

Escola Politécnica - USP

MAP3121 - Método Numéricos e Aplicações - 2021

Matheus Silva Coutinho - 10410156

Taiki Hashizume - 10791812

Exercício Programa 1

Autovalores e Autovetores de Matrizes Tridiagonais Simétricas

—

O Algoritmo QR

27 de Junho de 2021

1. Introdução	3
2. Algoritmo QR com e sem deslocamento espectrais	3
3. Sistema massa mola com 5 massas e 3 casos de posição inicial	5
4. Sistema massa mola com 10 massas e 3 casos de posição inicial	26
5. Conclusão	62

1. Introdução

O EP1 da disciplina foi escrito em linguagem Python 3. Ele consiste em receber uma matriz A quadrada n por n tridiagonal simétrica e uma matriz V identidade de mesma dimensão e aplicar o algoritmo QR primeiro sem deslocamentos espectrais e depois com eles para efeito de comparação. Ao final obtemos a matriz A diagonalizada em que sua diagonal principal possui todos os autovalores e a matriz V possui seus autovetores em cada coluna.

O EP1 propõe a utilização do algoritmo QR com deslocamentos espectrais para se chegar em um equacionamento de posição por tempo de um sistema massa mola com n massas e $n+1$ molas. Para isso a matriz A é composta por elementos relacionados às constantes elásticas das molas com sua lei de formação definida no enunciado. A seguir as tarefas feitas foram mais detalhadas e separadas em:

- Algoritmo QR com e sem deslocamentos (item a)
- Sistema massa mola com 5 massas e 3 casos de posição inicial (item b)
- Sistema massa mola com 10 massas e 3 casos de posição inicial (item c)

2. Algoritmo QR com e sem deslocamento espectrais

O algoritmo implementado analisa os betas de baixo para cima. Portanto se começa com β_{n-1} aplicando as rotações de givens até que ele tenha um valor menor que 10^{-6} . Quando isto ocorrer, o beta sendo analisado é aproximado para zero e parte-se para o β_{n-2} e assim por diante até se chegar ao β_1 . Quando isto ocorrer a matriz estará diagonalizada e os autovetores estarão na diagonal principal enquanto que a matriz V estará com seus autovetores em cada coluna.

Este algoritmo consiste em um corpo de código dentro de um loop “while” que checa duas condições: se o valor em módulo do beta analisado é menor do que 10^{-6} e se o beta sendo analisado é o último da matriz, no nosso caso β_1 . Existem três contadores envolvidos com esse loop: k , n e i . O contador k é persistente e conta os passos/iterações, isto é, quantas vezes o loop “while” foi realizado. O contador n também é persistente e serve para fornecer os índices dos betas sendo analisados através de “len(A)-n” já que o nosso algoritmo analisa os betas de baixo para cima. Por causa disso ele também é utilizado para fornecer os índices dos elementos da matriz A que entram no cálculo da Heurística de Wilkinson e deslocamento espectral. E por último o contador i é resetado para o valor de 1 a cada passo k . Ele é utilizado em outro loop “while” dentro do qual estamos analisando e fornece os índices dos elementos da matriz A que entram nos cálculos dos coeficientes c e s e que sofrem rotação de givens daquele operador Q_i .

Quanto a aplicação das rotações de givens $Q_i^{(k)}$ existe uma lista “store_Q” que armazena todas as matrizes Q de cada passo k . Ao final de todas as rotações daquele passo k resgatamos todas as matrizes Q_i e realizamos a transposição de cada uma. Multiplica-se a matriz A rotacionada por cada Q transposto e também a matriz V por cada Q transposto. O loop “while” dentro do loop que estamos analisando é responsável por aplicar as rotações em A , definir as matrizes de rotação de Givens Q e armazená-las em store_Q. Já o loop “for” é utilizado para multiplicar a matriz A e V por cada rotação Q daquele passo k .

Para o algoritmo com Heurística de Wilkinson e deslocamento espectral temos um condicional “if else” logo no começo do loop para definir μ_k . Antes das rotações fazemos a subtração da matriz A por $\mu_k \cdot I$ em que I é uma matriz identidade de mesma ordem de A. Logo depois das multiplicações pelas matrizes Q transpostas temos a adição de A pelo mesmo termo e isto compõe os deslocamentos espectrais. Para o caso em que não há Heurística de Wilkinson e deslocamento espectral apenas se desconsideram estes três passos apresentados neste parágrafo.

Por último, no fim do loop, existe um condicional “if” que checa se o beta analisado é menor do que 10^{-6} . Caso isto ocorra este β será igualado a zero e o contador n será incrementado por 1. Observe que ao sempre se incrementar o contador n quando esta condição é satisfeita, a condição do “while” que engloba o algoritmo QR acaba por ser satisfeita apenas quando o β_1 é menor que 10^{-6} , isto é, apenas quando o último β é zerado.

Partindo agora para os resultados da tarefa temos a seguir os resultados para matriz de ordem n=4:

Matriz A:

```
[[ 2. -1.  0.  0.]
 [-1.  2. -1.  0.]
 [ 0. -1.  2. -1.]
 [ 0.  0. -1.  2.]]
```

Matriz A Final:

```
[[ 3.61803399e+00 -2.91433544e-16 -1.34807895e-16 -1.97626213e-16]
 [ 0.00000000e+00  2.61803399e+00  1.55789421e-15  4.31261482e-17]
 [-3.80180036e-18 -4.44709129e-19  1.38196601e+00  4.31831727e-07]
 [ 1.75772900e-17 -8.14220402e-17  4.31831727e-07  3.81966011e-01]]
```

Matriz V Final:

```
[[ 0.37174803  0.60150096 -0.60150079 -0.37174829]
 [-0.60150096 -0.37174803 -0.37174777 -0.60150112]
 [ 0.60150096 -0.37174803  0.37174829 -0.60150079]
 [-0.37174803  0.60150096  0.60150112 -0.37174777]]
```

iterações k com deslocamento: 7

iterações k sem deslocamento: 45

Autovalores Analiticamente:

$\lambda_1=0.3819660112501051$ $\lambda_2=1.381966011250105$ $\lambda_3=2.618033988749895$ $\lambda_4=3.618033988749895$

Autovetores Analiticamente:

$v_1=[0.5877852522924731, 0.9510565162951535, 0.9510565162951536, 0.5877852522924732]$

$v_2=[0.9510565162951535, 0.5877852522924732, -0.587785252292473, -0.9510565162951536]$

$v_3=[0.9510565162951536, -0.587785252292473, -0.5877852522924734, 0.9510565162951535]$

$v_4=[0.5877852522924732, -0.9510565162951536, 0.9510565162951535, -0.5877852522924728]$

Podemos observar que os valores obtidos na diagonal da matriz A tem correspondência com os autovalores calculados analiticamente em até 10^{-6} . Além disso os autovetores também tem correspondência nessa mesma ordem, basta multiplicar cada um por um certo fator para observar que os valores calculados analiticamente são múltiplos dos obtidos numericamente. O passo de normalização não foi feito pois representa o mesmo vetor e o uso de bibliotecas de algebra computacional não é permitido neste EP. É possível observar também que o número de passos k é expressivamente menor nas iterações com deslocamento espectral, pois o β converge mais rápido para zero e consequentemente o α converge mais rápido para seu autovalor. Este fato fica cada vez mais evidente conforme a dimensão da matriz aumenta. Como os próximos casos possuem matrizes de ordem muito grande para visualização apresentaremos apenas as diferenças no número de iterações para com deslocamento e sem deslocamento para cada uma:

Matriz de ordem 8:

iterações k com deslocamento: 15

iterações k sem deslocamento: 143

Matriz de ordem 16:

iterações k com deslocamento: 31

iterações k sem deslocamento: 473

Matriz de ordem 32:

iterações k com deslocamento: 59

iterações k sem deslocamento: 1600

3. Sistema massa mola com 5 massas e 3 casos de posição inicial

Para esta tarefa desenvolvemos as equações de $x(t)$ para cada massa e obtemos os valores de posição de cada uma em um intervalo de tempo de 10s. Primeiro realizamos a operação da matriz V transposta multiplicada pela matriz de posições iniciais para obtermos a matriz $Y(0)$ que contém os índices que multiplicam os termos cossenos na equação de movimento:

$$Y(0) = V^T X(0)$$

Como partimos do instante zero e as velocidades iniciais são nulas temos que as equações de posição são da seguinte forma:

$$Y_1(t) = Y(0)_1 \cos(\omega_1 t)$$

$$Y_2(t) = Y(0)_2 \cos(\omega_2 t)$$

$$Y_3(t) = Y(0)_3 \cos(\omega_3 t)$$

$$Y_4(t) = Y(0)_4 \cos(\omega_4 t)$$

$$Y_5(t) = Y(0)_5 \cos(\omega_5 t)$$

Aplicamos todos os instantes de tempo para cada massa e obtemos uma matriz final com 5 massas e 400 valores de posição, isto é, uma matriz 5x400. O último passo agora é fazer a multiplicação de V por esta matriz 5x400 para obtermos a matriz X(t) com todos os valores de tempo:

$$X(t) = VY(t)$$

Como dito no enunciado do EP, os modos de frequência correspondem às posições iniciais que são múltiplos dos autovetores da matriz A e suas respectivas frequências são as raízes quadradas de cada autovalor correspondente (frequência angular em rad/s) dividido por 2π (frequência em Hz). Desta forma, para achar o modo de vibração com a maior frequência basta vasculhar nossa matriz A pelo maior autovalor, obter o autovetor correspondente e alimentar as posições iniciais com os elementos deste vetor.

Neste algoritmo em específico foi criado uma lista "store_X" que em cada posição armazena uma lista de valores de posição de cada massa que varia de $t=0s$ até $t=10s$. Assim a posição store_X[0], por exemplo, armazena todos os valores de posição da massa 1 de $t=0s$ até $t=10s$ com intervalos de 0.025s entre cada instante t. Isto é feito se criando uma lista T que armazena todos os instantes de tempo que serão utilizados e um loop "for" que aplica na equação de cada massa cada um dos valores. Para se chegar na equação de cada massa foram precisos os autovalores, para cálculo dos ω e os autovetores e posições iniciais para se chegar nos elementos de Y(0).

Para o cálculo das posições das molas em relação a sua posição de equilíbrio foi utilizado um loop "for" que itera sobre cada lista de posição das massas. Ele realiza a subtração da posição da massa atual pela posição da massa anterior para todos os instantes de tempo. Para o cálculo da primeira mola e da última mola a subtração é diferente. Para a primeira mola se subtrai a posição da massa 1 por zero para todos os instantes de tempo enquanto para a última mola se subtrai zero pela posição da última massa. As posições de cada mola para todos os instantes de tempo são guardadas em uma lista chamada "store_Spring". A seguir estão os resultados dos 3 casos:

- Caso 1 $X(0) = -2, -3, -1, -3, -1$:

K das molas:

[42, 44, 46, 48, 50, 52]

Matriz A:

```
[[ 43. -22.  0.  0.  0.]
 [-22.  45. -23.  0.  0.]
 [  0. -23.  47. -24.  0.]
 [  0.  0. -24.  49. -25.]
 [  0.  0.  0. -25.  51.]]
```

Matriz A Final:

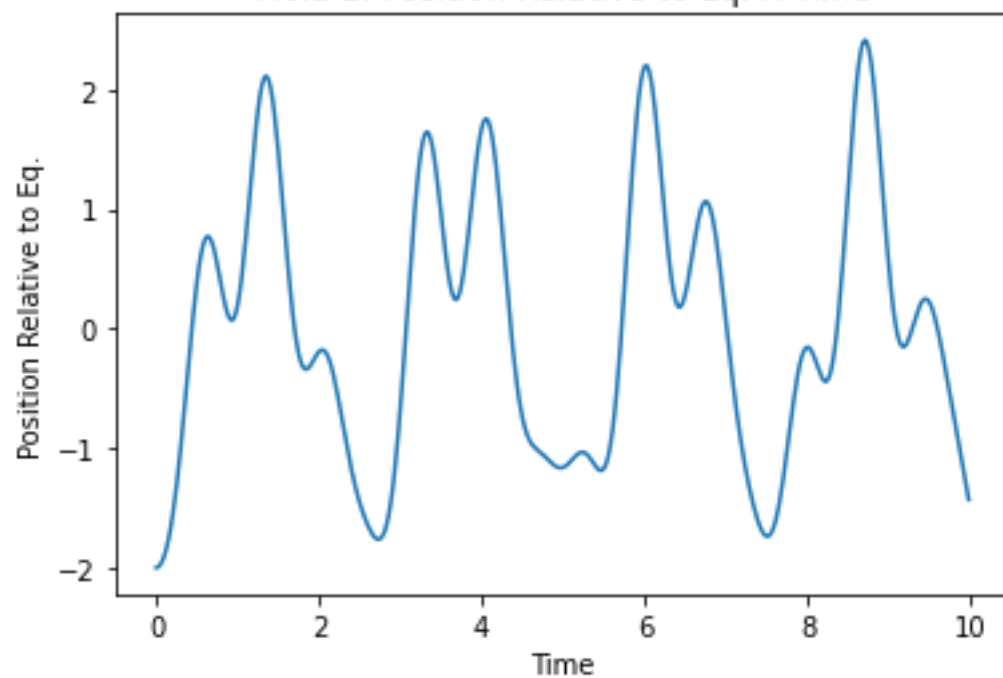
```
[[ 8.84450056e+01  5.78838094e-15 -6.80980129e-14  3.34743725e-14
  6.16571196e-15]
 [ 0.00000000e+00  7.01138311e+01 -4.60829189e-13  1.16752674e-13
  9.40125676e-15]
 [ 2.22706817e-16 -3.33487620e-17  4.67731863e+01 -1.34322149e-06
 -2.43423935e-14]
 [ 4.04275829e-16 -1.24525312e-15 -1.34322055e-06  2.33986332e+01
 -2.10284583e-07]
 [ 1.82314418e-16 -3.24321795e-16 -3.30493520e-16 -2.10284784e-07
  6.26934375e+00]]
```

Matriz V Final:

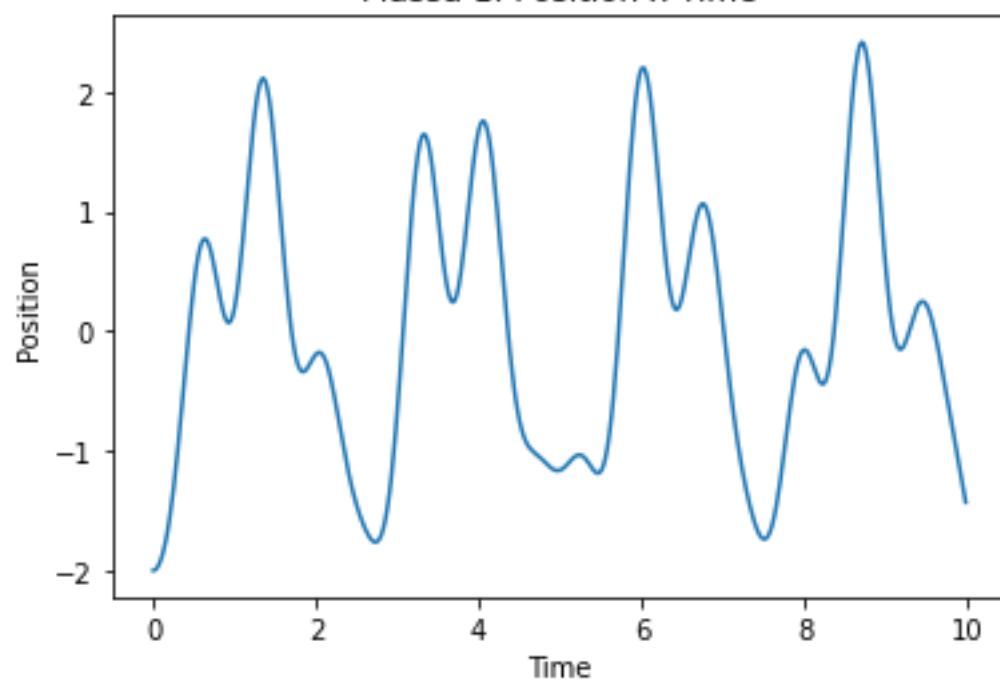
```
[[ 0.18933466 -0.4749761 -0.59882243 -0.53250392 -0.31048568]
 [-0.39110521  0.58538281  0.10270309 -0.47444571 -0.51837923]
 [ 0.55766111 -0.18485786  0.56486874  0.06375698 -0.57593403]
 [-0.588202 -0.38295964 -0.09308546  0.51737516 -0.4806437 ]
 [ 0.39271058  0.50089335 -0.55056532  0.46861374 -0.26863216]]
```

iterações k com deslocamento: 9

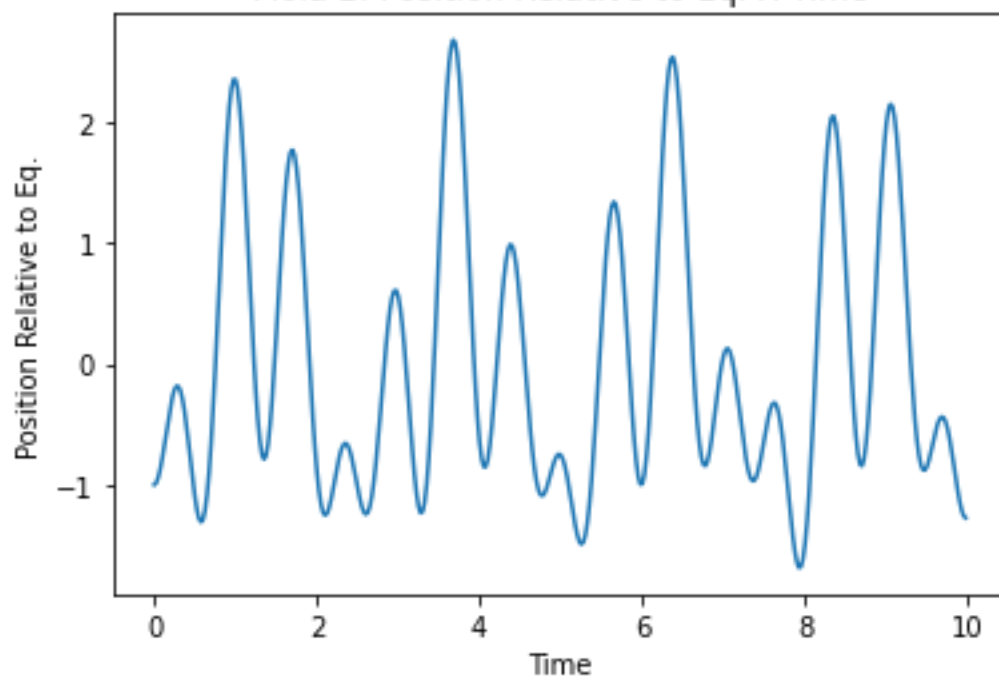
Mola 1: Position Relative to Eq. X Time



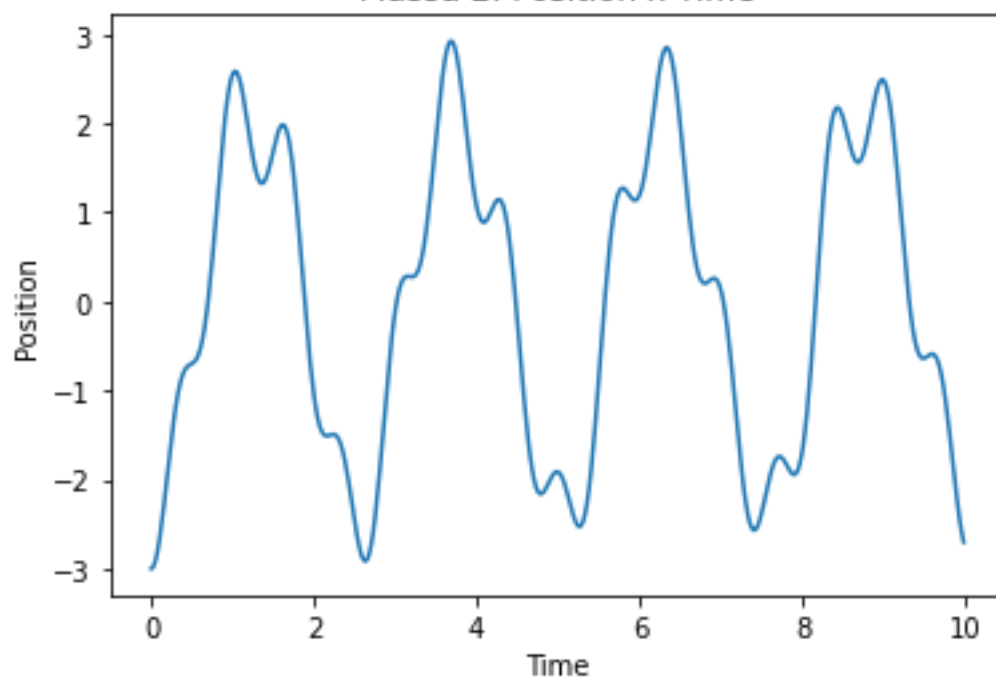
Massa 1: Position x Time

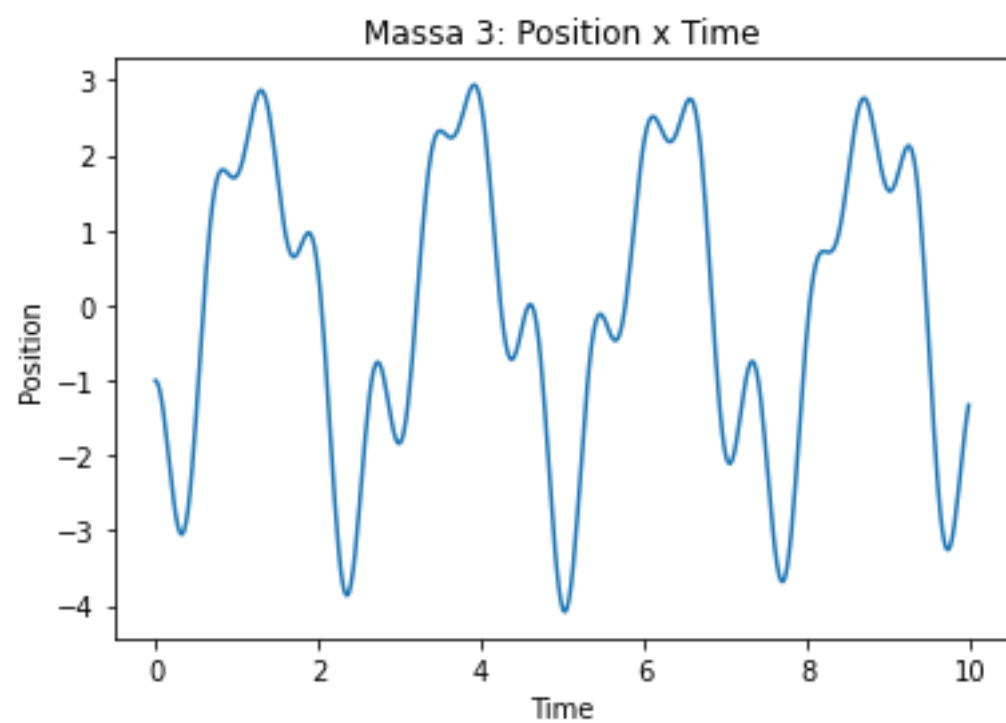
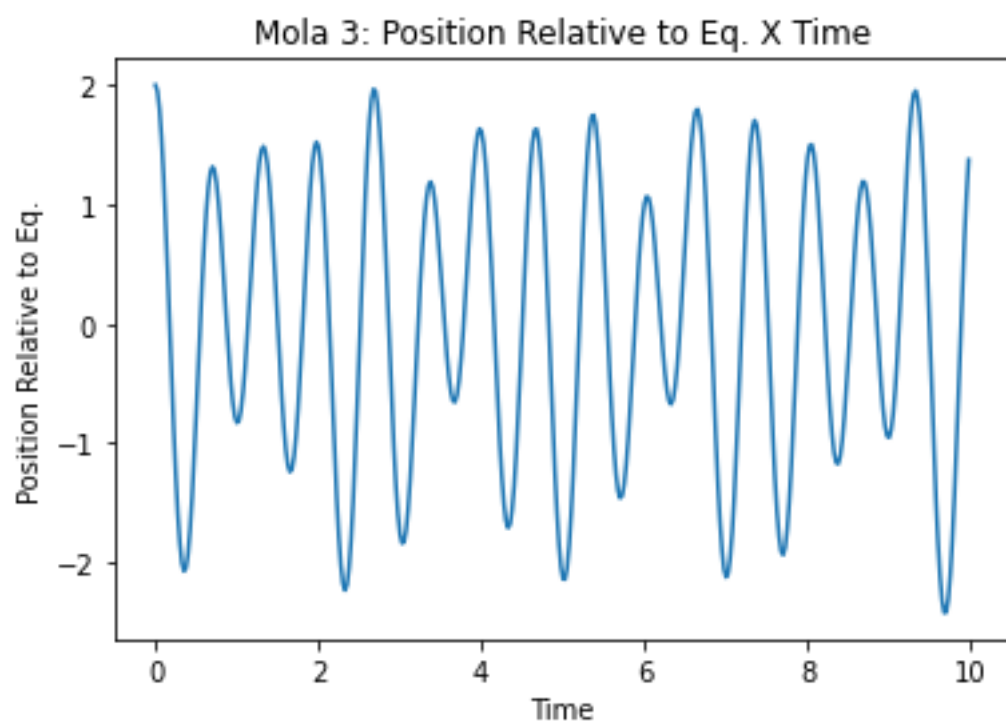


