



Programa de Pós-Graduação
Disciplina: Mecânica de Sistemas Inteligentes - COM783

Professor: Luã Guedes Costa

Aluno: Caio Alves Barcellos de Oliveira - DRE:124300813

Data de entrega: 25/08/2025

Lista I - Dinâmica Não linear

Questão 1:

Em sua grande maioria, sistemas dinâmicos não lineares não possuem soluções analíticas, ou, quando possuem, estas têm um escopo restrito. Por isso, é necessário recorrer a métodos numéricos para avaliar o comportamento desses sistemas. Nesse contexto, implemente os seguintes métodos numéricos:

- Runge-Kutta de quarta ordem (RK4) de passo fixo.
- Runge-Kutta-Dormand-Prince (DOPRI45) adaptativo de ordens 4 e 5, com controle automático de erro.

A implementação deve ser feita de forma generalizada, isto é, abstraída em funções ou classes. Explique por que essa abordagem é importante no contexto de modelagem computacional. Para validar os métodos, simule um oscilador harmônico linear descrito pela equação a seguir e compare os resultados obtidos com sua solução analítica correspondente. Faça uma análise de convergência da solução.

$$\ddot{x} + 2\zeta\omega_n\dot{x} + \omega_n^2x = \gamma \sin(\Omega t)$$

Por fim, discuta em quais contextos a utilização desses métodos é mais adequada e, nos casos em que não forem recomendados, quais outros métodos poderiam ser empregados como alternativa.

Solução

Para sistemas não lineares, a implementação de soluções numéricas são recorrentes. Portanto, o uso de soluções generalizadas extratificadas permite a implementação de um mesmo método numérico em diversos sistemas, sem modificar o código do método em si. Além disso, facilita a correção de erros e entendimento do código para terceiros.

Para a validação do método, foram feitas duas simulações principais, no entanto os parâmetros gerais para as duas são:

- Fator de amortecimento $\zeta = 0.5$
- Frequência natural $\omega_n = 1.0$
- Amplitude do forçamento $\gamma = 1.0$
- Frequência do forçamento $\Omega = 2.0$
- Posição inicial $x_0 = 1.0$

- Velocidade inicial $v_0 = 0.0$

A primeira simulação tem como parâmetros base o passo fixo para o método Runge Kutta de 4^a ordem (RK4) em $h = 0.001$ e tempo de duração da simulação em 50 segundos. Além disso, a tolerância do teste feito no método RungeKutta Dormand Prince (DOPRI45) foi de 10^{-6} . O resultado dos erros absolutos em escala logarítmica são expostos no gráfico abaixo. É possível notar uma estabilização mais rápida do erro no integrador DOPRI45, por volta de 36 segundos, enquanto o RK4 apresenta contínua redução do erro e não estabilização.

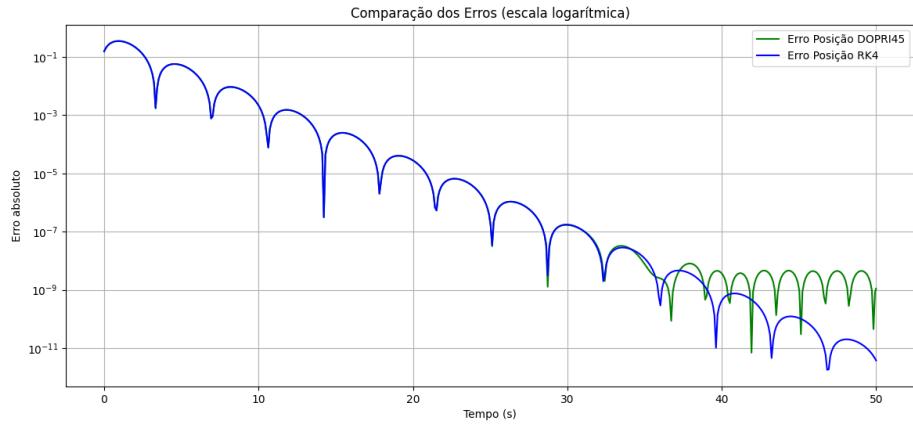


Figure 1: Erro absoluto (simulação 1).

Enquanto, ao aumentar o tempo para 100 segundos, ambos os métodos apresentam estabilidade no erro. No entanto, o erro absoluto do método RK4 é consideravelmente menor, embora sua estabilização ocorra aproximadamente 20 segundos após o DOPRI45.

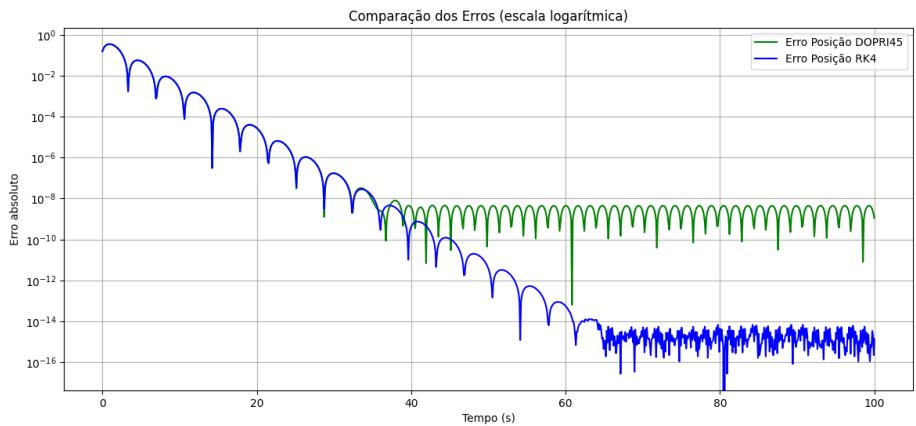


Figure 2: Erro absoluto (simulação 2).

Apesar disso, o resultado da simulação da posição x no tempo para ambos os casos é bem próxima, quase não sendo possível distinguir diferença na convergência para o resultado analítico, como mostram as figuras a seguir.

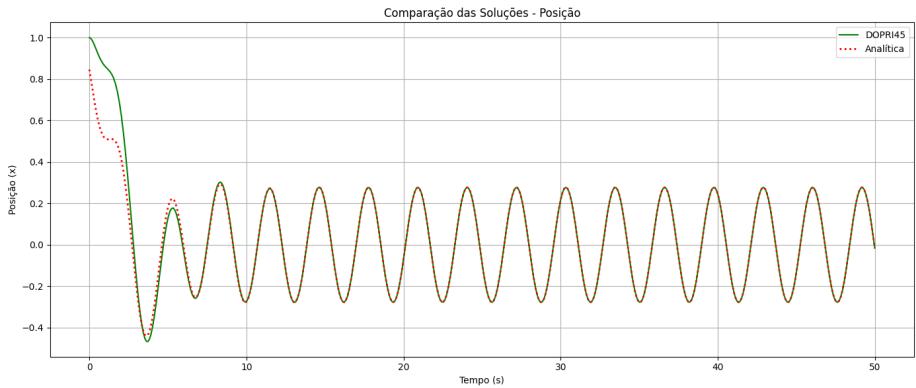


Figure 3: Posição no tempo para RK4 (simulação 1).

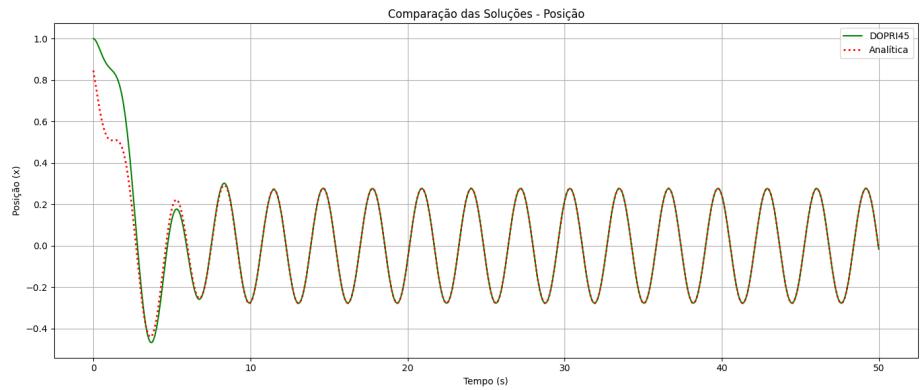


Figure 4: Posição no tempo para DOPRI45 (simulação 1).

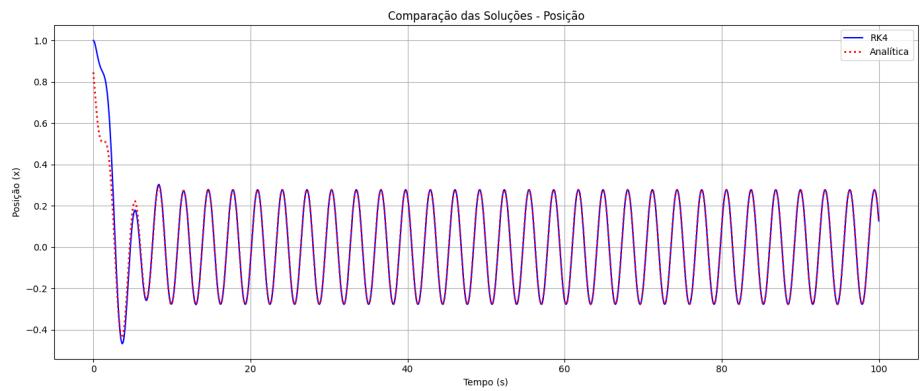


Figure 5: Posição no tempo para RK4 (simulação 2).

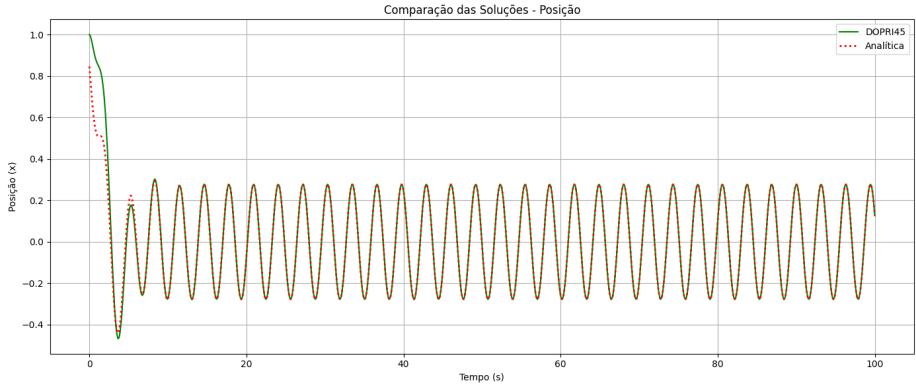


Figure 6: Posição no tempo para DOPRI45 (simulação 2).

Vale ressaltar que o método DOPRI45 variou entre passo mínimo de $h = 0.001$ e máximo de $h = 0.1$. Além disso, concluiu a simulação de 50 segundos em 503 passos e de 100 segundos em 1003 passos, enquanto o RK4 concluiu a simulação de 50 segundos em 50.000 passos e a de 100 segundos em 100.000 passos.

Por se tratar de um sistema simples, com apenas duas variáveis de estado (posição e velocidade), a simulação parece vantajosa a longo prazo para o RK4, visto que para um passo fixo menor do que o variável do DOPRI45, obteve erro estabilizado menor.

No entanto, para sistemas mais complexos o impacto no custo computacional deve ser levado em conta e o DOPRI45 se torna mais vantajoso.

Para efeito de comparação, foi feita a mesma simulação de 50 segundos, mas com o passo fixo igual ao passo máximo do DOPRI45. O resultado do erro e a comparação entre os gráficos das posições são apresentados abaixo.

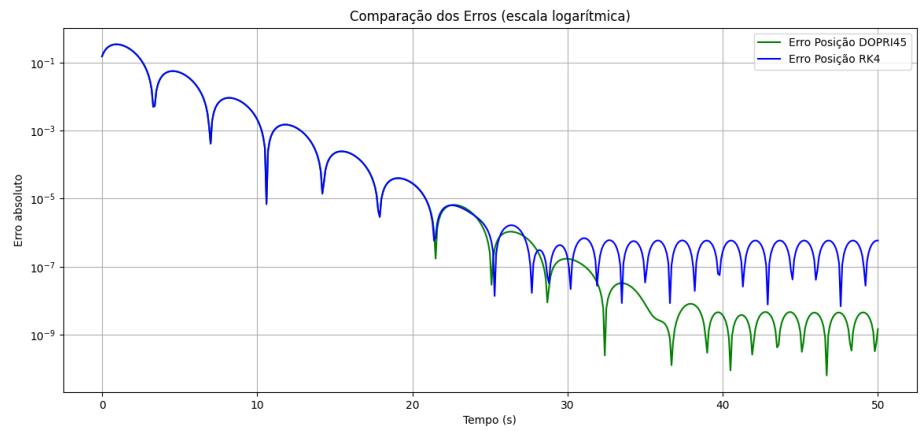


Figure 7: Erro absoluto (simulação 3).

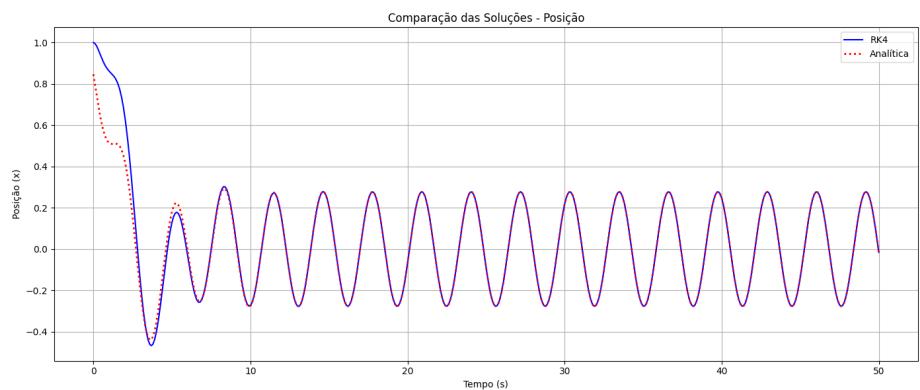


Figure 8: Posição no tempo para RK4 (simulação 3).

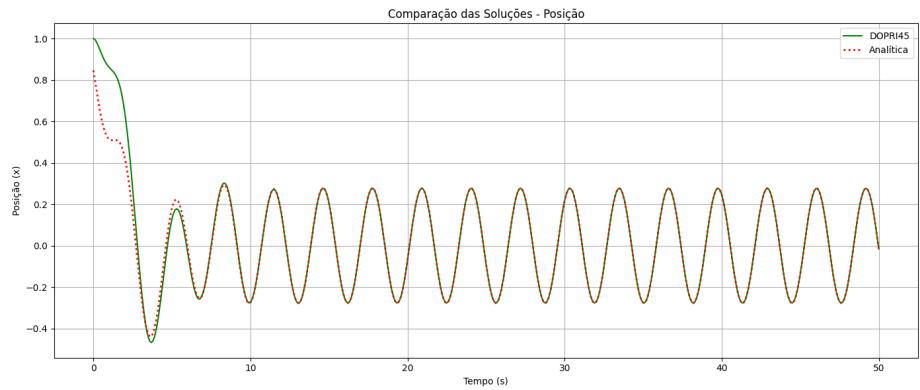


Figure 9: Posição no tempo para DOPRI45 (simulação 3).

Como resultado, é visivel que para um passo fixo maior, o método RK4 carece de precisão aumentando seu erro absoluto. Portanto, é possível concluir que o método depende da aplicação. O RK4 tem uma implementação mais simples, no entanto pode precisar de passos mais finos para obter um erro menor, além de ser mais lento em relação ao DOPRI45, que tem uma implementação mais complexa mas é visivelmente melhor para sistemas complexos.

