Exercício de Programa 4

Caio Vinícius Dadauto 7994808

16 de junho de 2013

Problema I

A partir das condições iniciais $y(0) = 0$ e $\dot{y}(0) = 0$ e, ainda, da equação diferencial ordinaria de segunda ordem,

$$\dot{z} = \dot{y} - y + t^3 - 3t^2 + 6t \quad (1)$$

$$z = \dot{y} \quad (2)$$

foram implementados dois programas com o intuito de determinar $y(5)$ e $\dot{y}(5)$. O primeiro programa aproxima tais valores aplicando o método de Euler, o qual segue logo abaixo.

Programa 1: Implementação para o método de Euler.

```
#include<stdio.h>
#include<math.h>
#define H 0.01

double funcao_z (double y_o, double t, double z_o) {
   return z_o + H * (z_o - y_o + pow(t, 3) - 3 * pow(t, 2) + 6 * t);
}

double funcao_y (double y_o, double z_o) {
   return y_o + H * z_o;
}

int main () {
   int
      i;
/* z = y ' */
   double
      y,
```

```
      z,
      z_o = 0,
      y_o = 0,
      t = 0;

   for (i = 0; i < 5/H; ++i) {
      z = funcao_z (y_o, t, z_o);
      y = y_o + H * z_o;
      y_o = y;
      z_o = z;
      t += H;
   }
   printf ("y = %f\ty' = %f\n", y_o, z_o);
   return 0;
}
```

Já o segundo programa aplica o método de Runge-Kutta clássico de quarta ordem que segue abaixo.

Programa 2: Implementação para o método de Runge-Kutta.

```
#include<stdio.h>
#include<math.h>
#define H 0.01

double funcao (double y, double t, double z) {
   return z - y + pow(t, 3) - 3 * pow(t, 2) + 6 * t;
}

int main () {
/* z = y' */
   int
      i;
   double
      k_1z,
      k_2z,
      k_3z,
      k_4z,
      k_1y,
      k_2y,
      k_3y,
      k_4y,
      z = 0,
      y = 0,
      t = 0;

   for (i = 0; i < 5/H; ++i) {
```

```
      k_1y = H * z;
      k_1z = H * funcao (y, t, z);
      k_2y = H * (z + k_1z / 2);
      k_2z = H * funcao (y + k_1y / 2, t + H / 2, z + k_1z / 2);
      k_3y = H * (z + k_2z / 2);
      k_3z = H * funcao (y + k_2y / 2, t + H / 2, z + k_2z / 2);
      k_4y = H * (z + k_3z);
      k_4z = H * funcao (y + k_3y, t + H, z + k_3z);
      z = z + (k_1z + 2 * k_2z + 2 * k_3z + k_4z) / 6;
      y = y + (k_1y +2 * k_2y + 2 * k_3y + k_4y) / 6;
      t += H;
   }
   printf ("y = %.15f\ty ' = %.15f\n", y, z);
   return 0;
}
```

Os valores obtidos para $y(5)$ e $\dot{y}(5)$, a partir do método de Euler e do método de Runge-Kutta, são apresentados a seguir:

Euler	$y(5) = 125.565125$ $\dot{y}(5) = 75.914657$
Runge-Kutta	$y(5) = 124.999999$ $\dot{y}(5) = 74.999999$

Como a solução da equação (1) é $y = t^3$, é claramente observavel (analisando os resultados acima) que o método de Runge-Kutta é muito mais acurado que o método de Euler, assim como era esperado.

Problema II

Item 1

Item 1 - A

Para determinar o diagrama de fase ($\dot{x} \times x$) dado pela solução da equação do potencial de poço duplo,

$$\ddot{x} - \frac{x}{2}(1 - x^2) = 0 \tag{3}$$

foi implementado um programa que aplica o método de Runge-Kutta de quarta ordem com um passo de 0.05. O qual é responsável por aproximar os pontos do plano $\dot{x} \times x$.