Mola 2: Position Relative to Eq. X Time

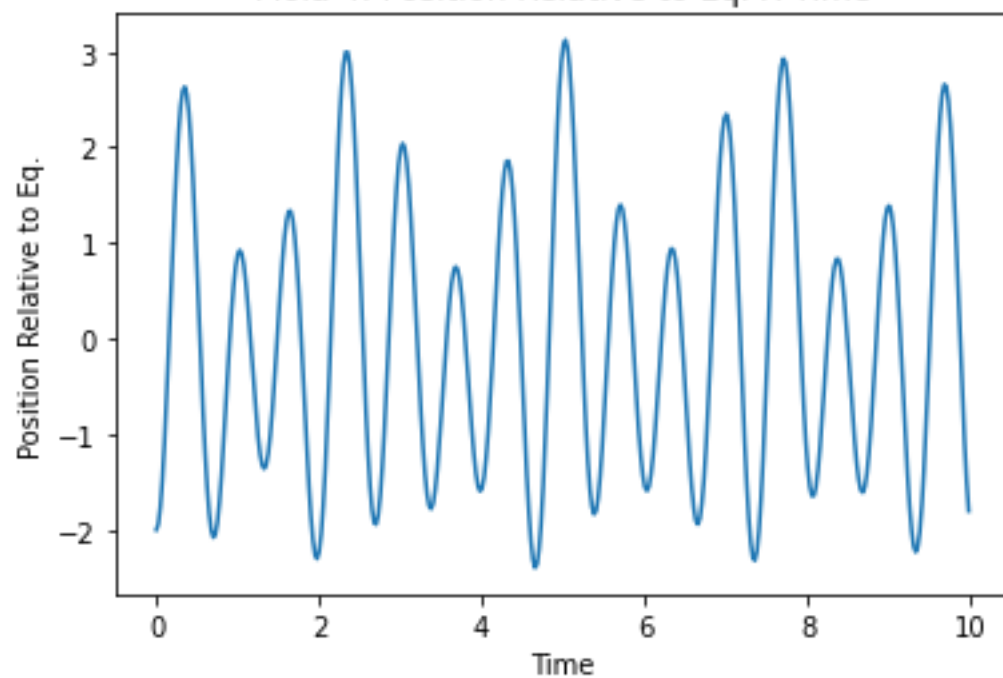


Massa 2: Position x Time

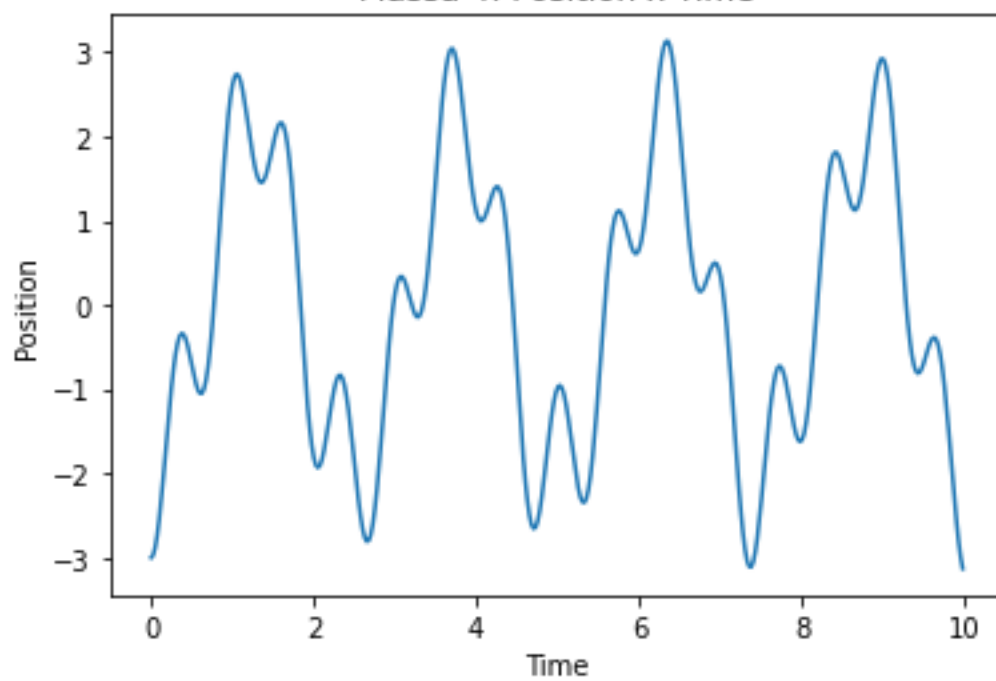




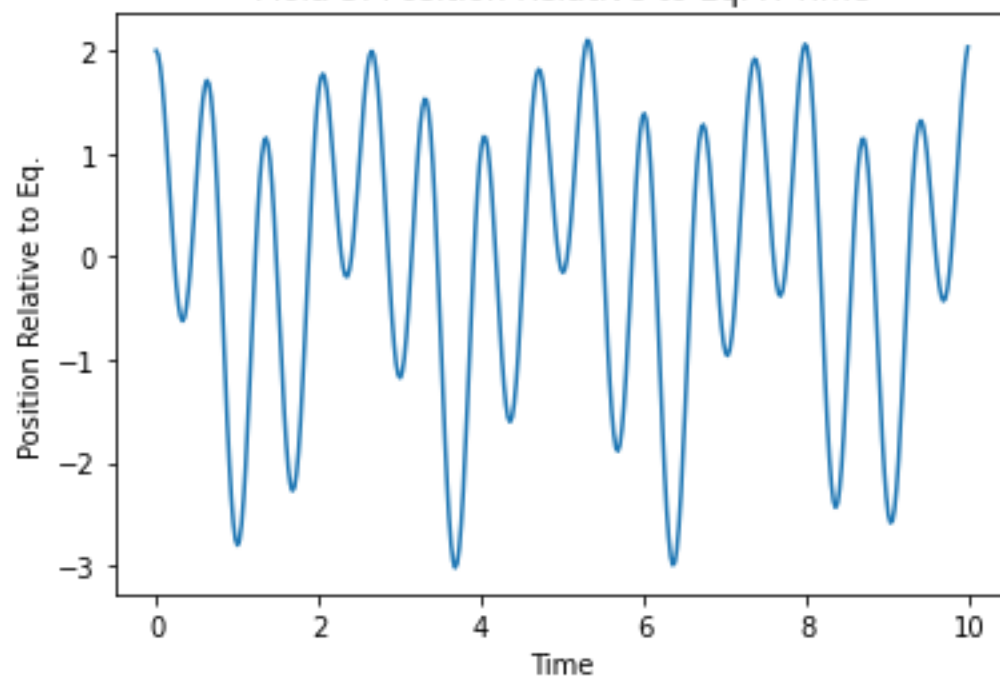
Mola 4: Position Relative to Eq. X Time



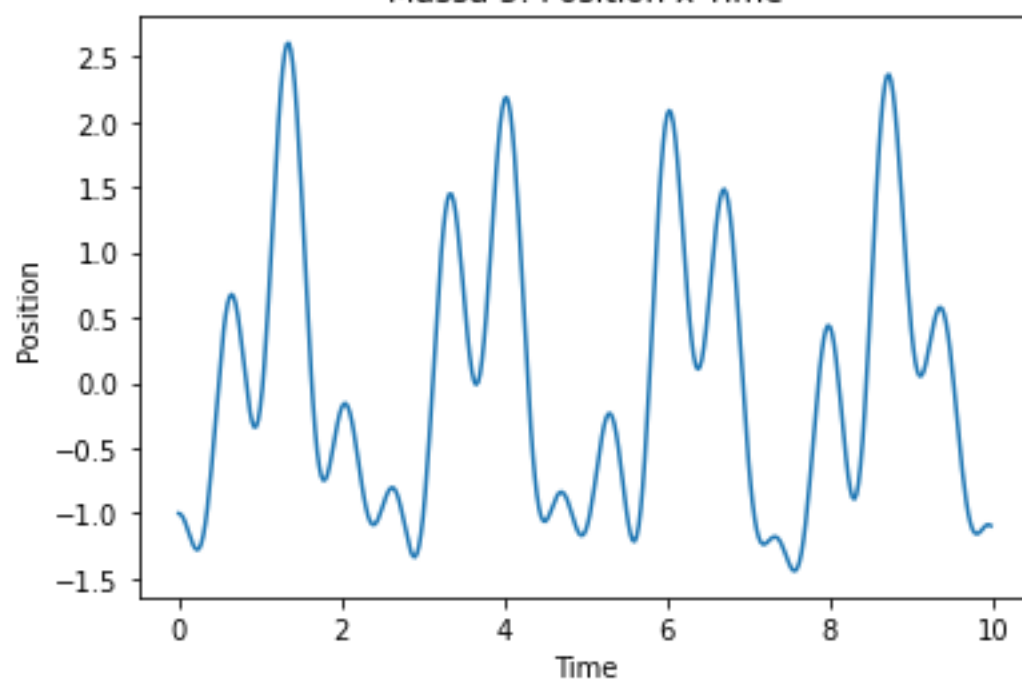
Massa 4: Position x Time

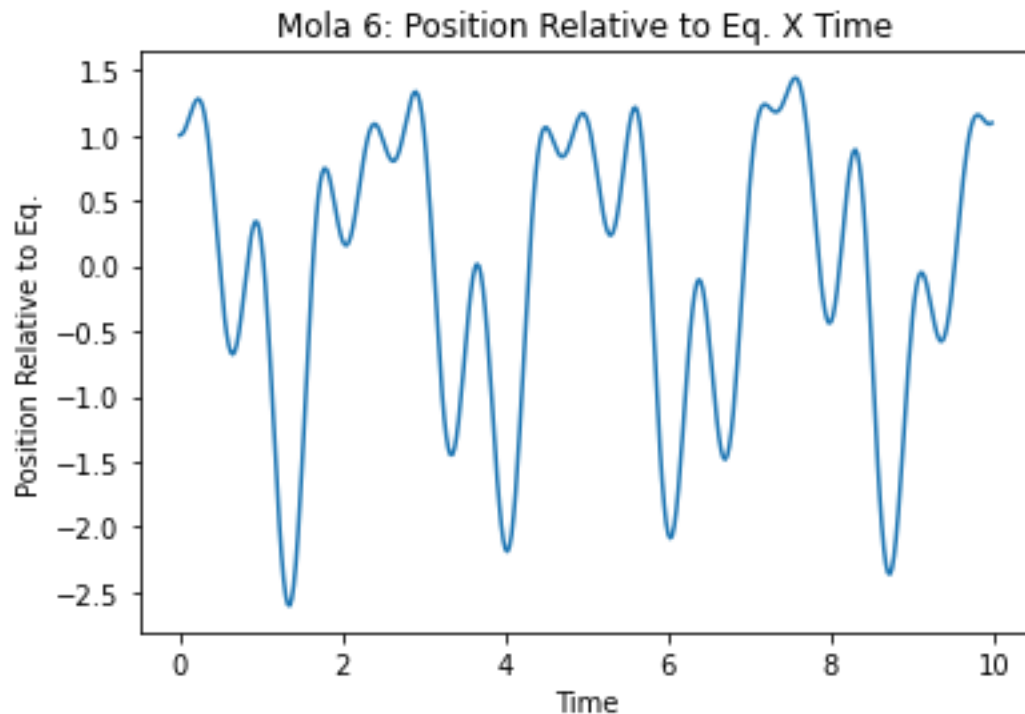


Mola 5: Position Relative to Eq. X Time



Massa 5: Position x Time





Neste caso as posições iniciais se diferem muito de qualquer autovetor. Sabemos que quando as posições iniciais são diferentes de um múltiplo de algum autovetor o movimento das massas é uma combinação linear de diferentes modos. Porém não é possível observar a olho nu alguma semelhança com algum modo de vibração do sistema.

- Caso 2 $X(0) = 1, 10, -4, 3, -2$:

K das molas:

[42, 44, 46, 48, 50, 52]

Matriz A:

```
[ [ 43. -22.  0.  0.  0.]
  [-22.  45. -23.  0.  0.]
  [  0. -23.  47. -24.  0.]
  [  0.  0. -24.  49. -25.]
  [  0.  0.  0. -25.  51.]]
```

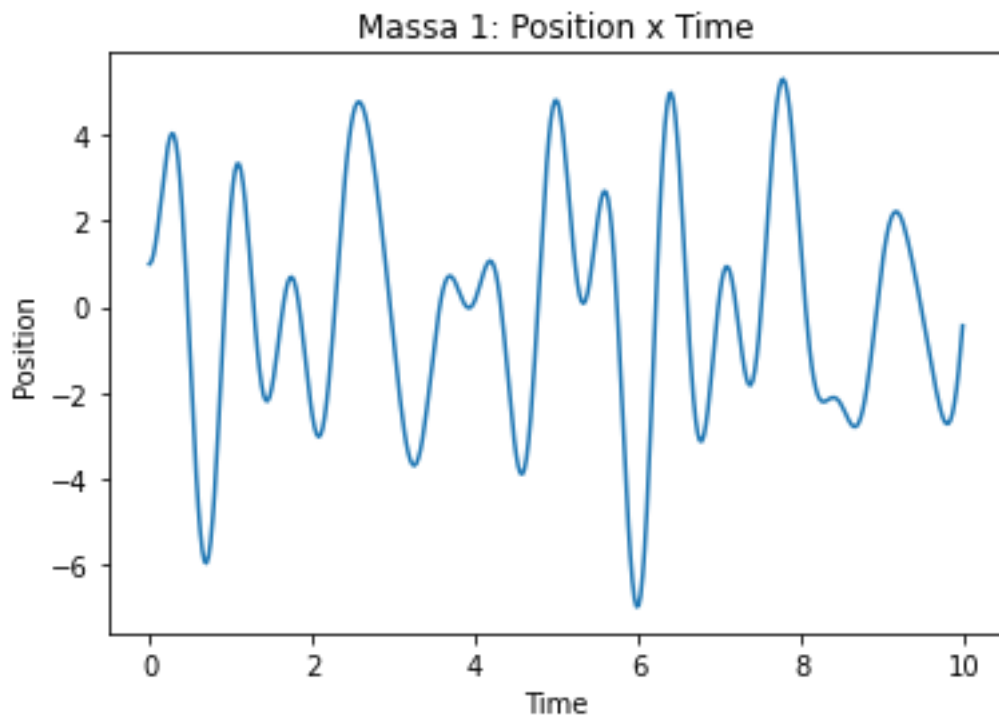
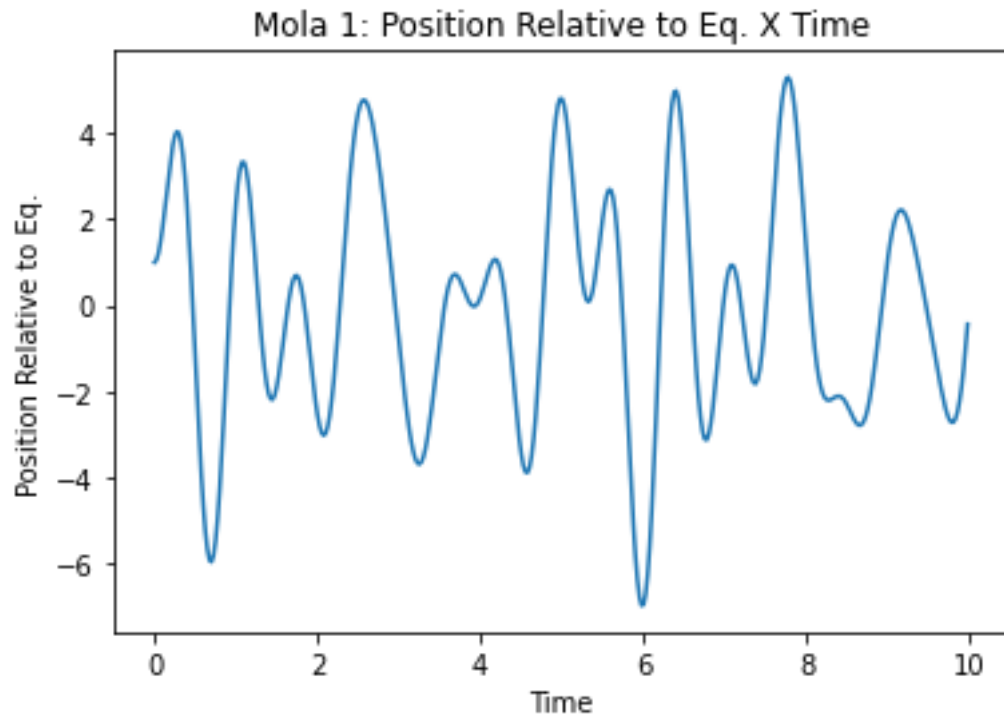
Matriz A Final:

```
[ [ 8.84450056e+01  5.78838094e-15 -6.80980129e-14  3.34743725e-14
   6.16571196e-15]
  [ 0.00000000e+00  7.01138311e+01 -4.60829189e-13  1.16752674e-13
   9.40125676e-15]
  [ 2.22706817e-16 -3.33487620e-17  4.67731863e+01 -1.34322149e-06
  -2.43423935e-14]
  [ 4.04275829e-16 -1.24525312e-15 -1.34322055e-06  2.33986332e+01
  -2.10284583e-07]
  [ 1.82314418e-16 -3.24321795e-16 -3.30493520e-16 -2.10284784e-07
   6.26934375e+00]]
```

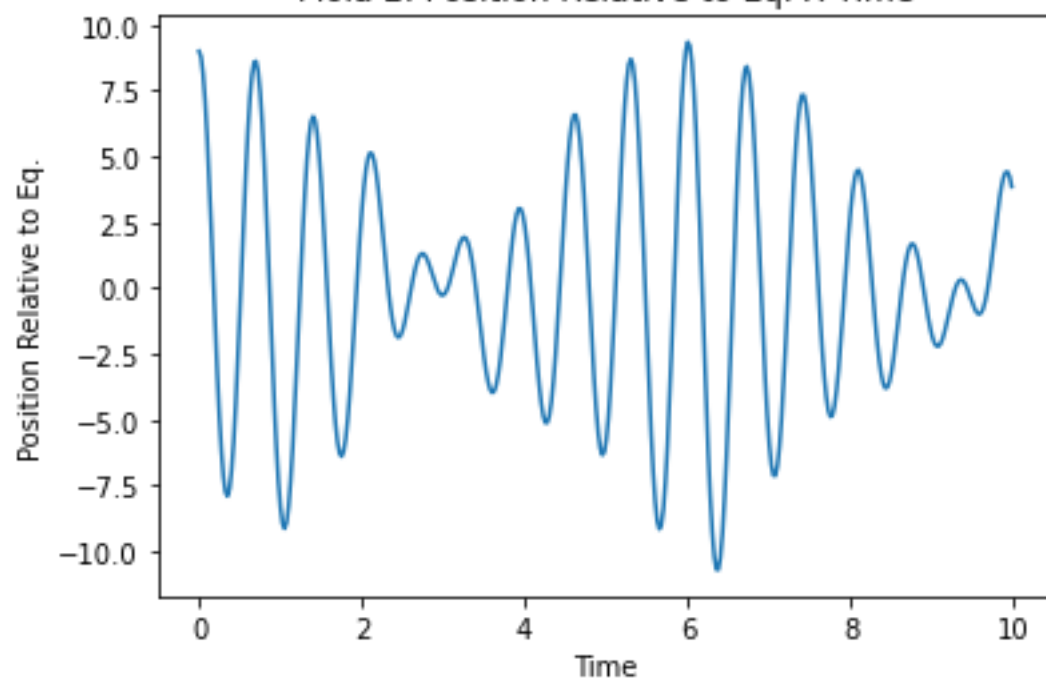
Matriz V Final:

```
[[ 0.18933466 -0.4749761 -0.59882243 -0.53250392 -0.31048568]
 [-0.39110521  0.58538281  0.10270309 -0.47444571 -0.51837923]
 [ 0.55766111 -0.18485786  0.56486874  0.06375698 -0.57593403]
 [-0.588202   -0.38295964 -0.09308546  0.51737516 -0.4806437 ]
 [ 0.39271058  0.50089335 -0.55056532  0.46861374 -0.26863216]]
```

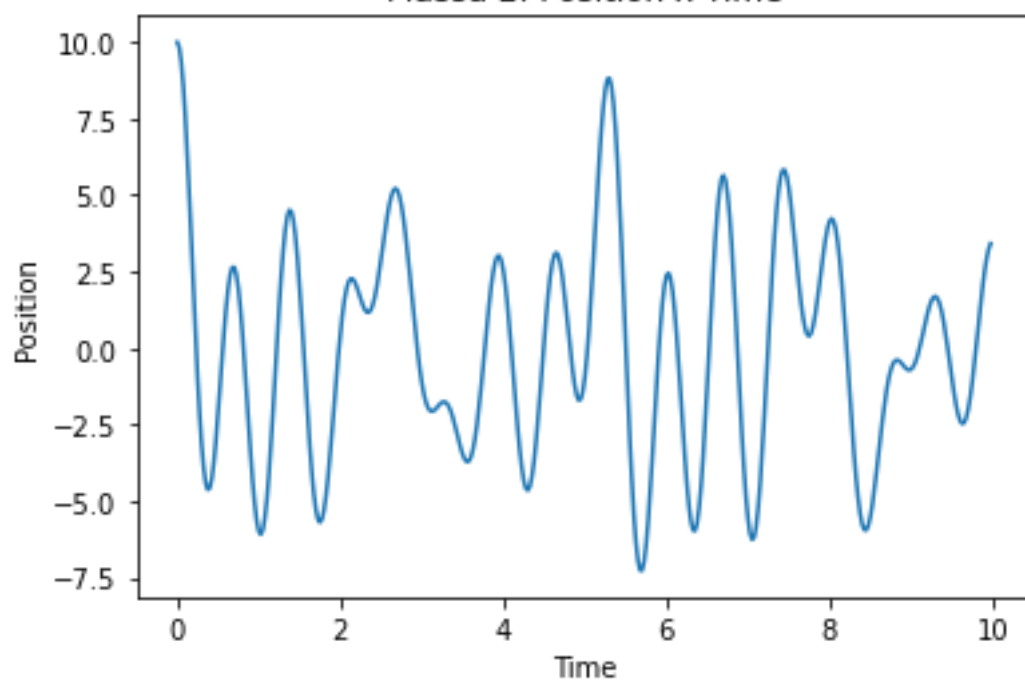
iterações k com deslocamento: 9



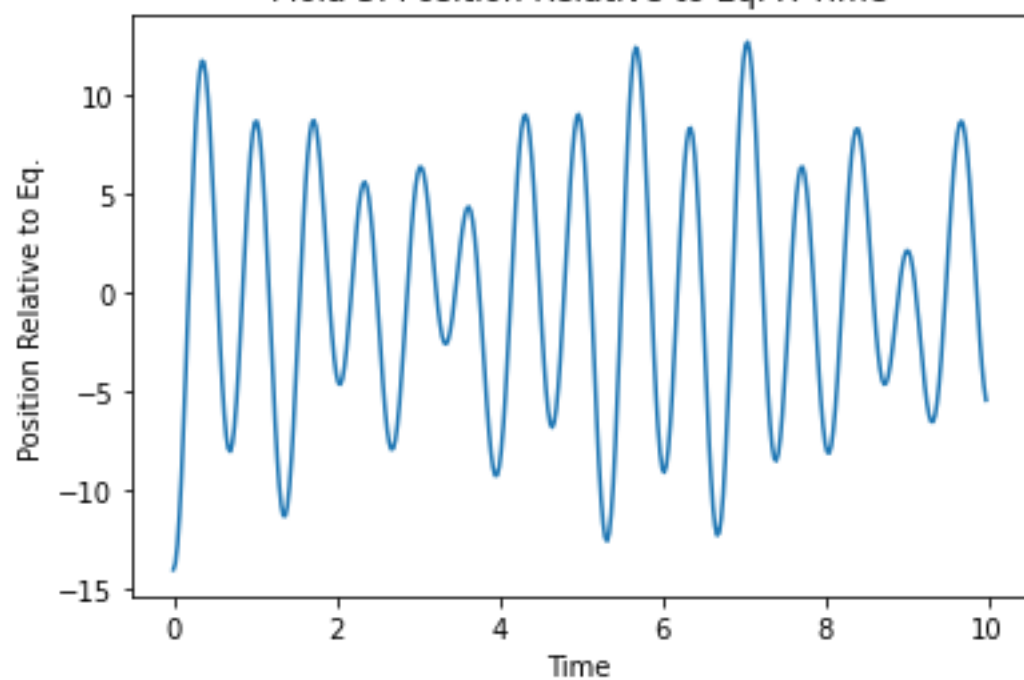
Mola 2: Position Relative to Eq. X Time



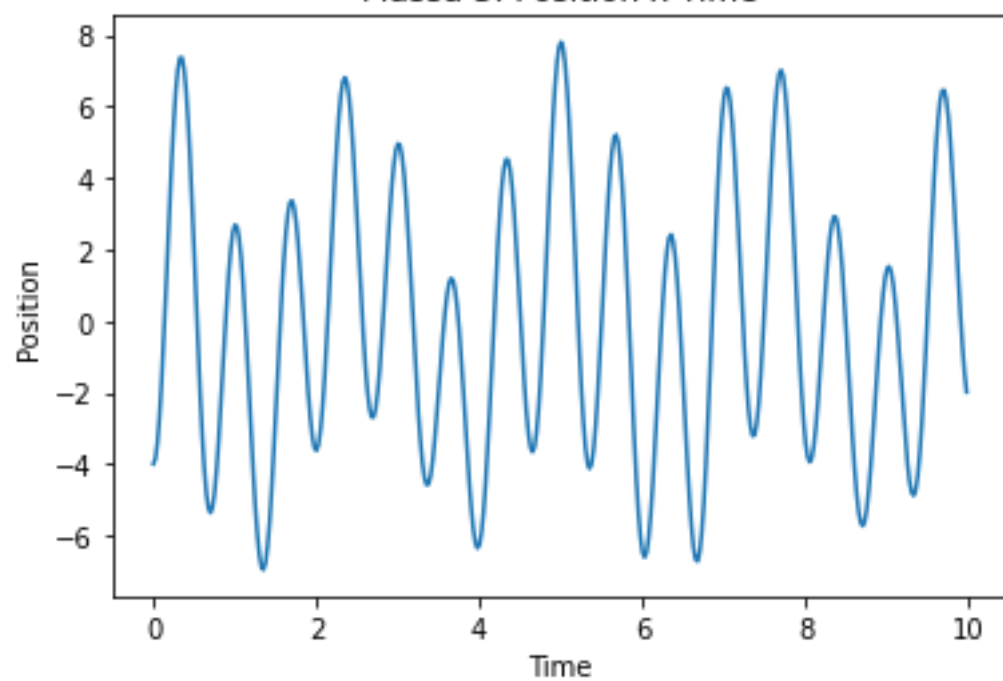
Massa 2: Position x Time



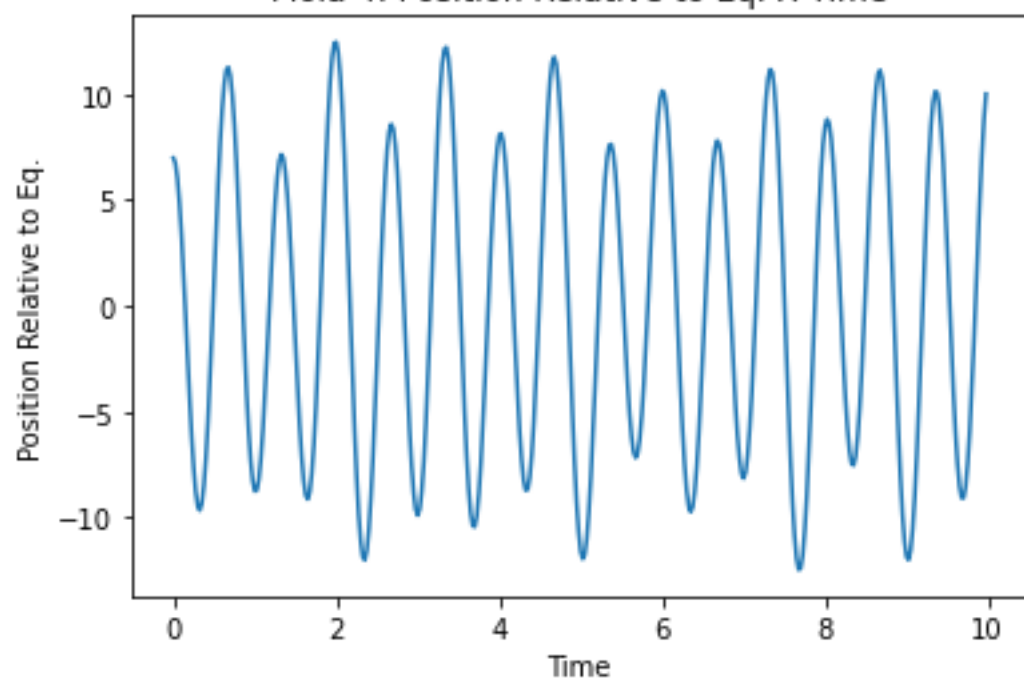
Mola 3: Position Relative to Eq. X Time



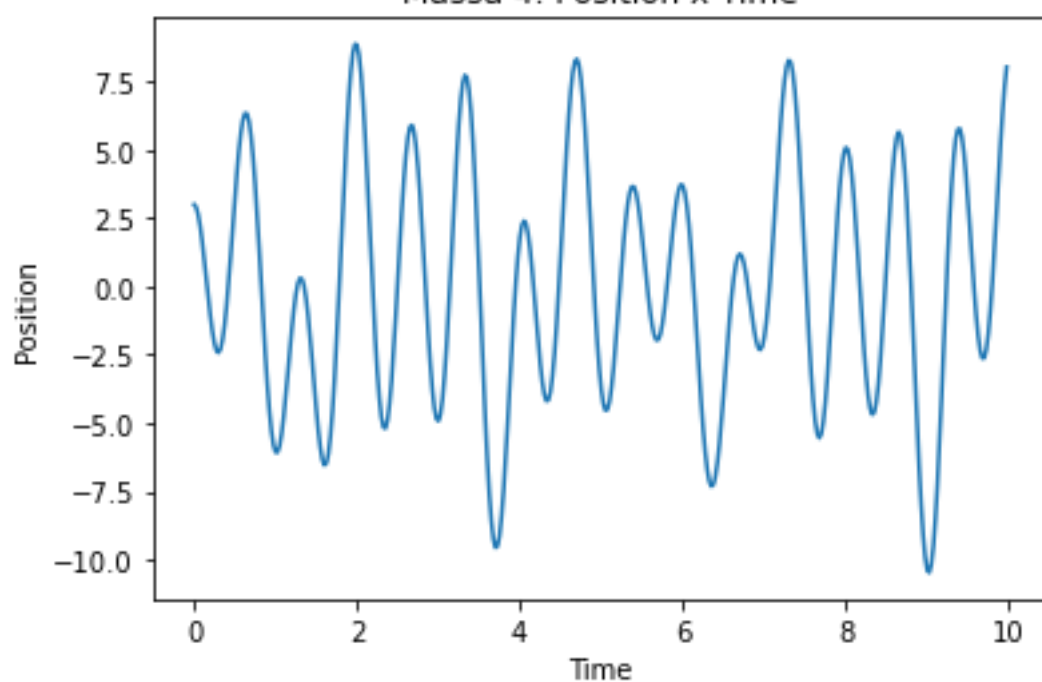
Massa 3: Position x Time



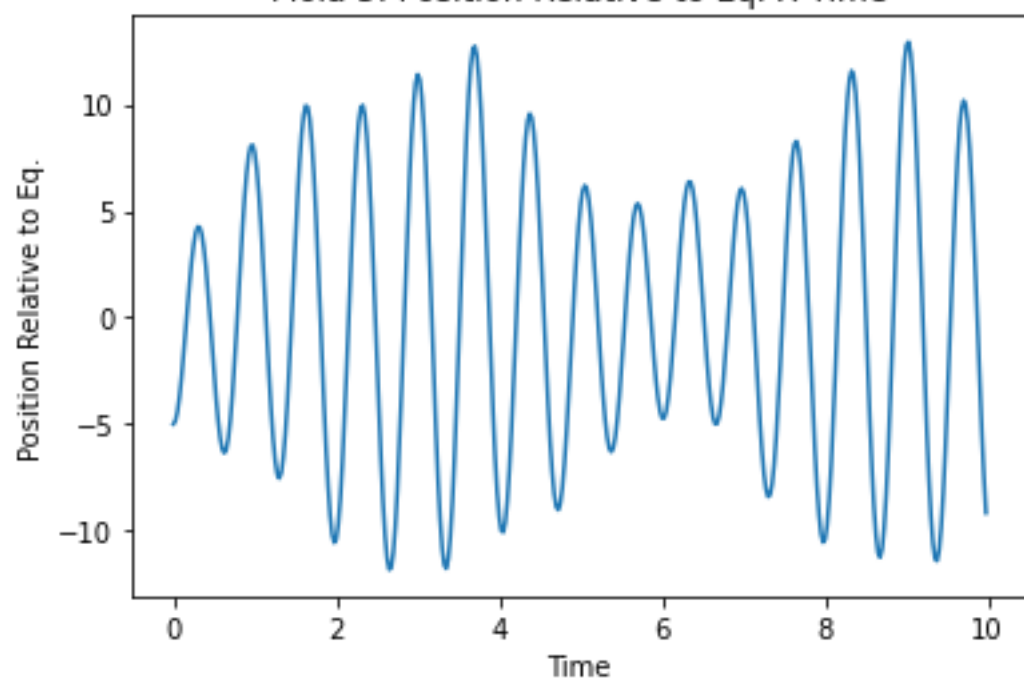
Mola 4: Position Relative to Eq. X Time



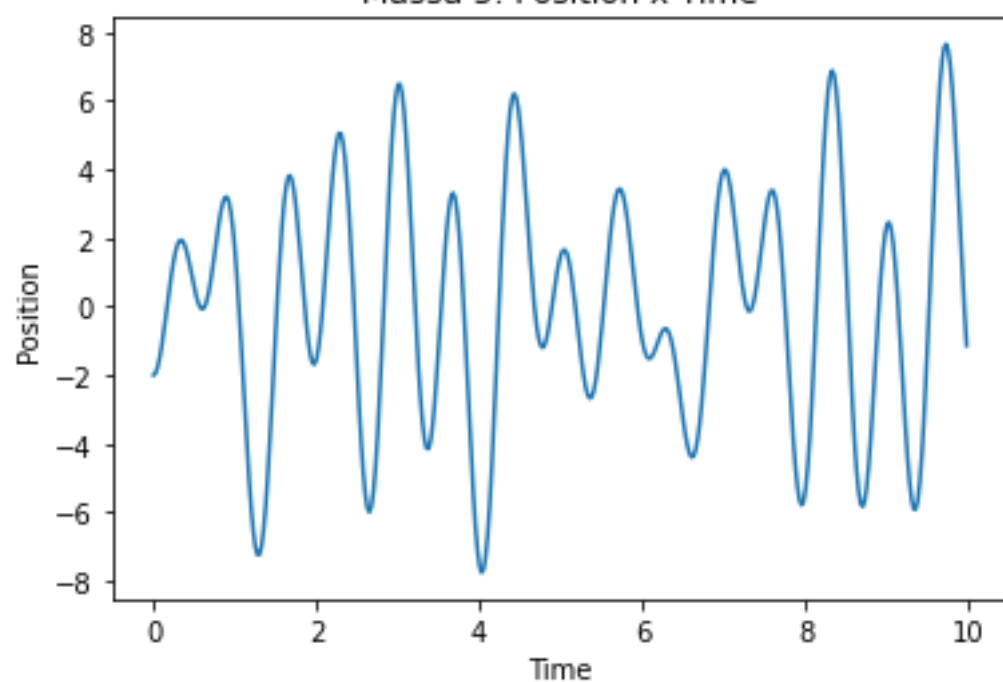
Massa 4: Position x Time

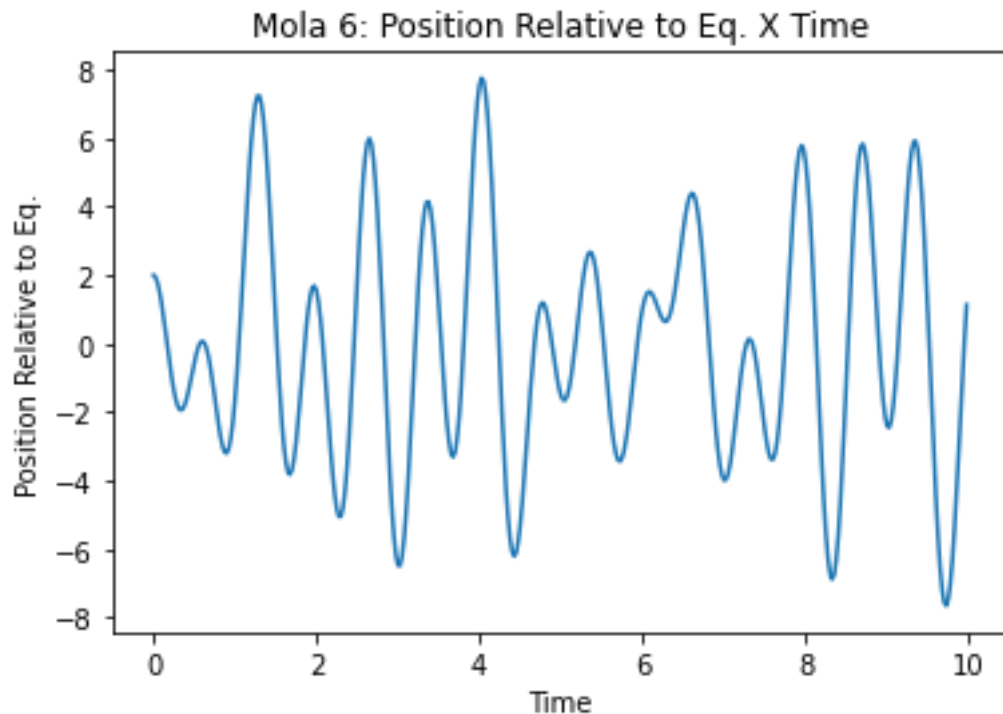


Mola 5: Position Relative to Eq. X Time



Massa 5: Position x Time





Podemos observar neste caso que os gráficos começam a tomar uma forma mais parecida a algum modo de vibração. A combinação da posição inicial, portanto, se aproxima mais de algum autovetor do sistema.

