

UNIVERSIDADE DE SÃO PAULO
ESCOLA POLITÉCNICA
MAP3122 - MÉTODOS NUMÉRICOS E APLICAÇÕES



Felipe Luís Korbes - 13682893
João Felipe de Souza Melo - 13682913

Professor: Alexandre Megiorin Roma

São Paulo
28 de março de 2024

Resumo

O objetivo deste relatório é analisar o comportamento da solução da órbita de 3 corpos celestes. Para atingir esse propósito, serão exploradas três técnicas numéricas: integração numérica utilizando o método de Runge-Kutta de 4 passos, Splines Cúbicos para conectar os pontos da solução discretizada, e a implementação do Algoritmo de Fatoração de Crout para encontrar soluções de um sistema linear tri-diagonal, necessária para a técnica dos Splines Cúbicos. Inicialmente, será realizada a modelagem matemática, acompanhada da explicação da motivação por trás da escolha do tema deste relatório. Em seguida, o problema em questão será discretizado e as técnicas numéricas serão detalhadamente explicadas. Por fim, será conduzida uma depuração no código computacional para assegurar que a implementação foi feita de maneira correta, apresentando limitações, detalhes e aplicações relevantes para as técnicas e o problema selecionado.

1 Introdução

A compreensão de corpos celestes é fundamental para o mundo moderno. Um fruto desse estudo é a estruturação do calendário ocidental, chamado calendário gregoriano [1] que tem o objetivo de alinhar o calendário com o movimento em torno do Sol e ajustar as datas para coincidir com os equinócios e solstícios. A compreensão de orbitas de planetas e satélites permitiu o ser humano a criar satélites artificiais o que elevou o impacto das telecomunicações em um nível global.

Um problema famoso relacionado a orbitas celestes é o problema dos 3 corpos [13] [5], onde há 3 corpos situados no espaço e cada corpo sofre a influência gravitacional dos outros dois corpos.

Apesar de não haver uma solução geral para esse problema, Karl F. Sundman [4] apresentou, em 1912, uma solução analítica na forma de uma série de potências, caracterizada por uma convergência extremamente lenta.

Um exemplo de problema dessa classe é a Terra, Lua e Sol. Onde o sol pode ser considerado um astro fixo no espaço por conta da quantidade imensa de massa se comparado com a terra e a lua.

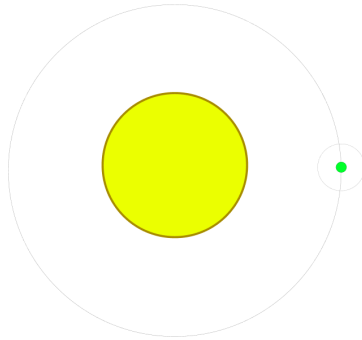


Figura 1: Problema dos 3 corpos terra-lua-sol fora de escala

É importante destacar que existem casos específicos do problema dos três corpos que possuem famílias de soluções conhecidas [7][11]. Contudo, por terem soluções exatas, esses casos também impõem restrições nas condições do sistema. O objetivo deste trabalho é utilizar um modelo genérico de três corpos, com as condições iniciais de um sistema com 3 corpos e aplicar métodos numéricos para aproximar suas órbitas.

Além disso, o problema dos três corpos pode ser generalizado para o caso de N corpos, ampliando significativamente sua aplicabilidade. Essa generalização é frequentemente empregada em diversas áreas, incluindo a detecção e monitoramento de asteroides [3], simulações em larga escala do universo, como a notável Simulação do Milênio [12], e outras aplicações pertinentes à astrofísica. Contudo, nesse trabalho, o problema será restringido a apenas 3 corpos.

2 Modelagem Matemática

O problema dos três corpos, em sua forma mais básica, faz uso da equação da gravitação universal de Newton, que na sua forma mais simples, modelada como a força entre os corpos m_i e m_n , obtém-se

$$\mathbf{F}_{gn} = -G \frac{m_i m_n}{r_{ni}^3} \mathbf{r}_{ni}. \quad (1)$$

Onde $r_{ni} = r_i - r_n$. A soma dos vetores \mathbf{F}_g , será a força gravitacional que age no corpo i , e pode ser escrita da forma

$$\mathbf{F}_g = -G \frac{m_i m_1}{r_{1i}^3} \mathbf{r}_{1i} - G \frac{m_i m_2}{r_{2i}^3} \mathbf{r}_{2i} - \dots - G \frac{m_i m_j}{r_{ji}^3} \mathbf{r}_{ji}. \quad (2)$$

Para $i \neq j$.

Dessa maneira, a fórmula 2 pode ser escrita como o somatório

$$\mathbf{F}_g = G m_i \sum_{\substack{j=1 \\ j \neq i}}^n \frac{m_j}{r_{ji}^3} \mathbf{r}_{ji}. \quad (3)$$

Dividindo ambos os lados da equação por m_i , obtemos a aceleração, que se expressa em termos da derivada da posição chega-se a expressão

$$\ddot{\mathbf{r}}_i = G \sum_{\substack{j=1 \\ j \neq i}}^n \frac{m_j}{r_{ji}^3} \mathbf{r}_{ji}. \quad (4)$$

Especificamente, ao se abordar o caso particular de três corpos, considerando a aceleração que cada corpo exerce sobre os outros presentes no sistema em cada novo instante, chega-se à tarefa de resolver um sistema de três equações diferenciais de segunda ordem, uma para cada corpo, conforme apresentado a seguir.

$$\begin{cases} \ddot{r}_1 &= -G m_2 \frac{(\mathbf{r}_1 - \mathbf{r}_2)}{|\mathbf{r}_1 - \mathbf{r}_2|^3} - G m_3 \frac{(\mathbf{r}_1 - \mathbf{r}_3)}{|\mathbf{r}_1 - \mathbf{r}_3|^3} \\ \ddot{r}_2 &= -G m_1 \frac{(\mathbf{r}_2 - \mathbf{r}_1)}{|\mathbf{r}_2 - \mathbf{r}_1|^3} - G m_3 \frac{(\mathbf{r}_2 - \mathbf{r}_3)}{|\mathbf{r}_2 - \mathbf{r}_3|^3} \\ \ddot{r}_3 &= -G m_2 \frac{(\mathbf{r}_3 - \mathbf{r}_2)}{|\mathbf{r}_3 - \mathbf{r}_2|^3} - G m_1 \frac{(\mathbf{r}_3 - \mathbf{r}_1)}{|\mathbf{r}_3 - \mathbf{r}_1|^3} \end{cases} \quad (5)$$

É evidente que cada corpo que compõe nosso sistema introduz 6 variáveis de estado - 3 para a posição e 3 para a velocidade. Assim, em um sistema com 3 corpos, encontra-se um total de 18 variáveis de estado.