O diagrama de fase foi plotado para três condições iniciais distintas de $\dot{x}(0)$ com $x(0) = -1$. Segue os diagramas obtido para cada uma das condições iniciais de $\dot{x}(0)$.

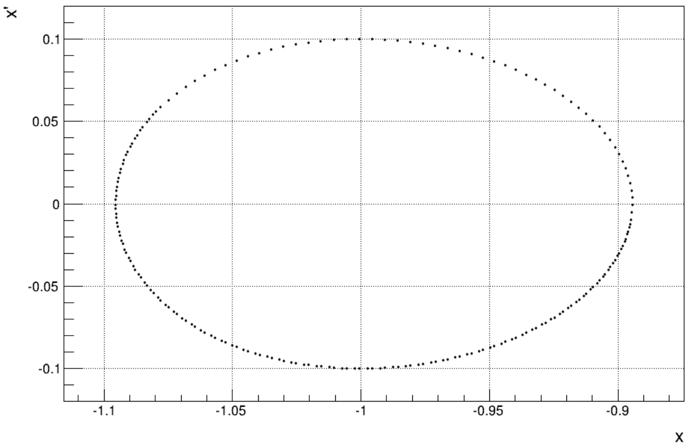

Figura 1: Diagrama de fase para $\dot{x}(0) = 0.1$.

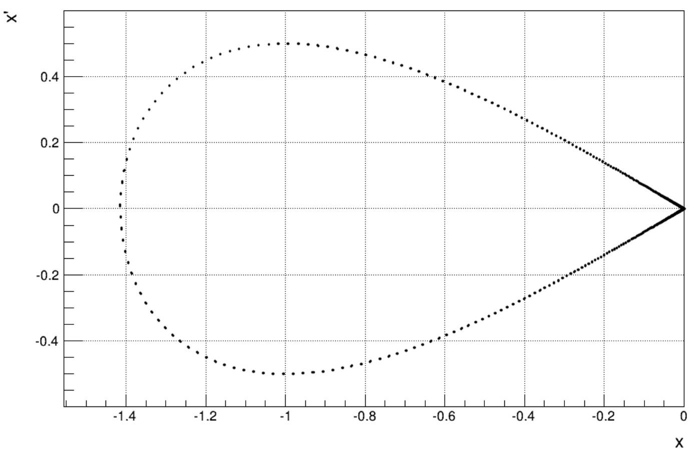

Figura 2: Diagrama de fase para $\dot{x}(0) = 0.5$.

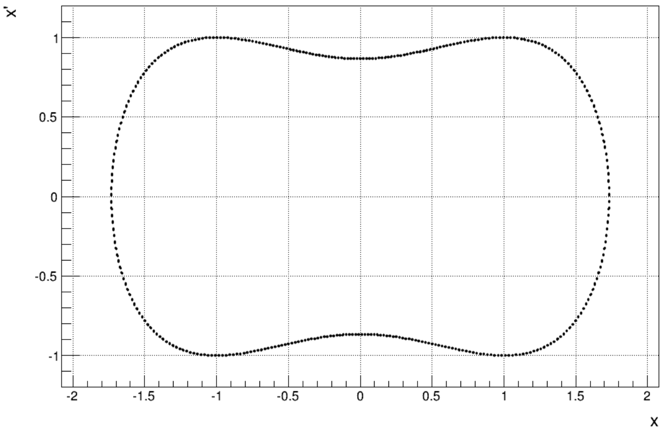

Figura 3: Diagrama de fase para $\dot{x}(0) = 1$.

Item 1 - B

Incluindo amortecimento a equeção (3), tem-se:

$$\ddot{x} + 2\gamma\dot{x} - \frac{x}{2}(1 - x^2) = 0 \tag{4}$$

Procedendo de forma semelhante ao item 1 - A, porém, agora, fixando $\dot{x}(0) = 1$ e $x(0) = -1$, foram plotados dois diagramas de fase, cada qual para um valor distinto de γ. Diagramas, estes, que seguem abaixo.