Questão 2:

O Mapa de Poincaré é uma ferramenta extremamente útil para avaliar a dinâmica de um sistema. Existem formas distintas de obtenção desses mapas, dependendo do sistema dinâmico. Implemente um procedimento numérico para obter o mapa de Poincaré para o caso particular de sistemas com excitação harmônica. Para isso, siga os passos descritos abaixo:

(a)

Com auxílio do método de Runge-Kutta de quarta ordem (RK4) para solução dos sistemas dinâmicos, avalie os mapas de Poincaré dos seguintes sistemas:

- Oscilador linear não dissipativo:

$$\ddot{x} + \omega_n^2 x = \gamma \sin(\Omega t)$$

- Oscilador biestável tipo Duffing:

$$\ddot{x} + 2\zeta \dot{x} - \alpha x + \beta x^3 = \gamma \sin(\Omega t)$$

- Pêndulo simples:

$$\ddot{\phi} + \zeta \dot{\phi} + \omega_n^2 \sin(\phi) = \gamma \sin(\Omega t)$$

No caso do oscilador linear não dissipativo, avalie diferentes relações entre a frequência natural, ω_n , e frequência de forçamento adimensional, Ω . Para os casos do oscilador tipo Duffing e o pêndulo simples, avalie diferentes combinações entre amplitude de forçamento adimensional, γ , e frequência de forçamento adimensional, Ω . Adicionalmente, discuta a topologia do espaço de fase de cada caso.

(b)

Tente implementar o item anterior utilizando o método Runge-Kutta-Dormand-Prince (DOPRI45) para solucionar os sistemas dinâmicos e obter seus respectivos mapas de Poincaré. Discuta as principais dificuldades durante a implementação, e, se necessário, descreva as ferramentas adicionais utilizadas para uma implementação bem-sucedida.

Solução

a) Runge Kutta de 4^a ordem

Para todos os problemas abaixo, foi utilizado o integrador numérico RK4 com passo fixo de 0.01 e tempo médio de 15 segundos para cada simulação.

Foram feitas simulações de 200 segundos para cada modelo e para remover o transiente, foram retirados 5 períodos no plot do mapa de Poincaré, exceto um caso que será comentado a seguir.

Oscilador linear não dissipativo

$$\ddot{x} + \omega_n^2 x = \gamma \sin(\Omega t)$$

Utilizando um código em python, é necessário escrever o sistema em equações de estado. **Considere o sistema descrito no seguinte formato:**

Definindo as variáveis de estado $u = x$ e $v = \dot{x}$, obtemos:

$$\begin{cases} \dot{u} = v \\ \dot{v} = -\omega_n^2 u + \gamma \sin(\Omega t) \end{cases}$$

Variando os valores de ω_n , e frequência de forçamento Ω , com as seguintes condições:

- $u_0 = 0.0$
- $v_0 = 0.0$
- $\omega_n = [0.5, 1.0]$
- $\Omega = [0.5, 1.0, 1.5, 2.0]$
- $\gamma = 1.0$

Os resultados para $\omega_n = 0.5$ variando o valor de Ω foram:

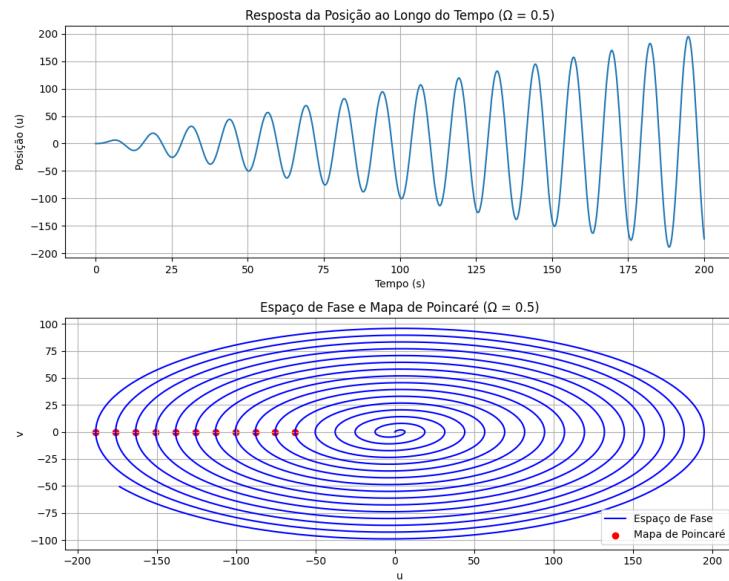


Figure 10: Mapa de Poincaré para Ω de 0.5

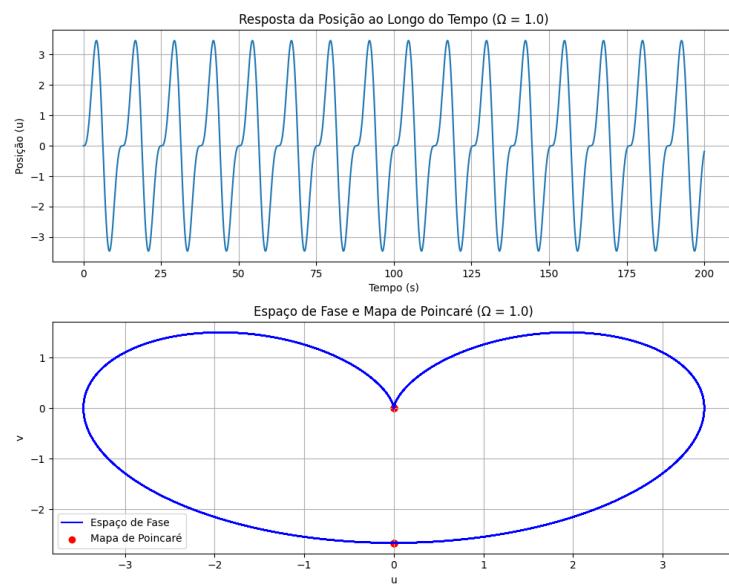


Figure 11: Mapa de Poincaré para Ω de 1.0

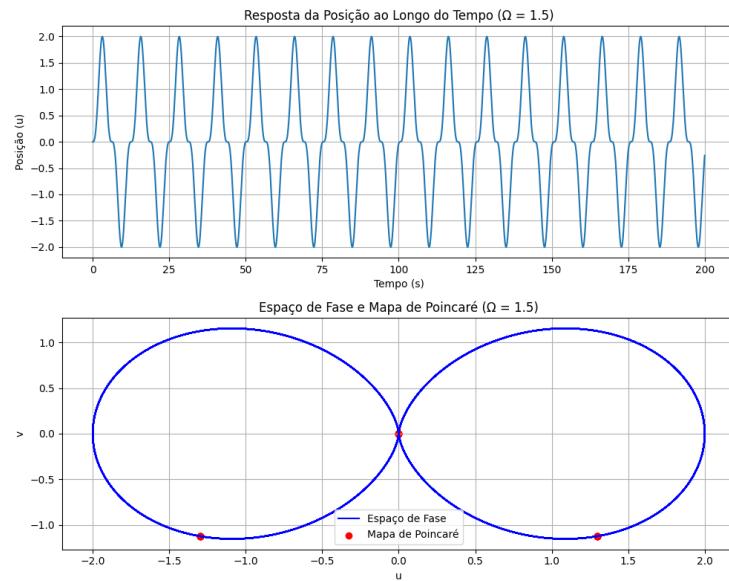


Figure 12: Mapa de Poincaré para Ω de 1.5

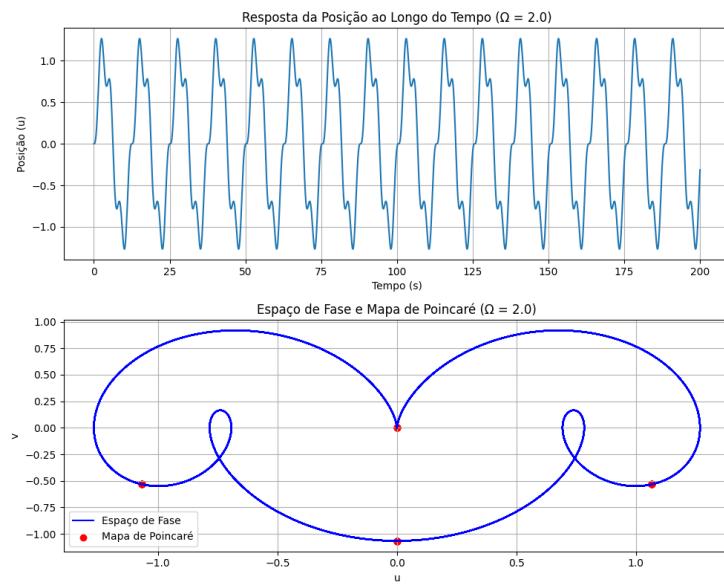


Figure 13: Mapa de Poincaré para Ω de 2.0

Os resultados para $\omega_n = 1.0$ variando o valor de Ω foram:

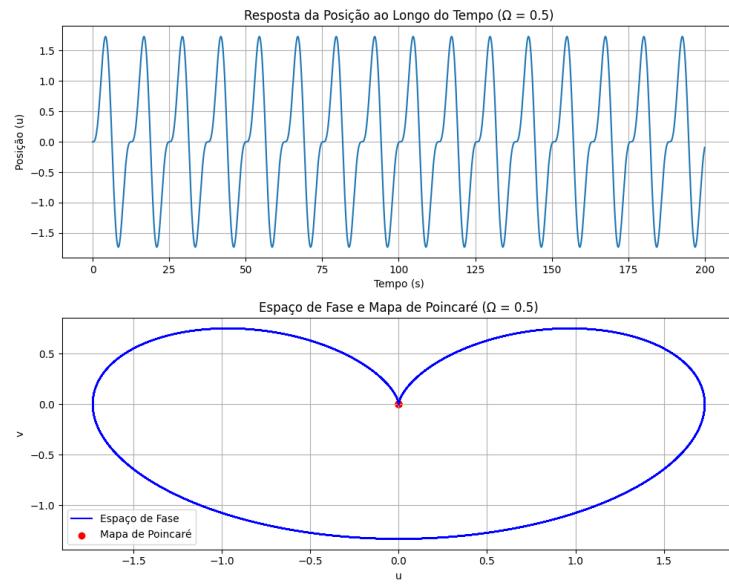


Figure 14: Mapa de Poincaré para Ω de 0.5

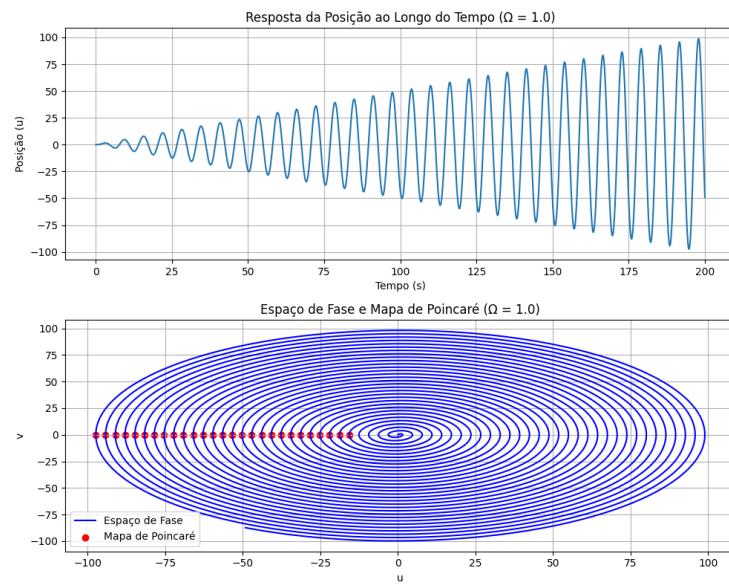


Figure 15: Mapa de Poincaré para Ω de 1.0

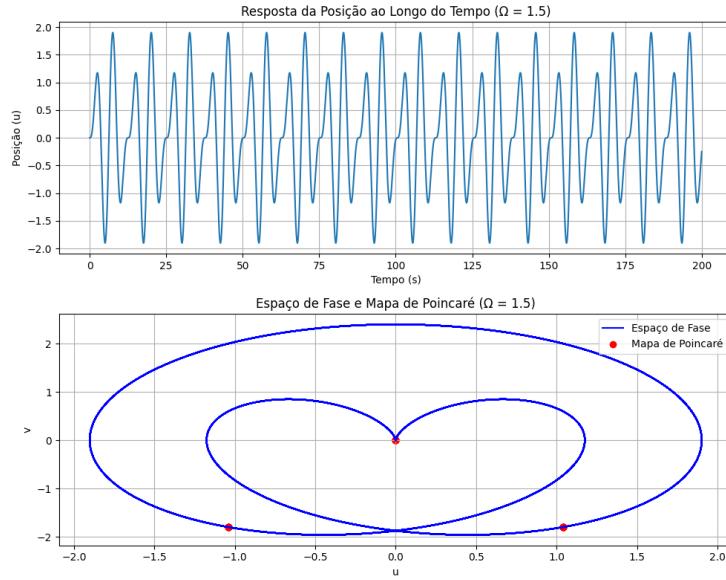


Figure 16: Mapa de Poincaré para Ω de 1.5

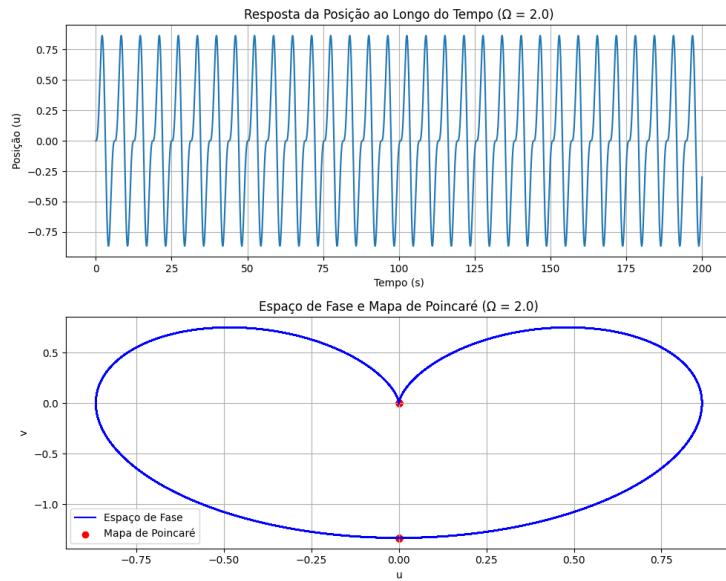


Figure 17: Mapa de Poincaré para Ω de 2.0

Devido a ausência de amortecimento, este sistema apresenta uma maior riqueza no sentido de mais períodos, comparado com os modelos abaixo. Além

disso, é interessante notar que quando a frequência de forçamento é igual a frequência natural do sistema ($\Omega = \omega_n$), o sistema entra em ressonância, sua amplitude aumenta de forma indefinida e o sistema se torna aperiódico com crescimento linear.