3 Metodologia Numérica

3.1 Discretização do Problema

Para aplicar técnicas de integração ao sistema 5, é necessário transformá-lo em um sistema de equações diferenciais de primeira ordem, chegando ao seguinte problema de Cauchy:

$$\left\{ \begin{array}{l} \dot{\mathbf{r}}_1 = \mathbf{v}_1 \\ \dot{\mathbf{v}}_1 = -Gm_2 \frac{(\mathbf{r}_1 - \mathbf{r}_2)}{|\mathbf{r}_1 - \mathbf{r}_2|^3} - Gm_3 \frac{(\mathbf{r}_1 - \mathbf{r}_3)}{|\mathbf{r}_1 - \mathbf{r}_3|^3} \\ \dot{\mathbf{r}}_2 = \mathbf{v}_2 \\ \dot{\mathbf{v}}_2 = -Gm_1 \frac{(\mathbf{r}_2 - \mathbf{r}_1)}{|\mathbf{r}_2 - \mathbf{r}_1|^3} - Gm_3 \frac{(\mathbf{r}_2 - \mathbf{r}_3)}{|\mathbf{r}_2 - \mathbf{r}_3|^3} \\ \dot{\mathbf{r}}_3 = \mathbf{v}_3 \\ \dot{\mathbf{v}}_3 = -Gm_2 \frac{(\mathbf{r}_3 - \mathbf{r}_2)}{|\mathbf{r}_3 - \mathbf{r}_2|^3} - Gm_1 \frac{(\mathbf{r}_3 - \mathbf{r}_1)}{|\mathbf{r}_3 - \mathbf{r}_1|^3} \\ r_1(t_0) = (x_{1;0}, y_{1;0}, z_{1;0}) ; v_1(t_0) = (v_{x1;0}, v_{y1;0}, v_{z1;0}) \\ r_2(t_0) = (x_{2;0}, y_{2;0}, z_{2;0}) ; v_2(t_0) = (v_{x2;0}, v_{y2;0}, v_{z2;0}) \\ r_3(t_0) = (x_{3;0}, y_{3;0}, z_{3;0}) ; v_3(t_0) = (v_{x3;0}, v_{y3;0}, v_{z3;0}) \end{array} \right. \quad (6)$$

Dessa equação, \mathbf{v}_1 , \mathbf{v}_2 e \mathbf{v}_3 representam os vetores velocidade de cada corpo, incluindo os três eixos cartesianos. Os valores $\dot{\mathbf{v}}_1$, $\dot{\mathbf{v}}_2$ e $\dot{\mathbf{v}}_3$ indicam os vetores de aceleração de cada corpo. Além disso, G denota a constante da gravitação universal, m_1 , m_2 e m_3 as massas individuais, \mathbf{r}_1 , \mathbf{r}_2 e \mathbf{r}_3 , não presentes no sistema acima, são os vetores de posição de cada corpo, e por fim, as condições iniciais para cada corpo do sistema. Todas as unidades estão no Sistema Internacional de medidas.

Para o caso de 3 corpos escolhido, o sistema consistirá em um corpo central que se move ao longo do eixo z , enquanto dois outros corpos de mesma massa o orbitam. Uma animação foi feita para demonstrar tal órbita usando as condições iniciais descritas abaixo, e pode ser encontrado neste link.

- **Corpo Central:**

$$\vec{r} = (0; 0; 0)$$

$$\vec{v} = (0; 0; 5e3)$$

$$m = 1.9885e30$$

- **Corpo Lateral 1:**

$$\vec{r} = (1.47095e11; 0; 0)$$

$$\vec{v} = (0; 30290; 0)$$

$$m = 5.972e27$$

• **Corpo Lateral 2:**

$$\vec{r} = (-1.47095e11; 0; 0)$$

$$\vec{v} = (0; -30290; 0)$$

$$m = 5.972e27$$

A resolução desse sistema envolverá o cálculo das velocidades \mathbf{v}_1 , \mathbf{v}_2 e \mathbf{v}_3 e das posições \mathbf{r}_1 , \mathbf{r}_2 e \mathbf{r}_3 dos corpos através do método de integração de Runge Kutta de 4 passos, onde nossas $f(t, y)$ serão as fórmulas vistas em 6.

3.2 Técnicas Numéricas

3.2.1 Método de Runge Kutta de 4 Passos

O método de Runge-Kutta de 4 passos proporciona uma integração numérica altamente precisa, com uma ordem de erro igual a 4 [9]. Isso significa que, à medida que diminui-se o passo de integração no problema em questão, o erro em relação à solução exata diminui com uma taxa de ordem 4. Este método é explícito, pois não depende de um valor futuro para o seu cálculo, e também é um método de passo único, pois não requer valores anteriores ao passo atual de integração. Para resolver o problema proposto na Equação 6, aplica-se essa técnica de integração, na forma vetorial, expressa por:

$$\phi(t, r, h) = \frac{1}{6}(\kappa_1 + 2\kappa_2 + 2\kappa_3 + \kappa_4) \text{ e } \begin{cases} \kappa_1 = f(t_k, y_k) \\ \kappa_2 = f(t_k + \frac{h}{2}, y_k + \frac{h}{2}\kappa_1) \\ \kappa_3 = f(t_k + \frac{h}{2}, y_k + \frac{h}{2}\kappa_2) \\ \kappa_4 = f(t_k + h, y_k + h\kappa_3) \end{cases} \quad (7)$$

Este processo se repetirá para todo o instante de tempo até $t = t_f$.