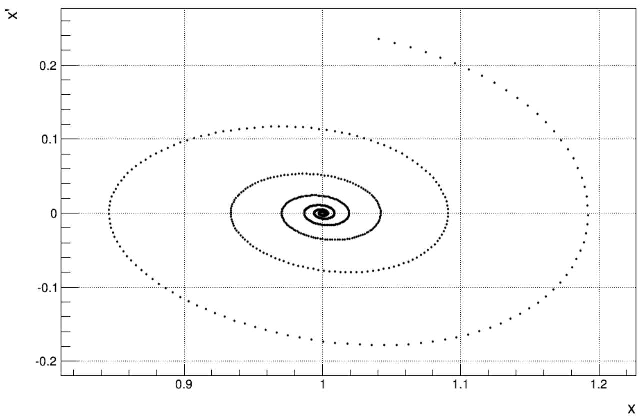

Figura 4: Diagrama de fase para $\gamma = 0.125$.

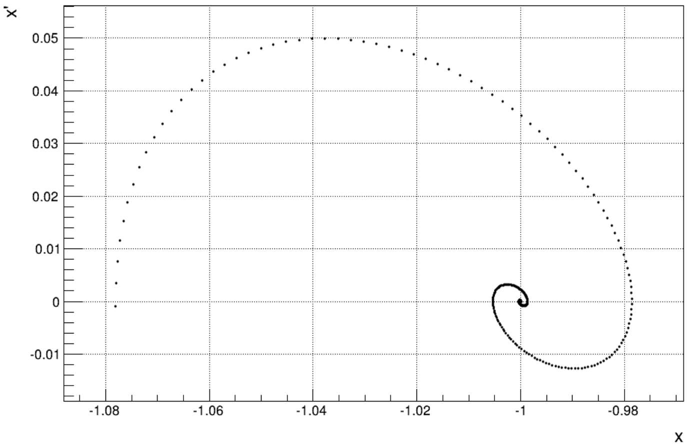

Figura 5: Diagrama de fase para $\gamma = 0.4$.

Item 1 - C

Forçando o sistema dado pela equação (4) com uma força F de período 2π, tem-se a seguinte equação:

$$\ddot{x} + 0.25\dot{x} - \frac{x}{2}(1 - x^2) = F\cos t \tag{5}$$

Tomando como condições iniciais $\dot{x}(0) = 1$ e $x(0) = -1$, foram plotados quatro diagramas de fase distintos, cada qual para um valor de F (0.22, 0.28, 0.35, 0.6). Ao plotar cada diagrama foram desconsiderados os primeiros 7000 pontos obtidos pela evolução do Runge-Kutta, com o intuito de eliminar o transiente.

Como exemplo de programa, é apresentado abaixo o código implementado para plotar o diagrama de fase quando F vale 0.6. Sendo que no programa para os demais diagramas ($F = 0.22$, $F = 0.28$, $F = 0.35$) foi modificado apenas o valor da força na função rotulada por "funcao".