Oscilador biestável tipo Duffing

$$\ddot{x} + 2\zeta\dot{x} - \alpha x + \beta x^3 = \gamma \sin(\Omega t)$$

Utilizando um código em python, é necessário escrever o sistema em equações de estado. **Considere o sistema descrito no seguinte formato:**

Definindo as variáveis de estado $u = x$ e $v = \dot{x}$, obtemos:

$$\begin{cases} \dot{u} = v \\ \dot{v} = -2\zeta v + \alpha u - \beta u^3 + \gamma \sin(\Omega t) \end{cases}$$

Variando os valores de γ , e frequência de forçamento Ω , com as seguintes condições:

- $u_0 = 0.0$
- $v_0 = 0.0$
- $\alpha = 1.0$
- $\beta = 1.0$
- $\zeta = 0.5$
- $\Omega = [0.5, 1.0, 1.5]$
- $\gamma = [0.5, 1.0]$

Os resultados para $\gamma = 0.5$ variando o valor de Ω foram:

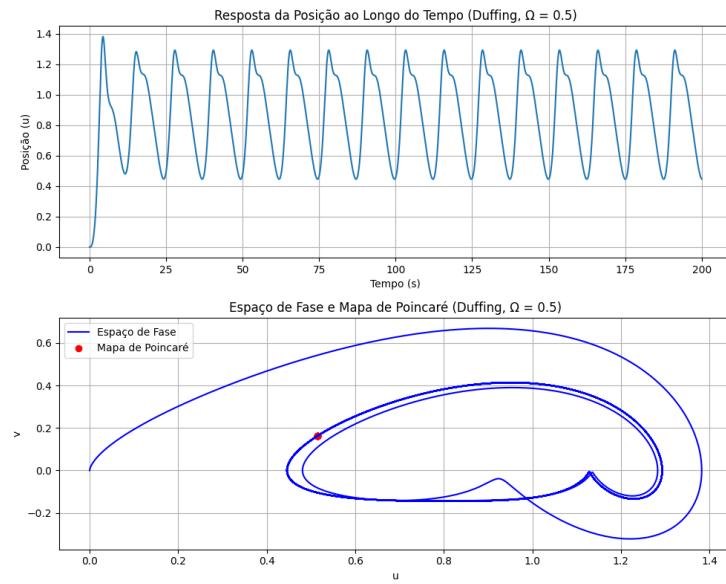


Figure 18: Mapa de Poincaré para Ω de 0.5

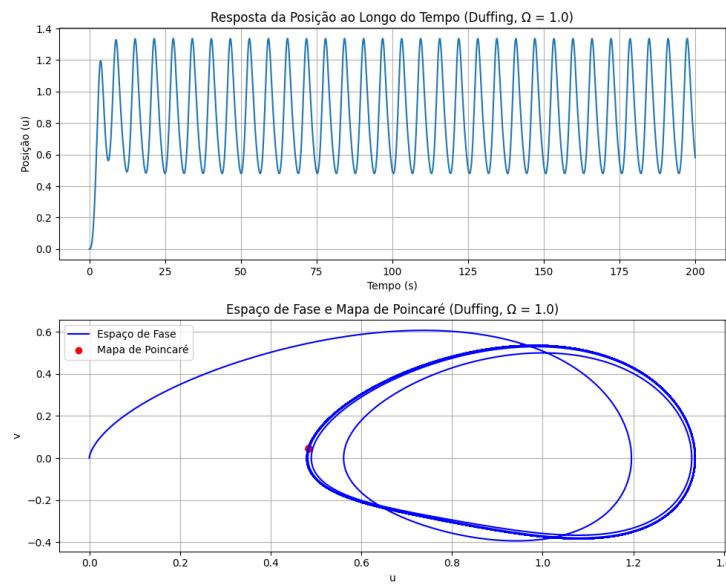


Figure 19: Mapa de Poincaré para Ω de 1.0

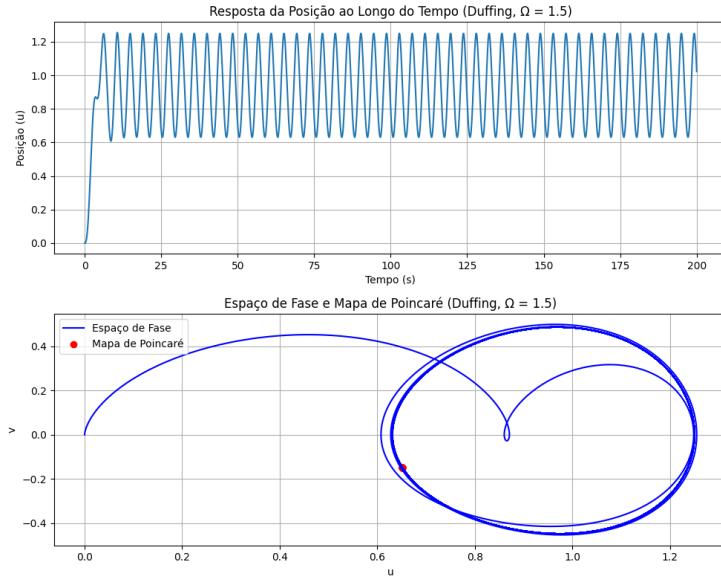


Figure 20: Mapa de Poincaré para Ω de 1.5

É interessante notar que para este valor de forçamento, o resultado de todos os casos é um período 1.

Os resultados para $\gamma = 1.0$ variando o valor de Ω foram:

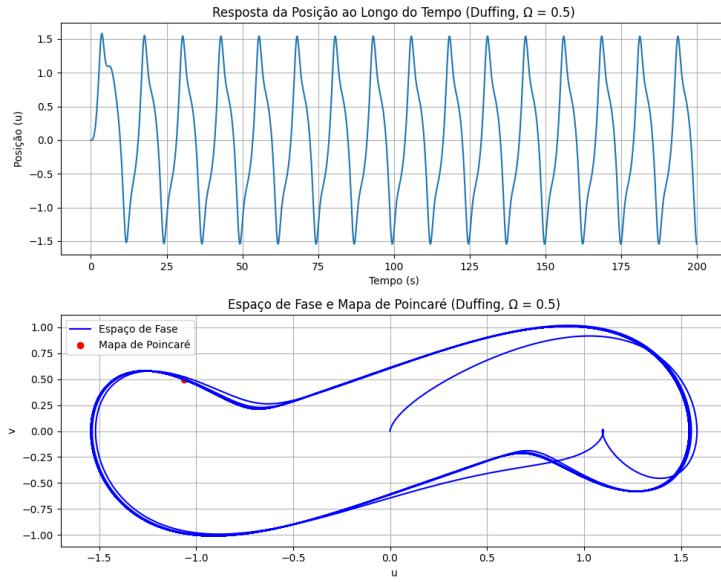


Figure 21: Mapa de Poincaré para Ω de 0.5

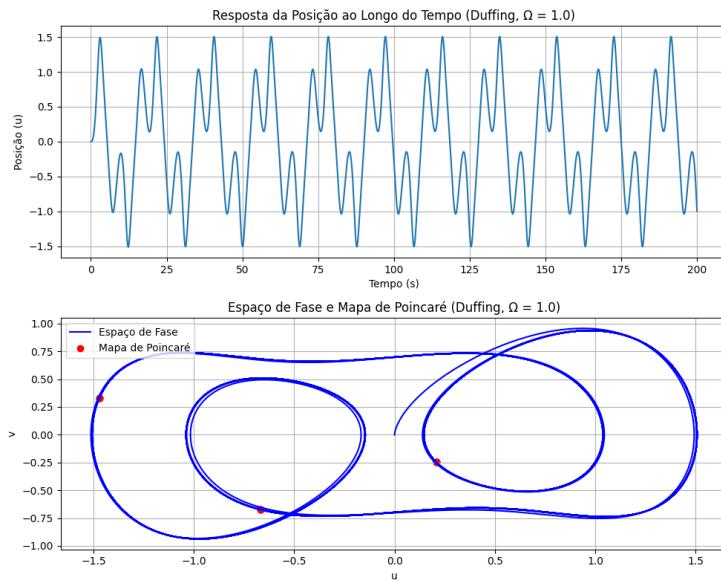


Figure 22: Mapa de Poincaré para Ω de 1.0

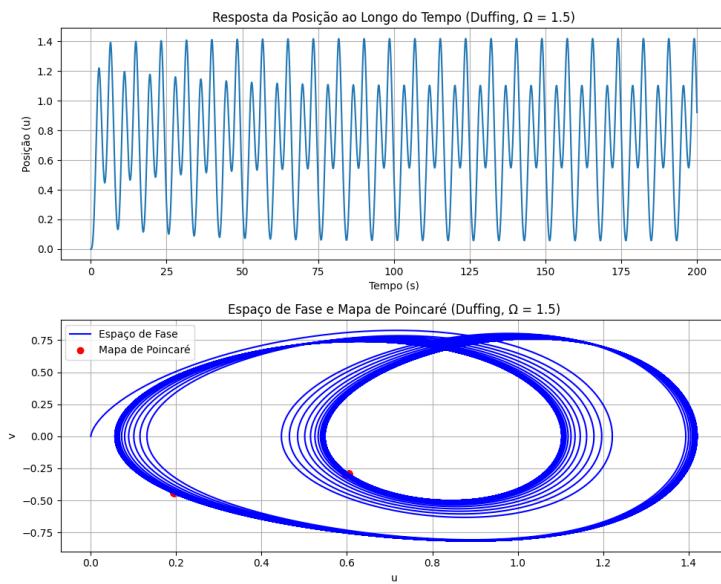


Figure 23: Mapa de Poincaré para Ω de 1.5

Diferente do caso anterior, ao aumentar a amplitude de forçamento o resultado deixa de ter período 1 para $\Omega = [1.0, 1.5]$.

É interessante notar que o transiente neste caso foi maior, enquanto no caso de $\gamma = 0.5$ foi utilizado um descarte dos 5 primeiros períodos, neste foi necessário o descarte dos 15 primeiros, apesar de isso só ser notado em $\Omega = 1.5$.

Pêndulo simples

$$\ddot{\phi} + \zeta \dot{\phi} + \omega_n^2 \sin(\phi) = \gamma \sin(\Omega t)$$

Utilizando um código em python, é necessário escrever o sistema em equações de estado. **Considere o sistema descrito no seguinte formato:**

Definindo as variáveis de estado $u = \phi$ e $v = \dot{\phi}$, obtemos:

$$\begin{cases} \dot{u} = v \\ \dot{v} = -\zeta v - \omega_n^2 \sin u + \gamma \sin(\Omega t) \end{cases}$$

Variando os valores de ω_n , e frequência de forçamento Ω , com as seguintes condições:

- $u_0 = 0.0$
- $v_0 = 0.0$
- $\omega_n = 1.0$
- $\Omega = [0.5, 1.0, 1.5]$
- $\zeta = 0.5$
- $\gamma = [0.5, 1.0]$

Os resultados para $\gamma = 0.5$ variando o valor de Ω foram:

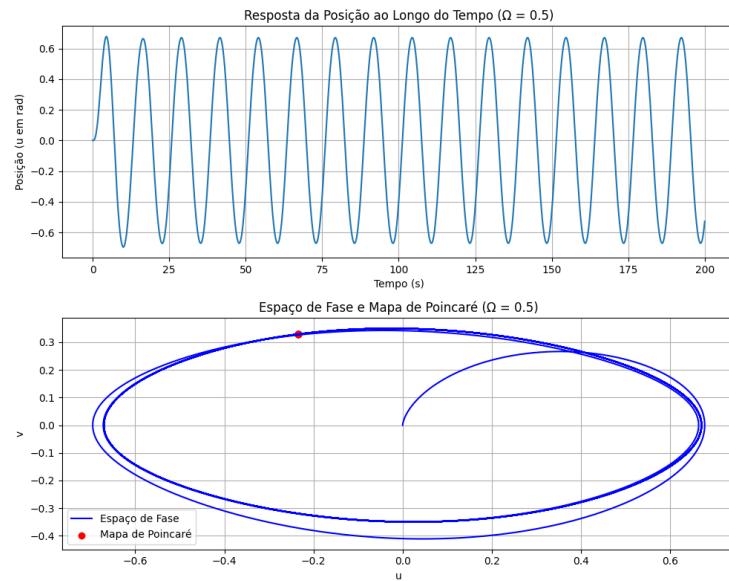


Figure 24: Mapa de Poincaré para Ω de 0.5

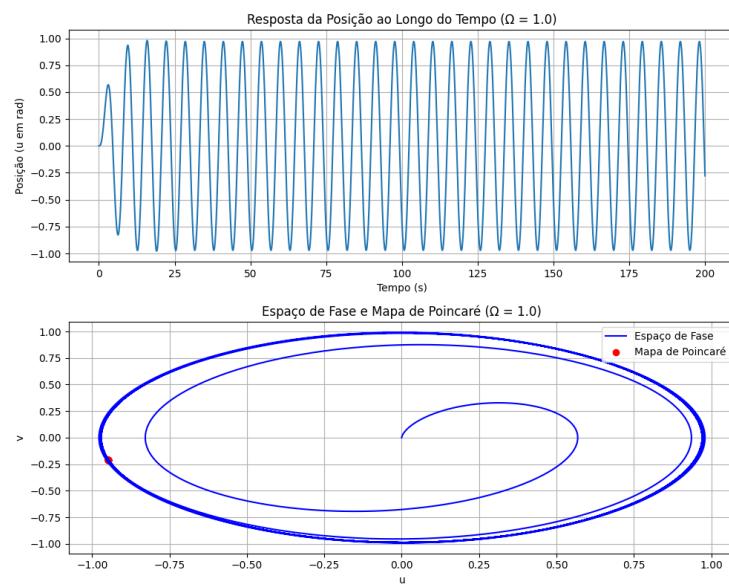


Figure 25: Mapa de Poincaré para Ω de 1.0

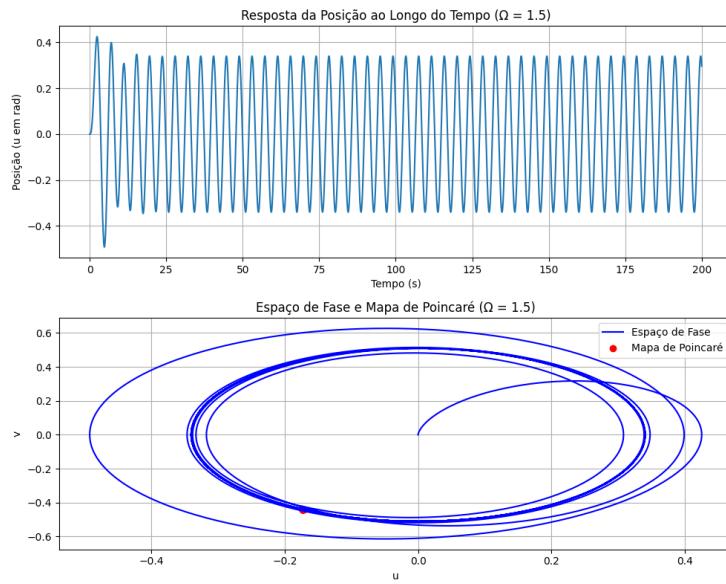


Figure 26: Mapa de Poincaré para Ω de 1.5

Os resultados para $\gamma = 1.0$ variando o valor de Ω foram:

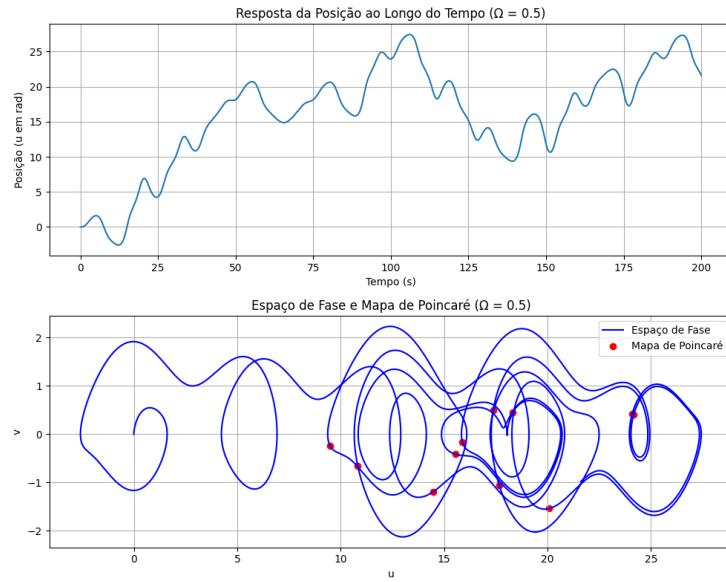


Figure 27: Mapa de Poincaré para Ω de 0.5

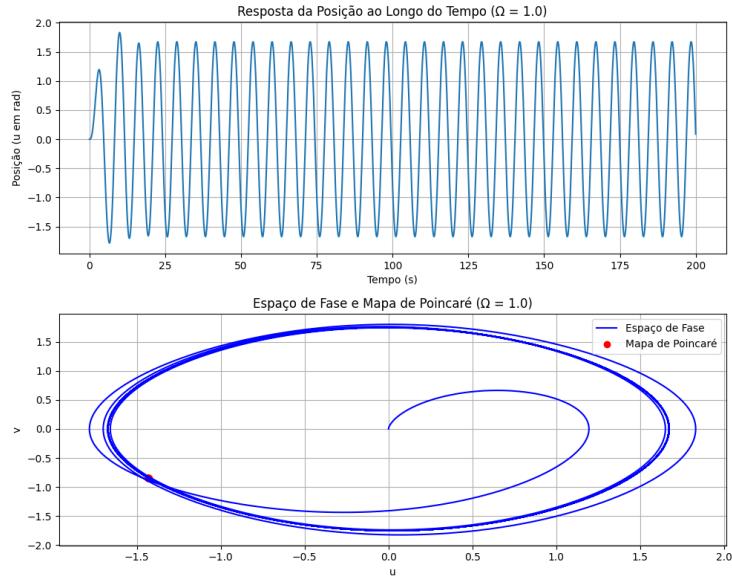


Figure 28: Mapa de Poincaré para Ω de 1.0

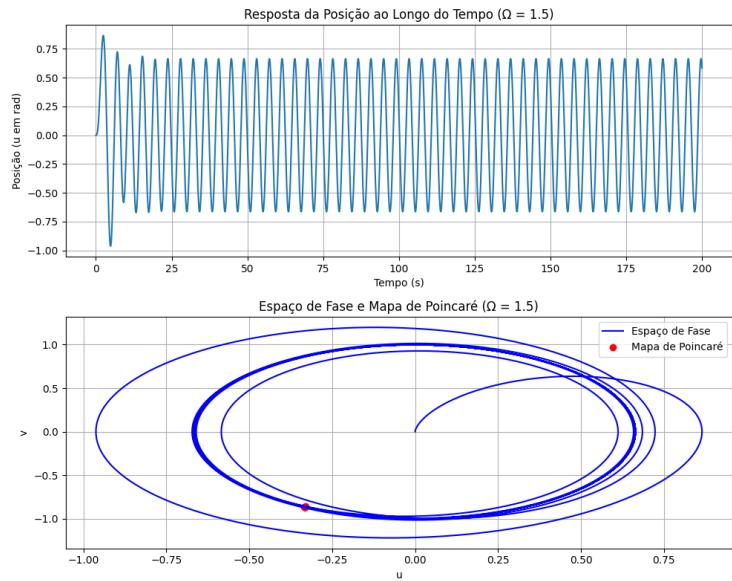


Figure 29: Mapa de Poincaré para Ω de 1.5

É interessante notar que ao aumentar a amplitude de forçamento γ para 1.0, o sistema para $\Omega = 0.5$ é o único que deixa de ter período 1 e se torna aperiódico,

enquanto os outros resultados seguem em período 1, tanto para $\gamma = 0.5$ quanto $\gamma = 1.0$. Isso se deve ao baixo fator de amortecimento, onde mesmo se aumentar o tempo o resultado segue aperiódico.

b) DOPRI45

No caso do Runge Kutta de 4^a ordem, o passo fixo viabilizava a análise do mapa de Poincaré nos termos $t_p = nT$, o passo h fixo permitia os instantes do integrador serem iguais aos do forçamento. Contudo, o DOPRI45 torna necessário uma interpolação, pois o passo variável não permite o tempo do forçamento ser o mesmo do passo (pois este é variável).

Neste caso, a alteração no código foram a inserção do novo integrador e um procedimento em que o mapa é construído no tempo exato do forçamento, mesmo que naquele instante o integrador não tenha varrido o ponto, tanto por ter um passo maior, quanto por ter um passo menor. Ou seja, é utilizado o intervalo da integração adaptativa e é avaliado o instante em que o tempo do forçamento estaria e é feita uma interpolação linear do tipo: $y_p = y_1 + \frac{t_p - t_1}{t_2 - t_1} (y_2 - y_1)$, onde $y(1) = y_1$ é o valor do sistema em t_1 , $y(2) = y_2$ é o valor do sistema em t_2 e t_p seria o tempo do forçamento.

O problema disso se encontra em assumir que a solução varia linearmente entre dois pontos consecutivos. Porém, se o passo for razoavelmente grande, e a dinâmica variar rápido demais, a interpolação linear pode induzir a um erro pequeno e a precisão decairia, gerando pontos próximos uns dos outros.

Para todos os problemas abaixo, foi utilizado o integrador numérico DOPRI com passo variável inicial de 0.01 e tolerância de 10^{-6} e tempo médio de 2 segundos para cada simulação.

Foram feitas simulações de 200 segundos para cada modelo e para remover o transiente, foram retirados 5 períodos no plot do mapa de Poincaré, exceto um caso que será comentado a seguir.

Oscilador linear não dissipativo

$$\ddot{x} + \omega_n^2 x = \gamma \sin(\Omega t)$$

Utilizando o mesmo procedimento anterior, escrevendo as equações de estado e seguindo os parâmetros abaixo: Variando os valores de ω_n , e frequência de forçamento Ω , com as seguintes condições:

- $u_0 = 0.0$
- $v_0 = 0.0$

- $\omega_n = [0.5, 1.0]$
- $\Omega = [0.5, 1.0, 1.5]$
- $\gamma = 1.0$

Os resultados para $\omega_n = 0.5$ variando o valor de Ω foram:

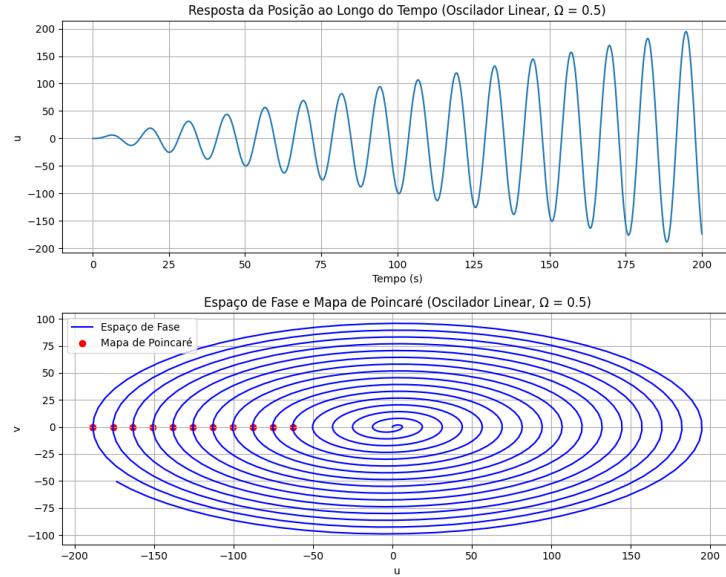


Figure 30: Mapa de Poincaré para Ω de 0.5

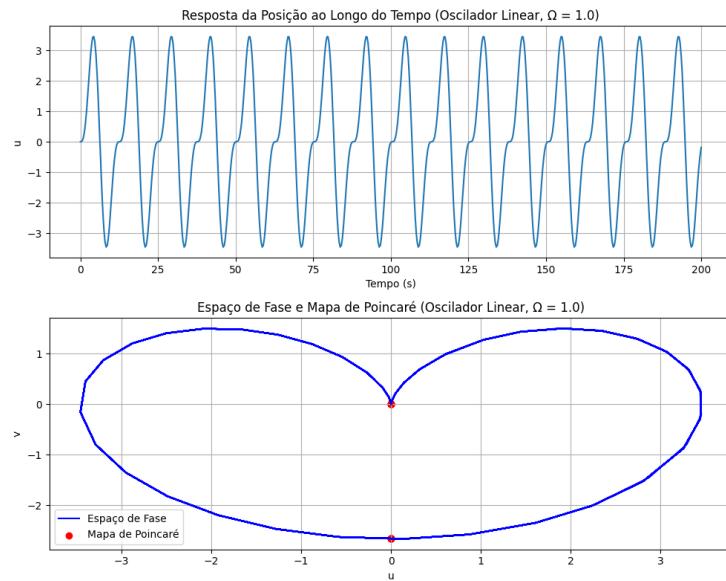


Figure 31: Mapa de Poincaré para Ω de 1.0

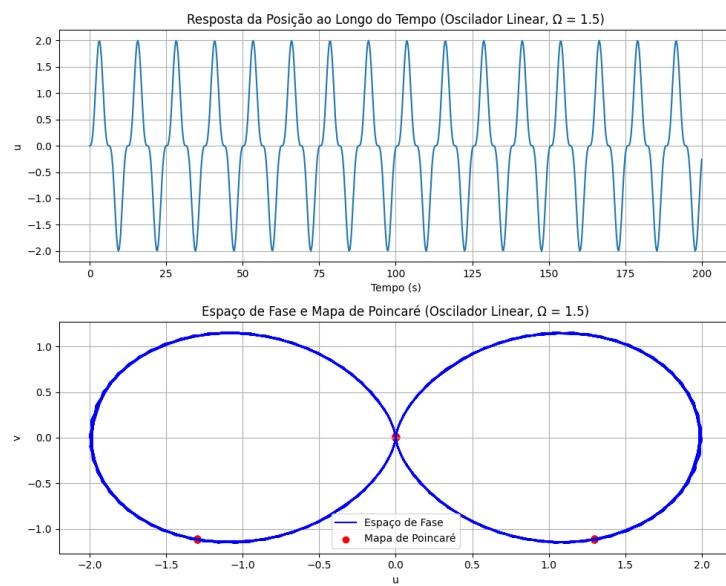


Figure 32: Mapa de Poincaré para Ω de 1.5

Os resultados para $\omega_n = 1.0$ variando o valor de Ω foram:

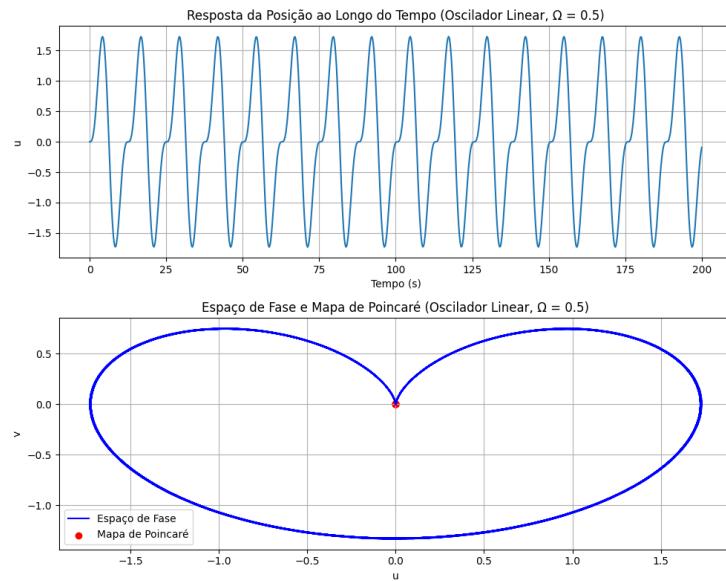


Figure 33: Mapa de Poincaré para Ω de 0.5

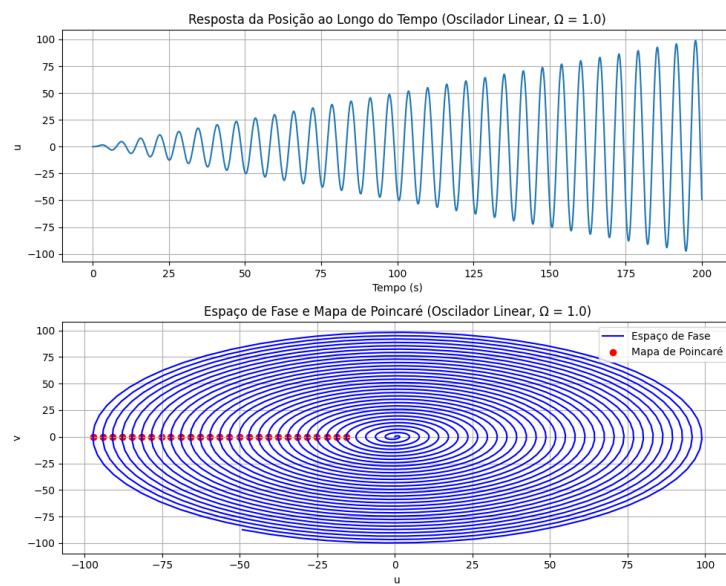


Figure 34: Mapa de Poincaré para Ω de 1.0

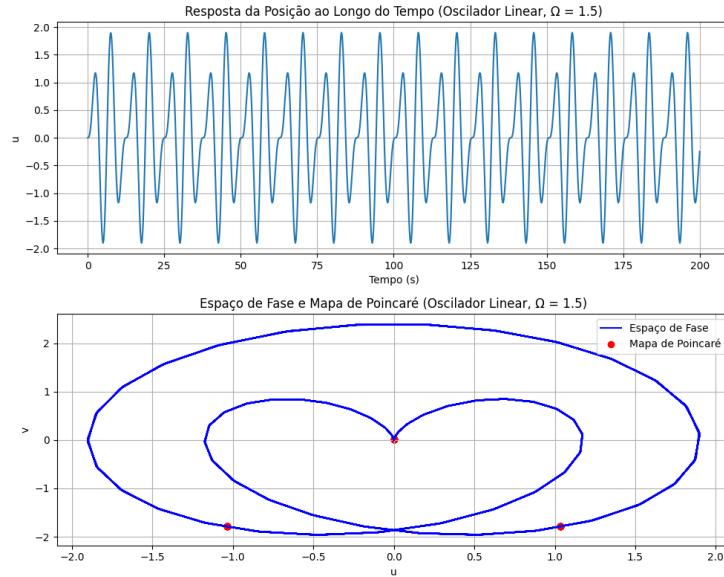


Figure 35: Mapa de Poincaré para Ω de 1.5

É interessante notar que apesar de ter obtido os pontos do Mapa de Poincaré de forma correta, o espaço de fase apresenta uma redução na qualidade do traço em relação ao passo fixo. Isso pode estar relacionado ao aumento do passo adaptativo.

Oscilador biestável tipo Duffing

$$\ddot{x} + 2\zeta\dot{x} - \alpha x + \beta x^3 = \gamma \sin(\Omega t)$$

Utilizando o mesmo procedimento anterior, escrevendo as equações de estado e seguindo os parâmetros abaixo: Variando os valores de γ , e frequência de forçamento Ω , com as seguintes condições:

- $u_0 = 0.0$
- $v_0 = 0.0$
- $\alpha = 1.0$
- $\beta = 1.0$
- $\zeta = 0.1$
- $\Omega = [0.5, 1.0, 1.5]$
- $\gamma = [0.5, 1.0]$

Para o sistema de Duffing, o transiente se torna longo e foi necessário utilizar um tempo de 1000 segundos para haver a possibilidade de visualizar a periodicidade. Além disso, houve um aumento da exclusão de 5 para 15 períodos removidos e assim o resultado ocorreria sem o transiente.