3.2.2 Splines Cúbicos Vinculados

O método dos Splines Cúbicos permite realizar a interpolação dos pontos discretos obtidos por meio do método de integração mencionado em 3.2.1. Essa técnica é particularmente interessante, pois permite a interpolação de uma quantidade arbitrária de pontos, evitando as oscilações nas bordas de um intervalo associadas ao fenômeno de Runge [10], que descreve oscilações nas extremidades de um intervalo quando utiliza-se interpolação polinomial de ordens elevadas.

Para que um Spline Cúbico seja usado, necessita-se que alguns critérios sejam satisfeitos:

1. $s_k(x_k) = y_k$ e $s_k(x_{k+1}) = y_{k+1}$
2. $s_k(x_{k+1}) = s_{k+1}(x_{k+1})$
3. $s'_k(x_{k+1}) = s'_{k+1}(x_{k+1})$
4. $s''_k(x_{k+1}) = s''_{k+1}(x_{k+1})$

Resumindo, precisa-se que cada polinômio passe por dois pontos consecutivos, duas equações consecutivas tem que coincidir no ponto que as conecta, a primeira derivada de duas equações consecutivas tem que ser iguais no ponto que as conecta, e a segunda derivada de duas equações consecutivas tem que ser iguais no ponto que as conecta.

O que resulta desse método são $n - 1$ polinômios da forma:

$$s_k(x) = a_k + b_k(x - x_k) + c_k(x - x_k)^2 + d_k(x - x_k)^3. \quad (8)$$

Onde a_k , b_k , c_k e d_k são coeficientes a serem encontrados. Rescrevendo a fórmula acima na forma

$$h_{k-1}c_{k-1} + 2(h_{k-1} - h_k)c_k + h_k c_{k+1} = \frac{3}{h_k}(a_{k+1} - a_k) - \frac{3}{h_{k-1}}(a_k - a_{k-1}), \quad (9)$$

chega-se em $n - 1$ equações onde os tem-se apenas os c_n coeficientes para encontrar, onde $h_k = x_{k+1} - x_k$, ou seja, a distância entre pontos consecutivos da abscissa em questão. Além disso, $a_k = y_k$, onde y_k são os valores das ordenadas para cada ponto. Assim, chega-se a um sistema de equações que sempre será possível e determinado, e que pode ser escrito na forma $AX = B$, onde, para um Spline Cúbico Vinculado, com as condições de borda iguais a $s'_k(x_0) = f'(x_0)$ e $s'_k(x_n) = f'(x_n)$, temos

$$A = \begin{bmatrix} 2h_0 & h_0 & 0 & 0 & 0 & \dots & 0 & 0 \\ h_0 & 2(h_0 + h_1) & h_1 & 0 & 0 & \dots & 0 & 0 \\ 0 & h_1 & 2(h_1 + h_2) & h_2 & 0 & \dots & 0 & 0 \\ \vdots & & & & & & & \\ 0 & 0 & 0 & 0 & \dots & h_{n-2} & 2(h_n + h_{n-1}) & h_{n-1} \\ 0 & 0 & 0 & 0 & \dots & 0 & h_{n-1} & 2h_{n-1} \end{bmatrix}, \quad (10)$$

$$X = \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ \vdots \\ c_{n-1} \\ c_n \end{bmatrix}, \quad (11)$$

$$B = \begin{bmatrix} 3(a_1 - a_0)/h_0 - 3f'(a) \\ 3(a_2 - a_1)/h_1 - 3(a_1 - a_0)/h_0 \\ 3(a_3 - a_2)/h_2 - 3(a_2 - a_1)/h_1 \\ \vdots \\ 3(a_n - a_{n-1})/h_{n-1} - 3(a_{n-1} - a_{n-2})/h_{n-2} \\ 3f'(b) - 3(a_n - a_{n-1})/h_{n-1} \end{bmatrix}. \quad (12)$$

Uma vez encontrados os coeficientes c_0 a c_n , consegue-se encontrar, b_k e d_k com as seguintes fórmulas

$$b_k = \frac{1}{h_k}(a_{k+1} - a_k) - \frac{h_k}{3}(2c_k + c_{k+1}) \quad (13)$$

e

$$d_k = \frac{c_{k+1} - c_k}{3h_k}. \quad (14)$$

Um algoritmo de implementação desse método se encontra abaixo:

Algorithm 1 Spline Cúbico Vinculado

```
1: ENTRADAS:  $n : x_0, x_1, \dots, x_n$ ;  $a_0 = f(x_0), a_1 = f(x_1), \dots, a_n = f(x_n)$ ;  $PO = f'(x_0)$ ;  $UO = f'(x_n)$ 
2: procedure SPLINECUBICOVINCULADO( $n, x_i, a_i, PO, UO$ )
3:   for  $i = 0$  até  $n - 1$  do
4:      $h_i \leftarrow x_{i+1} - x_i$ 
5:   end for
6:    $\alpha_0 \leftarrow 3(a_1 - a_0)/h_0 - 3PO$ 
7:    $\alpha_n \leftarrow 3UO - 3(a_n - a_{n-1})/h_{n-1}$ 
8:   for  $i = 1, 2, \dots$  até  $n - 1$  do
9:      $\alpha_i \leftarrow \frac{3}{h_i}(a_{i+1} - a_i) - \frac{3}{h_{i-1}}(a_i - a_{i-1})$ 
10:  end for
11:   $l_0 \leftarrow 2h_0, \mu_0 \leftarrow 0.5, z_0 \leftarrow \alpha_0/l_0$ 
12:  for  $i = 1, 2, \dots$  até  $n - 1$  do
13:     $l_i \leftarrow 2(x_{i+1} - x_{i-1}) - h_{i-1}\mu_{i-1}$ 
14:     $\mu_i \leftarrow \frac{h_i}{l_i}, z_i \leftarrow \alpha_i - h_{i-1}z_{i-1}/l_i$ 
15:  end for
16:   $l_n \leftarrow h_{n-1}(2 - \mu_{n-1}), z_n \leftarrow \alpha_n - h_{n-1}z_{n-1}/l_n, c_n \leftarrow z_n$ 
17:  for  $j = n - 1, n - 2, \dots$  até  $0$  do
18:     $c_j \leftarrow z_j - \mu_j c_{j+1}$ 
19:     $b_j \leftarrow (a_{j+1} - a_j)/h_j - h_j(c_{j+1} + 2c_j)/3$ 
20:     $d_j \leftarrow (c_{j+1} - c_j)/3h_j$ 
21:  end for
22:  SAÍDA  $(a_j, b_j, c_j, d_j)$  para  $j = 0, 1, \dots, n - 1$ 
23:  PARE.
24: end procedure
```