Programa 3: Implementação para plotar o diagrama de fase da equação (5) quando $F = 0.6$.

```c
#include<stdio.h>
#include<math.h>
#define H 0.05

double funcao (double x, double t, double v) {
   return (x / 2) * (1 - pow(x, 2)) - (0.25 * v) + 0.6 * cos( t );
}

int main () {
   int
      i;
   FILE
      *arquivo;
/* v = x ' */
   double
      k_1v,
      k_2v,
      k_3v,
      k_4v,
      k_1x,
      k_2x,
      k_3x,
      k_4x,
      x = -1,
      v = 1,
      t = 0;

   arquivo = fopen ("graf_for.txt", "w");
   for (i = 0; i < 420/H; ++i) {
      k_1x = H * v;
      k_1v = H * funcao (x, t, v);
      k_2x = H * (v + k_1v / 2);
      k_2v = H * funcao (x + k_1x / 2, t + H / 2, v + k_1v / 2);
      k_3x = H * (v + k_2v / 2);
      k_3v = H * funcao (x + k_2x / 2, t + H / 2, v + k_2v / 2);
      k_4x = H * (v + k_3v);
      k_4v = H * funcao (x + k_3x, t + H, v + k_3v);
      if (i > 7000)
         fprintf (arquivo, "%f\t%f\n", x, v);
      v = v + (k_1v +2 * k_2v + 2 * k_3v + k_4v) / 6;
      x = x + (k_1x +2 * k_2x + 2 * k_3x + k_4x) / 6;
      t += H;
   }
   fclose (arquivo);
```

```
    return 0;
}
```

Segue os diagramas de fase obtidos para cada valor distinto de F.

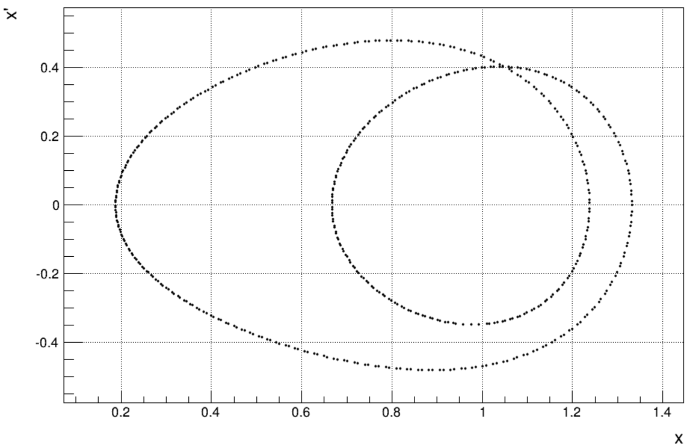

Figura 6: Diagrama de fase para $F = 0.22$.

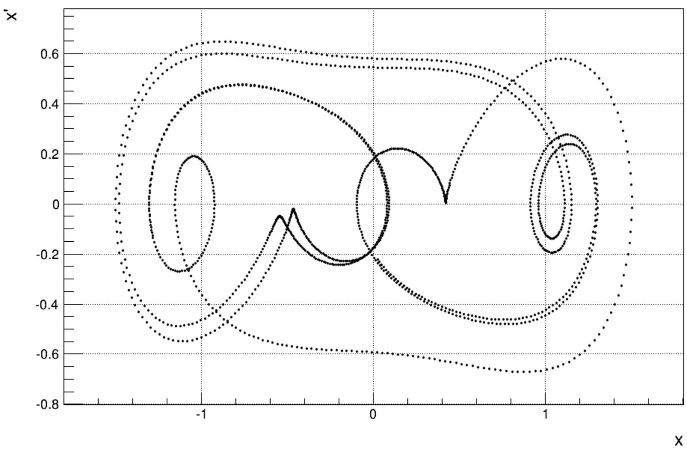

Figura 7: Diagrama de fase para $F = 0.28$.

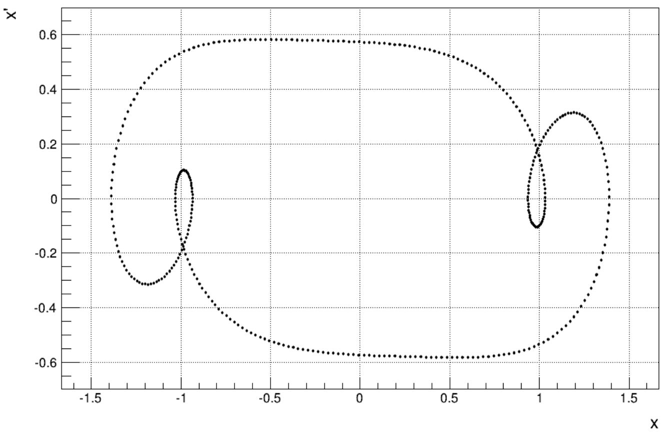

Figura 8: Diagrama de fase para $F = 0.35$.