Os resultados para $\gamma = 0.5$ variando o valor de Ω foram:

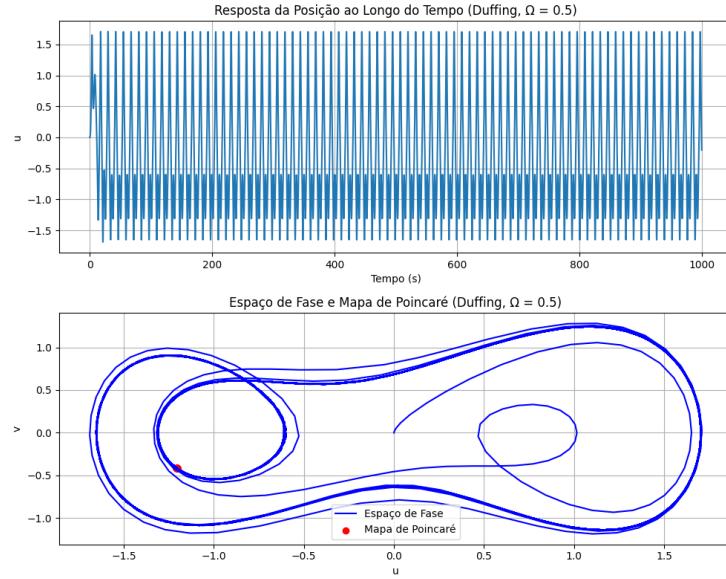


Figure 36: Mapa de Poincaré para Ω de 0.5

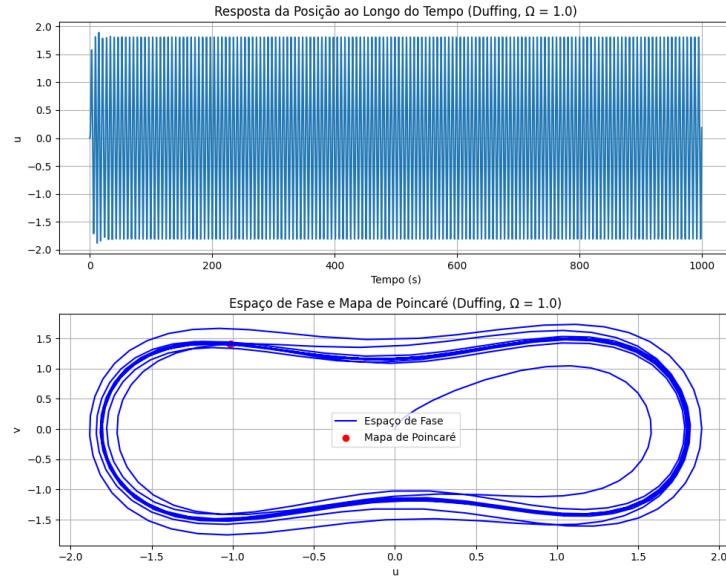


Figure 37: Mapa de Poincaré para Ω de 1.0

É interessante notar que a dinâmica abaixo tem um período longo além do transiente, e mesmo assim, por conta do passo adaptativo, se torna inviável em relação ao passo fixo analisar para um aumento de frequência de forçamento.

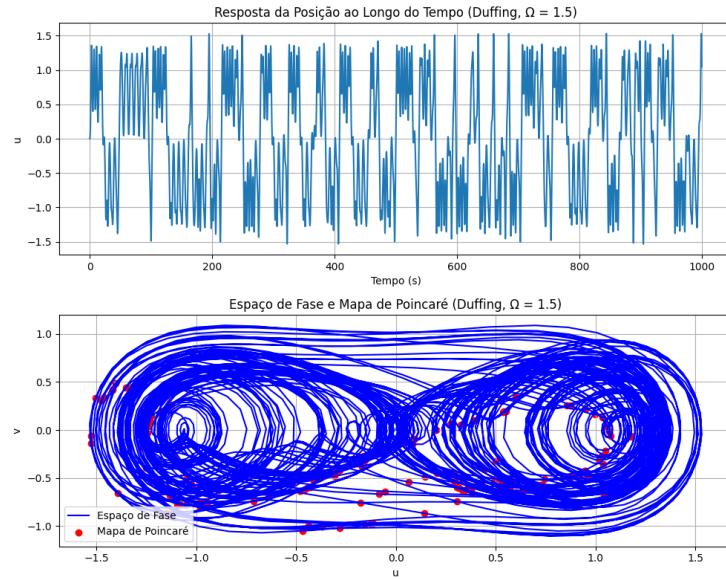


Figure 38: Mapa de Poincaré para Ω de 1.5

Ao aumentar a amplitude de forçamento, é verificado que para a frequência de forçamento de $\Omega = 0.5$ o método também se torna pouco efetivo, enquanto $\Omega = 1.0$ foi possível obter um resultado condizente com o de passo fixo e, por fim, quando aumenta $\Omega = 1.5$ o sistema apresenta um transiente muito longo e pontos muito próximos, exibindo a dificuldade do método em convergir a interpolação.

Os resultados para $\gamma = 1.0$ variando o valor de Ω foram:

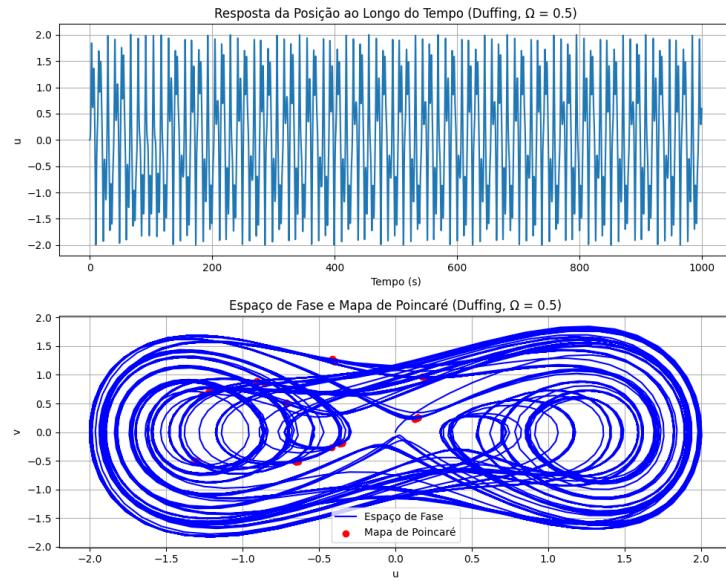


Figure 39: Mapa de Poincaré para Ω de 0.5

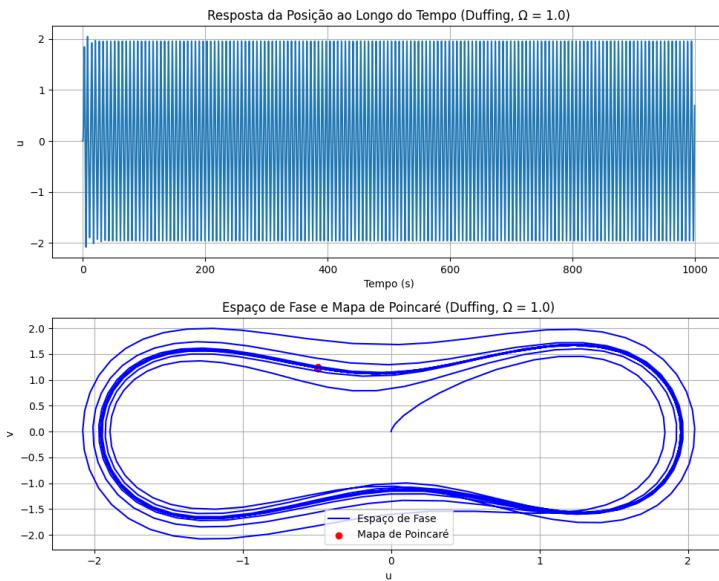


Figure 40: Mapa de Poincaré para Ω de 1.0

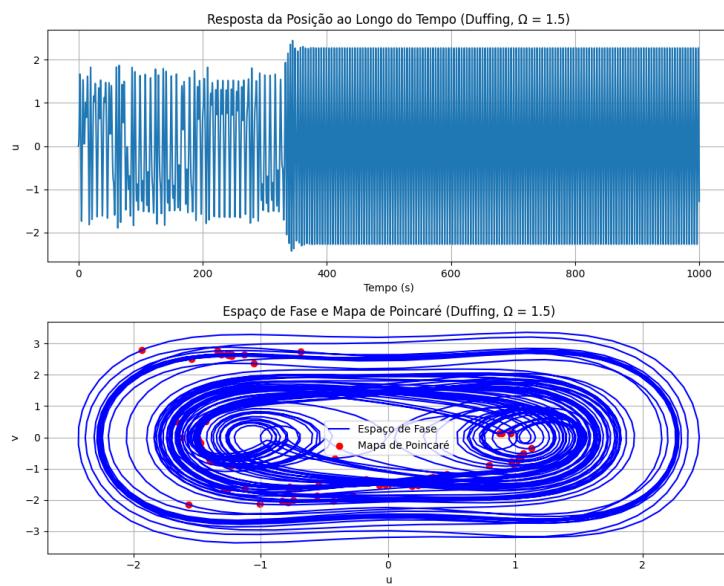


Figure 41: Mapa de Poincaré para Ω de 1.5

Pêndulo simples

$$\ddot{\phi} + \zeta \dot{\phi} + \omega_n^2 \sin(\phi) = \gamma \sin(\Omega t)$$

Utilizando o mesmo procedimento anterior, escrevendo as equações de estado e seguindo os parâmetros abaixo: Variando os valores de γ , e frequência de forçamento Ω , com as seguintes condições:

- $u_0 = 0.0$
- $v_0 = 0.0$
- $\omega_n = 1.0$
- $\Omega = [0.5, 1.0, 1.5]$
- $\zeta = 0.5$
- $\gamma = [0.5, 3.0]$

Os resultados para $\gamma = 0.5$ variando o valor de Ω foram:

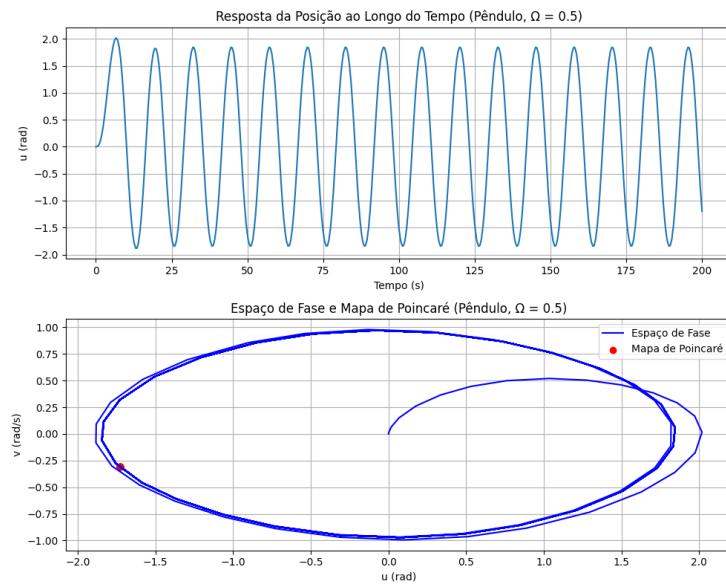


Figure 42: Mapa de Poincaré para Ω de 0.5

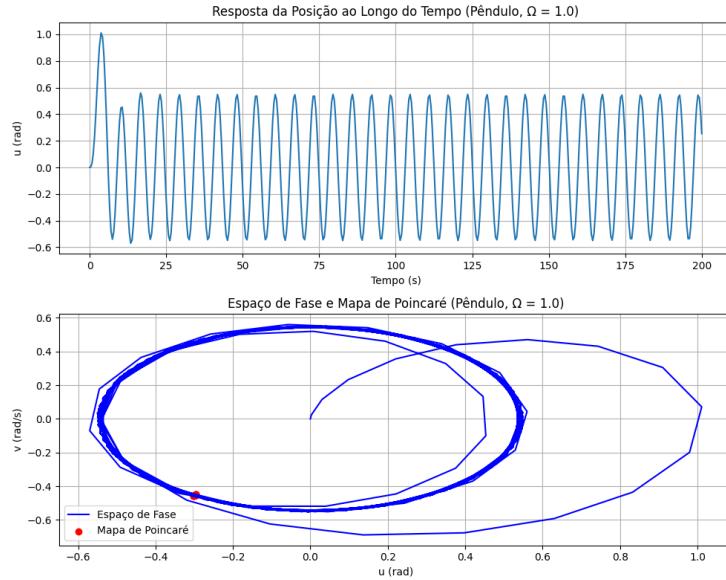


Figure 43: Mapa de Poincaré para Ω de 1.0

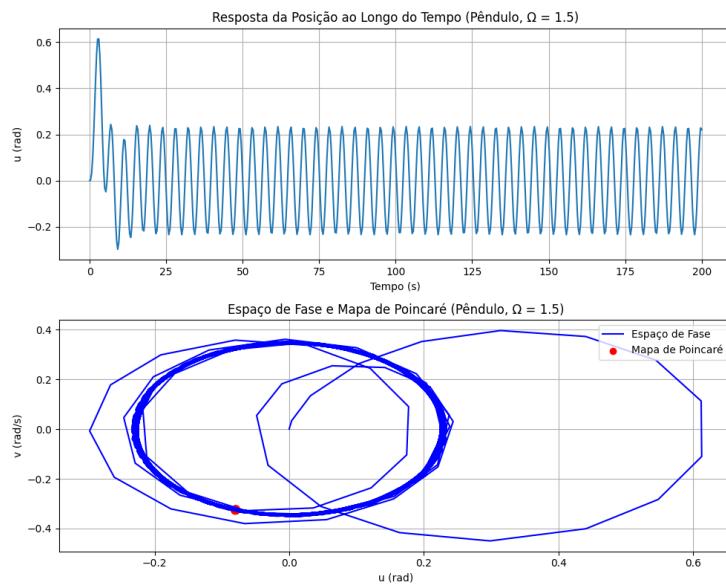


Figure 44: Mapa de Poincaré para Ω de 1.5

É interessante notar que no caso do pêndulo, o método funciona de forma razoável, exceto quando há um aumento tanto do forçamento, quanto da frequência

de forçamento. No caso $\Omega = 0.5$ o método funciona, no entanto $\Omega = 1.0$ e $\Omega = 1.5$, a dinâmica tem um transiente maior e o método não acompanha, bem como a mudança do passo permite os pontos se acumularem próximos. Os resultados para $\gamma = 3.0$ variando o valor de Ω foram:

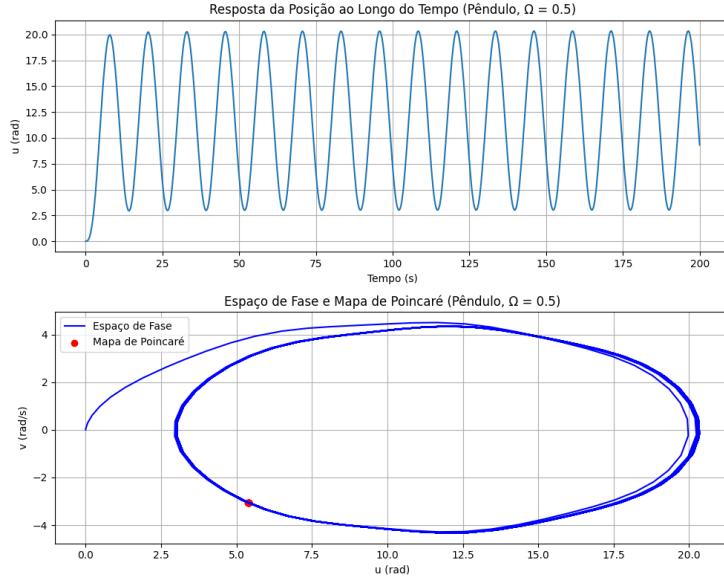


Figure 45: Mapa de Poincaré para Ω de 0.5

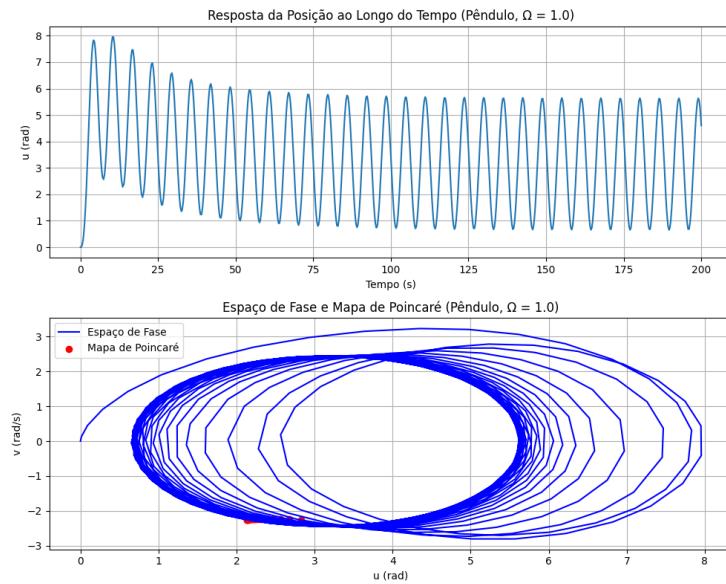


Figure 46: Mapa de Poincaré para Ω de 1.0

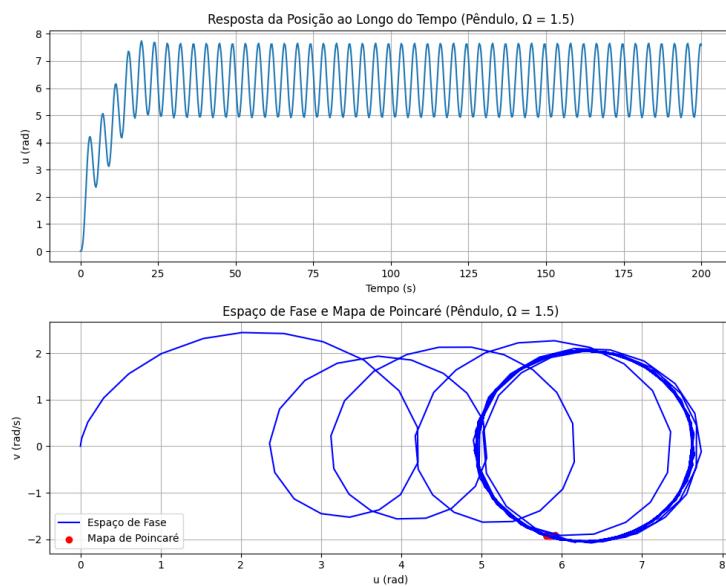


Figure 47: Mapa de Poincaré para Ω de 1.5

Questão 3

Determine os pontos de equilíbrio e avalie a natureza da estabilidade desses pontos para os sistemas listados abaixo. Realize a análise de forma analítica, sempre que possível, e, quando não for possível, implemente a solução numericamente. Se houver, discuta os casos em que a natureza dos pontos de equilíbrio não pode ser determinada pela análise de autovalores e autovetores.

(a)

Oscilador tipo Duffing:

$$\ddot{x} + 2\zeta\dot{x} + \alpha x + \beta x^3 = 0$$

(b)

Pêndulo simples:

$$\ddot{\phi} + \omega_n^2 \sin(\phi) = 0$$

(c)

Sistema de Lorenz:

$$\begin{cases} \dot{x} = \sigma(y - x) \\ \dot{y} = x(\rho - z) - y \\ \dot{z} = xy - \beta z \end{cases}$$

(d)

Sistema multiestável com 2 graus de liberdade:

$$\begin{cases} \ddot{x}_1 + 2\zeta_1\dot{x}_1 - 2\zeta_2(\dot{x}_2 - \dot{x}_1) + (1 + \alpha_1)x_1 + \beta_1 x_1^3 - \rho\Omega_s^2(x_2 - x_1) = 0 \\ \rho\ddot{x}_2 + 2\zeta_2(\dot{x}_2 - \dot{x}_1) + \alpha_2 x_2 + \beta_2 x_2^3 + \rho\Omega_s^2(x_2 - x_1) = 0 \end{cases}$$

Generalize a implementação numérica para este tipo de avaliação, de modo que seja flexível e aplicável a qualquer sistema dinâmico.

Solução:

(a) Oscilador tipo Duffing

Equação:

$$\ddot{x} + 2\zeta\dot{x} + \alpha x + \beta x^3 = 0$$

Considere o sistema descrito no seguinte formato (redução para um sistema de primeira ordem) Definindo as variáveis de estado $x_1 = x$ e $x_2 = \dot{x}$, obtemos:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -2\zeta x_2 - \alpha x_1 - \beta x_1^3 \end{cases}$$

Obtendo os pontos de equilíbrio: Resolvendo $\dot{x}_1 = \dot{x}_2 = 0$:

$$x_2 = 0 \quad \text{e} \quad \alpha x_1 + \beta x_1^3 = 0 \implies x_1(\alpha + \beta x_1^2) = 0$$

Solucionando o sistema, obtemos 3 pontos de equilíbrio:

- $(x_1, x_2) = (0, 0)$
- $(x_1, x_2) = (0, \sqrt{-\alpha/\beta})$, (se $\alpha/\beta < 0$)
- $(x_1, x_2) = (0, -\sqrt{-\alpha/\beta})$, (se $\alpha/\beta < 0$)

Análise de estabilidade: Matriz Jacobiana: $A = Df(x_1, x_2)$, $D = \begin{bmatrix} \frac{\partial}{\partial x_1} & \frac{\partial}{\partial x_2} \\ \frac{\partial}{\partial x_2} & \frac{\partial}{\partial x_1} \end{bmatrix}$

$$J = \begin{bmatrix} 0 & 1 \\ -\alpha - 3\beta x_1^2 & -2\zeta \end{bmatrix}$$

Em seguida, é preciso encontrar os autovalores da matriz Jacobiana do sistema, de forma que $\det(A - \lambda I) = 0$. Sendo assim:

$$\begin{vmatrix} -\lambda & 1 \\ -\alpha - 3\beta x_1^2 & -2\zeta - \lambda \end{vmatrix} = 0$$

Os autovalores desta matriz são $\lambda_{1,2} = \frac{-2\zeta \pm \sqrt{4\zeta^2 - 4\alpha - 12\beta x_1^2}}{2}$, sendo necessário agora avaliar os pontos fixos encontrados anteriormente, agora nesta linearização. Como a linearização não depende de x_2 e de qualquer forma seu valor é sempre zero, a avaliação depende dos valores de pontos fixos de x_1 :

- Para $x_1 = 0$:

$$\lambda_{1,2} = -\zeta \pm \sqrt{\zeta^2 - \alpha}$$

- Se $\alpha < 0$ e $\zeta > \sqrt{\zeta^2 - \alpha}$: $\operatorname{Re}(\lambda_1 < 0)$ e $\operatorname{Re}(\lambda_2 < 0)$, assim como $\operatorname{Im}(\lambda_1 = 0)$ e $\operatorname{Im}(\lambda_2 = 0)$, logo se trata de um ponto do tipo **sorvedouro estável**.
- Se $\alpha < 0$ e $\zeta < \sqrt{\zeta^2 - \alpha}$: $\operatorname{Re}(\lambda_1 > 0)$ e $\operatorname{Re}(\lambda_2 < 0)$, assim como $\operatorname{Im}(\lambda_1 = 0)$ e $\operatorname{Im}(\lambda_2 = 0)$, logo se trata de um **ponto fixo instável**.

- Se $\alpha > 0$ e $\zeta < \sqrt{\alpha}$: $Re(\lambda_1 < 0)$ e $Re(\lambda_2 < 0)$, assim como $Im(\lambda_1 > 0)$ e $Im(\lambda_2 < 0)$, logo se trata de um ponto do **espiral (foco) estável**.
- Se $\alpha > 0$ e $\zeta > \sqrt{\alpha}$: $Re(\lambda_1 > 0)$ e $Re(\lambda_2 < 0)$, assim como $Im(\lambda_1 = 0)$ e $Im(\lambda_2 = 0)$, logo se trata de um **ponto fixo instável**.
- Se $\alpha < 0$ e $\zeta = 0$: $Re(\lambda_1 > 0)$ e $Re(\lambda_2 < 0)$, assim como $Im(\lambda_1 = 0)$ e $Im(\lambda_2 = 0)$, logo se trata uma **sela instável**.
- Se $\alpha > 0$ e $\zeta = 0$: $Re(\lambda_1 = 0)$ e $Re(\lambda_2 = 0)$, assim como $Im(\lambda_1 > 0)$ e $Im(\lambda_2 < 0)$, logo se trata de um ponto de centro (indefinido) não sendo possível avaliar por meio da linearização.

- Para $x_1 = \pm\sqrt{-\alpha/\beta}$:

$$\lambda_{1,2} = -\zeta \pm \sqrt{\zeta^2 + 2\alpha}$$

- Se $\alpha > 0$ e $\zeta \geq 0$ ou $\zeta < 0$: $Re(\lambda_1 > 0)$ e $Re(\lambda_2 < 0)$, assim como $Im(\lambda_1 = 0)$ e $Im(\lambda_2 = 0)$, logo a variação do ponto fixo entre **estável, instável e sela instável** dependendo de como ζ se relaciona com α . Se $\zeta < \mp\sqrt{\zeta^2 + 2\alpha}$, $Re(\lambda_1 > 0)$ e $Re(\lambda_2 > 0)$, então o ponto fixo é uma **fonte instável**. Se $\zeta > \mp\sqrt{\zeta^2 + 2\alpha}$, ou $Re(\lambda_1 < 0)$ e $Re(\lambda_2 < 0)$, então o ponto fixo é um **sorvedouro estável**. Se $\zeta^2 + 2\alpha = 0$ e $zeta > 0$, então $Re(\lambda_1 < 0)$ e $Re(\lambda_2 < 0)$, então o ponto fixo é um **sorvedouro instável**. Se $\zeta^2 + 2\alpha = 0$ e $zeta < 0$, então $Re(\lambda_1 < 0)$ e $Re(\lambda_2 < 0)$, então o ponto fixo é uma **fonte instável**. Por fim, se $zeta = 0$, , então $Re(\lambda_1 > 0)$ e $Re(\lambda_2 < 0)$, então o ponto fixo é uma **sela instável**.

(b) Pêndulo simples sem dissipação

Equação:

$$\ddot{\phi} + \omega_n^2 \sin(\phi) = 0$$

Considere o sistema descrito no seguinte formato (redução para um sistema de primeira ordem) Definindo as variáveis de estado $\phi_1 = \phi$ e $\dot{\phi}_2 = \dot{\phi}$, obtemos:

$$\begin{cases} \dot{\phi}_1 = \phi_2 \\ \dot{\phi}_2 = -\omega_n^2 \sin(\phi_1) \end{cases}$$

Obtendo os pontos de equilíbrio: Resolvendo $\dot{\phi}_1 = \dot{\phi}_2 = 0$:

$$\phi_2 = 0 \quad \text{e} \quad \sin(\phi_1) = 0 \implies \phi_1 = n\pi, \quad n \in \mathbb{Z}$$

Solucionando o sistema, obtemos infinitos pontos de equilíbrio, mas considerando o sistema com $\phi_2 \in [0, 2\pi]$ uma volta completa, os pontos fixos pontem ser:

- $(\phi_1, \phi_2) = (n\pi, 0)$ para n par
- $(\phi_1, \phi_2) = (n\pi, 0)$ para n ímpar

De forma que outros valores de n são repetições pares ou ímpares desses dois.

Análise de estabilidade: Matriz Jacobiana $A = Df(x_1, x_2)$, $D = \begin{bmatrix} \frac{\partial}{\partial \phi_1} & \frac{\partial}{\partial \phi_2} \\ \frac{\partial}{\partial \phi_1} & \frac{\partial}{\partial \phi_2} \end{bmatrix}$

$$J = \begin{bmatrix} 0 & 1 \\ -\omega_n^2 \cos(\phi_1) & 0 \end{bmatrix}$$

Em seguida, é preciso encontrar os autovalores da matriz Jacobiana do sistema, de forma que $\det(A - \lambda I) = 0$. Sendo assim:

$$\begin{vmatrix} -\lambda & 1 \\ -\omega_n^2 \cos(\phi_1) & -\lambda \end{vmatrix} = 0$$

Os autovalores desta matriz são $\lambda_{1,2} = \pm \sqrt{-\omega_n^2 \cos(\phi_1)}$, sendo necessário agora avaliar os pontos fixos encontrados anteriormente, agora nesta linearização. Como a linearização não depende de ϕ_2 , a análise depende de ϕ_1 , onde n ser par ou ímpar varia o valor do cosseno, mas não depende do valor do parâmetro ω_n , exceto se $\omega_n = 0$:

- Para $\phi_1 = 0$ (ou n par e $\omega_n \neq 0$):

$$\lambda = \pm i\omega_n \text{ é um } \mathbf{centro \ estável}$$

- Para $\phi_1 = \pi$ (ou n ímpar e $\omega_n \neq 0$):

$$\lambda = \pm \omega_n \text{ é uma } \mathbf{sela \ instável}$$

(c) Sistema de Lorenz

Equações:

$$\begin{cases} \dot{x} = \sigma(y - x) \\ \dot{y} = x(\rho - z) - y \\ \dot{z} = xy - \beta z \end{cases}$$

Obtendo os pontos de equilíbrio: Resolvendo $\dot{x} = \dot{y} = \dot{z} = 0$:

$$y = x, \quad x(\rho - z - 1) = 0, \quad x^2 - \beta z = 0$$

Soluções:

- $(\dot{x}, \dot{y}, \dot{z}) = (0, 0, 0)$
- $(\dot{x}, \dot{y}, \dot{z}) = (\pm \sqrt{\beta(\rho - 1)}, \pm \sqrt{\beta(\rho - 1)}, \rho - 1)$

Análise de estabilidade: Matriz Jacobiana: $A = Df(x_1, x_2)$, $D =$

$$\begin{bmatrix} \frac{\partial}{\partial x} & \frac{\partial}{\partial y} & \frac{\partial}{\partial z} \\ \frac{\partial}{\partial x} & \frac{\partial}{\partial y} & \frac{\partial}{\partial z} \\ \frac{\partial}{\partial x} & \frac{\partial}{\partial y} & \frac{\partial}{\partial z} \end{bmatrix}$$

$$J = \begin{bmatrix} -\sigma & \sigma & 0 \\ \rho - z & -1 & -x \\ y & x & -\beta \end{bmatrix}$$

- Para $(0, 0, 0)$:

$$J = \begin{bmatrix} -\sigma & \sigma & 0 \\ \rho & -1 & 0 \\ 0 & 0 & -\beta \end{bmatrix}, \quad \lambda = -\beta, \quad \text{e soluções de } \lambda^2 + (\sigma+1)\lambda + \sigma(1-\rho) = 0$$

- Se $\rho < 1$: Estável
- Se $\rho > 1$: Instável

- Para pontos não triviais ($\rho > 1$): A análise requer cálculo numérico dos autovalores.

Para tal, foi criado um código utilizando o método de autovalores do Jacobiano para classificar os pontos fixos.

O código foi validado com os modelos dos itens a) e b), além disso para que seja o mais geral possível na classificação de sistemas, o Jacobiano do sistema funciona com Diferenças Finitas Progressivas, onde ele avalia a uma função $\dot{x} = f$ no ponto atual e cria uma perturbação em uma dimensão j , calcula a função nessa perturbação e então aproxima a coluna j -ésima por diferenças finitas. Desta forma, não se faz necessário calcular o jacobiano do sistema para cada sistema que se insere no código, tornando o código mais genérico. Em contrapartida, há uma sequência de erros numéricos, como serão vistos após.