3.2.3 Método de Crout para Resolução de Matriz Tridiagonal

A abordagem para a resolução de uma matriz tridiagonal apresenta um algoritmo altamente eficiente na solução de sistemas lineares, devido à estrutura específica dessa matriz [6]. Uma matriz tridiagonal é caracterizada por ter valores na diagonal principal e nas duas diagonais adjacentes, assemelhando-se ao sistema $AX = B$ discutido na Seção 3.2.2.

O processo de resolução de uma matriz tridiagonal consiste fatorar a matriz principal A em duas matrizes L e U , conforme mostrado abaixo.

$$L = \begin{bmatrix} l_{11} & 0 & 0 & 0 & \dots & 0 & 0 \\ l_{21} & l_{22} & 0 & 0 & \dots & 0 & 0 \\ \vdots & & & & & & \\ 0 & 0 & 0 & 0 & \dots & l_{n-1,n-2} & 0 \\ 0 & 0 & 0 & 0 & \dots & l_{n,n-1} & l_{nn} \end{bmatrix} \quad (15)$$

$$U = \begin{bmatrix} 1 & u_{12} & 0 & 0 & \dots & 0 & 0 \\ 0 & 1 & u_{23} & 0 & \dots & 0 & 0 \\ \vdots & & & & & & \\ 0 & 0 & 0 & 0 & \dots & u_{n-1,n} & 1 \\ 0 & 0 & 0 & 0 & \dots & 0 & 1 \end{bmatrix} \quad (16)$$

A obtenção das matrizes 15 e 16 é realizada através das seguintes operações:

$$a_{11} = l_{11}; \quad (17)$$

$$a_{i,i-1} = l_{i,i-1}, \quad \text{para } i = 2, 3, \dots, n; \quad (18)$$

$$a_{ii} = l_{i,i-1}u_{i-1,i} + l_{ii}, \quad \text{para } i = 2, 3, \dots, n; \quad (19)$$

$$a_{i,i+1} = l_{ii}u_{i,i+1}, \quad \text{para } i = 1, 2, \dots, n-1. \quad (20)$$

Aplicando as operações 17, 18, 19 e 20 na matriz A (10), chegamos aos sistemas $Lz = B$ e $Ux = z$, onde B é a matriz 12 e x são as soluções do nosso sistema. A seguir, são apresentados os dois sistemas em questão.

$$\begin{bmatrix} l_{11} & 0 & 0 & 0 & \dots & 0 & 0 \\ l_{21} & l_{22} & 0 & 0 & \dots & 0 & 0 \\ \vdots & & & & & & \\ 0 & 0 & 0 & 0 & \dots & l_{n-1,n-2} & 0 \\ 0 & 0 & 0 & 0 & \dots & l_{n,n-1} & l_{nn} \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_{n-1} \\ z_n \end{bmatrix} = B \quad (21)$$

resolvendo o sistema para z_n com $n = 0, 1, 2, \dots, n$, e colocando-o no sistema $Ux = z$, temos

$$\begin{bmatrix} 1 & u_{12} & 0 & 0 & \dots & 0 & 0 \\ 0 & 1 & u_{23} & 0 & \dots & 0 & 0 \\ \vdots & & & & & & \\ 0 & 0 & 0 & 0 & \dots & u_{n-1,n} & 1 \\ 0 & 0 & 0 & 0 & \dots & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{n-1} \\ x_n \end{bmatrix} = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_{n-1} \\ z_n \end{bmatrix} \quad (22)$$

Onde x_n para $n = 0, 1, 2, \dots, n$ são as soluções do nosso sistema.

Uma implementação desse método se encontra no algoritmo abaixo.

Algorithm 2 Algoritmo para Resolver o Sistema $AX = B$

```
1: ENTRADAS: A dimensão  $n$ ; as entradas de  $A$ .
2: procedure RESOLVESISTEMALINEAR( $n, A$ )
3:    $l_{11} = a_{11}$ ;
4:    $u_{12} = a_{12}/l_{11}$ ;
5:    $z_1 = a_{1,n+1}/l_{11}$ ;
6:   for  $i = 2, 3, \dots$  to  $n - 1$  do
7:      $l_{i,i-1} = a_{i,i-1}$ ;
8:      $l_{ii} = a_{ii} - l_{i,i-1}u_{i-1,i}$ ;
9:      $u_{i,i+1} = a_{i,i+1}/l_{ii}$ ;
10:     $z_i = a_{i,n+1} - l_{i,i-1}z_{i-1}/l_{ii}$ .
11:  end for
12:   $l_{n,n-1} = a_{n,n-1}$ ;
13:   $l_{nn} = a_{nn} - l_{n,n-1}u_{n-1,n}$ ;
14:   $z_n = (a_{n,n+1} - l_{n,n-1}z_{n-1})/l_{nn}$ ;
15:   $x_n = z_n$ ;
16:  for  $i = n - 1, \dots$  até 1 do
17:     $x_i = z_i - u_{i,i+1}x_{i+1}$ .
18:  end for
19:  SAÍDAS:  $(x_1, \dots, x_n)$ ;
20:  PARE
21: end procedure
```

4 Resultados

4.1 Verificação Computacional

Nesta seção, será abordada a verificação computacional do código, na qual serão testadas as implementações em código de todos os métodos numéricos apresentados na seção 3. Para cada exemplo, serão fornecidos modelos com soluções conhecidas para validar o funcionamento do código.