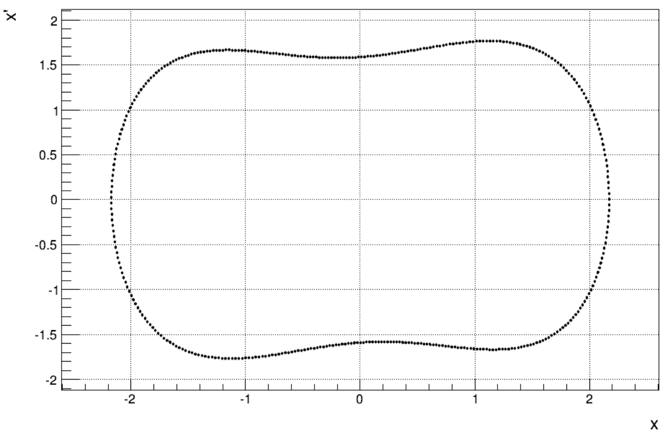

Figura 9: Diagrama de fase para $F = 0.6$.

No item 1 - A os atratores são dados pelas curvas definidas nos diagramas de fase das figuras 1, 2 e 3. No item 1 -B os atratores são os pontos, no plano $\dot{x} \times x$, (1; 0) e (-1; 0) para $\gamma = 0.125$ e para $\gamma = 0.4$, respectivamente. Por fim, no item 1 - C os atratores, para $F = 0.22$, $F = 0.35$ e $F = 0.6$, são dados pelas curvas definidas nos diagramas de fase das figuras 6, 8 e 9. Contudo, para $F = 0.28$ o atrator é caótico como se pode observar na figura 7.

Item 2

Partindo da equação (5) e das mesmas condições iniciais do item 1 - C é possível plotar um diagrama de bifurcação. Bastando para isso plotar sussecivas secções de Poincaré a cada período do elemento forçador F (que nesse caso é 2π) para um numero discreto de valores de F dentro de um intervalo.

Assim, foi implementado um programa que toma $[0; 0.7]$ como o intervalo de F, o qual progride em passos de 0.0005 de forma a discretizar o intervalo. A cada valor de F são plotadas 100 secções de Poincaré por periodo de F. O tempo t, por sua vez, varia de 0 a 2π (período de F) em passos de 0.002π de forma que para se completar um periodo seja necessario 1000 interações de Runge-Kutta. É ainda relalizada 200000 interações de Runge-Kutta a cada novo valor de F de forma a eliminar a região de transiente. Tal programa segue logo abaixo.