Para a solução dos pontos fixos, foi utilizado um método de Newton-Raphson em um loop que trabalha até a convergência ou um número máximo de iterações, utilizando chutes iniciais do usuário.

Além disso foi utilizado uma classificação de estabilidade baseada em autovalores, onde:

- **Ponto tipo sorvedouro estável (poço):** $\text{Re}(\lambda_1) < 0$, $\text{Re}(\lambda_2) < 0$, $\text{Im}(\lambda_1) = 0$ e $\text{Im}(\lambda_2) = 0$.
- **Ponto tipo fonte instável:** $\text{Re}(\lambda_1) > 0$, $\text{Re}(\lambda_2) > 0$, $\text{Im}(\lambda_1) = 0$ e $\text{Im}(\lambda_2) = 0$.
- **Ponto tipo Sela instável:** $\text{Re}(\lambda_1) > 0$, $\text{Re}(\lambda_2) < 0$, $\text{Im}(\lambda_1) = 0$ e $\text{Im}(\lambda_2) = 0$.

- **Ponto tipo Centro estável:** $\operatorname{Re}(\lambda_1) = 0$, $\operatorname{Re}(\lambda_2) = 0$, $\operatorname{Im}(\lambda_1) > 0$ e $\operatorname{Im}(\lambda_2) < 0$.
- **Ponto tipo espiral (foco) estável:** $\operatorname{Re}(\lambda_1) < 0$, $\operatorname{Re}(\lambda_2) < 0$, $\operatorname{Im}(\lambda_1) > 0$ e $\operatorname{Im}(\lambda_2) < 0$.
- **Ponto tipo espiral (foco) instável:** $\operatorname{Re}(\lambda_1) > 0$, $\operatorname{Re}(\lambda_2) > 0$, $\operatorname{Im}(\lambda_1) > 0$ e $\operatorname{Im}(\lambda_2) < 0$.

O código segue em anexo, e foram utilizados os seguintes parâmetros na análise:

- $\sigma = 10.0$
- $\rho = 28.0$
- $\beta = 2.67$

Os pontos fixos encontrados neste sistema foram:

- Ponto 1: $(x, y, z) = (0, 0, 0)$ de autovalores $(-22.827, 11.82, -2.66)$
- Ponto 2: $(x, y, z) = (8.485, 8.485, 27.00)$ de autovalores $(-13.85, 0.0939 + 10.194j, 0.0939 - 10.194j)$
- Ponto 3: $(x, y, z) = (-8.485, -8.485, 27.00)$ de autovalores $(-13.85, 0.0939 + 10.194j, 0.0939 - 10.194j)$

Como se tratad e um sistema de mais de 2 dimensões, se torna difícil classificar sua estabilidade pelo método sugerido.

Vale ressaltar que é possível considerar

- Estável: $\sigma_s = \lambda$ tal que $\operatorname{Re}(\lambda) < 0$
- Instável: $\sigma_u = \lambda$ tal que $\operatorname{Re}(\lambda) > 0$
- Centro: $\sigma_c = \lambda$ tal que $\operatorname{Re}(\lambda) = 0$

Sendo assim, é possível assumir

- Ponto 1: Instável (tem em sua parte real 2 autovalores positivos)
- Ponto 2: Instável (tem em sua parte real 1 autovalor positivo)
- Ponto 3: Instável (tem em sua parte real 1 autovalor positivo)

(d) Sistema multiestável com 2 graus de liberdade

Equações:

$$\begin{cases} \ddot{x}_1 + 2\zeta_1 \dot{x}_1 - 2\zeta_2 (\dot{x}_2 - \dot{x}_1) + (1 + \alpha_1)x_1 + \beta_1 x_1^3 - \rho\Omega_s^2(x_2 - x_1) = 0 \\ \rho\ddot{x}_2 + 2\zeta_2 (\dot{x}_2 - \dot{x}_1) + \alpha_2 x_2 + \beta_2 x_2^3 + \rho\Omega_s^2(x_2 - x_1) = 0 \end{cases}$$

Transformação para sistema de primeira ordem: Definindo $x_3 = \dot{x}_1$ e $x_4 = \dot{x}_2$, obtemos:

$$\begin{cases} \dot{x}_1 = x_3 \\ \dot{x}_2 = x_4 \\ \dot{x}_3 = -2\zeta_1 x_3 + 2\zeta_2 (x_4 - x_3) - (1 + \alpha_1)x_1 - \beta_1 x_1^3 + \rho\Omega_s^2(x_2 - x_1) \\ \dot{x}_4 = -\frac{2\zeta_2}{\rho}(x_4 - x_3) - \frac{\alpha_2}{\rho}x_2 - \frac{\beta_2}{\rho}x_2^3 - \Omega_s^2(x_2 - x_1) \end{cases}$$

Obtendo os pontos de equilíbrio: Resolvendo $\dot{x}_1 = \dot{x}_2 = \dot{x}_3 = \dot{x}_4 = 0$:

$$x_3 = x_4 = 0$$

e as equações restantes são não lineares em x_1 e x_2 . Solução numérica é necessária.

Para tal, foi criado um código utilizando o método de autovalores do Jacobiano para classificar os pontos fixos.

O código foi validado com os modelos dos itens a) e b), além disso para que seja o mais geral possível na classificação de sistemas, o Jacobiano do sistema funciona com Diferenças Finitas Progressivas, onde ele avalia a uma função $\dot{x} = f$ no ponto atual e cria uma perturbação em uma dimensão j , calcula a função nessa perturbação e então aproxima a coluna j -ésima por diferenças finitas. Desta forma, não se faz necessário calcular o jacobiano do sistema para cada sistema que se insere no código, tornando o código mais genérico. Em contrapartida, há uma sequência de erros numéricos, como serão vistos após.

Para a solução dos pontos fixos, foi utilizado um método de Newton-Raphson em um loop que trabalha até a convergência ou um número máximo de iterações, utilizando chutes iniciais do usuário.

Além disso foi utilizado uma classificação de estabilidade baseada em autovalores, onde:

- **Ponto tipo sorvedouro estável (poço):** $\text{Re}(\lambda_1) < 0$, $\text{Re}(\lambda_2) < 0$, $\text{Im}(\lambda_1) = 0$ e $\text{Im}(\lambda_2) = 0$.
- **Ponto tipo fonte instável:** $\text{Re}(\lambda_1) > 0$, $\text{Re}(\lambda_2) > 0$, $\text{Im}(\lambda_1) = 0$ e $\text{Im}(\lambda_2) = 0$.
- **Ponto tipo Sela instável:** $\text{Re}(\lambda_1) > 0$, $\text{Re}(\lambda_2) < 0$, $\text{Im}(\lambda_1) = 0$ e $\text{Im}(\lambda_2) = 0$.

- **Ponto tipo Centro estável:** $\operatorname{Re}(\lambda_1) = 0$, $\operatorname{Re}(\lambda_2) = 0$, $\operatorname{Im}(\lambda_1) > 0$ e $\operatorname{Im}(\lambda_2) < 0$.
- **Ponto tipo espiral (foco) estável:** $\operatorname{Re}(\lambda_1) < 0$, $\operatorname{Re}(\lambda_2) < 0$, $\operatorname{Im}(\lambda_1) > 0$ e $\operatorname{Im}(\lambda_2) < 0$.
- **Ponto tipo espiral (foco) instável:** $\operatorname{Re}(\lambda_1) > 0$, $\operatorname{Re}(\lambda_2) > 0$, $\operatorname{Im}(\lambda_1) > 0$ e $\operatorname{Im}(\lambda_2) < 0$.

O código segue em anexo, e foram utilizados os seguintes parâmetros na análise:

- $\zeta_1 = 0.025$
- $\zeta_2 = 0.025$
- $\alpha_1 = -2.0$
- $\alpha_2 = -1.0$
- $\beta_1 = 1.0$
- $\beta_2 = 1.0$
- $\rho = 0.5$
- $\Omega = 1.0$

Os pontos fixos encontrados neste sistema foram: beginitemize

Ponto 1: $(x_1, x_2, x_3, x_4) = (0, 0, 0, 0)$ Sela instável.

Ponto 2: $(x_1, x_2, x_3, x_4) = (1, 1, 0, 0)$ Espiral (foco) estável.

Ponto 3: $(x_1, x_2, x_3, x_4) = (-1, -1, 0, 0)$ Espiral (foco) instável.

Por um erro numérico, o sistema assumiu um 4º ponto de equilíbrio como sela instável em Ponto 4: $(x_1, x_2, x_3, x_4) = (-0.0003, 0.0003, 0, 0)$ Sela instável. Sendo possível distinguir utilizando a bacia de atração apresentada no item c) da questão 4.

Questão 4

Implemente a bacia de atração de pontos de equilíbrio (pontos fixos) estáveis para todos os sistemas descritos nas letras (a), (b) e (d) da questão anterior utilizando os seguintes espaços ou subespaços:

- (a) Espaço de estados $x \times \dot{x}$
- (b) Espaço de estados $\phi \times \dot{\phi}$
- (c) Subespaços de estados: $x_1 \times x_2$, $x_1 \times \dot{x}_1$, e $x_2 \times \dot{x}_2$

(Bônus) Faça uma implementação generalizada para permitir a análise de qualquer sistema dinâmico, além de identificar automaticamente os atratores (pontos fixos) dentro do domínio avaliado. Discuta as dificuldades encontradas durante a implementação e comente sobre outras possíveis abordagens para a identificação de pontos de equilíbrio que não foram adotadas na solução final.

Solução:

(a) Oscilador tipo Duffing

Equação:

$$\ddot{x} + 2\zeta\dot{x} + \alpha x + \beta x^3 = 0$$

Para o sistema multiestável tipo duffing, foram utilizados os seguintes parâmetros de simulação:

- $\alpha = 1.0$
- $\beta = 1.0$
- $\zeta = 0.1$
- Tempo de evolução de cada ponto da malha: 300 segundos

As simulações foram feitas com uma malha de 50 pontos e outra, para melhor resultado, de 250 pontos, ambas em cada direção. Como resultado, o aumento de pontos melhora a qualidade da figura, no entanto aumenta o tempo de simulação.

No caso a simulação de 50 pontos teve um custo computacional de 120 segundos e a simulação de 250 teve um custo de aproximadamente 1 hora, não sendo efetiva para resultados rápidos. Segue abaixo o resultado das simulações.

Vale ressaltar que o código em anexo está para 50 pontos, para que a validação seja viável. Vale notar que os pontos de equilíbrio condizem com os encontrados na questão 3, através do método numérico para a solução e classificação dos pontos de equilíbrio.

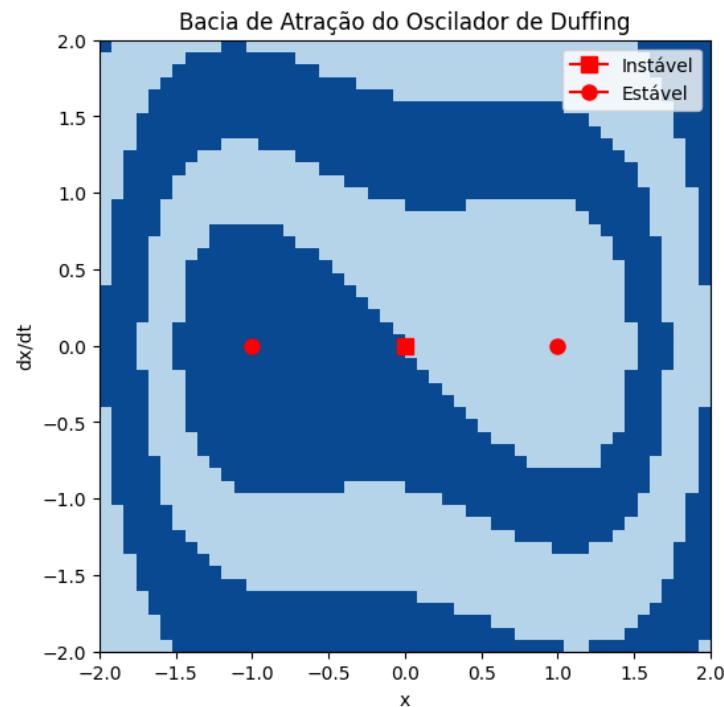


Figure 48: Bacia x por \dot{x} com malha de 50 por 50.

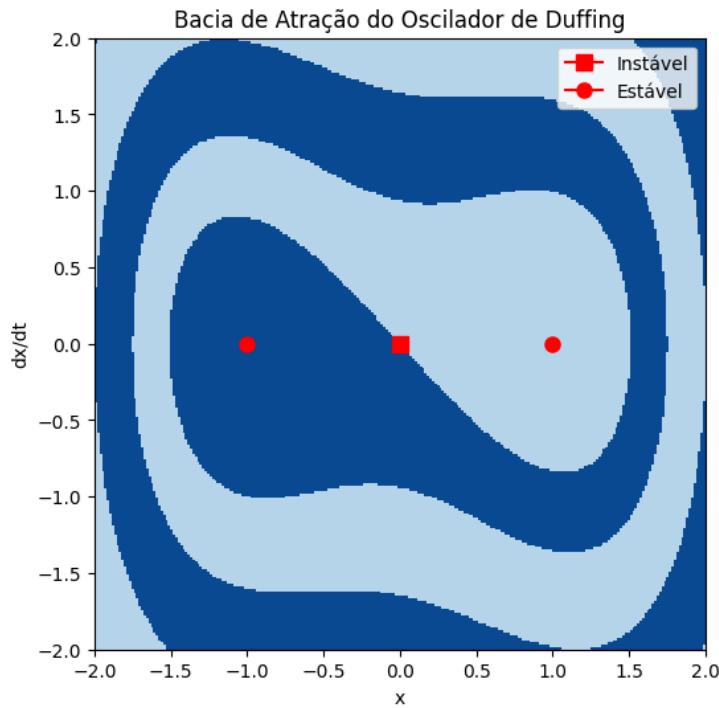


Figure 49: Bacia x por \dot{x} com malha de 250 por 250.

(b) Pêndulo simples com dissipação (modificado devido a presença de centro atrapalhar a solução da bacia de atração)

Equação:

$$\ddot{\phi} + \zeta \dot{\phi} + \omega_n^2 \sin(\theta) = 0]$$

Como a ausência de amortecimento no pêndulo simples sugerido na questão 4, item b) apresenta um ponto fixo de centro, se torna inviável a análise da bacia de atração, visto que por não ter amortecimento o pêndulo oscila em certa região constantemente.

Agora utilizando a equação acima, o sistema foi simulado com os seguintes parâmetros de simulação:

- $\omega_n = 1.0$
- $\zeta = 0.2$
- Tempo de evolução de cada ponto da malha: 300 segundos

As simulações foram feitas com uma malha de 50 pontos e outra, para melhor resultado, de 250 pontos, ambas em cada direção. Como resultado, o aumento de pontos melhora a qualidade da figura, no entanto aumenta o tempo de simulação.

No caso a simulação de 50 pontos teve um custo computacional de 120 segundos e a simulação de 250 teve um custo de aproximadamente 1 hora, não sendo efetiva para resultados rápidos. Segue abaixo o resultado das simulações.

Vale ressaltar que o código em anexo está para 50 pontos, para que a validação seja viável. Vale notar que os pontos de equilíbrio condizem com os encontrados na questão 3, através do método numérico para a solução e classificação dos pontos de equilíbrio.