4.1.1 Verificação do Método de Runge Kutta de 4 Passos

Para verificar o método de Runge-Kutta descrito em 3.2.1, foi selecionada uma equação diferencial de terceira ordem com solução conhecida, extraída de *Exact Solutions for Ordinary Differential Equations*, Andrei D. Polyanin et al.[8]. A equação escolhida é a seguinte:

$$y''' + 6y'' + 12y' + 8y = 0, \quad y(0) = 1, \quad y'(0) = 2, \quad y''(0) = 0 \quad (23)$$

Contudo, para aplicarmos o método de Runge-Kutta nesta EDO, é necessário primeiro decompor-lá em um sistema de equações diferenciais de primeira ordem. Através de uma substituição de variáveis onde $y = y_1$, $y' = y_2$, $y'' = y_3$ e $y''' = y_3'$, podemos re-escrever a formula acima da seguinte forma:

$$\begin{cases} y_1' = y_2 \\ y_2' = y_3 \\ y_3' = -6y_3 - 12y_2 - 8y_1 \\ y_1(0) = 1, \quad y_2(0) = 2, \quad y_3(0) = 0 \end{cases} \quad (24)$$

Cuja solução exata é igual a

$$y(t) = e^{-2t} + 4te^{-2t} + 6t^2e^{-2t} \quad (25)$$

Aplicando o método a este sistema por meio de iterações sucessivas com intervalos de tempo cada vez menores, chegamos aos seguintes resultados que demonstram a ordem e convergência do método:

n	Passo(Delta t)	$ e(t, h) $	$q = \log_2\left(\frac{ e(t, 2h) }{ e(t, h) }\right)$
4	2.500000	14911938.931450	
8	1.250000	41.436132	18.457147
16	0.625000	0.000001	25.497815
32	0.312500	0.000000	4.863427
64	0.156250	0.000000	4.338900
128	0.078125	0.000000	4.164391
256	0.039062	0.000000	4.081843
512	0.019531	0.000000	4.040879

Tabela 1: Delta t, erro absoluto e ordem de convergência das aproximações da equação 23

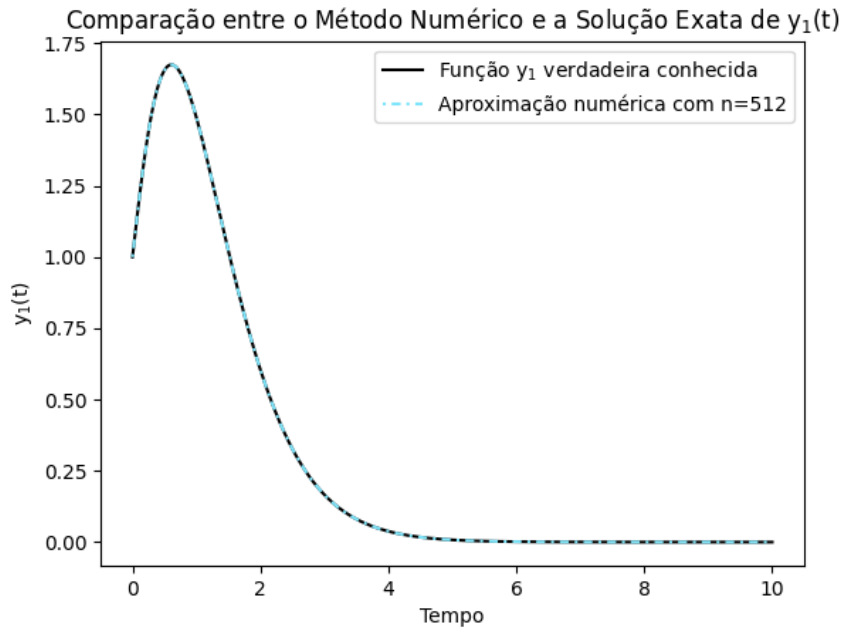


Figura 2: Comparação do método de Runge Kutta com a solução real do problema escolhido 23

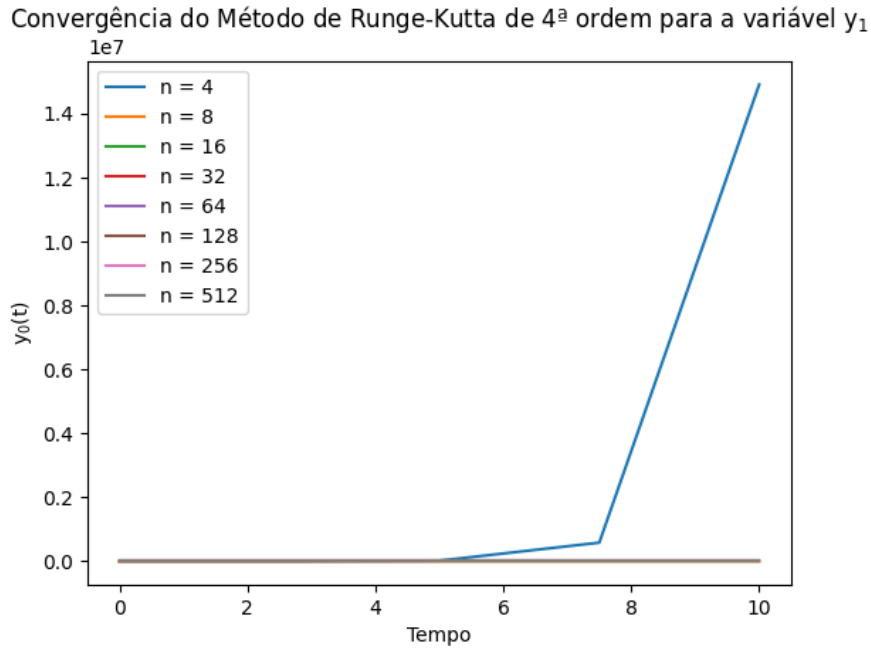


Figura 3: Convergência da equação 23 usando o método de Runge Kutta

Com base na tabela e nas figuras apresentadas acima, conclui-se que a implementação computacional do método de Runge-Kutta de 4 passos está funcionando conforme o esperado. Além disso, o método foi testado para as outras duas soluções da EDO escolhida. No entanto, optou-se por não incluir as tabelas e gráficos dessas soluções neste documento, a fim de manter o relatório conciso.