- Caso 3 correspondente ao modo de maior frequência:

K das molas:

[42, 44, 46, 48, 50, 52]

Matriz A:

```
[ [ 43. -22.  0.  0.  0.]
  [-22.  45. -23.  0.  0.]
  [  0. -23.  47. -24.  0.]
  [  0.  0. -24.  49. -25.]
  [  0.  0.  0. -25.  51.]]
```

Matriz A Final:

```
[ [ 8.84450056e+01  5.78838094e-15 -6.80980129e-14  3.34743725e-14
   6.16571196e-15]
  [ 0.00000000e+00  7.01138311e+01 -4.60829189e-13  1.16752674e-13
   9.40125676e-15]
  [ 2.22706817e-16 -3.33487620e-17  4.67731863e+01 -1.34322149e-06
  -2.43423935e-14]
  [ 4.04275829e-16 -1.24525312e-15 -1.34322055e-06  2.33986332e+01
  -2.10284583e-07]
  [ 1.82314418e-16 -3.24321795e-16 -3.30493520e-16 -2.10284784e-07
   6.26934375e+00]]
```

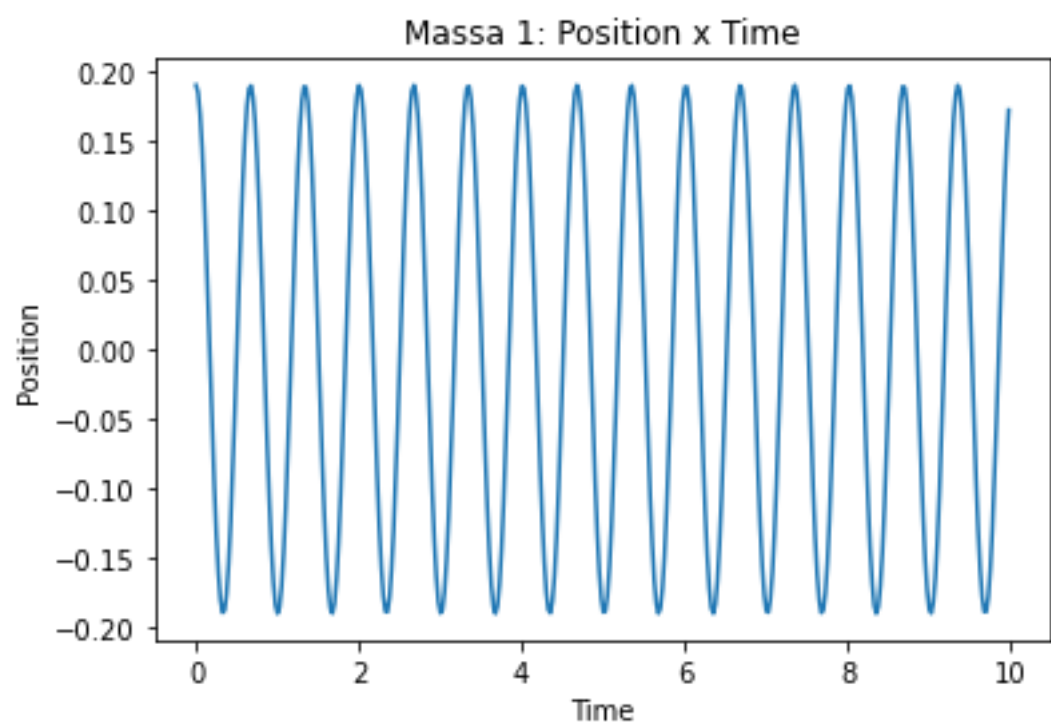
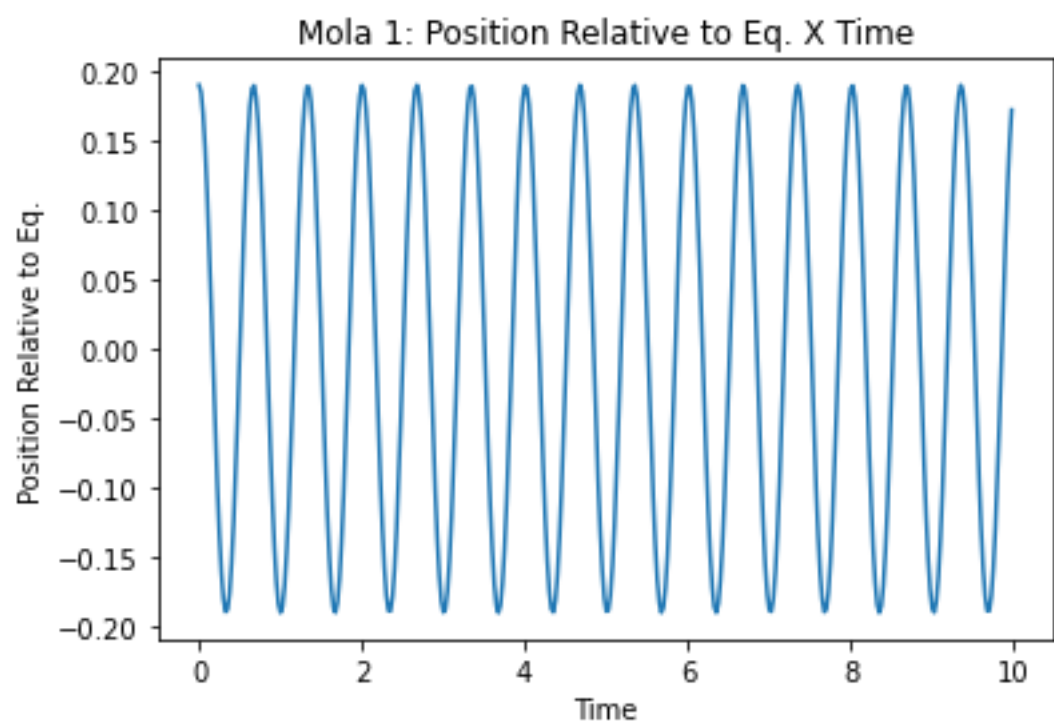
Matriz V Final:

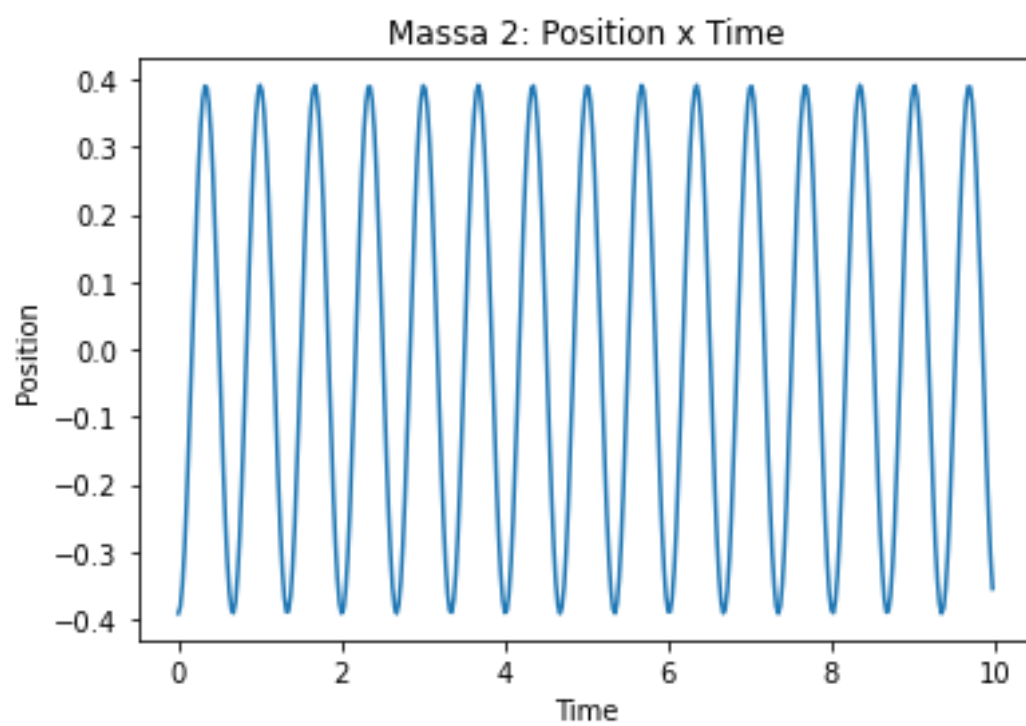
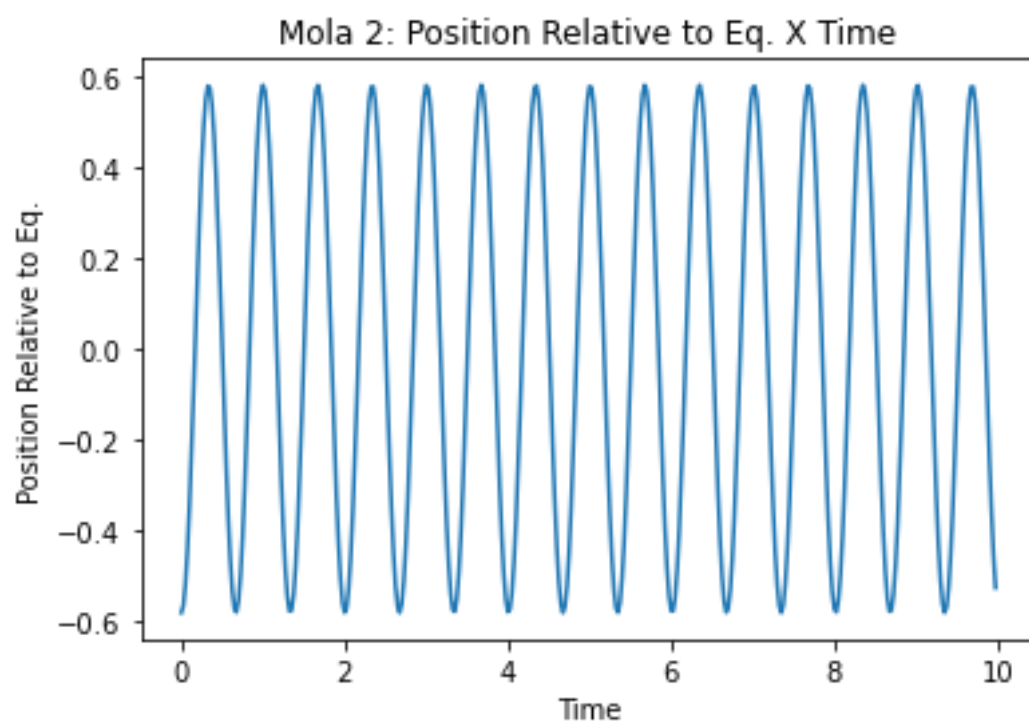
```
[[ 0.18933466 -0.4749761 -0.59882243 -0.53250392 -0.31048568]
 [-0.39110521  0.58538281  0.10270309 -0.47444571 -0.51837923]
 [ 0.55766111 -0.18485786  0.56486874  0.06375698 -0.57593403]
 [-0.588202   -0.38295964 -0.09308546  0.51737516 -0.4806437 ]
 [ 0.39271058  0.50089335 -0.55056532  0.46861374 -0.26863216]]
```

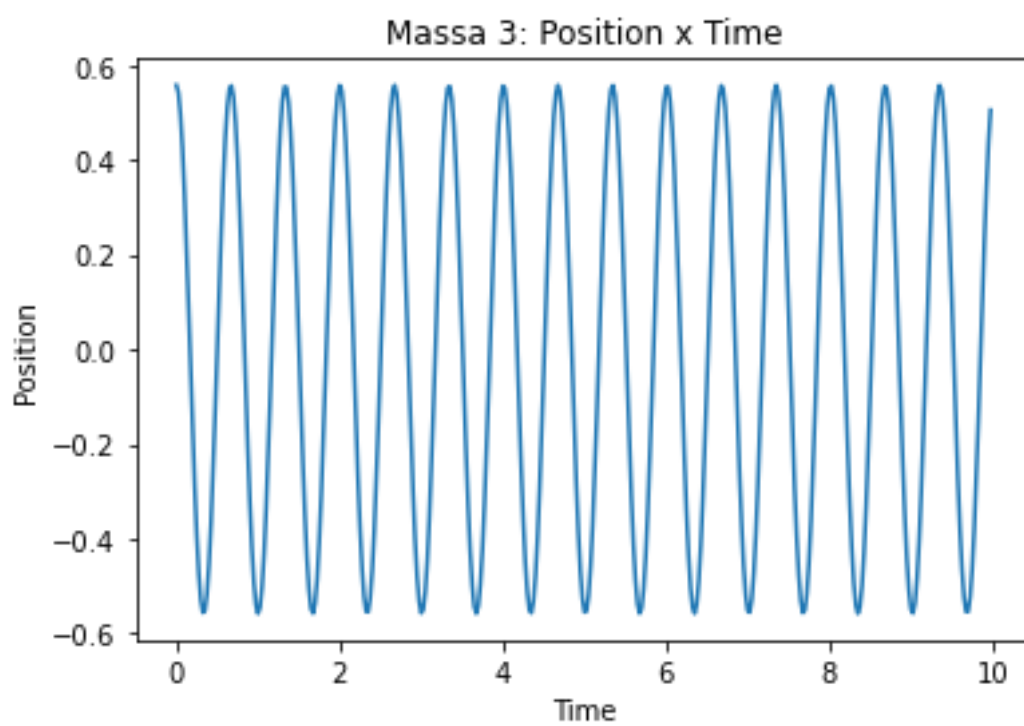
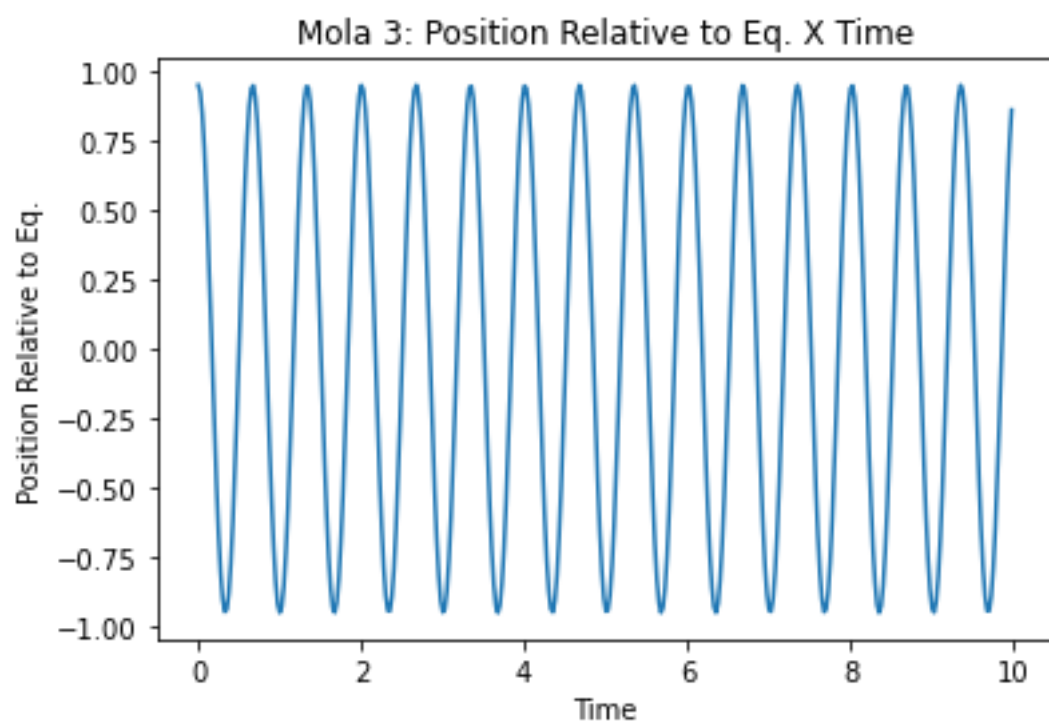
iterações k com deslocamento: 9

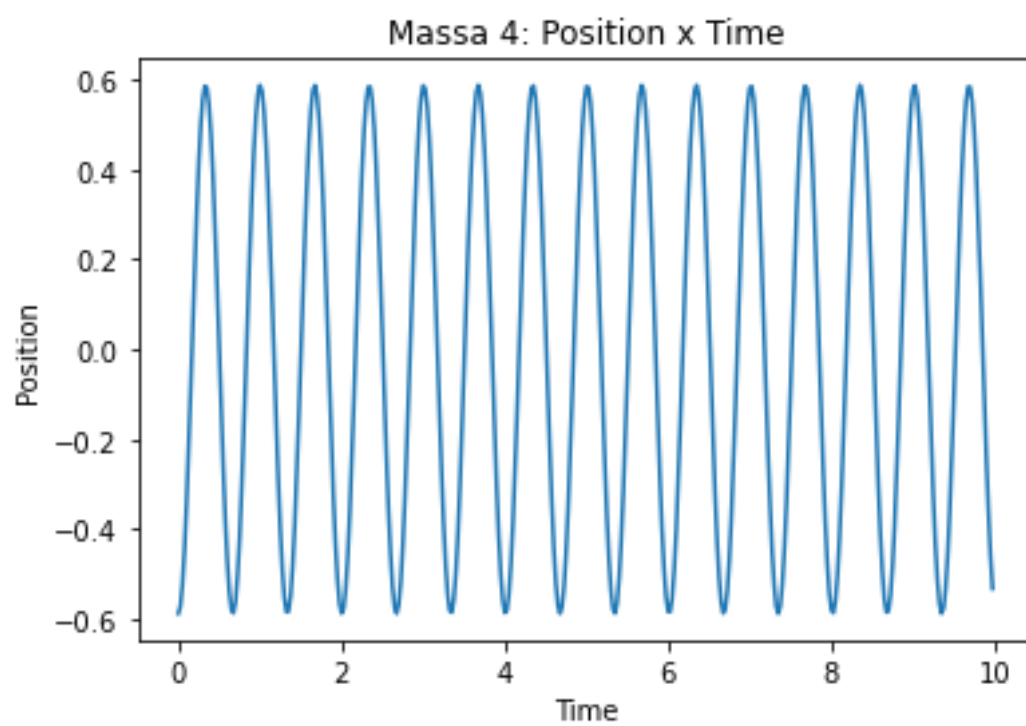
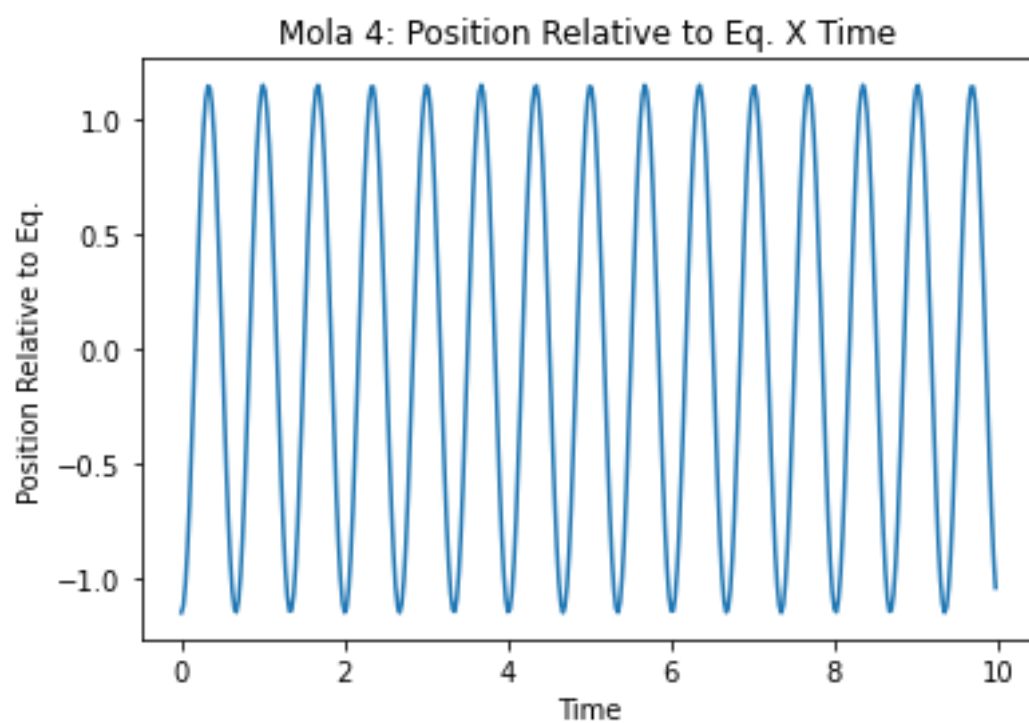
Maior Autovalor e sua frequência: $\lambda=88.44500562036657$ $f=1.496775923168591$
5

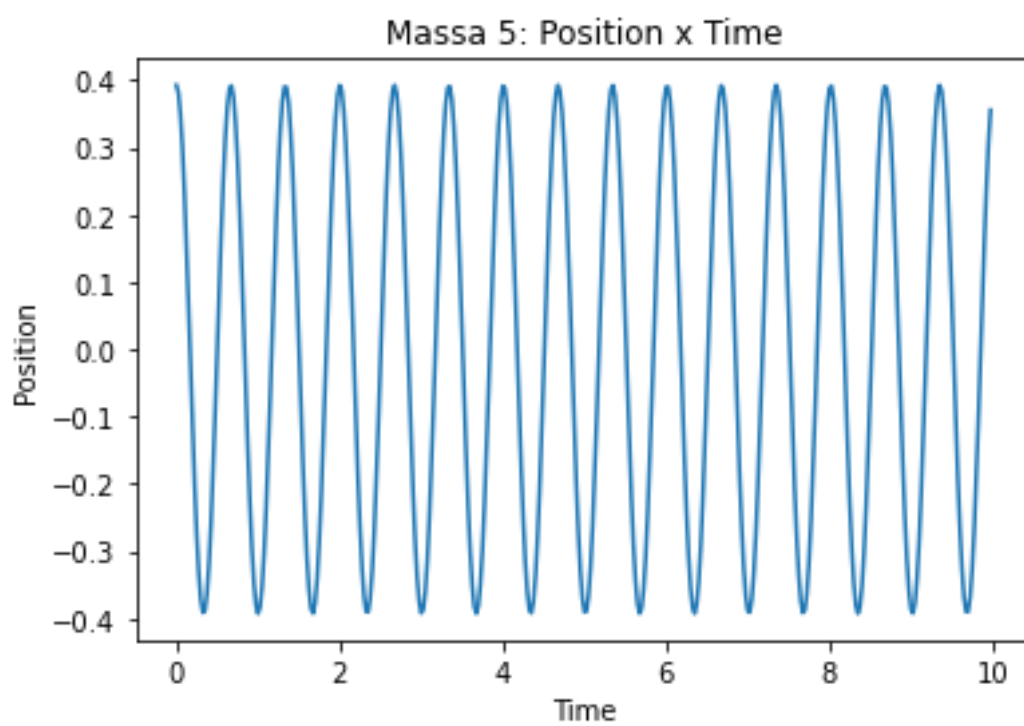
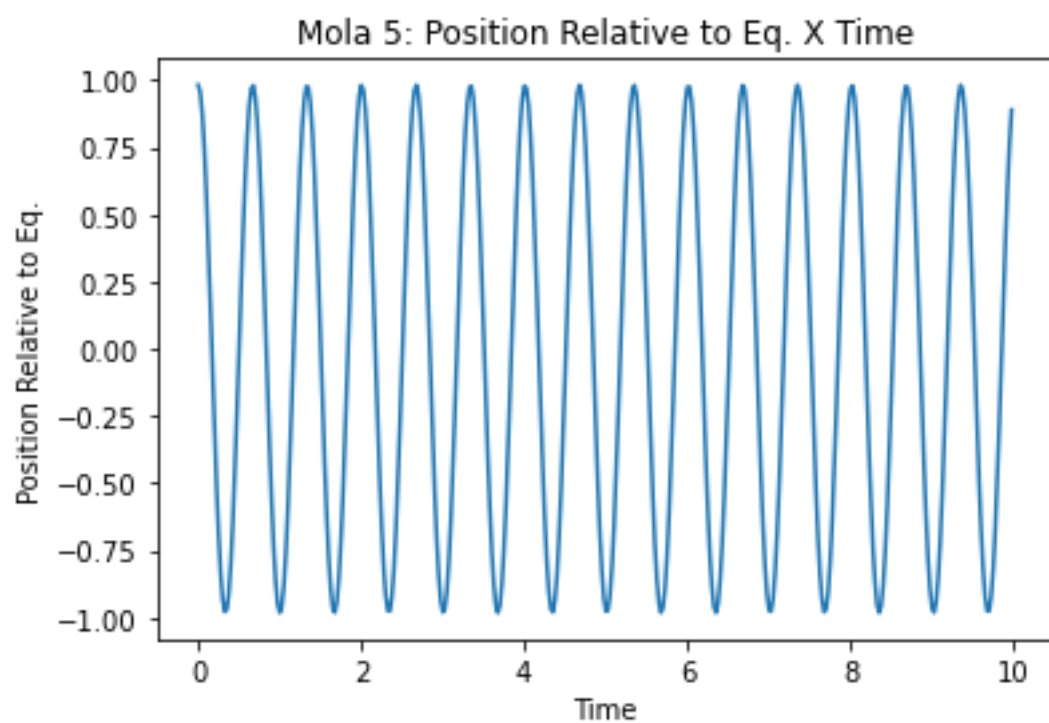
Posições iniciais (Autovetor correspondente): [0.18933466 -0.39110521 0
.55766111 -0.588202 0.39271058]

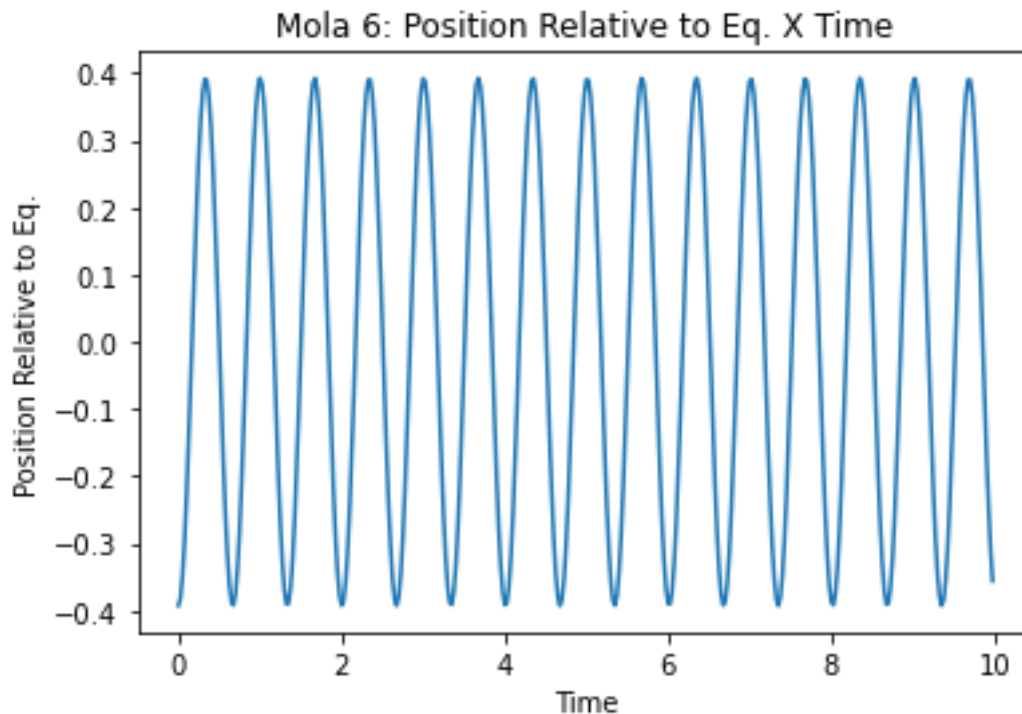












Os outros modos de vibração e suas respectivas frequências são:

$f=1.3326680344183768$ Autovetor= $[-0.4749761042860255, 0.5853828128471872, -0.18485786148930233, -0.3829596375780954, 0.5008933517347197]$

$f=1.0884753789933004$ Autovetor= $[-0.5988224288056447, 0.10270308605544064, 0.5648687429201433, -0.09308545611115204, -0.5505653240137749]$

$f=0.7698664180935881$ Autovetor= $[-0.5325039218892617, -0.4744457086012574, 0.06375697872911724, 0.517375157475226, 0.46861373950539015]$

$f=0.39850261287318234$ Autovetor= $[-0.31048568377649105, -0.5183792287236151, -0.5759340334428541, -0.48064370259136613, -0.26863215682278785]$

Temos agora o modo de vibração com a maior frequência. Podemos observar que todas as massas vibram na mesma frequência equivalente à $f=1.4967759231685915$. Como explicado anteriormente isto ocorre por conta das posições iniciais serem múltiplos dos elementos de um autovetor, neste caso o autovetor correspondente ao maior autovalor. Os outros modos de vibração e suas respectivas frequências estão apresentadas acima também.