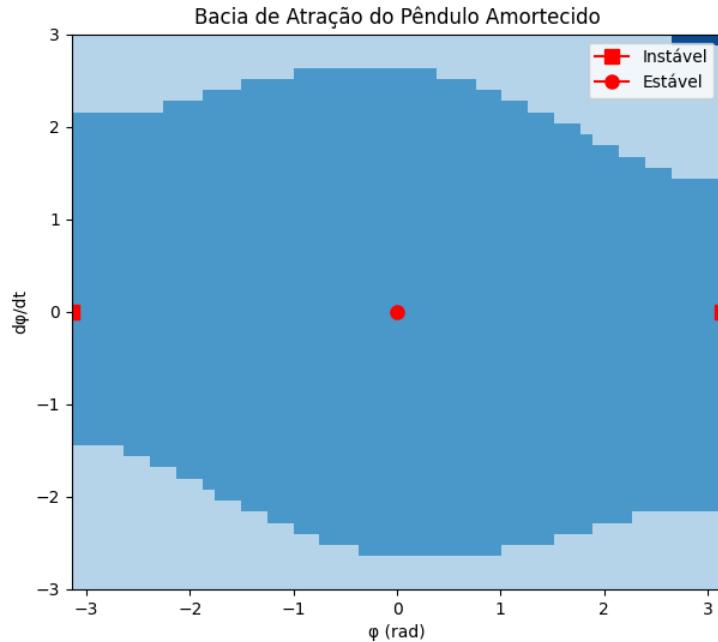


Figure 50: Bacia ϕ por $\dot{\phi}$ com malha de 50 por 50.

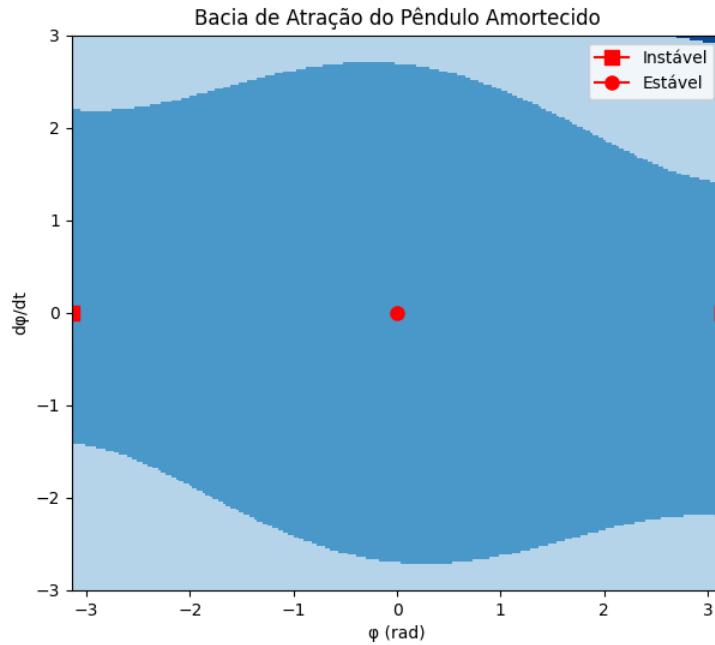


Figure 51: Bacia $\dot{\phi}$ por ϕ com malha de 250 por 250.

(c) Sistema multiestável com 2 graus de liberdade

Equações:

$$\begin{cases} \ddot{x}_1 + 2\zeta_1 \dot{x}_1 - 2\zeta_2(\dot{x}_2 - \dot{x}_1) + (1 + \alpha_1)x_1 + \beta_1 x_1^3 - \rho\Omega_s^2(x_2 - x_1) = 0 \\ \rho\ddot{x}_2 + 2\zeta_2(\dot{x}_2 - \dot{x}_1) + \alpha_2 x_2 + \beta_2 x_2^3 + \rho\Omega_s^2(x_2 - x_1) = 0 \end{cases}$$

Para o sistema multiestável, foram utilizados os seguintes parâmetros de simulação:

- $\zeta_1 = 0.025$
- $\zeta_2 = 0.025$
- $\alpha_1 = -2.0$
- $\alpha_2 = -1.0$
- $\beta_1 = 1.0$
- $\beta_2 = 1.0$
- $\rho = 0.5$

- $\Omega = 1.0$
- Tempo de evolução de cada ponto da malha: 300 segundos

As simulações foram feitas com uma malha de 50 pontos e outra, para melhor resultado, de 250 pontos, ambas em cada direção.

Como resultado, o aumento de pontos melhora a qualidade da figura, no entanto aumenta o tempo de simulação. No caso a simulação de 50 pontos teve um custo computacional de 9 minutos e a simulação de 250 teve um custo de aproximadamente 3 horas, não sendo efetiva para resultados rápidos. Segue abaixo o resultado das simulações. Vale ressaltar que o código em anexo está para 50 pontos, para que a validação seja viável.

Segue abaixo o resultado das simulações. Vale ressaltar que o código em anexo está para 50 pontos, para validação rápida. Vale notar que os pontos de equilíbrio condizem com os encontrados na questão 3, através do método numérico para a solução e classificação dos pontos de equilíbrio.

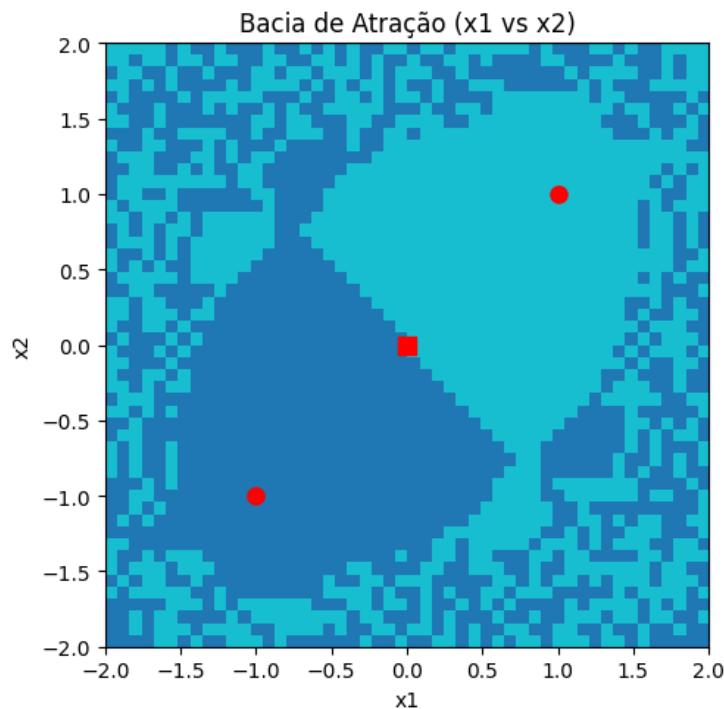


Figure 52: Bacia x_1 por x_2 com malha de 50 por 50.

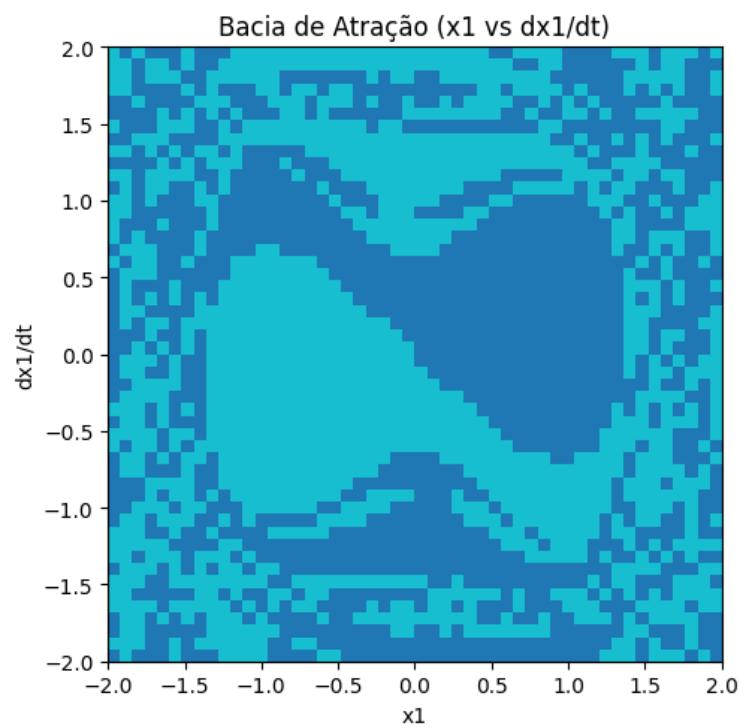


Figure 53: Bacia x_1 por \dot{x}_1 com malha de 50 por 50.

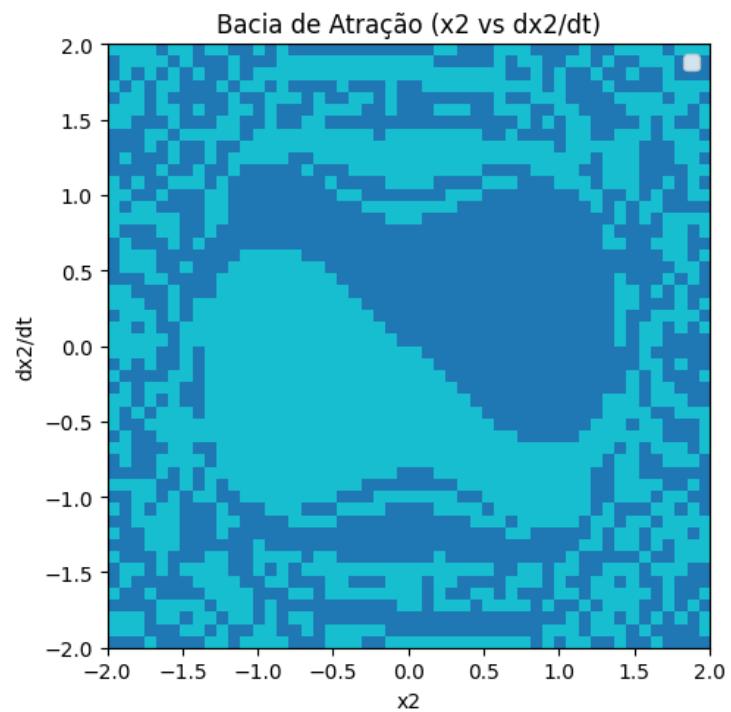


Figure 54: Bacia x_2 por \dot{x}_2 com malha de 50 por 50.

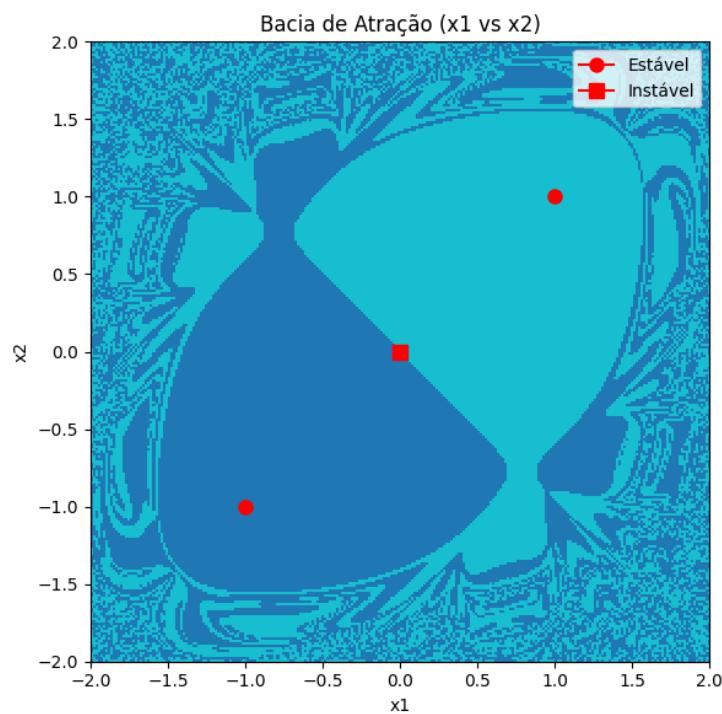


Figure 55: Bacia x_1 por x_2 com malha de 250 por 250.

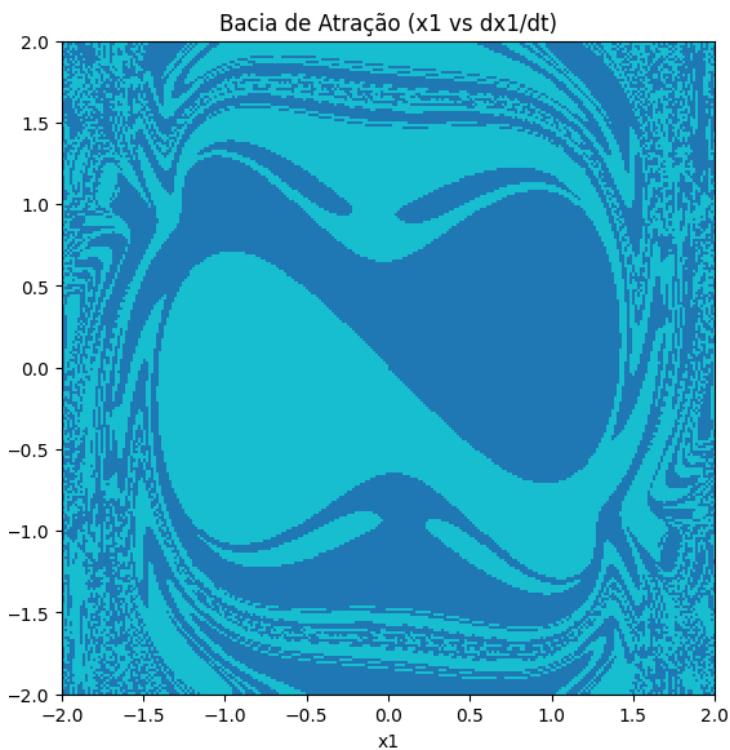


Figure 56: Bacia x_1 por \dot{x}_1 com malha de 250 por 250.

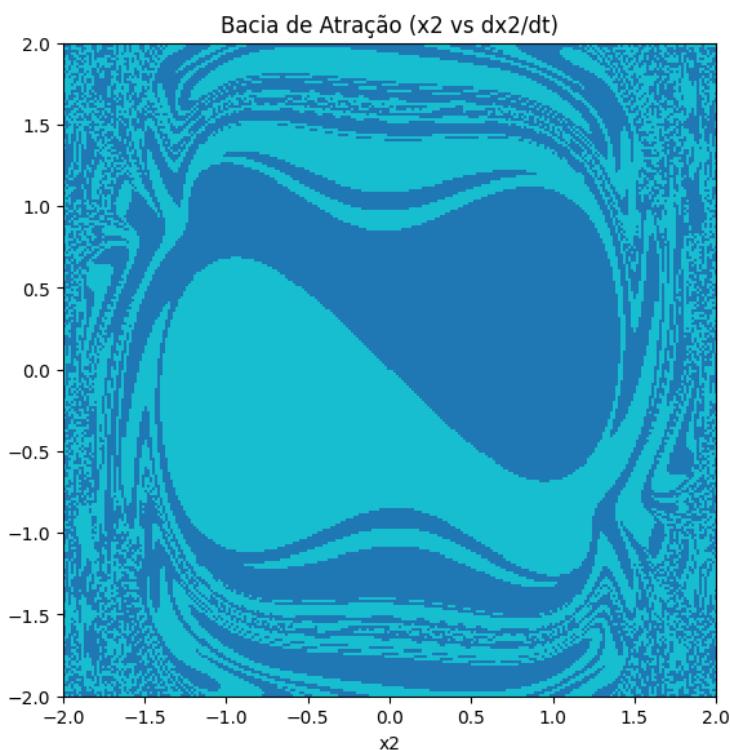


Figure 57: Bacia x_2 por \dot{x}_2 com malha de 250 por 250.

Questão 5 (Bônus)

Utilizando o conceito de mapas de Poincaré, crie um algoritmo que classifique automaticamente as respostas dinâmicas de um sistema dinâmico submetido a estímulos harmônicos. O algoritmo precisa ser robusto e aceitar qualquer sistema dinâmico como entrada.