+np.pi)%(2*np.pi))-np.pi
        w[n] = w[n-1] + (k1w + 2*k2w + 2*k3w + k4w)/6
    return x, w

```

```

N_Periodos = 50
NP=100
dt = P/NP
t = np.arange(0,N_Periodos*P, dt)
Nt = t.size
x1 = np.zeros(Nt)
w1 = np.zeros(Nt)
x2 = np.zeros(Nt)
w2 = np.zeros(Nt)

```

#Condições iniciais do pêndulo 1

```
x1[0] = 0
w1[0] = 0
#Condições iniciais do pêndulo 2
x2[0] = 0
w2[0] = 0.001
rk4(x1, w1, t, dt)
rk4(x2, w2, t, dt)

#Para plotar o mapa de Poincaré
plt.title("ω(t) para 2 pêndulos (A = 0.6)")
plt.plot(t, w1, 'r')
plt.plot(t, w2, 'b')
plt.xlabel("Tempo")
plt.ylabel("ω")
#plt.xlim([-pi, pi])
#plt.ylim([-3, 3])
```