4. Sistema massa mola com 10 massas e 3 casos de posição inicial

Para esta tarefa os conceitos usados são os mesmos utilizados no item anterior, porém agora para uma matriz de ordem maior:

- Caso 1 $X(0) = -2, -3, -1, -3, -1, -2, -3, -1, -3, -1$:

K das molas:

[38, 42, 38, 42, 38, 42, 38, 42, 38, 42, 38]

Matriz A:

```
[ 40. -21.  0.  0.  0.  0.  0.  0.  0.  0.]
[-21.  40. -19.  0.  0.  0.  0.  0.  0.  0.]
[  0. -19.  40. -21.  0.  0.  0.  0.  0.  0.]
[  0.  0. -21.  40. -19.  0.  0.  0.  0.  0.]
[  0.  0.  0. -19.  40. -21.  0.  0.  0.  0.]
[  0.  0.  0.  0. -21.  40. -19.  0.  0.  0.]
[  0.  0.  0.  0.  0. -19.  40. -21.  0.  0.]
[  0.  0.  0.  0.  0.  0. -21.  40. -19.  0.]
[  0.  0.  0.  0.  0.  0.  0. -19.  40. -21.]
[  0.  0.  0.  0.  0.  0.  0.  0. -21.  40.]
```

Matriz A Final:

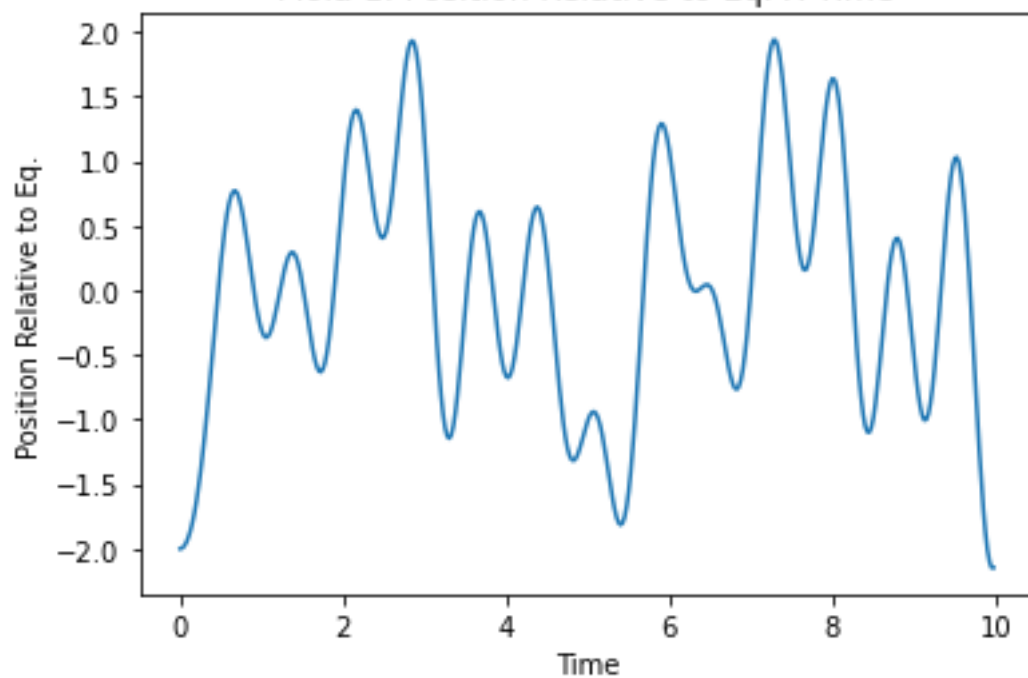
```
[ 7.83987480e+01  9.47633661e-16 -5.83533533e-16  3.63724107e-15
 -5.41954136e-15  7.25962940e-15 -3.42238051e-15 -9.33362852e-15
 -1.73546928e-15  8.43679341e-12]
[ 0.00000000e+00  7.37299279e+01 -4.62636583e-15  8.24039574e-15
 -3.52096647e-14  5.60506940e-15 -3.97965931e-15 -7.89495755e-15
  2.42841838e-16  1.95223432e-11]
[ 4.74877169e-17 -5.94497538e-18  6.63937585e+01  1.32035483e-06
  1.85445089e-13 -2.29730250e-15  6.48316360e-16 -2.90073220e-16
 -1.10537153e-15 -3.75458995e-11]
[-1.09397677e-16  7.36406243e-17  1.32035486e-06  5.70639508e+01
  2.19220090e-07  2.62993928e-15 -1.46336483e-15  1.52547472e-15
 -3.50769396e-15  7.12931228e-11]
[ 1.59465725e-16  4.97493542e-16  1.25465042e-16  2.19221814e-07
  4.70013722e+01 -4.53343474e-06  6.65201080e-15 -3.32043115e-15
  7.64453543e-16 -1.29109409e-10]
[-5.26811402e-16 -5.32213037e-16  9.45270780e-16 -5.33038162e-16
 -4.53343474e-06  3.29986278e+01 -1.94350551e-04  1.47755086e-14
 -3.64326837e-15  3.91329676e-10]
[-3.47899812e-16  1.68742221e-18 -9.96963659e-16  1.31990462e-16
 -4.93694870e-16 -1.94350551e-04  2.29360492e+01  2.18800125e-06
  1.66602726e-14 -1.26300747e-09]
[-1.83366032e-16 -6.66936637e-16 -4.55360442e-16 -7.04789815e-16
 -1.50197796e-17  1.41566349e-16  2.18800140e-06  1.36062415e+01
  5.30852287e-06  6.05616761e-09]
[ 2.76720478e-16 -2.31255296e-16  6.41577625e-17 -1.25452449e-15
 -5.85125141e-16  4.37861694e-16  8.32717145e-16  5.30852301e-06
  6.27007209e+00  4.75197022e-07]
[-2.31395668e-16 -7.35137925e-16  9.48185005e-16  2.55815089e-16
  8.81122438e-16 -4.32179164e-16 -1.12499058e-15  6.96649650e-17
  5.58740020e-07  1.60125202e+00]
```

Matriz V Final:

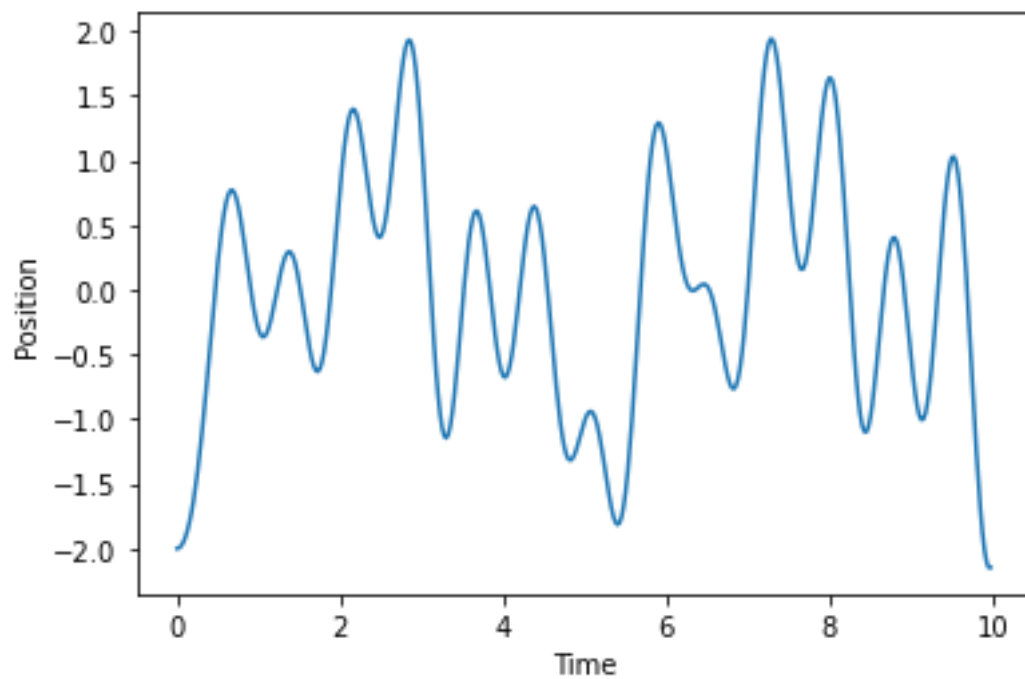
```
[ [ 0.12524522 -0.24007379 -0.33455077 -0.39821223 -0.39519756 -0.39520499
    -0.39820448 -0.33455074 -0.24007404  0.12524519]
  [-0.22901235  0.38560341  0.4204787  0.32357498  0.13175826 -0.13176459
    -0.32357228 -0.42047854 -0.38560374  0.22901232]
  [ 0.32440201 -0.41920135 -0.21433934  0.14952573  0.38824524  0.38824787
    0.14951832 -0.21433898 -0.41920154  0.32440197]
  [-0.38597125  0.3244365  -0.11104165 -0.41425829 -0.24865022  0.24865838
    0.41425345  0.11104204 -0.32443646  0.38597122]
  [ 0.42149319 -0.11263113  0.39115438  0.20678121 -0.33748716 -0.33748295
    0.20678757  0.39115454 -0.11263089  0.42149318]
  [-0.42149319 -0.11263113 -0.39115444  0.20678108  0.33748695 -0.33749116
    -0.20677472  0.39115428  0.11263137  0.4214932 ]
  [ 0.38597125  0.3244365  0.11104176 -0.41425827  0.24865037  0.24864221
    -0.41426311  0.11104137  0.32443654  0.38597128]
  [-0.32440201 -0.41920135  0.2143393  0.14952581 -0.38824498  0.38824235
    -0.14953322 -0.21433966  0.41920116  0.32440205]
  [ 0.22901235  0.38560341 -0.42047879  0.32357486 -0.13175834 -0.13175201
    0.32357756 -0.42047895  0.38560308  0.22901239]
  [-0.12524522 -0.24007379  0.33455088 -0.39821216  0.39519729 -0.39518986
    0.39821991 -0.3345509  0.24007353  0.12524524]]
```