4.1.2 Verificação do Método dos Splines Cúbicos

Para verificar o método dos Splines Cúbicos 3.2.2, foi inicialmente implementado o Spline Cúbico Natural e, em seguida, o Spline Cúbico Vinculado, ambos aplicados à função exponencial. Essa escolha foi motivada pelo fato de que o Spline Vinculado, quando aplicado à função exponencial, é praticamente idêntico, tornando difícil distinguir entre a função real dos Splines. A motivação para usar a função exponencial veio de um exemplo tirado de Burden et al. [6], devido à sua implementação simples quanto à sua fácil verificação. Usando os pontos $(0, 1)$, $(1, e)$, $(2, e^2)$, $(3, e^3)$, chega-se ao seguinte resultando usando Splines Cúbicos Naturais.

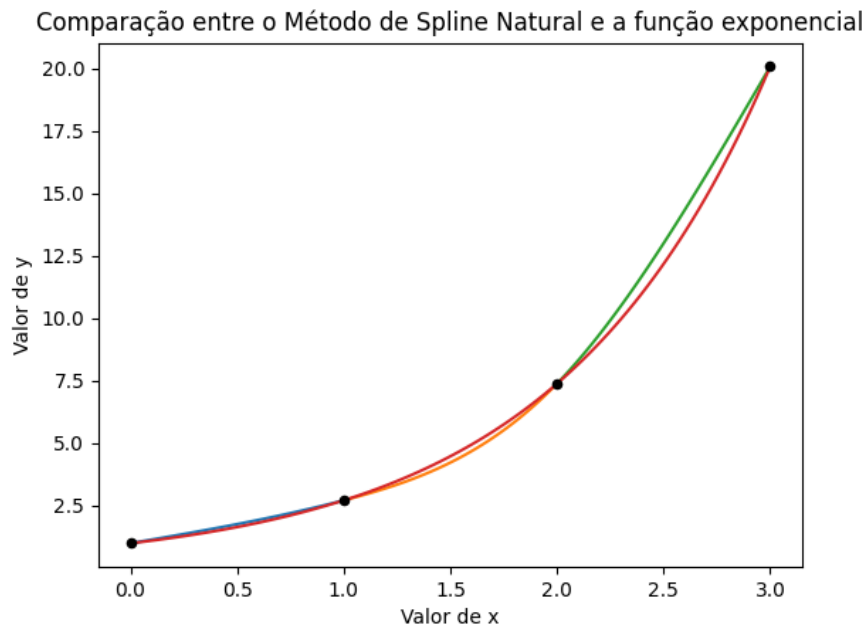


Figura 4: Comparação entre a função exponencial e sua aproximação usando Splines Cúbicos Naturais

Por outro lado, quando aplicado os Splines Cúbicos Vinculados a função exponencial, chega-se a um gráfico que é basicamente idêntico a função exponencial.

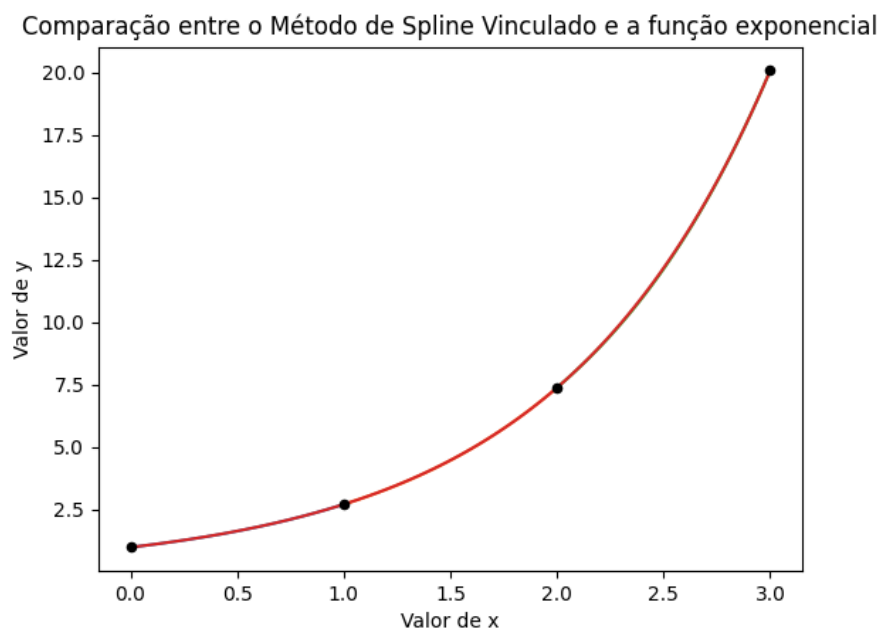


Figura 5: Comparação entre a função exponencial e sua aproximação usando Splines Cúbicos Vinculados

Destaca-se que, neste relatório será usado o método dos Splines Cúbicos Vinculados. Além disso, o processo de implementação deste método envolve a resolução

de um sistema linear, cujo será abordado na seção que segue.