Programa 4: Implementação para plotar o diagrama de fase para a equação (5).

```
#include<stdio.h>
#include<math.h>
#define  H 6.283185E-3

double funcao (double x, double t, double v, double f) {
   return (x / 2) * (1 - pow(x, 2)) - (0.25 * v) + f * cos( t );
}

void runge_kuta (double *x, double *t, double *v, double *f) {
   double
      k_1v,
      k_2v,
      k_3v,
      k_4v,
      k_1x,
      k_2x,
      k_3x,
      k_4x;

   k_1x = H * *v;
   k_1v = H * funcao (*x, *t, *v, *f);
```

```
   k_2x = H * (*v + k_1v / 2);
   k_2v = H * funcao (*x + k_1x / 2, *t + H / 2, *v + k_1v / 2, *f);
   k_3x = H * (*v + k_2v / 2);
   k_3v = H * funcao (*x + k_2x / 2, *t + H / 2, *v + k_2v / 2, *f);
   k_4x = H * (*v + k_3v);
   k_4v = H * funcao (*x + k_3x, *t + H, *v + k_3v, *f);
   *v = *v + (k_1v +2 * k_2v + 2 * k_3v + k_4v) / 6;
   *x = *x + (k_1x +2 * k_2x + 2 * k_3x + k_4x) / 6;
   *t += H;
}

int main () {
   int
      k,
      j,
      i;
   FILE
      *arquivo;
/* v = x ' */
   double
      f  = 0,
      x = -1,
      v = 1,
      t = 0;

   arquivo = fopen ("graf_birf.txt", "w");
   for (k = 0; k <= 1400; ++k) {
      for (j = 0; j < 200000; ++j) {
         runge_kuta (&x, &t, &v, &f);
      }
      for (i = 0; i < 100; ++i) {
         for (j = 0; j < 1000; ++j) {
            runge_kuta (&x, &t, &v, &f);
         }
         fprintf (arquivo, "%f\t%f\n", f, x);
         printf ("%d\n", k);
      }
      f += 0.0005;
      x = -1;
      v = 1;
      t = 0;
   }
   fclose (arquivo);
   return 0;
}
```

A partir do programa acima, obteve-se o diagrama de bifurcação apresentado na figura 10.

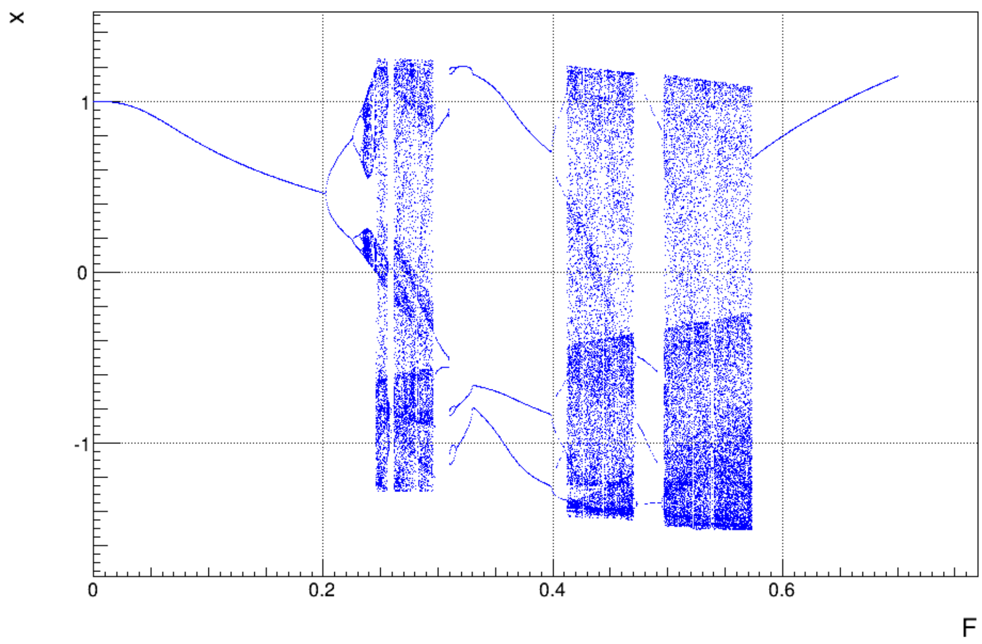

Figura 10: Diagrama de bifurcação para a equação (5).