iterações k com deslocamento: 19

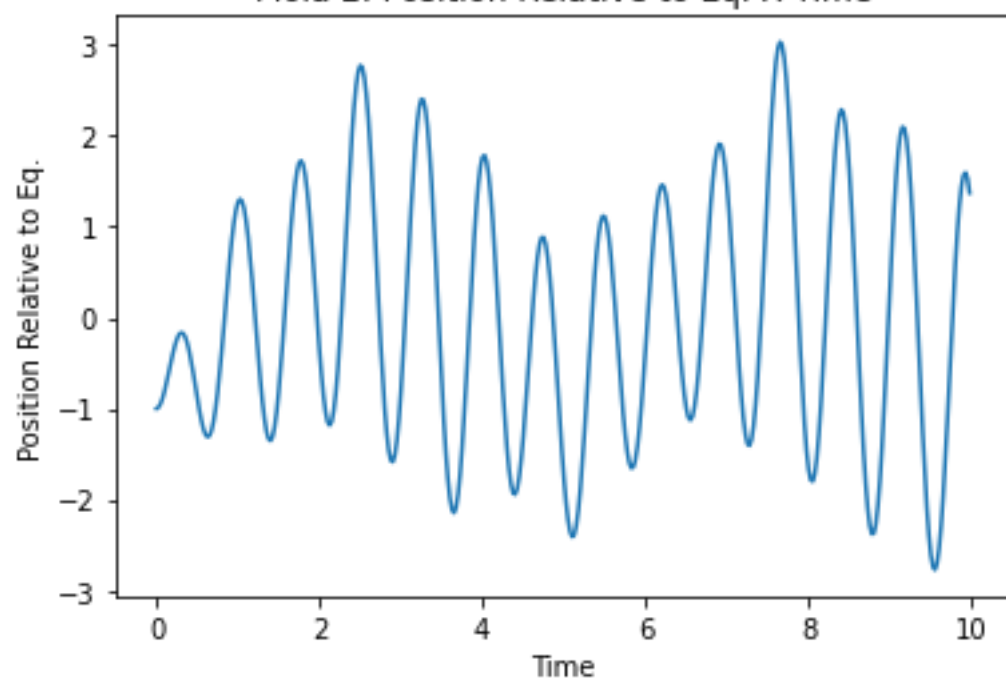
Mola 1: Position Relative to Eq. X Time



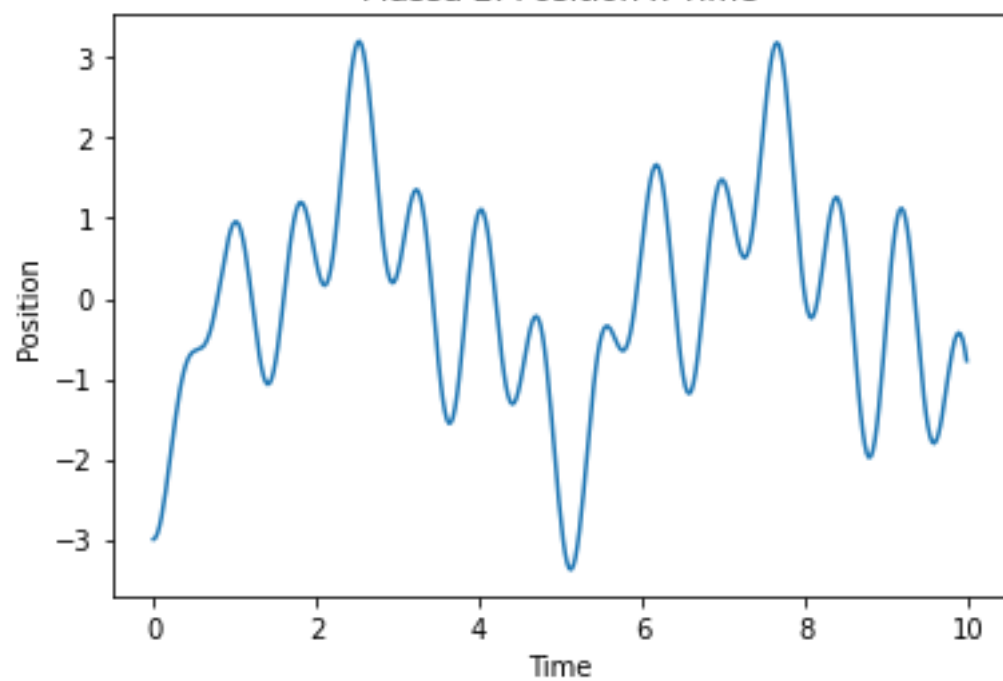
Massa 1: Position x Time



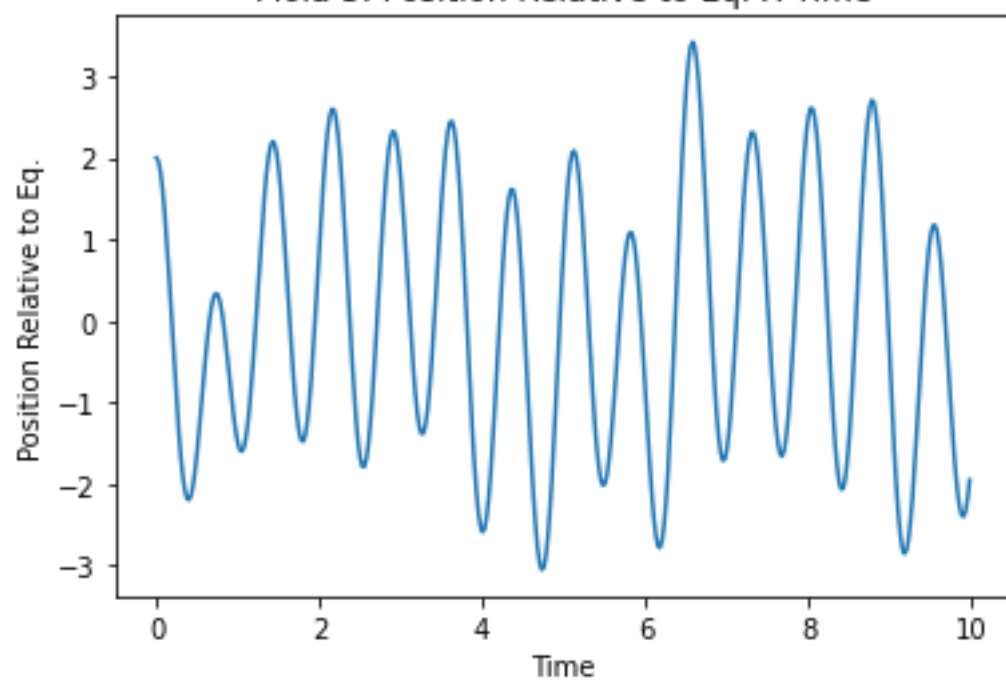
Mola 2: Position Relative to Eq. X Time



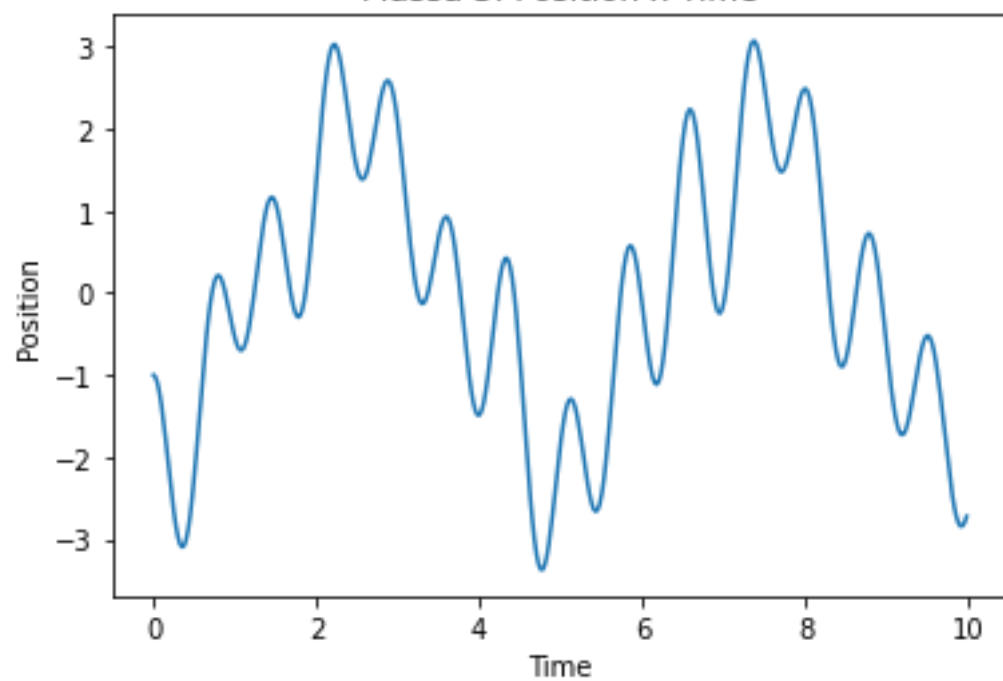
Massa 2: Position x Time



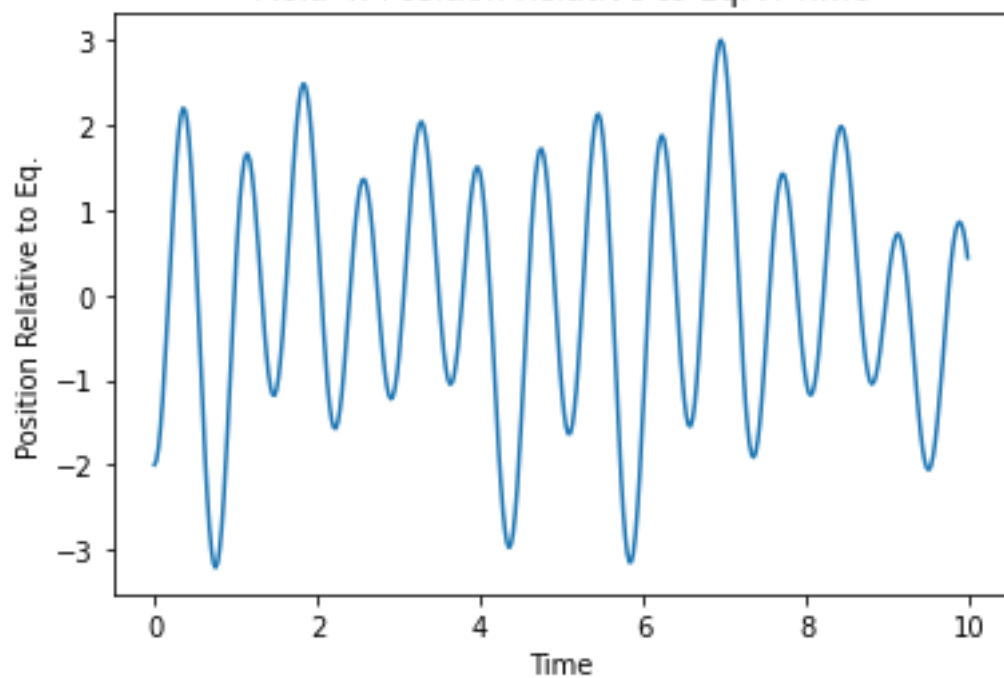
Mola 3: Position Relative to Eq. X Time



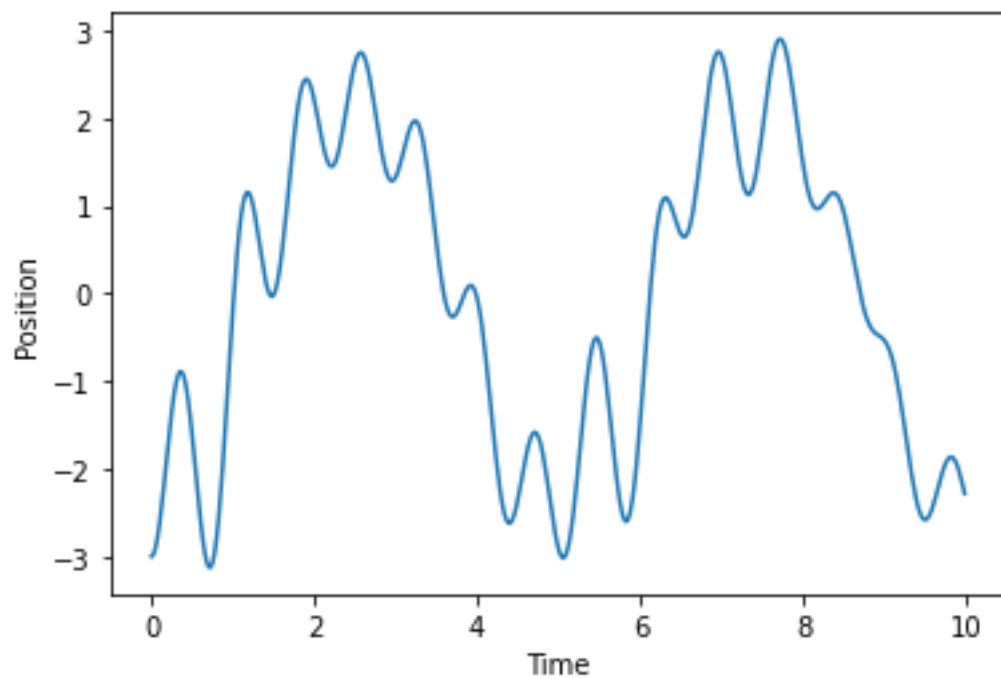
Massa 3: Position x Time



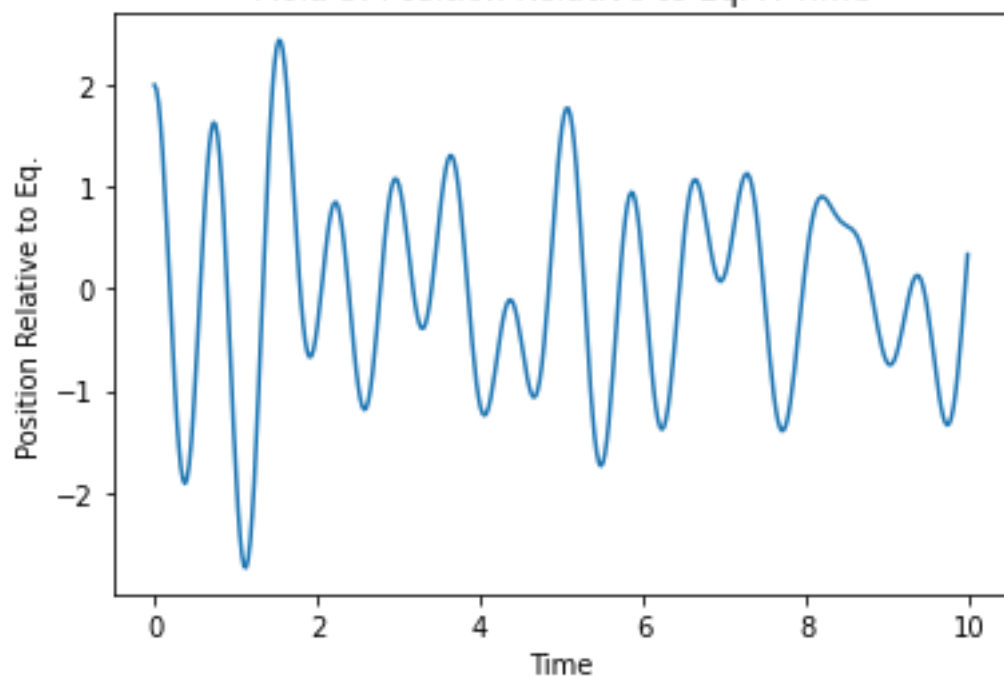
Mola 4: Position Relative to Eq. X Time



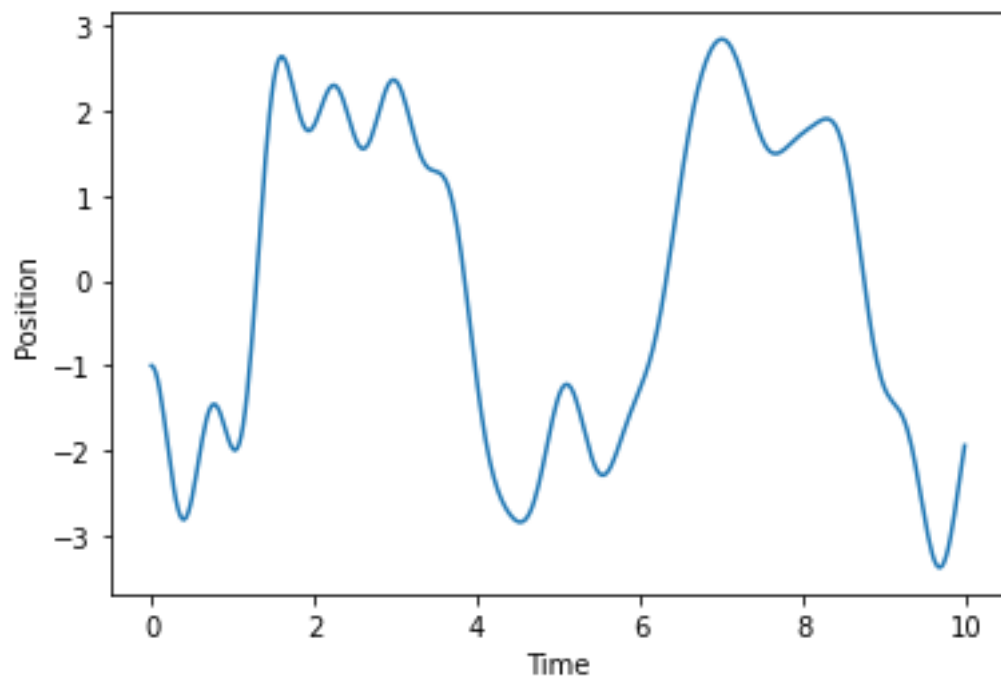
Massa 4: Position x Time



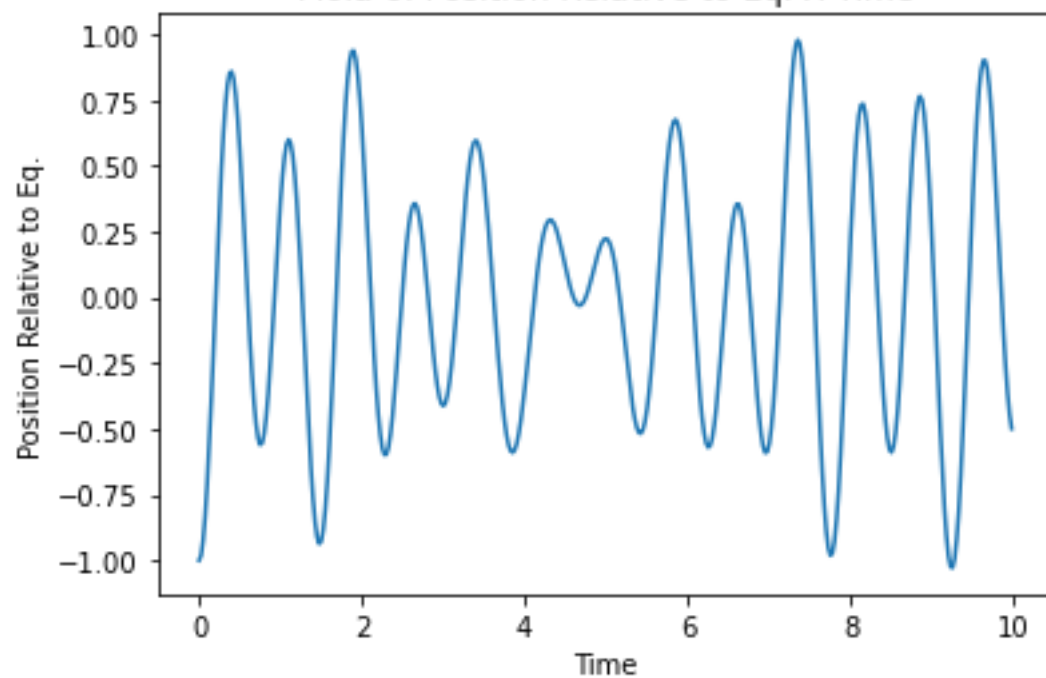
Mola 5: Position Relative to Eq. X Time



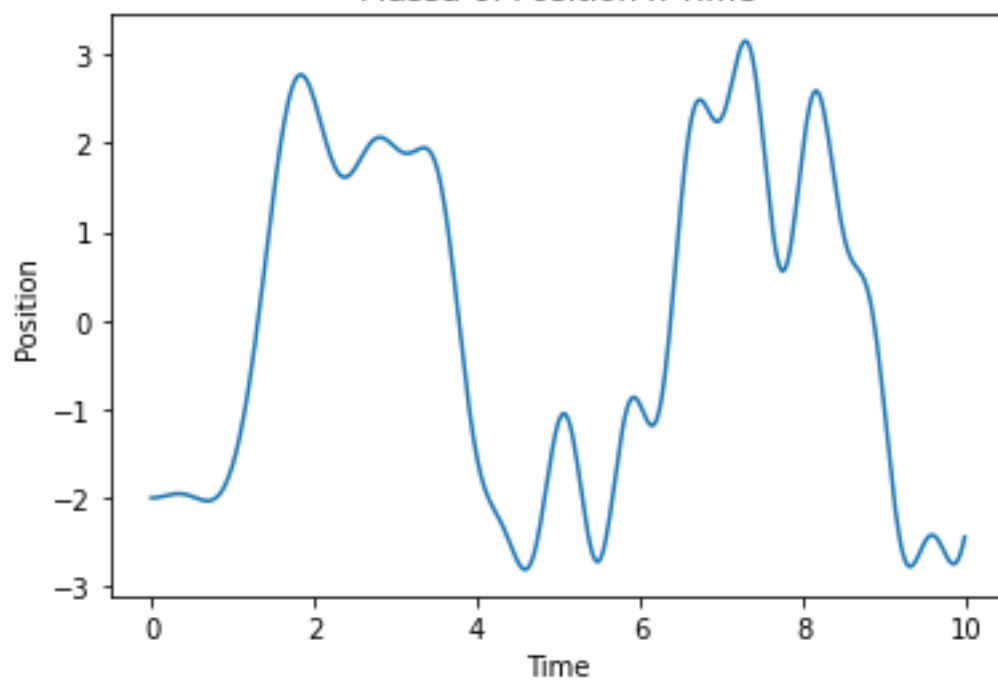
Massa 5: Position x Time



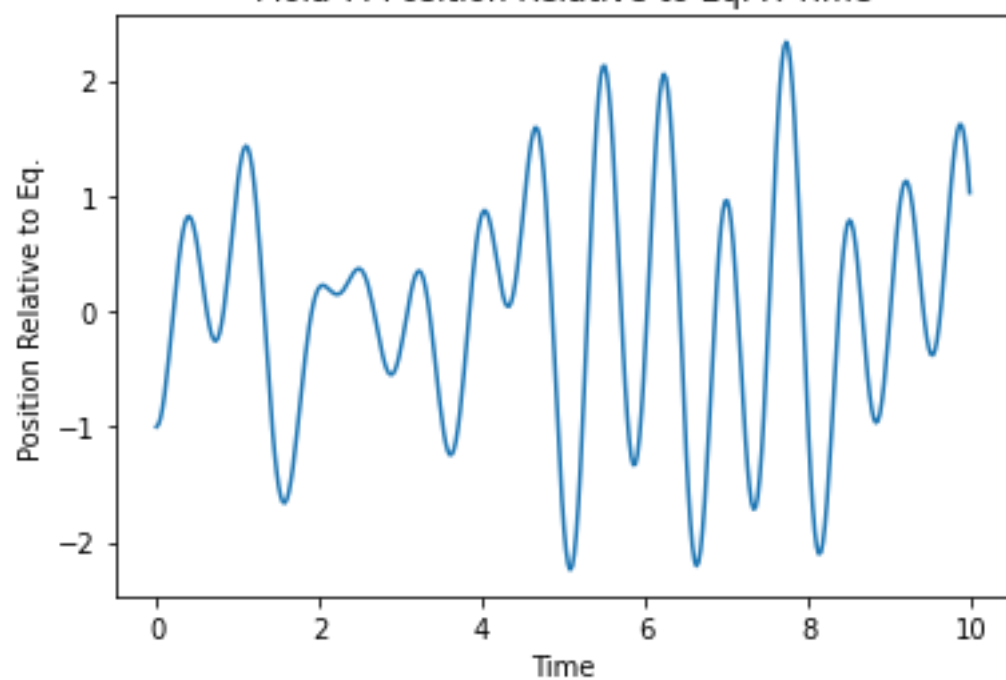
Mola 6: Position Relative to Eq. X Time



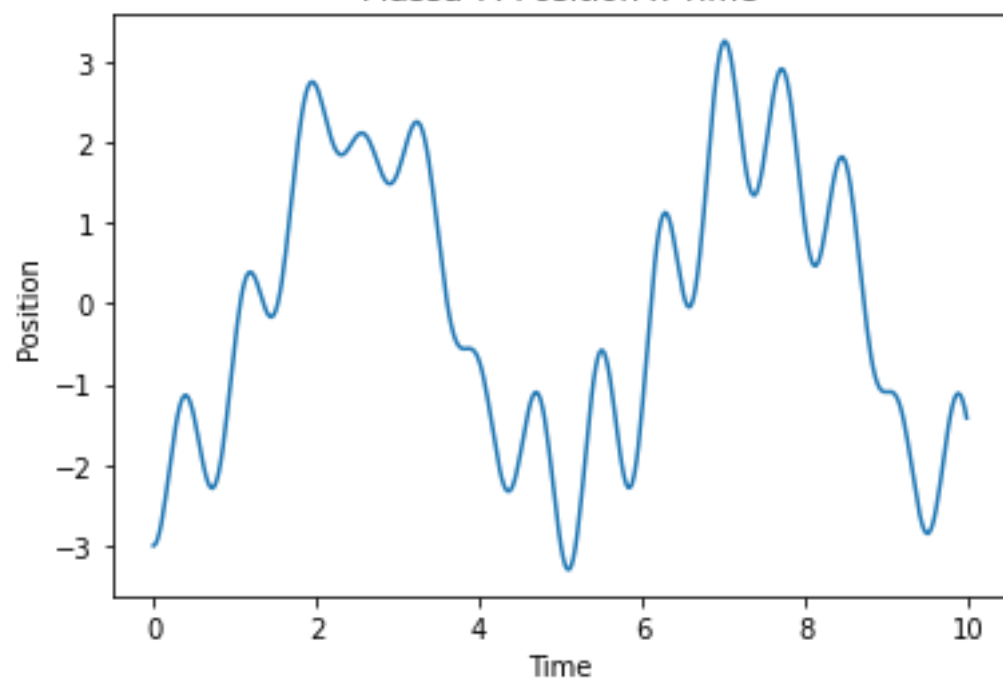
Massa 6: Position x Time

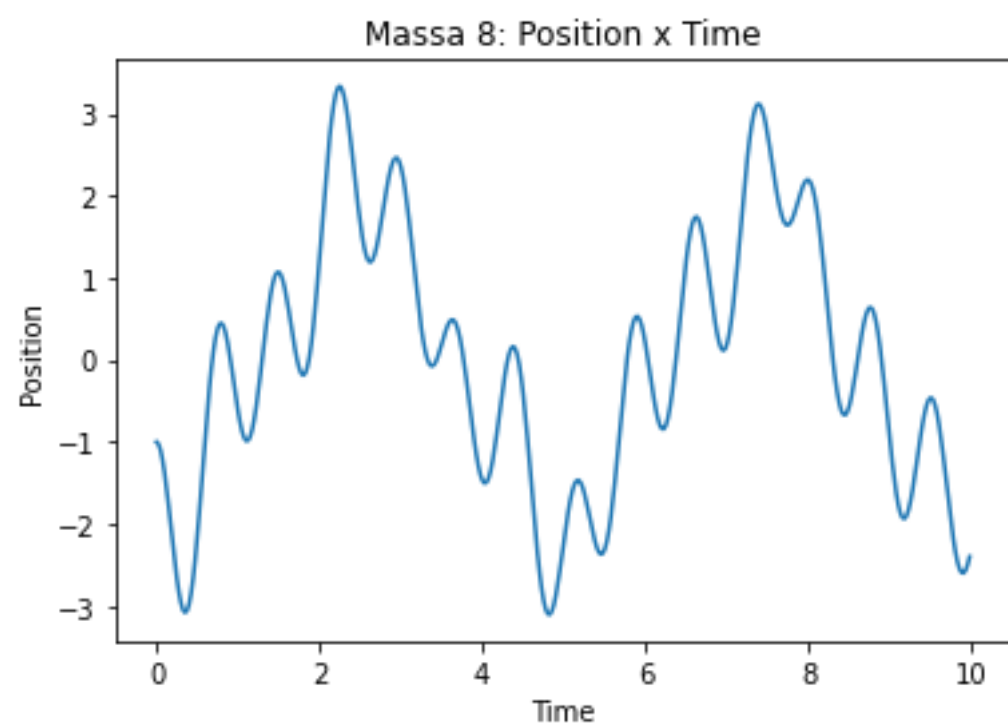
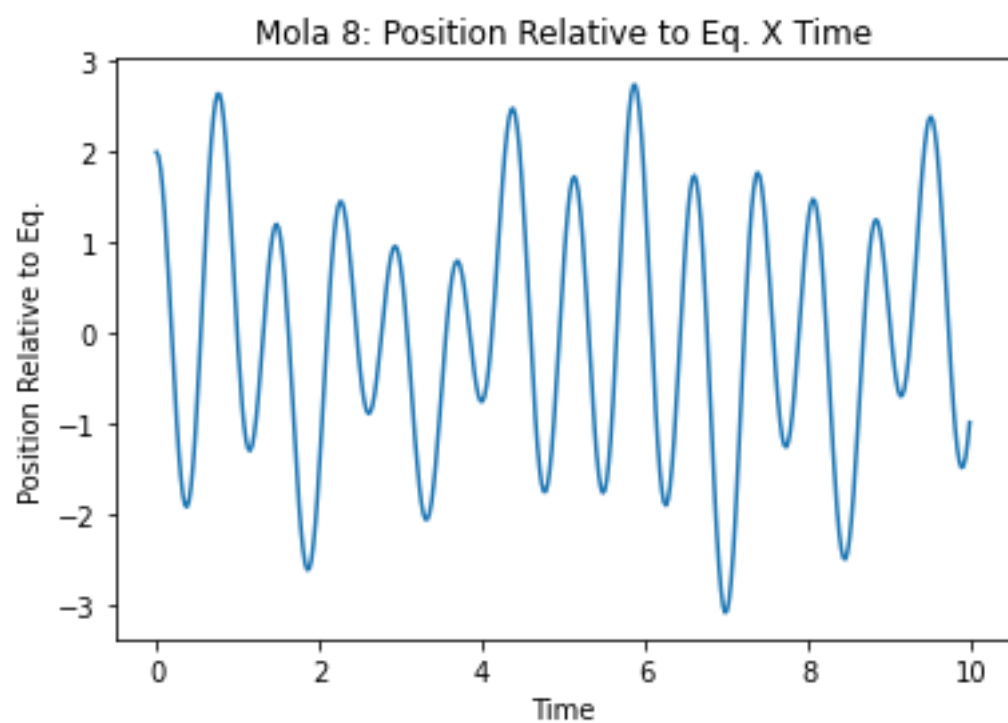


Mola 7: Position Relative to Eq. X Time

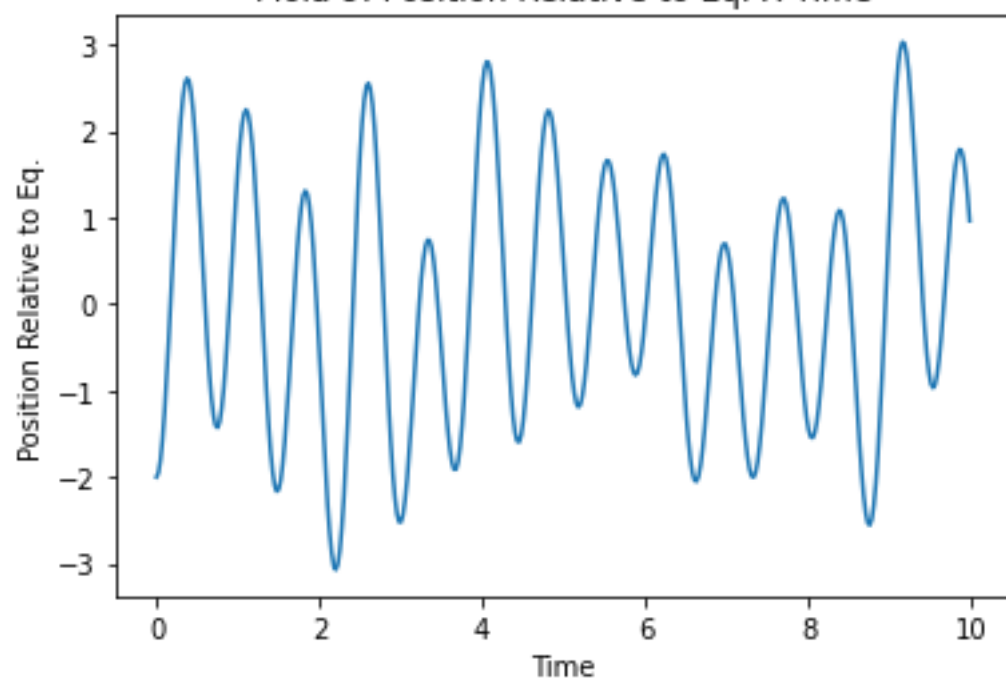


Massa 7: Position x Time

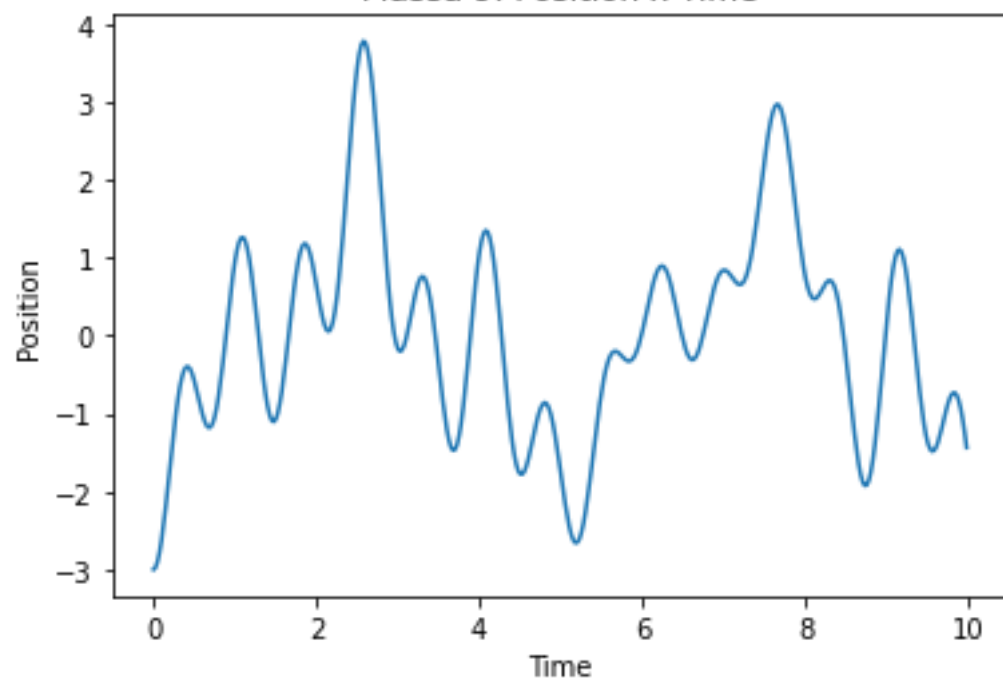




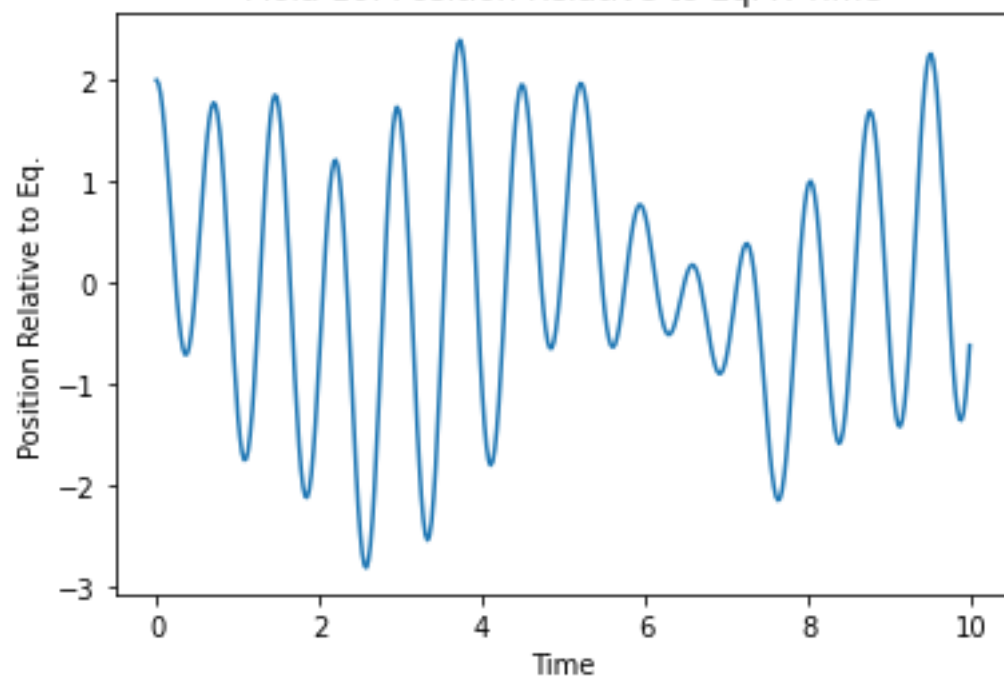
Mola 9: Position Relative to Eq. X Time



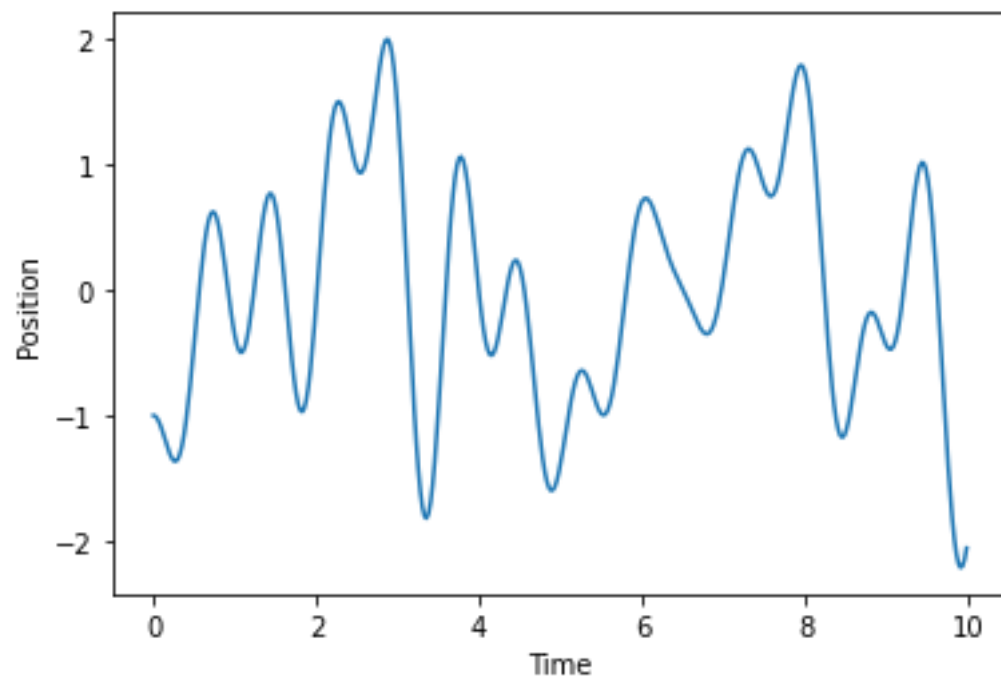
Massa 9: Position x Time

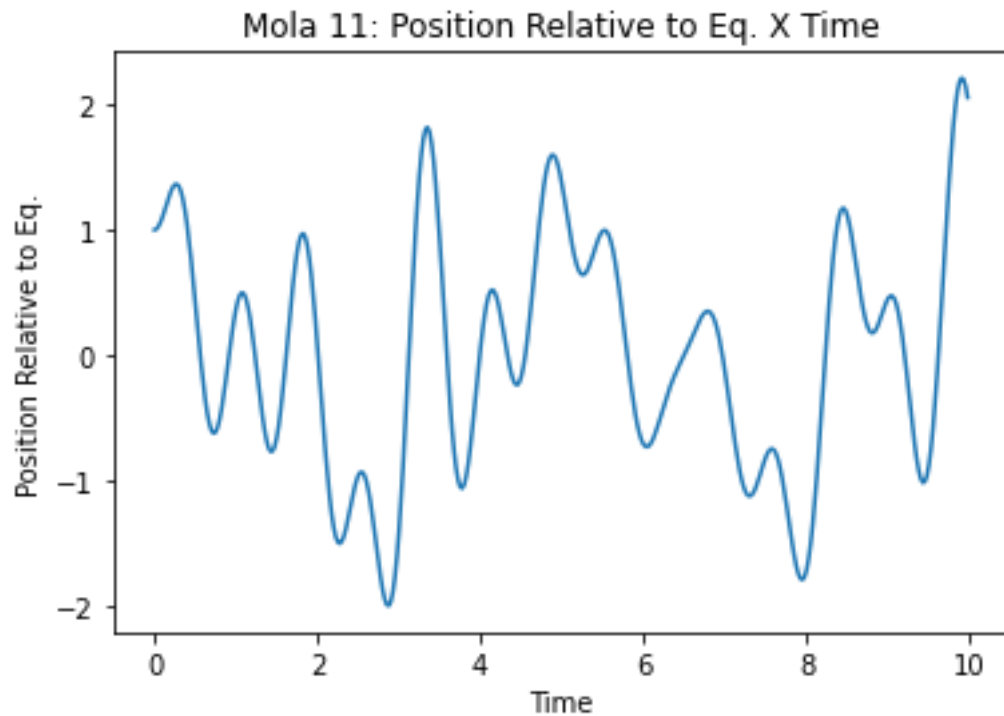


Mola 10: Position Relative to Eq. X Time



Massa 10: Position x Time



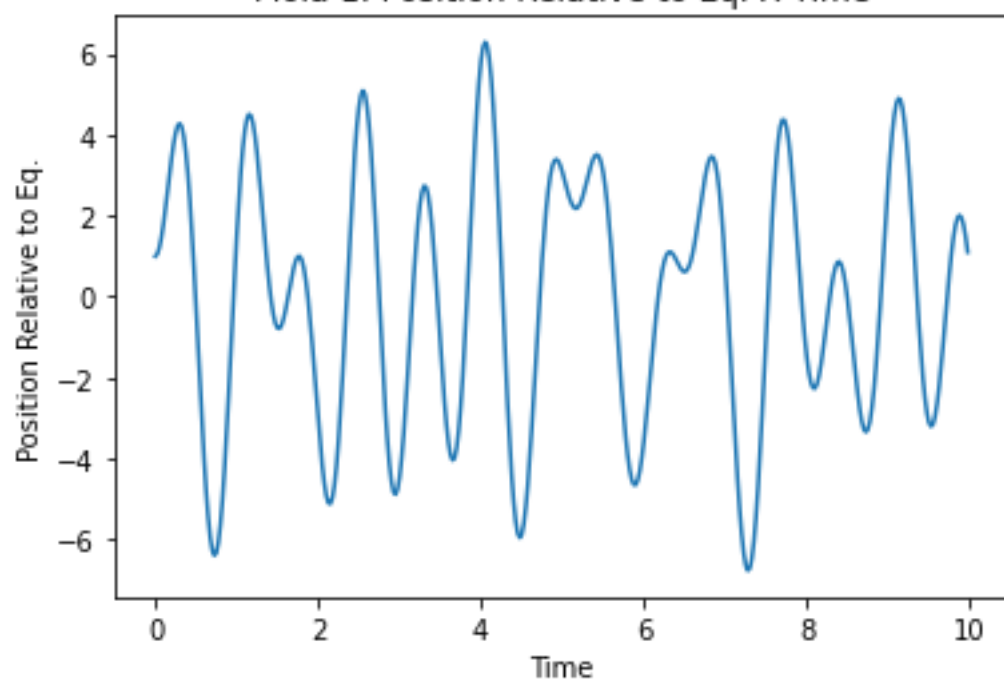


Comparando o caso 1 com 5 massas e o caso 1 com 10 massas podemos observar que ambos não possuem semelhança visível com algum modo de vibração. Além disso vemos que existem mais picos e mais vales dentro do mesmo intervalo de tempo.

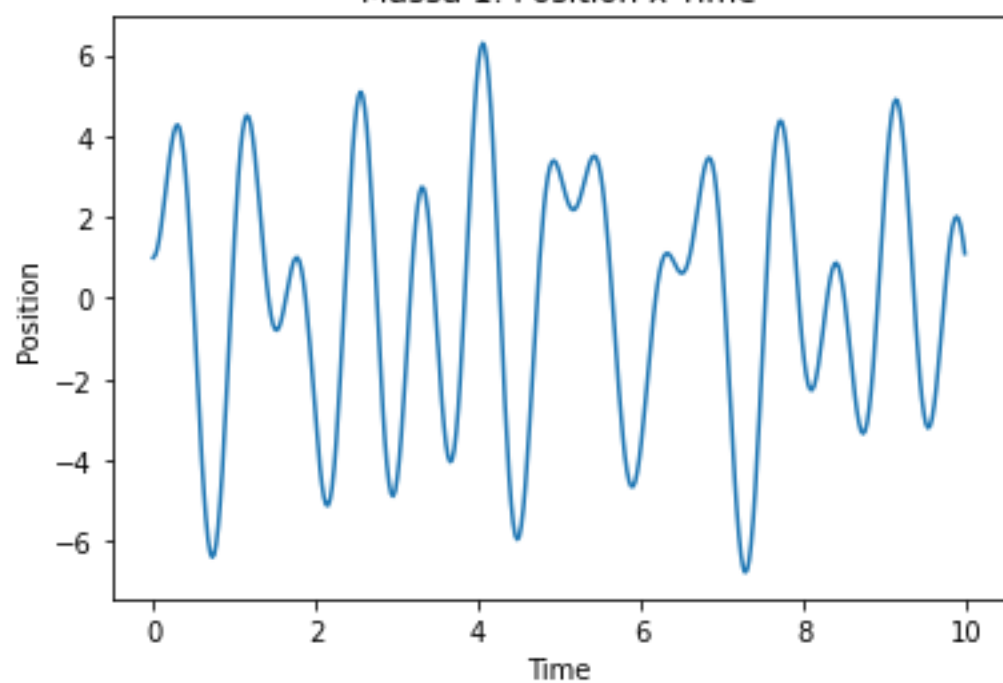
- Caso 2 $X(0) = 1, 10, -4, 3, -2, 1, 10, -4, 3, -2$:

Como a matriz A e V será a mesma para os próximos casos apresentaremos apenas os gráficos.