4.1.3 Verificação do Método de Crout

O método de Crout para fatoração e resolução de uma matriz tridiagonal, conforme discutido na seção 3.2.3, foi novamente testado utilizando a função exponencial como exemplo, extraído de Burden et al. [6]. Como o método dos Splines Cúbicos envolve a resolução de um sistema linear, podemos inferir que o método de Crout já foi testado na seção anterior e está funcionando conforme o esperado. No entanto, conforme mencionado na seção anterior, o sistema linear resultante do método dos Splines Cúbicos Vinculados será resolvido, comparando os resultados obtidos pela implementação própria com uma biblioteca resolutiva de sistemas lineares disponível no Python [2]. O sistema linear a ser resolvido para a função exponencial é:

$$\begin{bmatrix} 2 & 1 & 0 & 0 \\ 1 & 4 & 1 & 0 \\ 0 & 1 & 4 & 1 \\ 0 & 0 & 1 & 2 \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} 3(e-2) \\ 3(e^2-2e+1) \\ 3(e^3-2e^2+e) \\ 3e^2 \end{bmatrix} \quad (26)$$

Resolvendo para c_0 , c_1 , c_2 e c_3 usando os dois tipos de resoluções mencionadas, obtém-se os seguintes resultados:

c_i	Implementação Própria	numpy.linalg
0	0.4446824757874891	0.4446824757874892
1	1.2654805084250225	1.2654805084250222
2	3.3508728405124235	3.3508728405124235
3	9.408147749525288	9.408147749525288

Tabela 2: Comparação entre a implementação própria do método de Crout com a resolução fornecida pelo NumPy

Com isso, concluí-se que o método de Crout implementado está funcionando.

4.2 Aplicação

Para aplicação ao problema dos 3 corpos escolhidos pela dupla, serão demonstrados os resultados prévios obtidos para a posição e velocidade de um dos corpos lateiras no eixo x , aplicando todas as técnicas numéricas mencionadas na sub-seção de Técnicas Numéricas 3.2. Serão utilizadas as condições iniciais descritas na parte de discretização do problema 3.1 para um intervalo total de 3 anos.

Do método de Runge Kutta de 4 Passos tem-se:

n	Passo (Delta t)	$\eta(t, h)$	$\log_2(\frac{\eta(t, 2h) - \eta(t, h)}{\eta(t, h) - \eta(t, h/2)})$
4	1250000.000000	77446942537.69884	
8	625000.000000	77448379100.21362	
16	312500.000000	77448459631.416550	-4.186948
32	156250.000000	77448464052.893173	-4.129648
64	78125.000000	77448464305.485031	-4.076105
128	39062.500000	77448464320.460815	-4.041244
256	19531.250000	77448464321.370422	-4.019002
512	9765.625000	77448464321.426529	-4.005110

Tabela 3: Delta t, aproximação da solução e ordem de convergência do problema escolhido para a posição no eixo x de um dos planetas

n	Passo (Delta t)	$\eta(t, h)$	$\log_2(\frac{\eta(t, 2h) - \eta(t, h)}{\eta(t, h) - \eta(t, h/2)})$
4	1250000.000000	-25198.914717084743	
8	625000.000000	-25197.45439856724	
16	312500.000000	-25197.361810	-4.009263
32	156250.000000	-25197.356060	-4.009107
64	78125.000000	-25197.355703	-4.005640
128	39062.500000	-25197.355681	-4.003073
256	19531.250000	-25197.355679	-4.000860
512	9765.625000	-25197.355679	-4.007656

Tabela 4: Delta t, aproximação da solução e ordem de convergência do problema escolhido para a velocidade no eixo x de um dos corpos

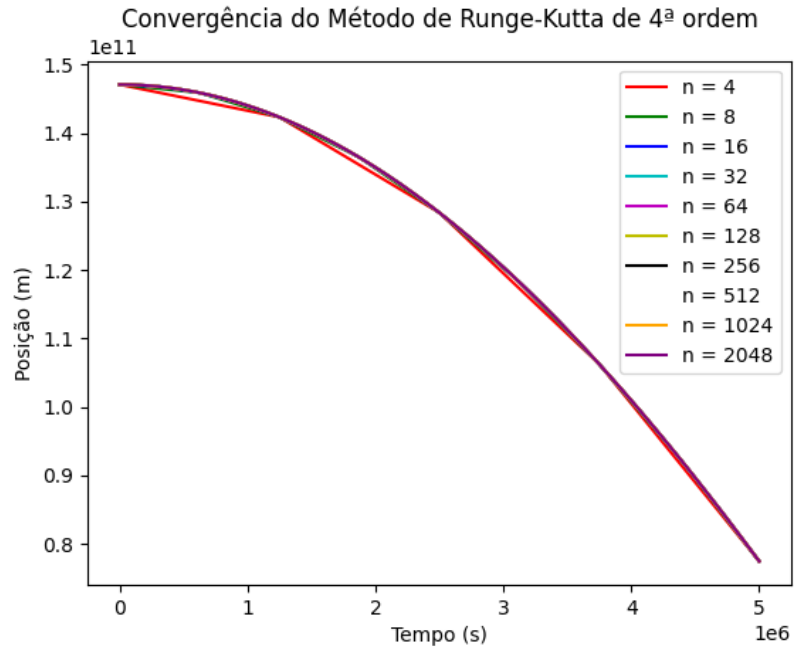


Figura 6: Convergência do problema escolhido para a posição no eixo x de um dos corpos