Com o objetivo de determinar a constante de Feigenbaum (δ) a figura 10 foi ampliada de forma a analisar o intervalo $[0.232; 0.234]$ de F e o intervalo $[0.2; 0.25]$ de X, tal ampliação é apresentada na figura 11. Nesta figura são indicados dois intervalos (Δ_1 e Δ_2), que correspondem a variação de F em duas bifurcações concecutivas. Assim, tem-se que a constante de Feigenbaum é dada, aproximadamente, por:

$$\delta = \frac{\Delta_1}{\Delta_2} \tag{6}$$

A partir da leitura do gráfico apresentado na figura 11 é possível determinado o valor de $\Delta_1 = 1.64 \cdot 10^{-3}$ e $\Delta_2 = 3.8 \cdot 10^{-4}$. Portanto, partindo da equação (6), tem-se que:

$$\delta = 4.31579$$

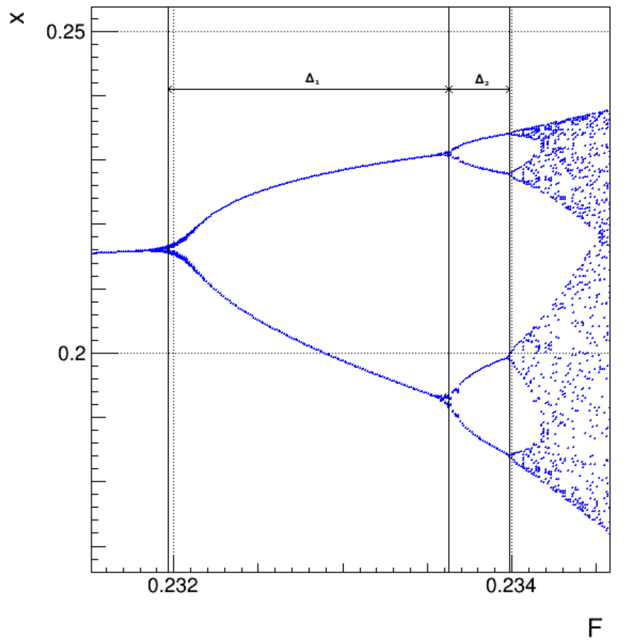

Figura 11: Ampliação do diagrama de bifurcação apresentado na figura 10.

Problema II

Para F constante, igual a 0.27, e nas mesmas condições iniciais do item 2, é possível sobrepor 2000 secções de Poincaré, onde cada secção é tomada após um ciclo do elemento forçador F.

De forma semelhante ao item 2, foram realizadas inicialmente 200000 interações de Runge-Kutta com o intuito de eleminar o transiente. A sobreposição de 2000 secções de Poincaré é apresentada na figura 12.

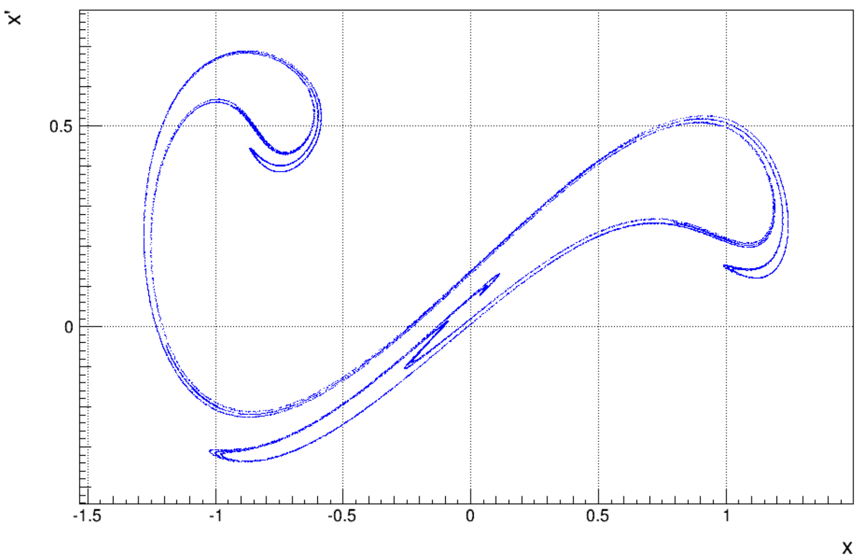

Figura 12: Sobreposição de 2000 secções de Poincaré.