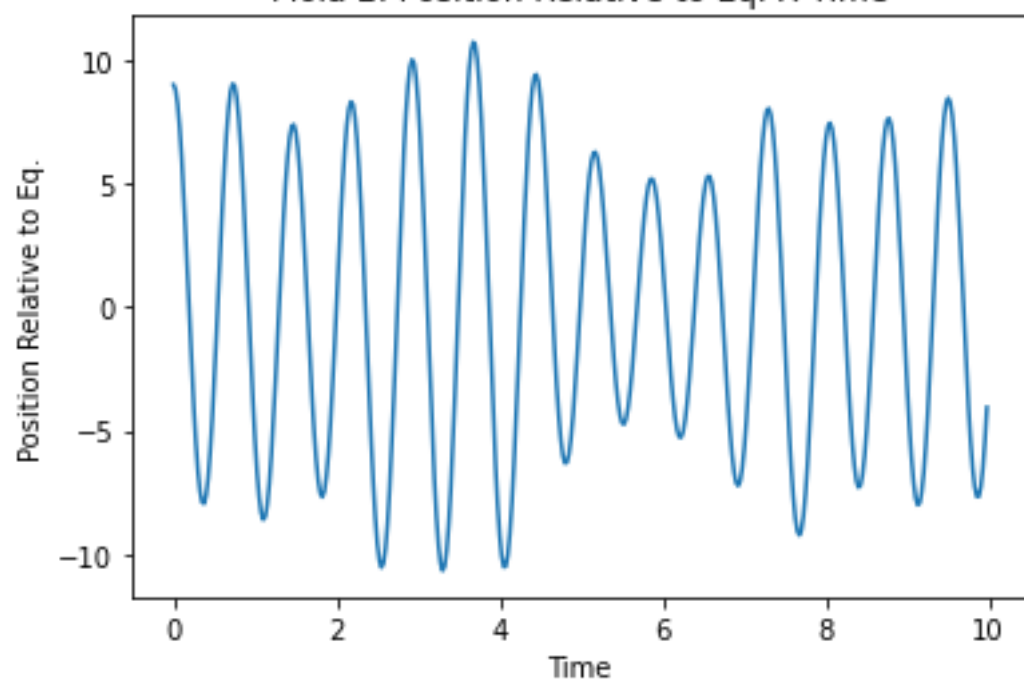
Mola 1: Position Relative to Eq. X Time



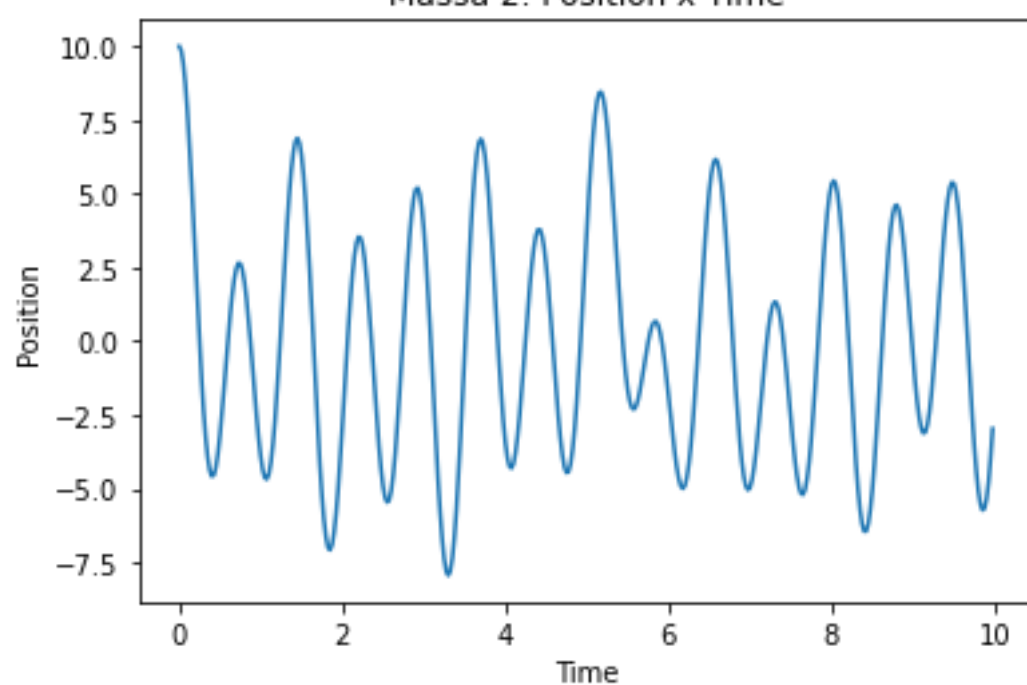
Massa 1: Position x Time

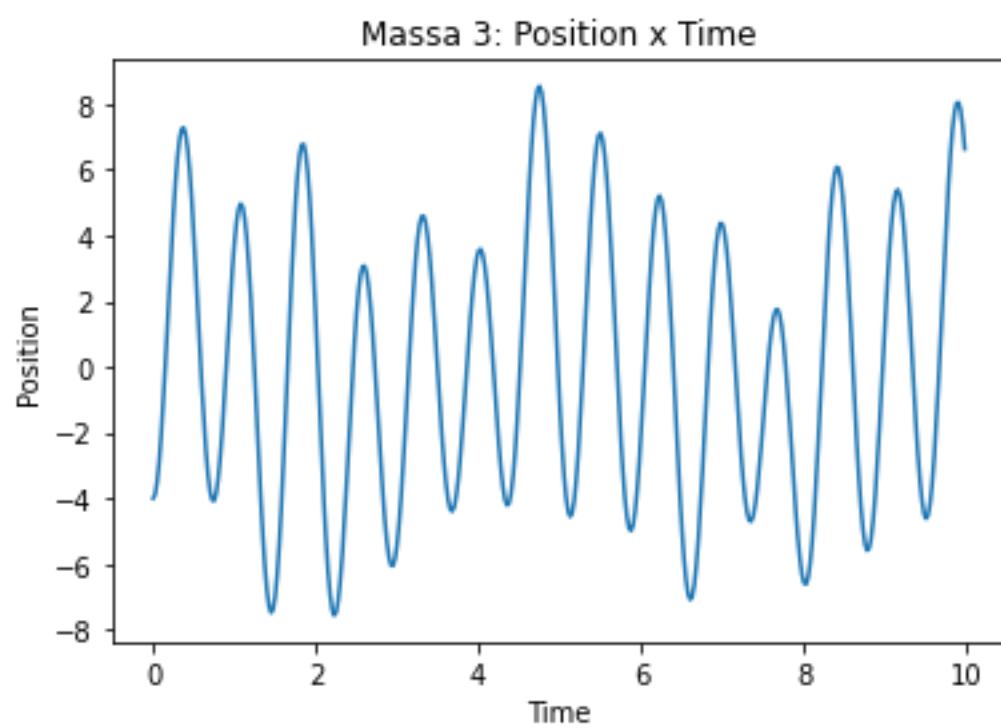
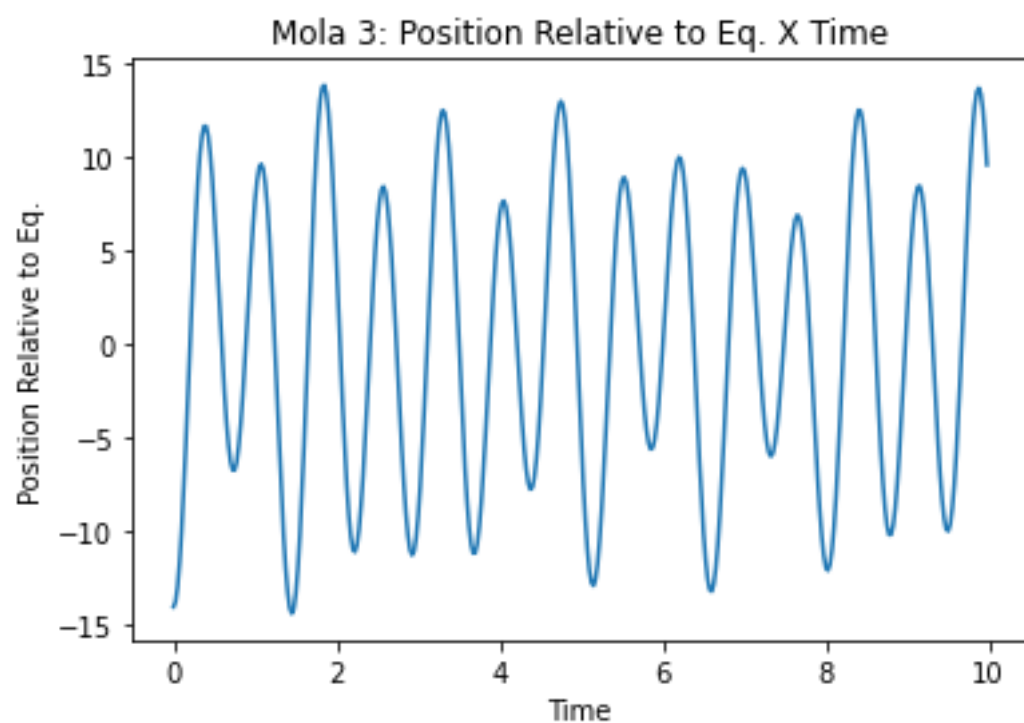


Mola 2: Position Relative to Eq. X Time

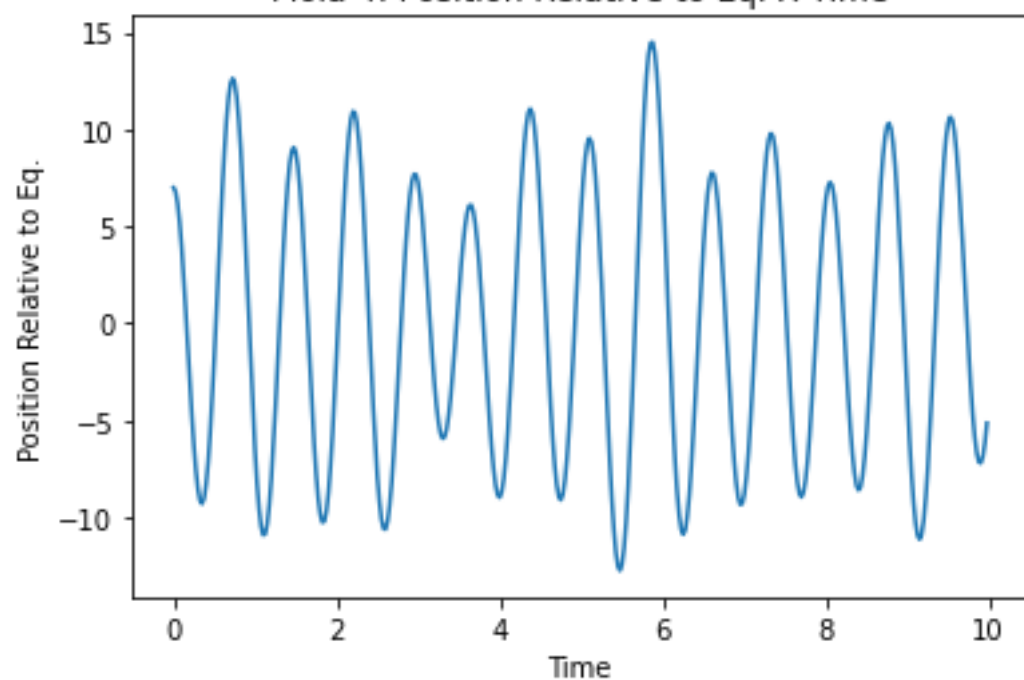


Massa 2: Position x Time

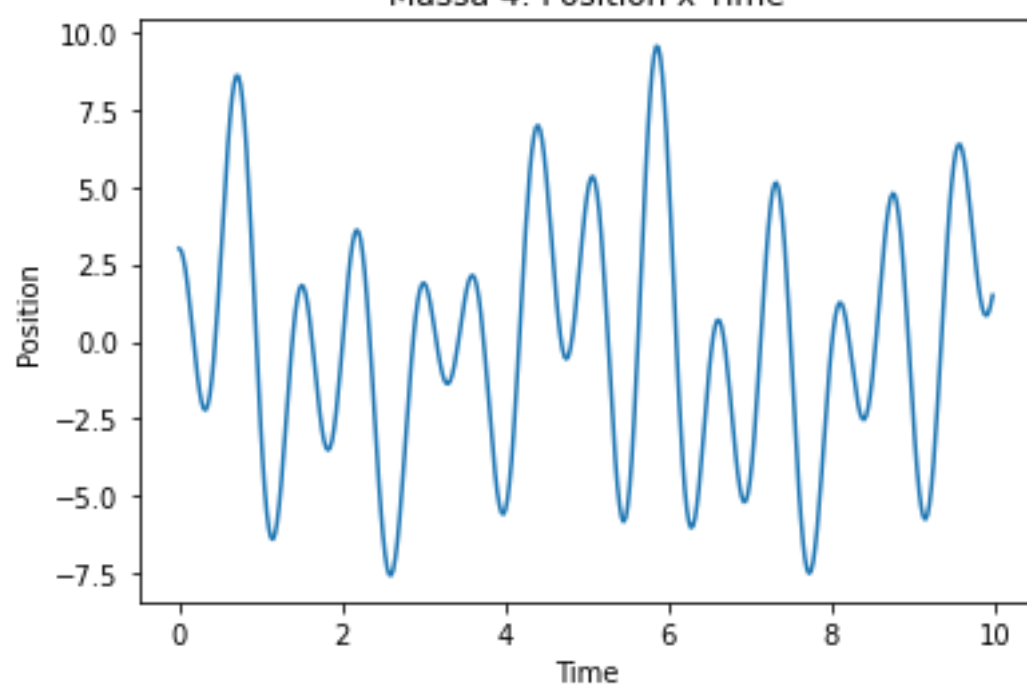




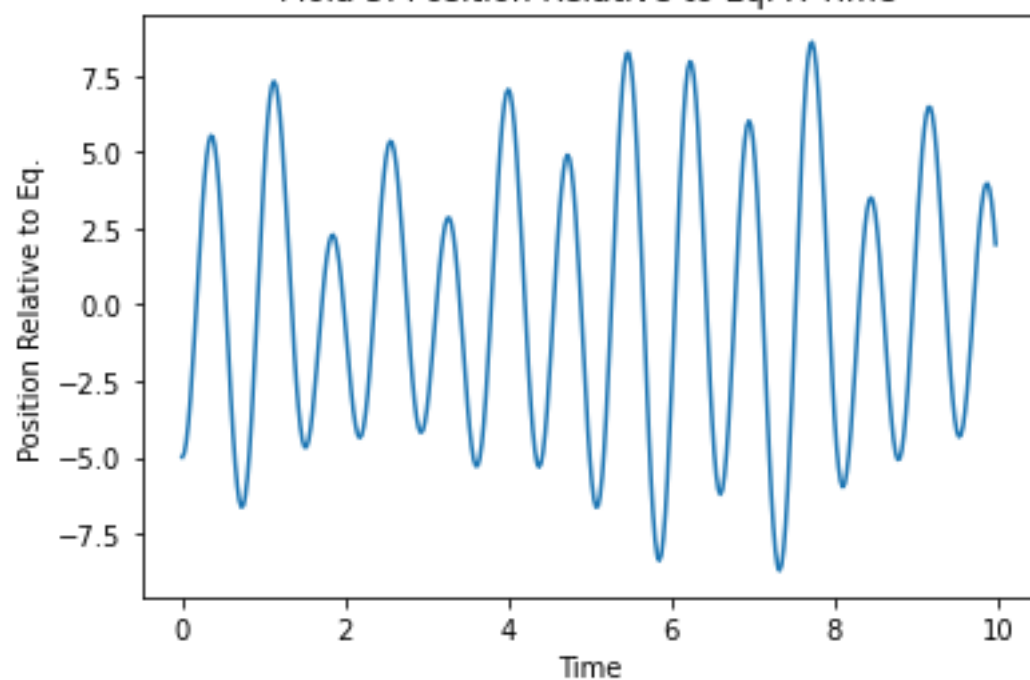
Mola 4: Position Relative to Eq. X Time



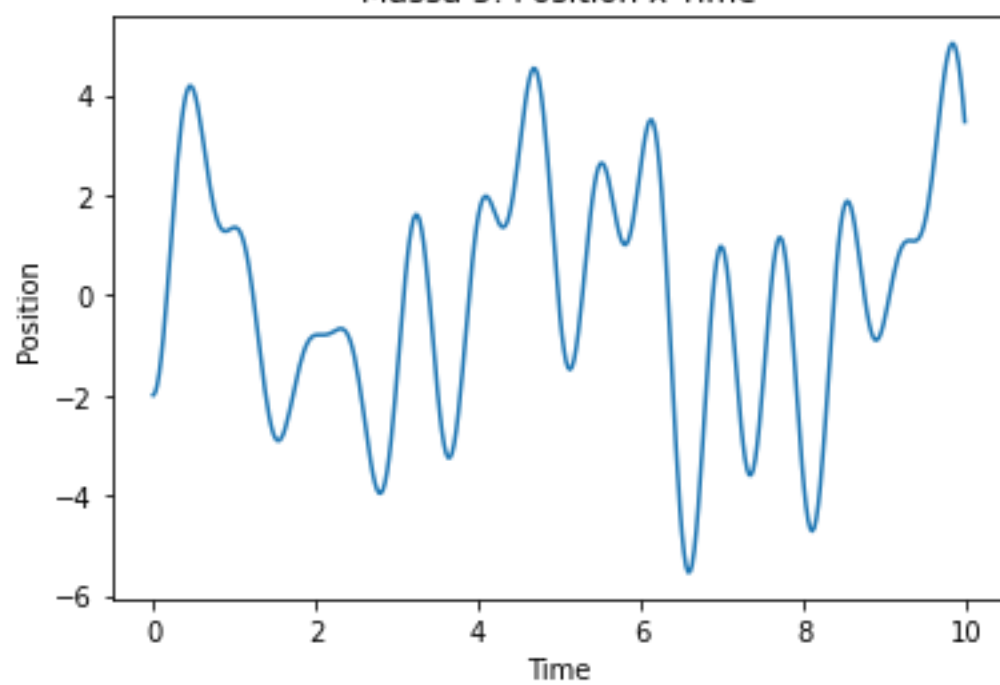
Massa 4: Position x Time



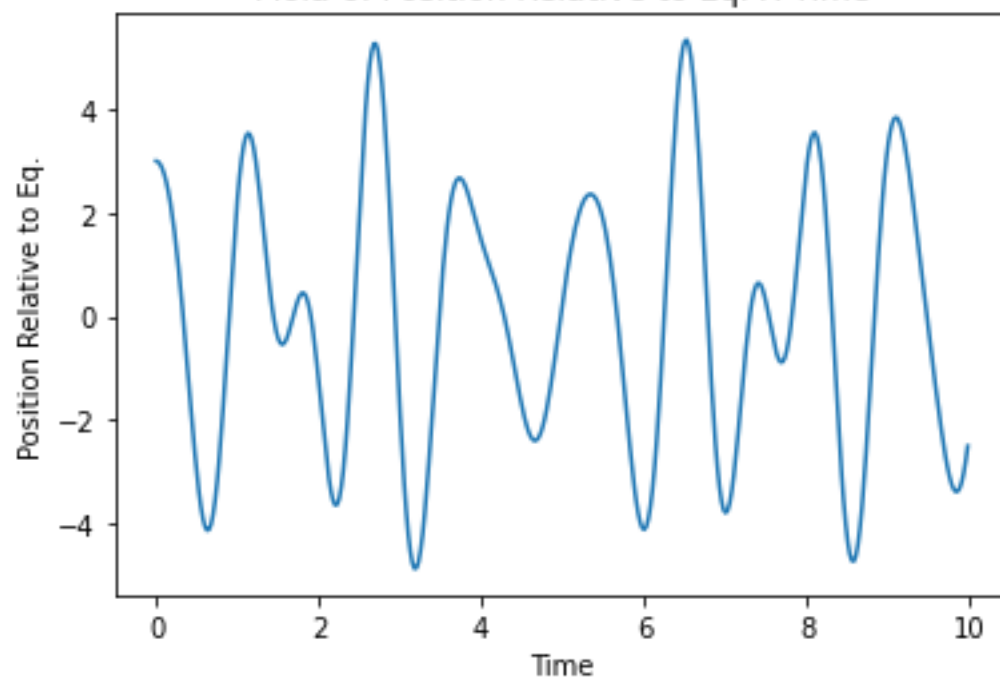
Mola 5: Position Relative to Eq. X Time



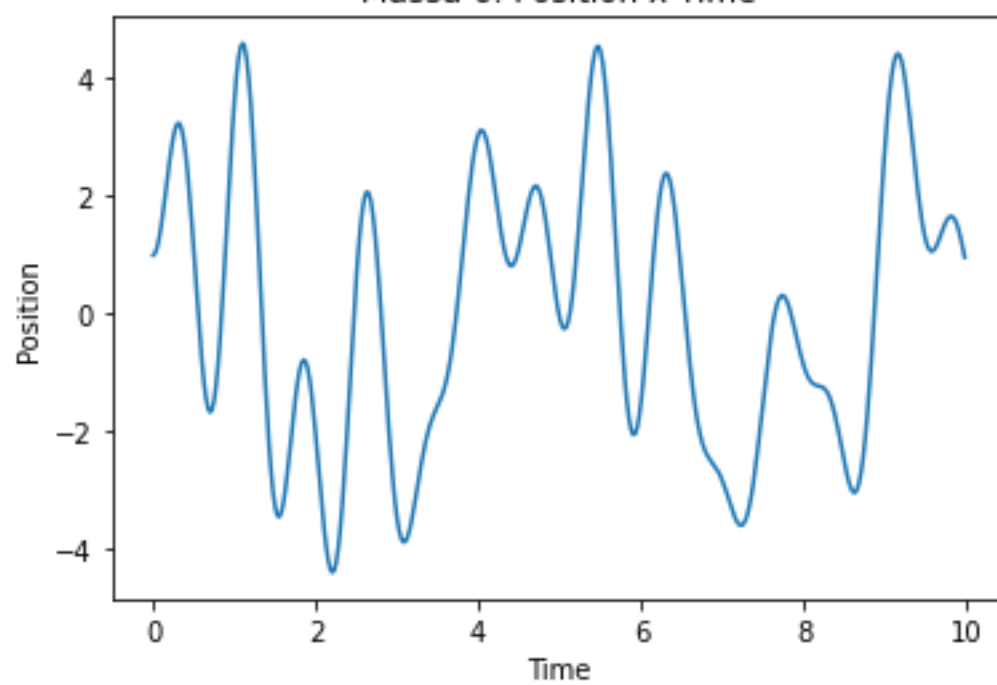
Massa 5: Position x Time



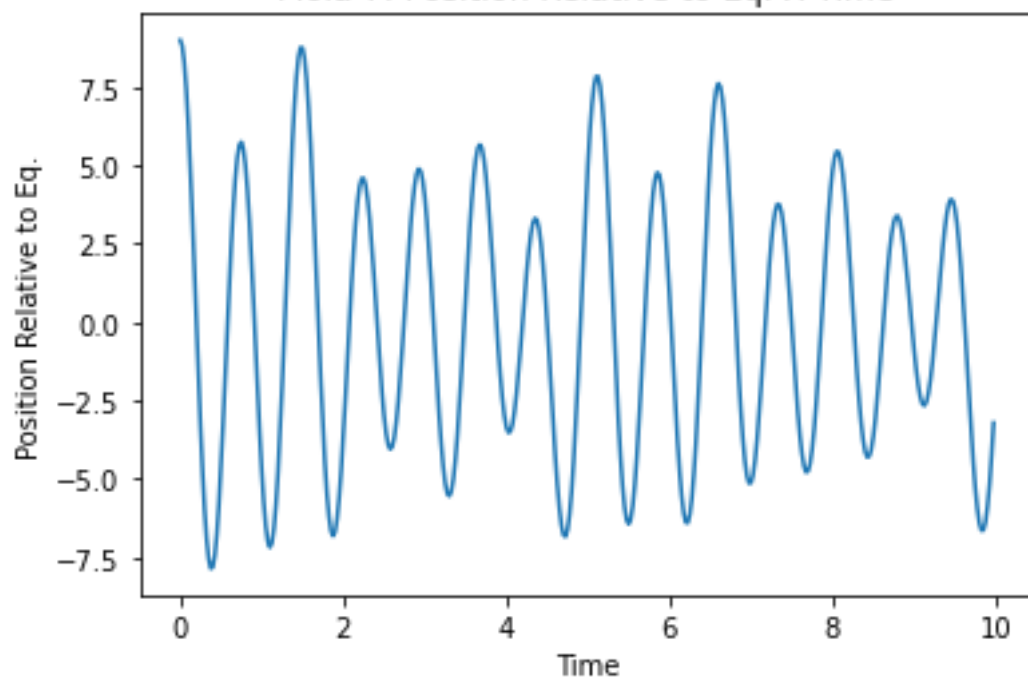
Mola 6: Position Relative to Eq. X Time



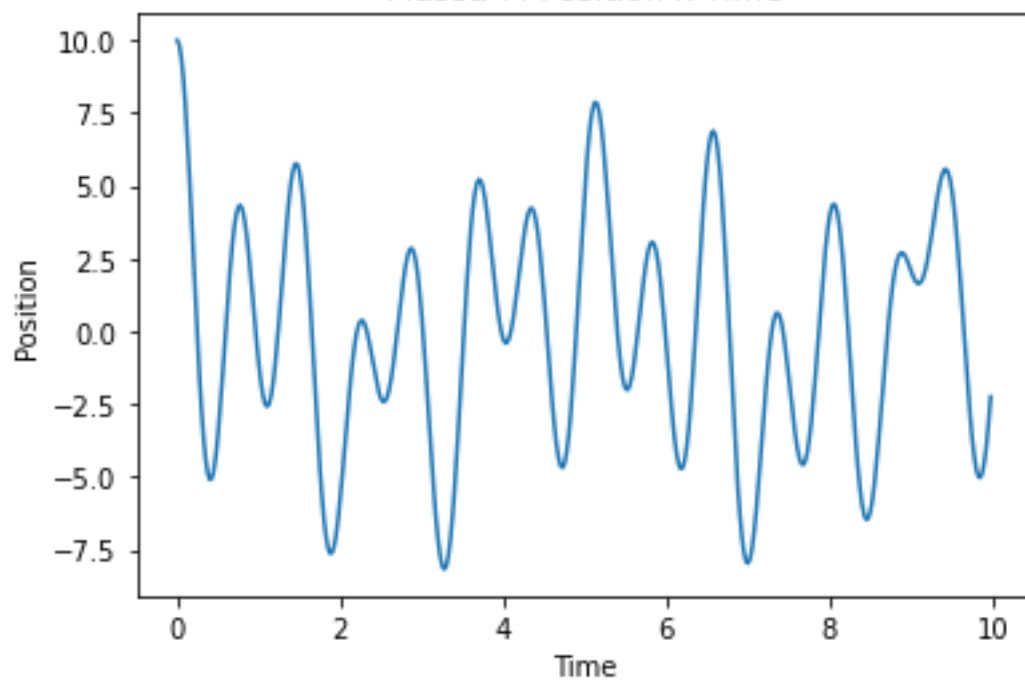
Massa 6: Position x Time



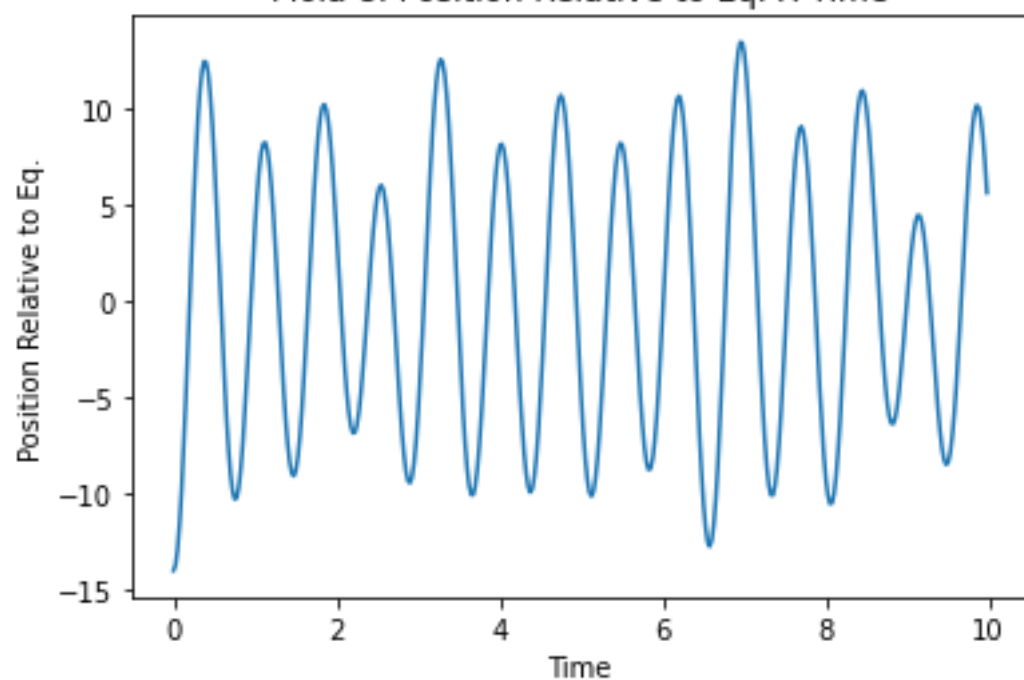
Mola 7: Position Relative to Eq. X Time



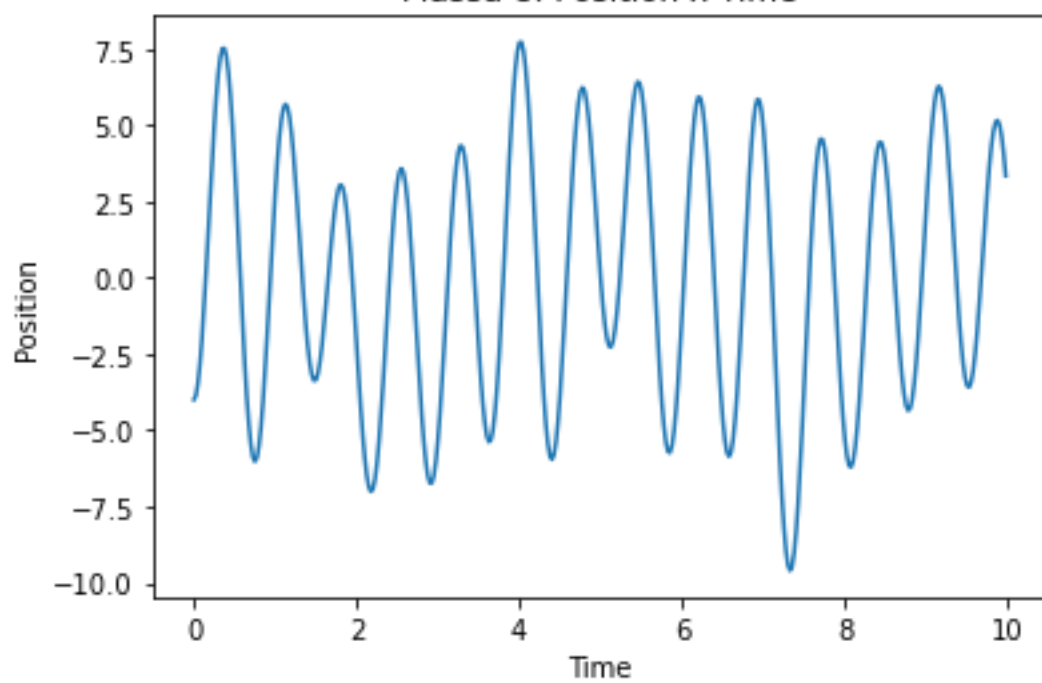
Massa 7: Position x Time



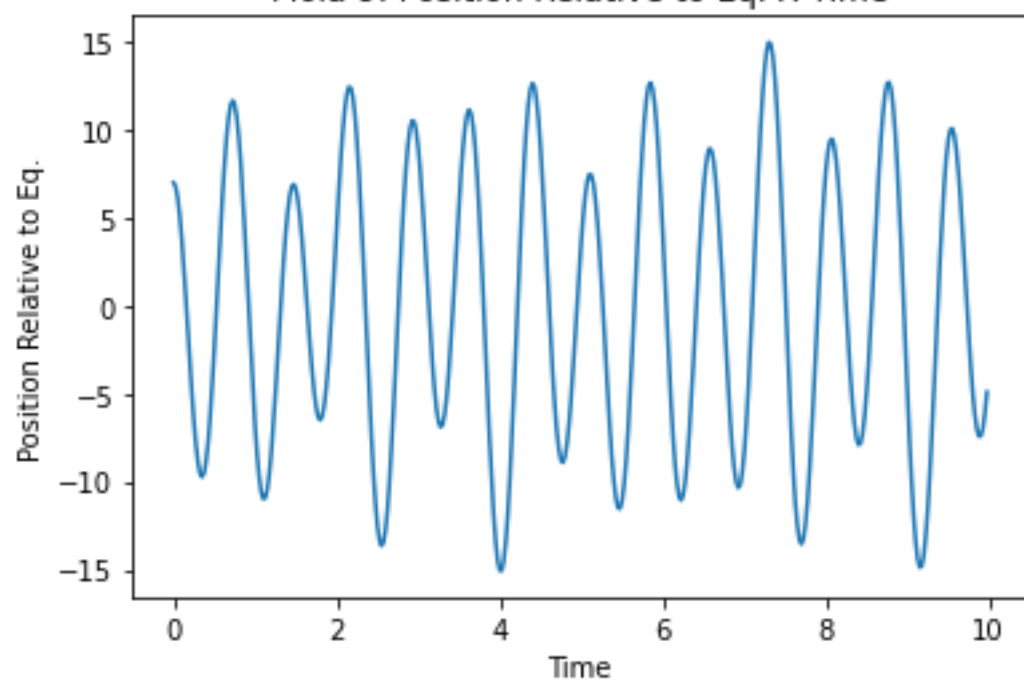
Mola 8: Position Relative to Eq. X Time



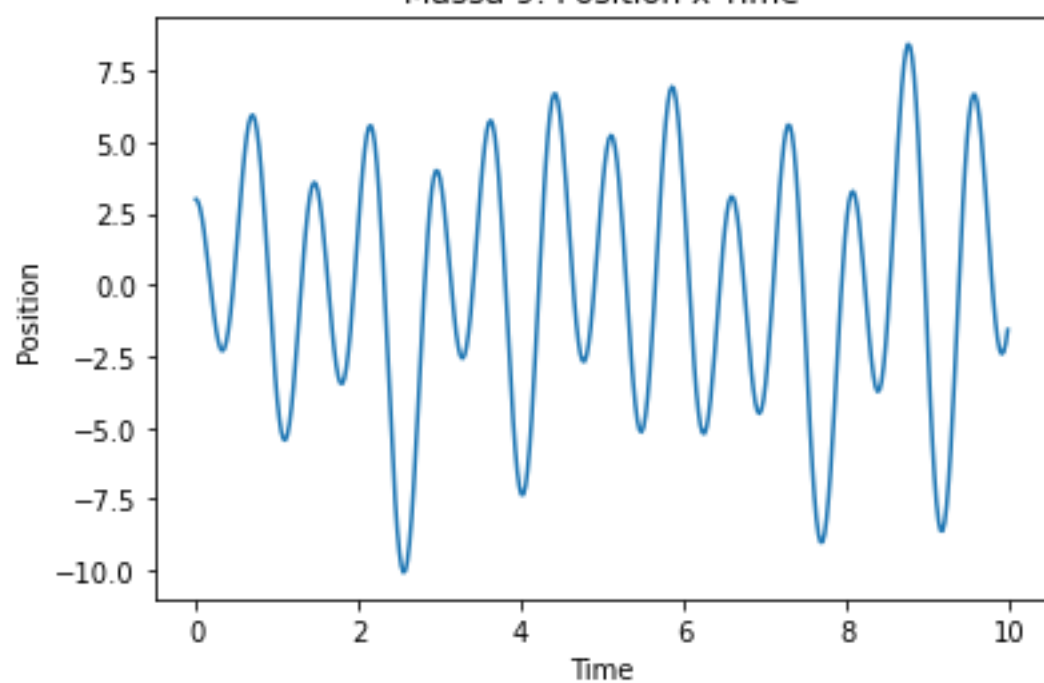
Massa 8: Position x Time



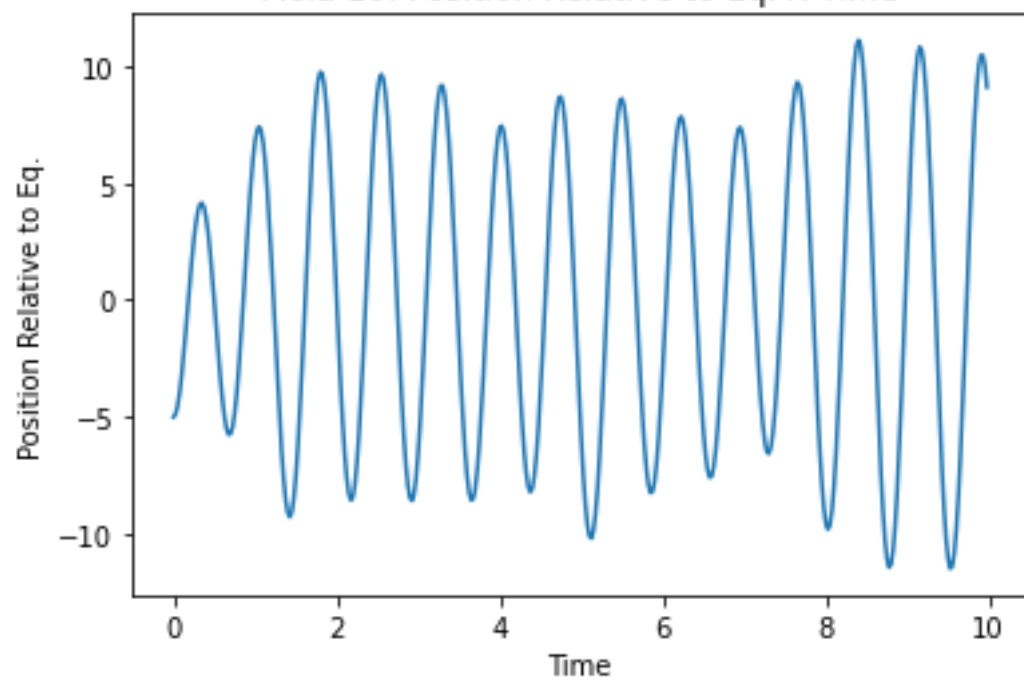
Mola 9: Position Relative to Eq. X Time



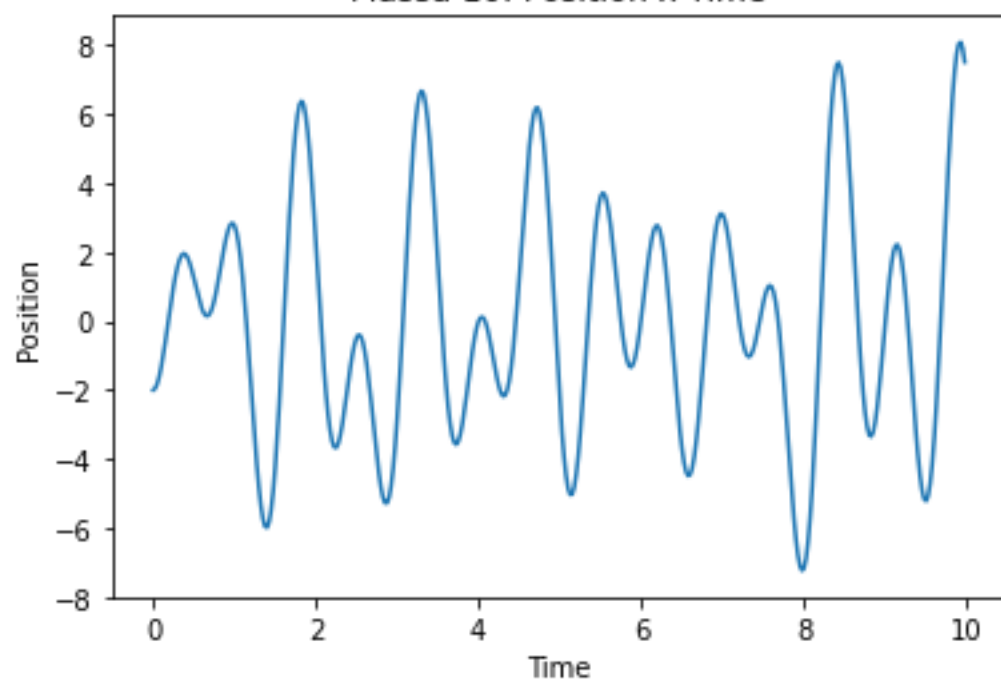
Massa 9: Position x Time

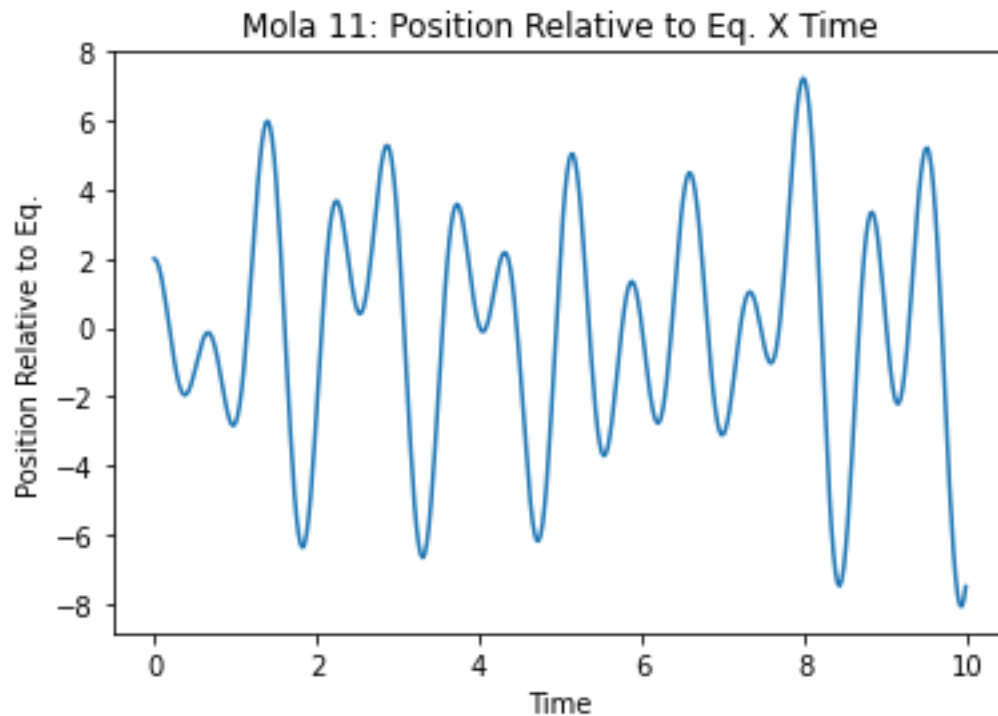


Mola 10: Position Relative to Eq. X Time



Massa 10: Position x Time





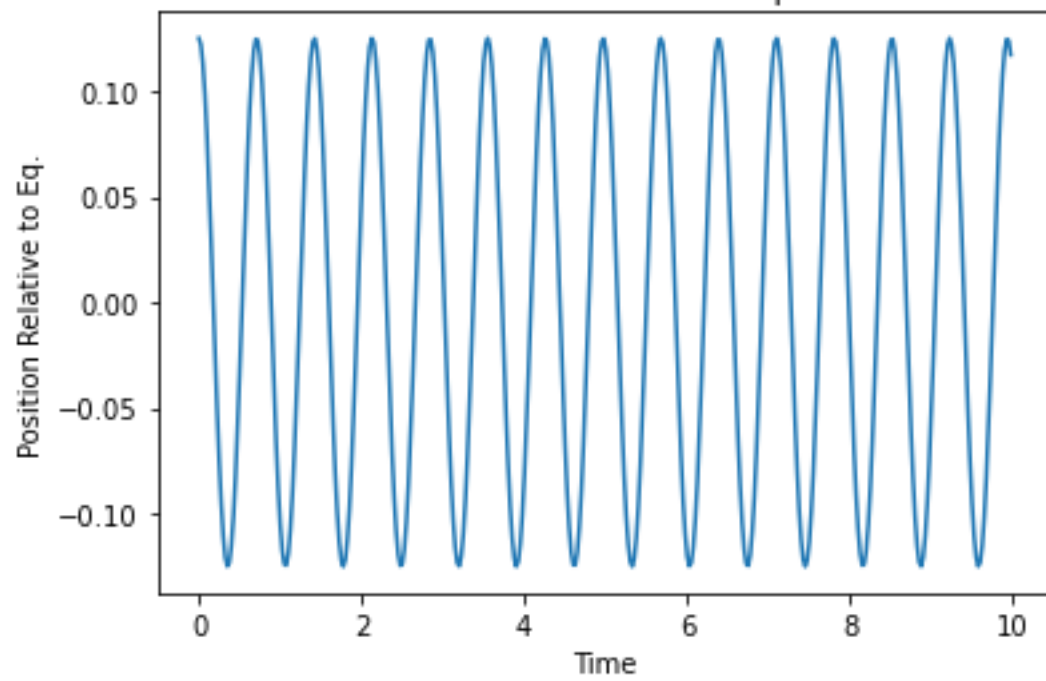
Como no caso 2 para 5 massas, o caso para 10 massas mostra que possui uma semelhança com um modo de vibração, isto é, as posições iniciais estão mais próximas de um autovetor do sistema.

- Caso 3 correspondente ao modo de maior frequência com 10 massas:

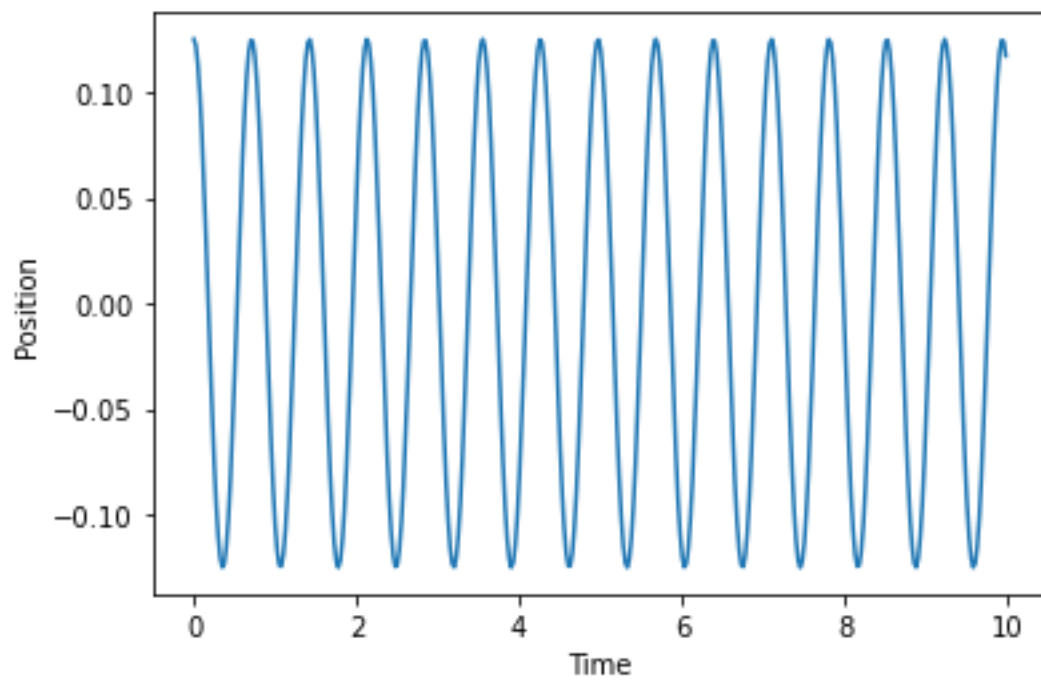
Maior Autovalor e sua frequência: $\lambda=78.39874797945029$ $f=1.409206686522176$

Posições iniciais (Autovetor correspondente): [0.12524522 -0.22901235 0
.32440201 -0.38597125 0.42149319 -0.42149319
0.38597125 -0.32440201 0.22901235 -0.12524522]

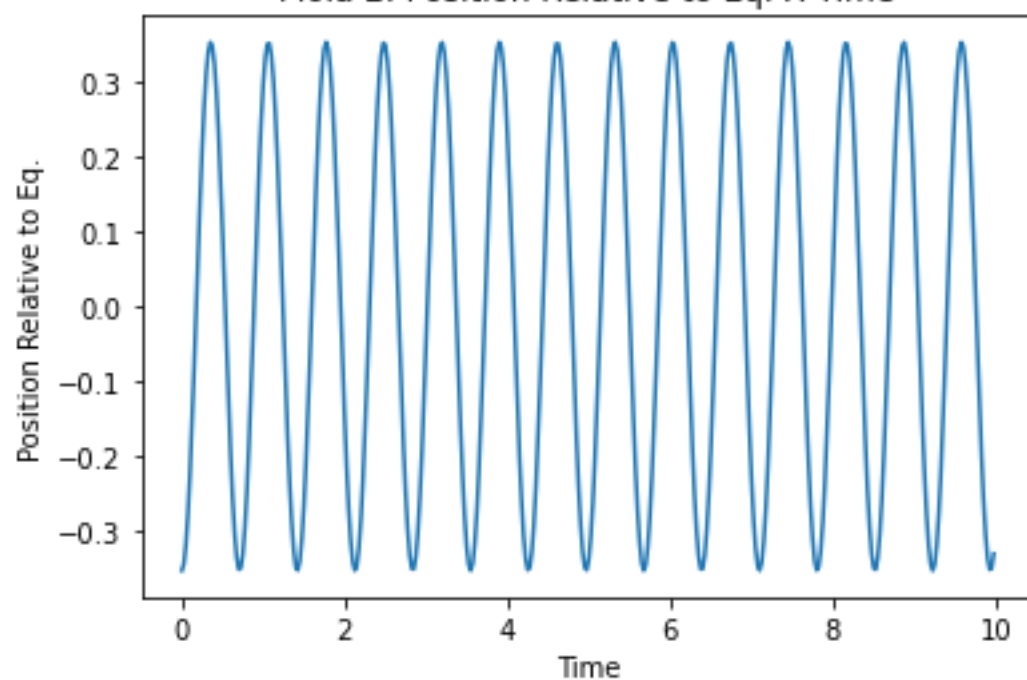
Mola 1: Position Relative to Eq. X Time



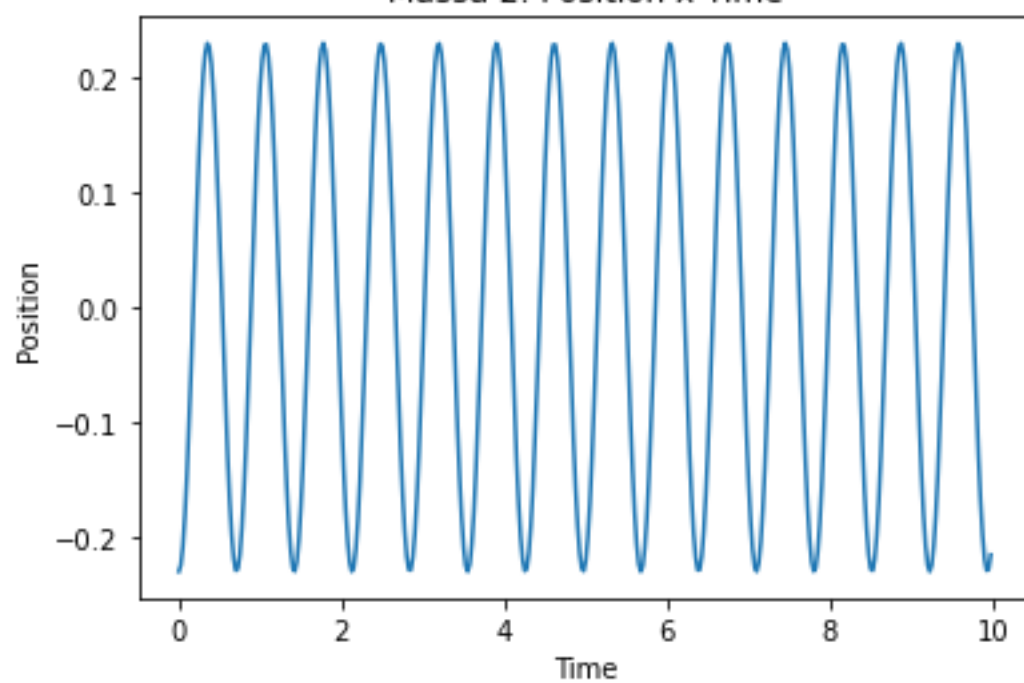
Massa 1: Position x Time

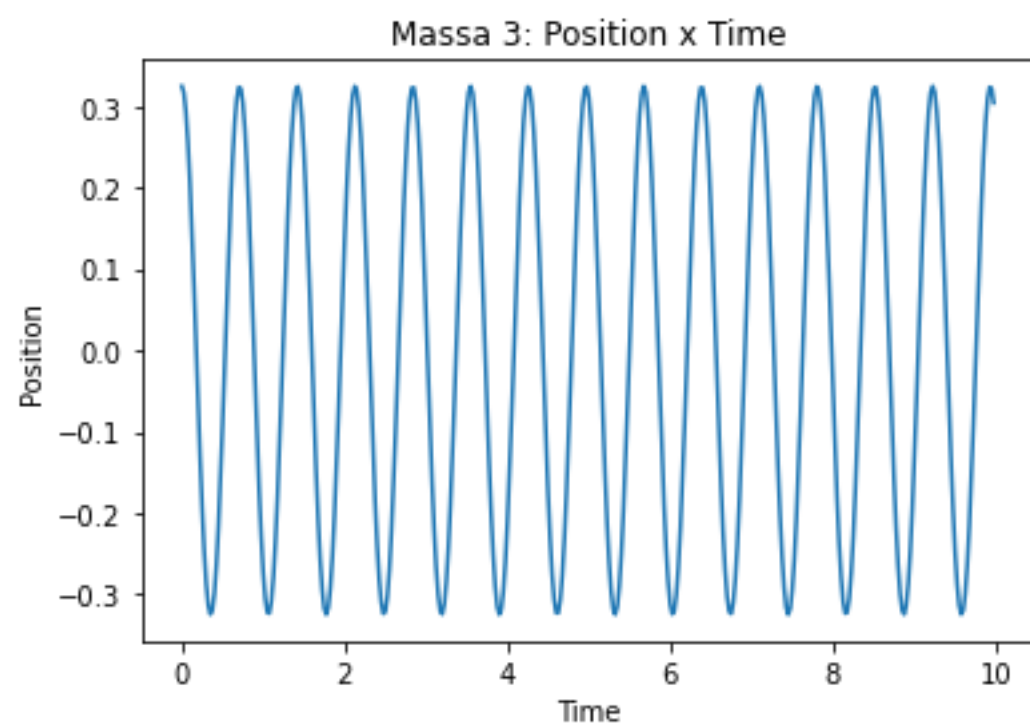
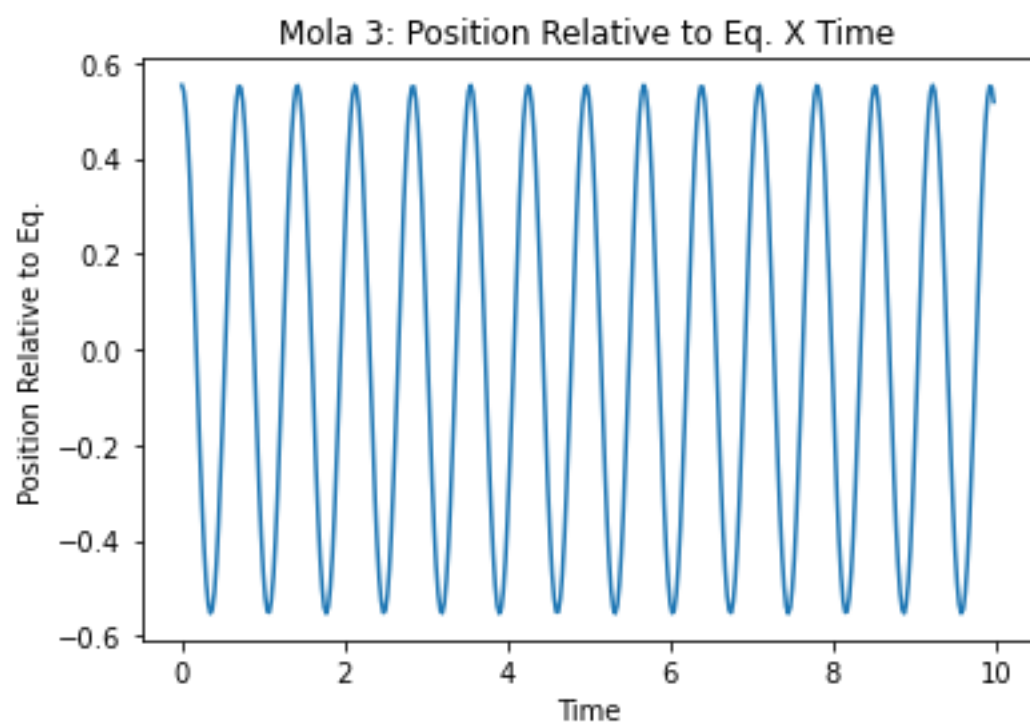


Mola 2: Position Relative to Eq. X Time

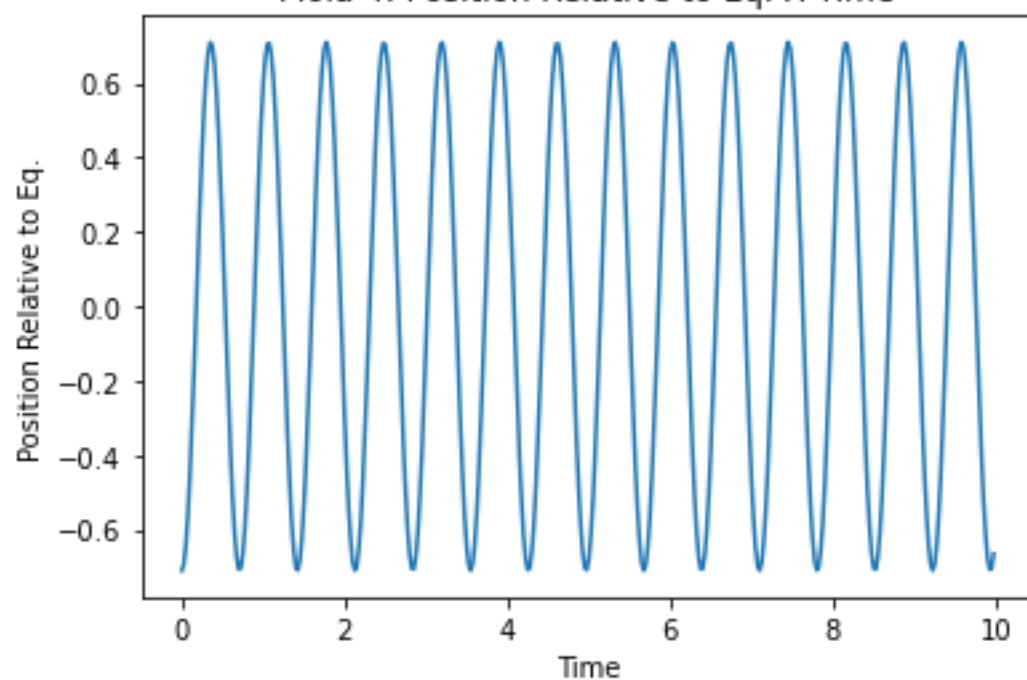


Massa 2: Position x Time

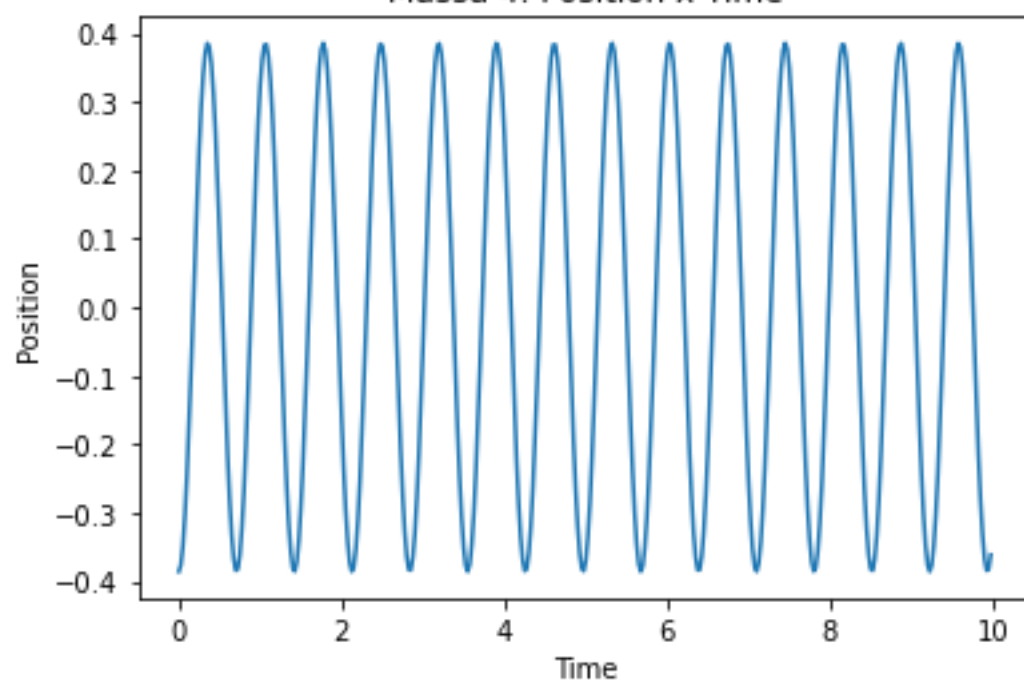




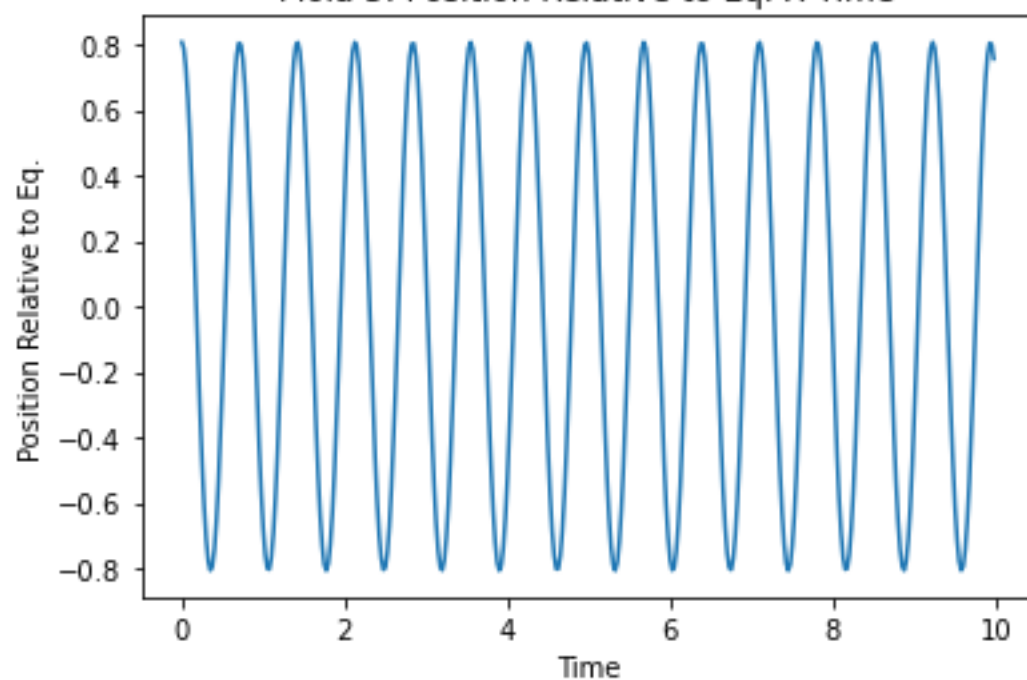
Mola 4: Position Relative to Eq. X Time



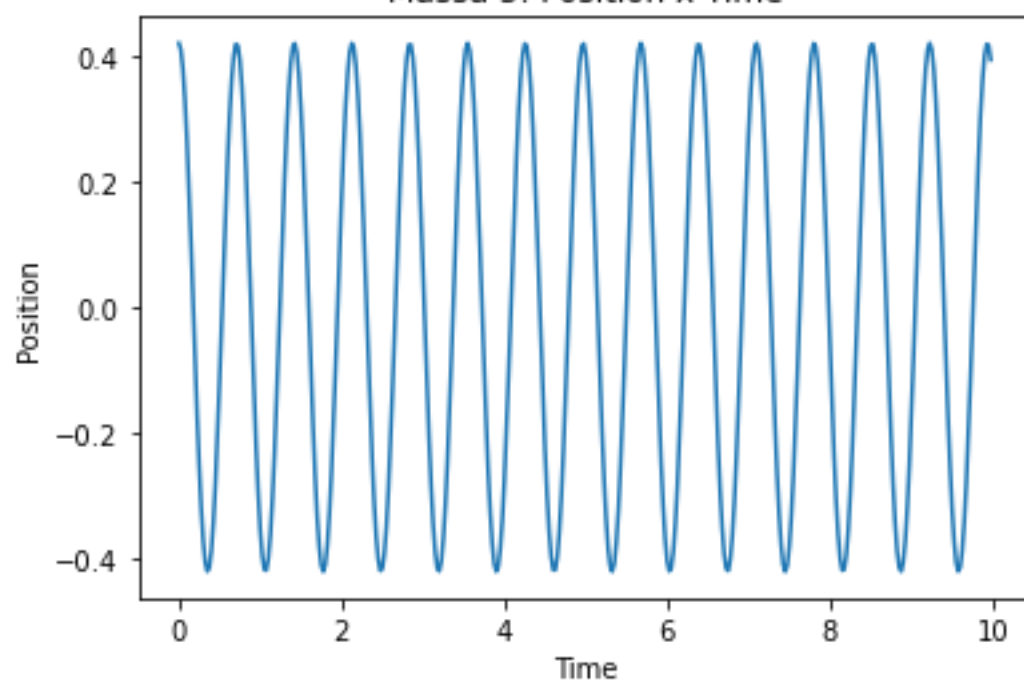
Massa 4: Position x Time



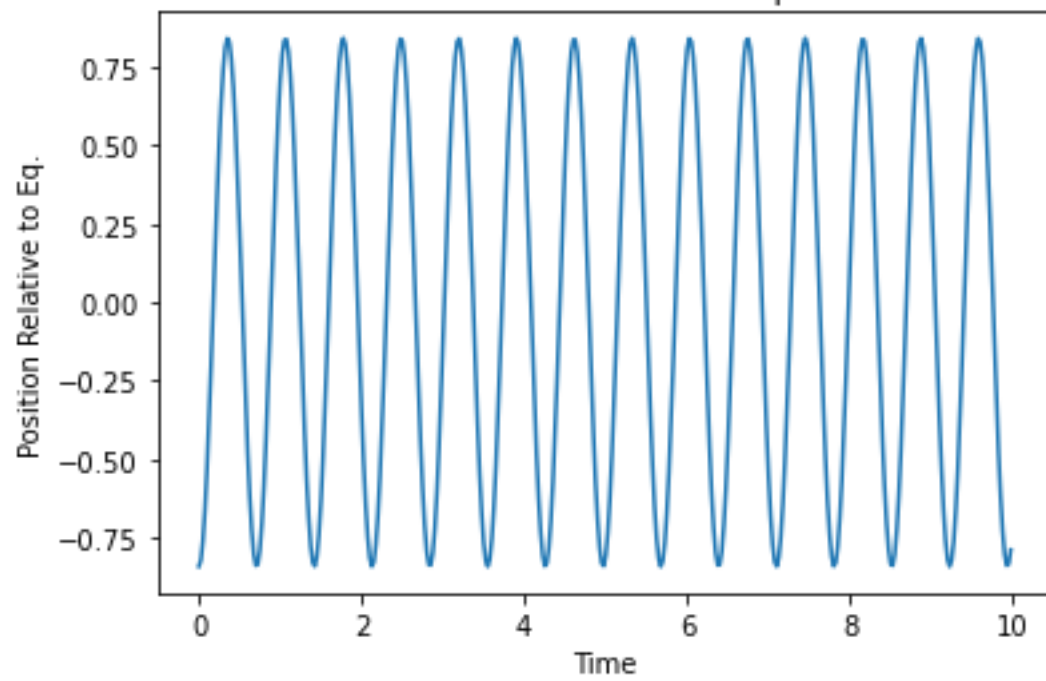
Mola 5: Position Relative to Eq. X Time



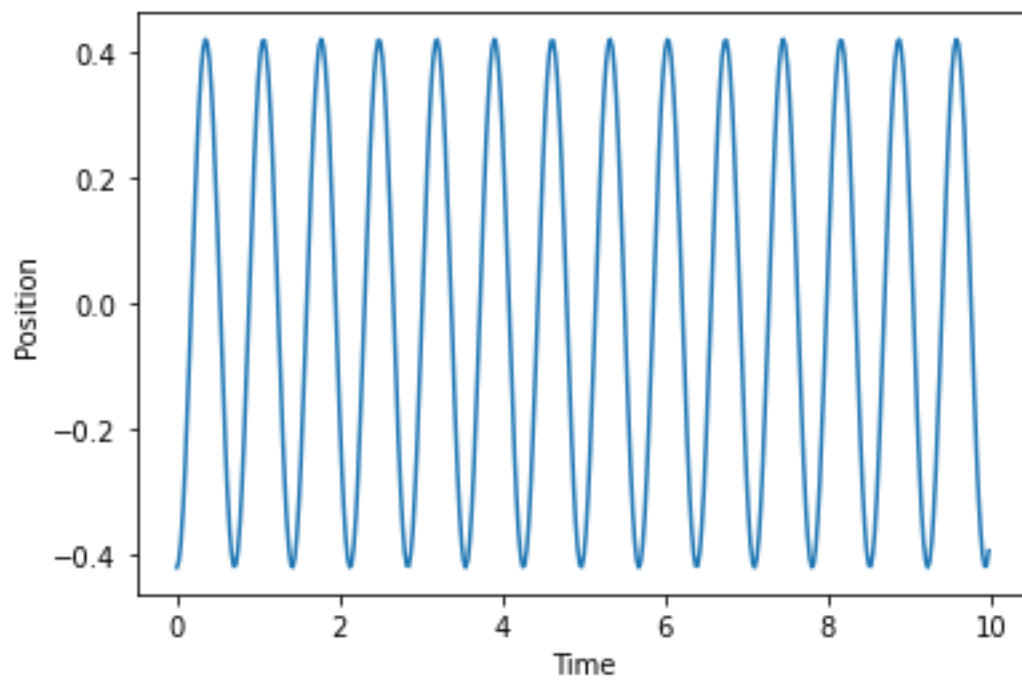
Massa 5: Position x Time



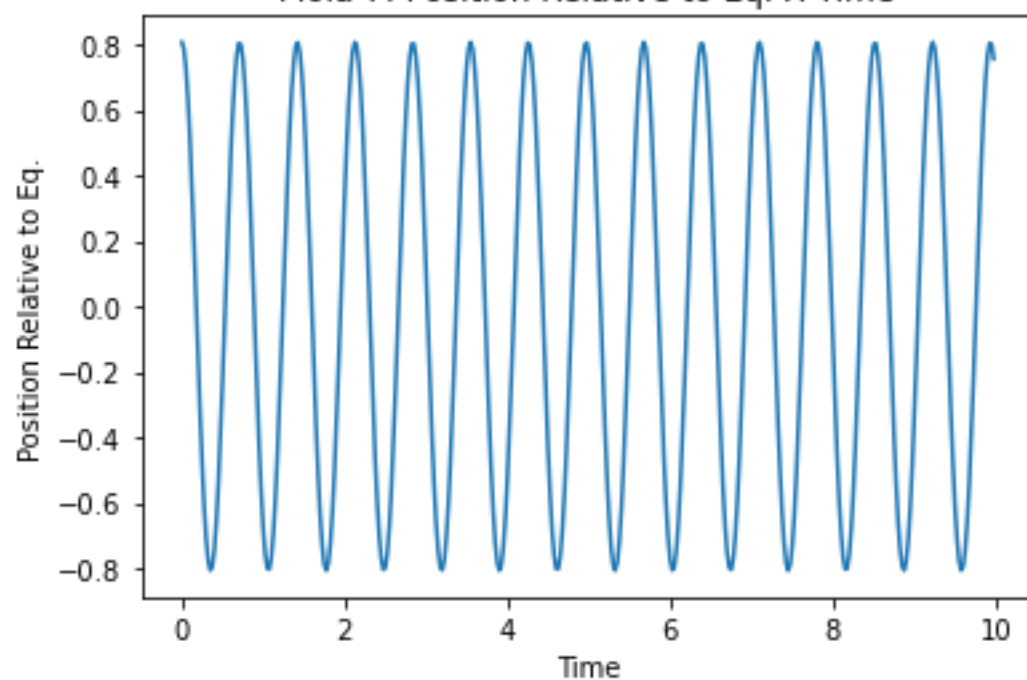
Mola 6: Position Relative to Eq. X Time



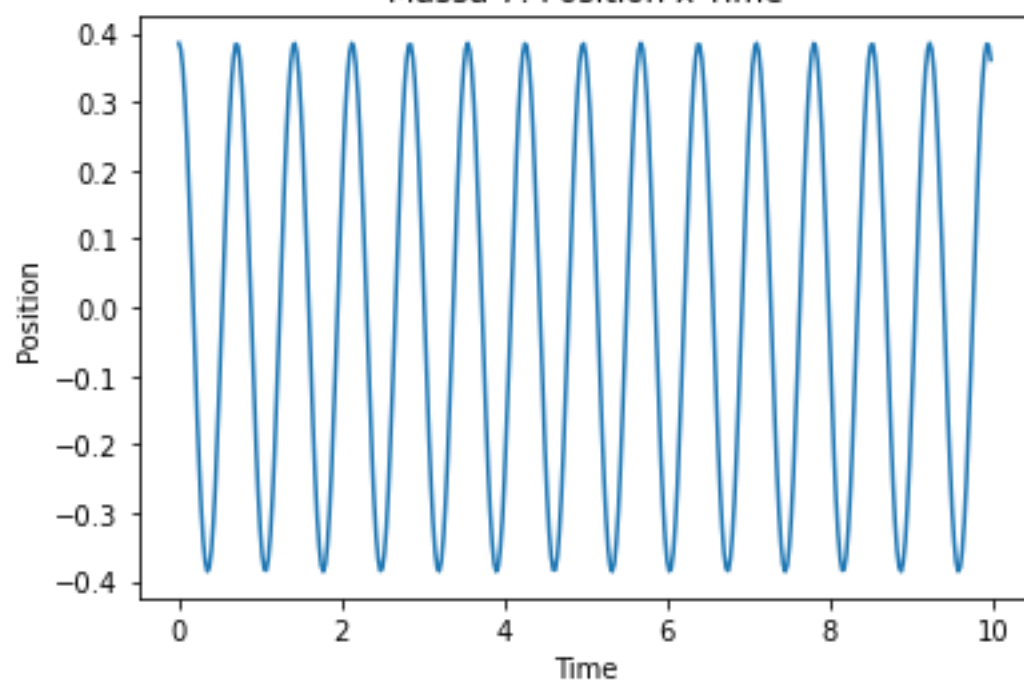
Massa 6: Position x Time



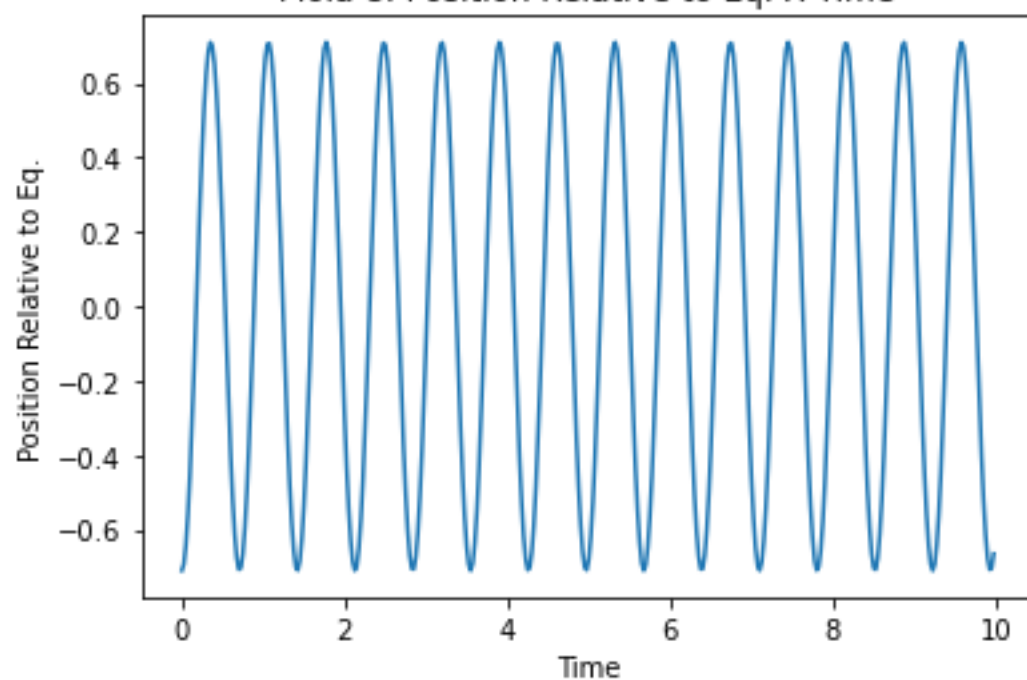
Mola 7: Position Relative to Eq. X Time



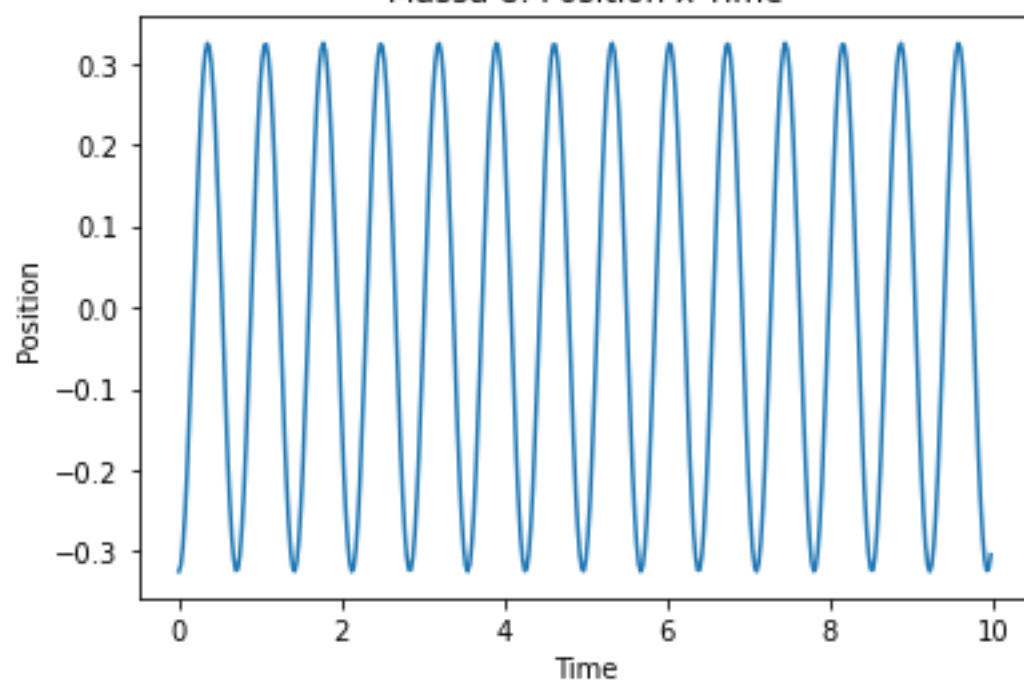
Massa 7: Position x Time

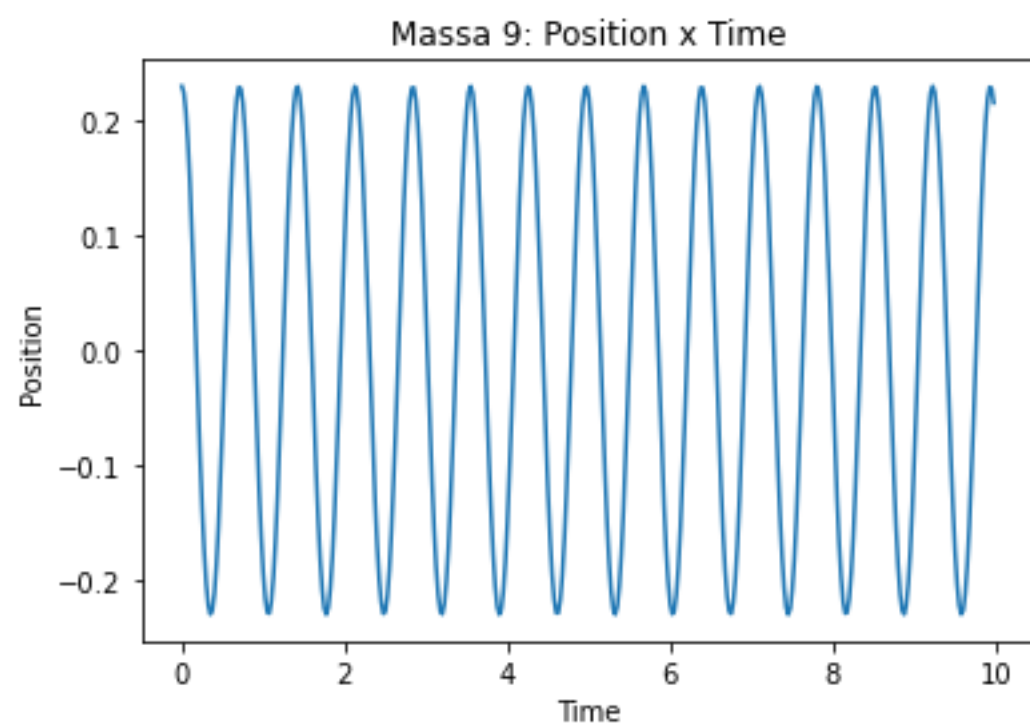
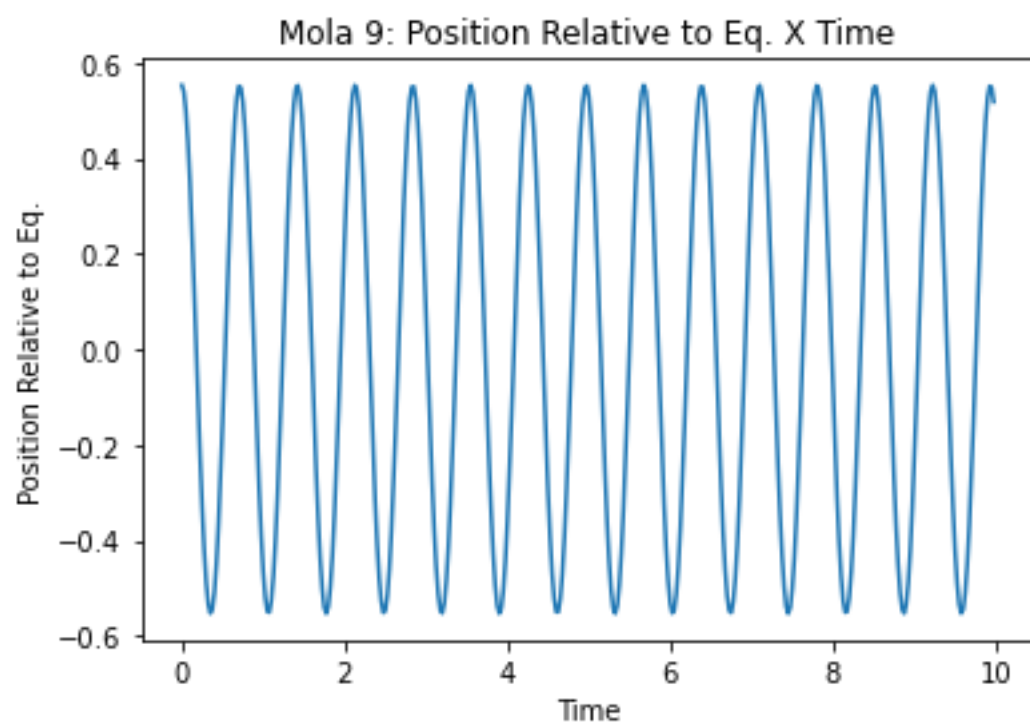


Mola 8: Position Relative to Eq. X Time

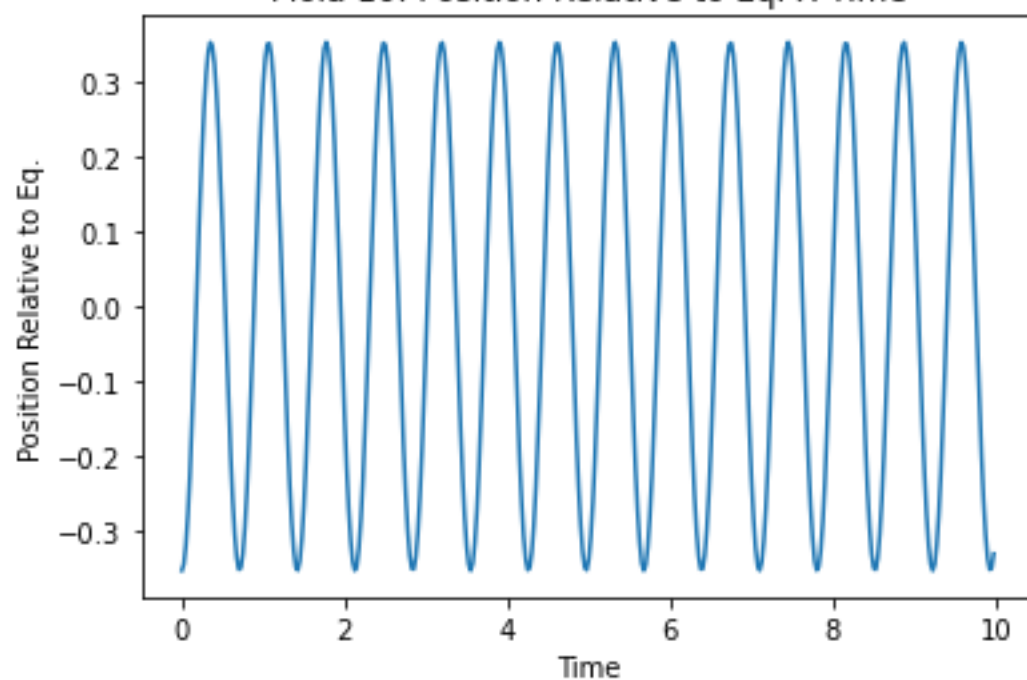


Massa 8: Position x Time

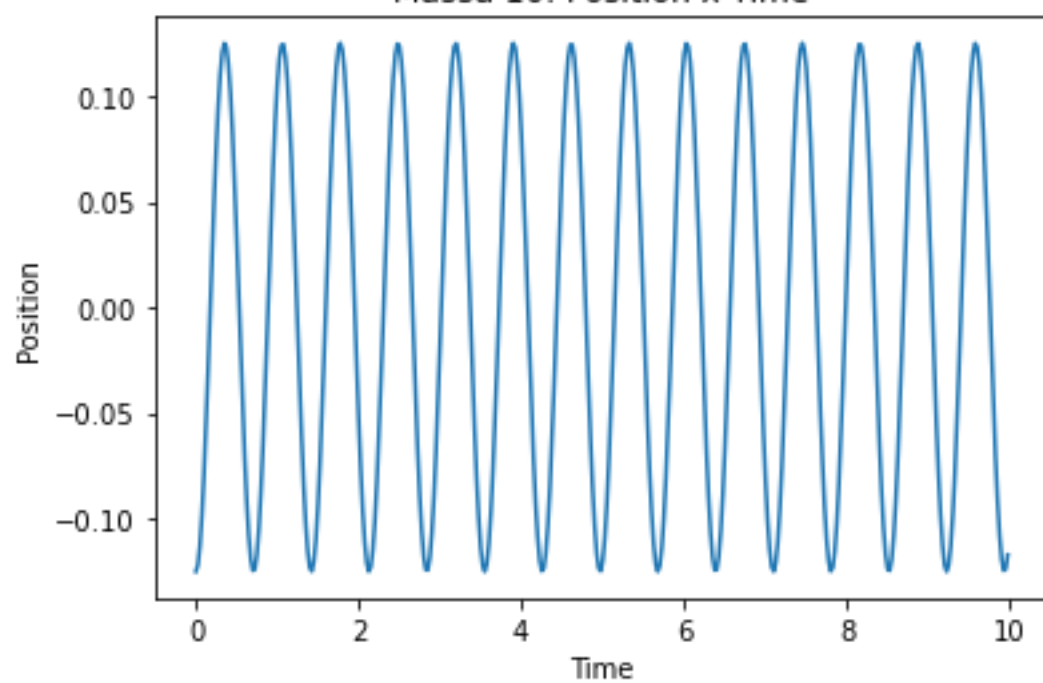


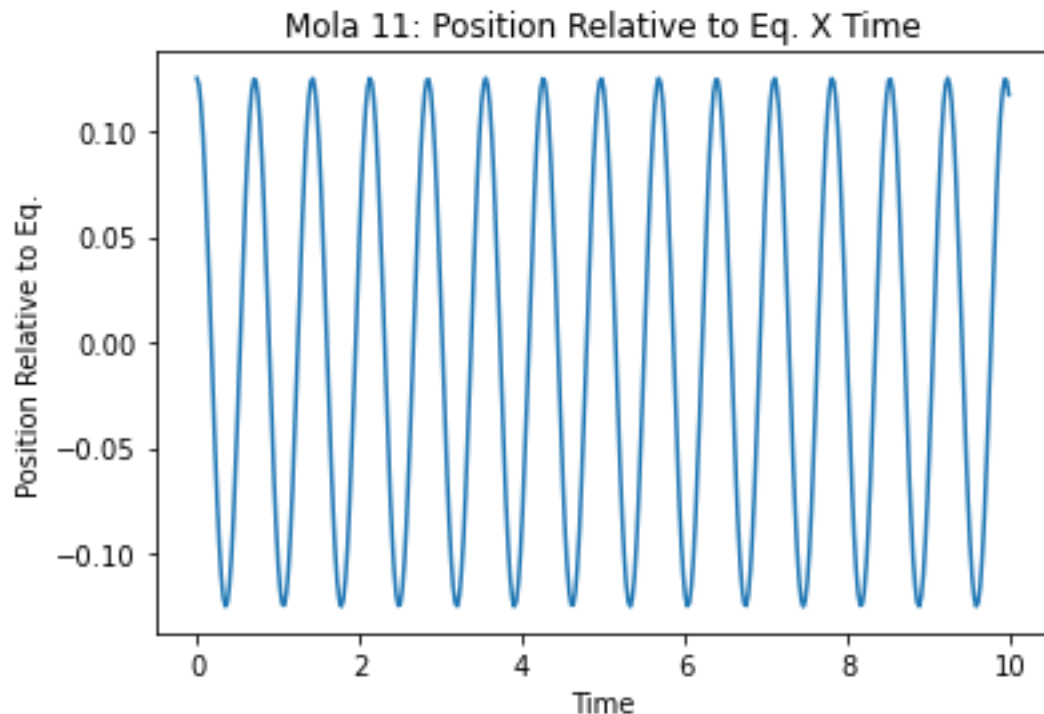


Mola 10: Position Relative to Eq. X Time



Massa 10: Position x Time





Os outros modos de vibração e suas respectivas frequências são:

$f=1.366601950586915$ Autovetor=[-0.2400737895682307, 0.38560341021292466, -0.4192013498005402, 0.3244365006284553, -0.11263112793486961, -0.1126311279348732, 0.32443650062845714, -0.4192013498005394, 0.38560341021292055, -0.24007378956822642]

$f=1.296832120623628$ Autovetor=[-0.334550765579963, 0.42047869685748596, -0.2143393433504271, -0.11104164643532921, 0.3911543765951952, -0.39115443512205955, 0.11104176368747465, 0.21433930102901935, -0.42047878844189895, 0.3345508782901273]

$f=1.2022673411036113$ Autovetor=[-0.39821223489790664, 0.3235749767109901, 0.1495257298647322, -0.41425828841691537, 0.20678120653843765, 0.20678108112129076, -0.41425826782169073, 0.14952580744771862, 0.32357486343953495, -0.3982121574258975]

$f=1.0911272459974262$ Autovetor=[-0.39519756328213546, 0.1317582639634498, 0.3882452373374126, -0.2486502244596614, -0.3374871572513861, 0.33748694773842386, 0.24865036741476992, -0.388244979429256, -0.13175833517791988, 0.39519729003926124]

$f=0.9142565317205804$ Autovetor=[-0.3952049897857414, -0.13176459179372874, 0.388247870580267, 0.24865837745160516, -0.3374829493660978, -0.3374911555116782, 0.24864221432294237, 0.3882423460334092, -0.13175200730456874, -0.3951898633868716]

$f=0.7622184158477692$ Autovetor=[-0.3982044847050731, -0.32357227658700005, 0.1495183202755275, 0.41425344954351007, 0.20678757037237966, -0.2067747172094999, -0.4142631065381928, -0.1495332169839269, 0.3235775634427793, 0.39821990747196134]

```
f=0.5870690962446335 Autovetor=[-0.3345507416652106, -0.42047853962238046, -0.21433898394962772, 0.11104203678224656, 0.39115453564136343, 0.39115427565191613, 0.11104137295003058, -0.21433966075865538, -0.4204789459072362, -0.3345509023293311]
```

```
f=0.3985257601446745 Autovetor=[-0.24007404439993782, -0.38560373778447443, -0.41920153791627734, -0.3244364595621224, -0.1126308877915994, 0.11263136807782374, 0.3244365416947634, 0.41920116168438015, 0.3856030826413384, 0.24007353473638404]
```

```
f=0.20139559952478217 Autovetor=[0.12524519047350036, 0.2290123152551859, 0.3244019664822125, 0.38597121741500856, 0.42149317698046695, 0.42149319993080037, 0.38597128350551896, 0.3244020518352418, 0.22901239371233825, 0.12524523929357473]
```

Temos agora o modo de vibração com a maior frequência para o caso de 10 massas. Podemos observar que todas as massas vibram na mesma frequência equivalente à $f=1.409206686522176$ por conta da posição inicial sendo um múltiplo de um dos autovetores.

Os outros modos de vibração e suas respectivas frequências se encontram acima.

5. Conclusão

O problema de massa mola possui inúmeros desafios e aplicações em diversos ramos da engenharia. Diversos sistemas podem ser simplificados, reduzidos ou simplesmente análogos a um sistema massa mola. Nos foi proposto um método de classificação composto por um algoritmo simples e extremamente poderoso, capaz de produzir excelentes resultados. Pode-se perceber que o trabalho extenso com matrizes de grandes dimensões pode levar a programas que requerem um maior poder de processamento e tempo de execução. Dessa forma, se mostra latente a necessidade de implementações eficientes dos algoritmos de manipulação de matrizes. A linguagem Python caracteriza uma ferramenta muito útil na implementação do algoritmo proposto, no entanto apresenta um maior tempo de execução devido ao seu caráter interpretado, em comparação com linguagens compiladas como C, C++ ou Julia. O uso da biblioteca Numpy representa um salto em desempenho do algoritmo, uma vez que recorre à chamada de funções compiladas de C.