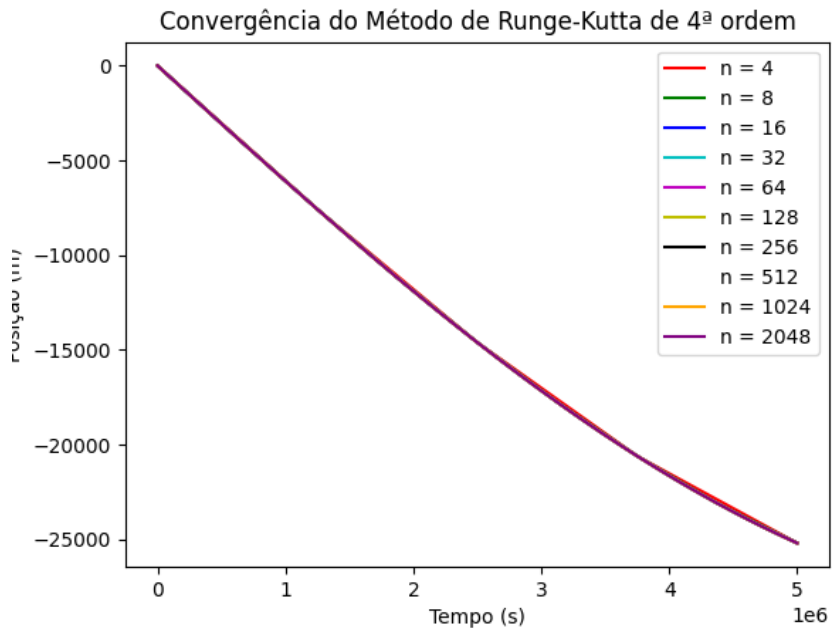


Figura 7: Convergência do problema escolhido para a velocidade no eixo x de um dos corpos

Do que segue, aplicando o método dos Splines Cúbicos Vinculados para a posição e velocidade no eixo x do corpo escolhido, tem-se:

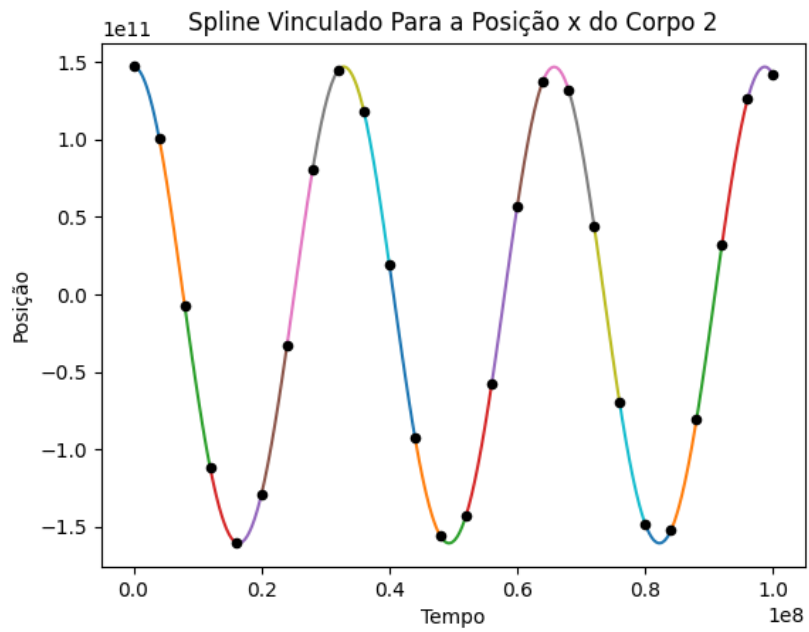


Figura 8: Aplicação do método dos Splines Cúbicos Vinculados para a posição x de um dos corpos

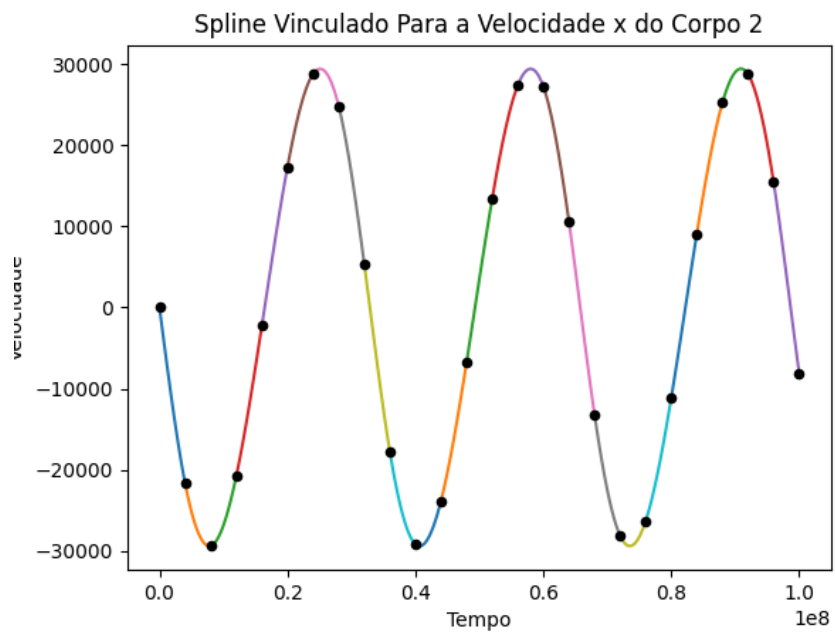


Figura 9: Aplicação do método dos Splines Cúbicos Vinculados para a velocidade no eixo x de um dos corpos

Referências

- [1] Calendário gregoriano: o que é, como surgiu e características. <https://www.significados.com.br/calendario-gregoriano/>. Acessado em 29 de fevereiro de 2024.
- [2] Linear algebra (numpy.linalg). Acessado em: 24 de Março de 2024.
- [3] The spaceguard survey: Report of the nasa international near-earth-object detection workshop. Tech. Report 19920025001, NASA, January 1992.
- [4] BARROW-GREEN, J. The dramatic episode of sundman. *Historia Mathematica* 37 (2010), 164–203.
- [5] BATE, R. R., MUELLER, D. D., AND WHITE, J. E. *Fundamentals of Astrodynamics*. 1971.
- [6] BURDEN, R. L., AND FAIRES, J. D. *Numerical Analysis*, ninth ed. Brooks/Cole, Cengage Learning, 2011.
- [7] MURRAY, C. D., AND DERMOTT, S. F. *Solar System Dynamics*. Cambridge University Press, 2000.
- [8] POLYANIN, A. D., AND ZAITSEV, V. F. *Handbook of Exact Solutions for Ordinary Differential Equations*, 2nd ed. Chapman & Hall/CRC Press, Boca Raton, London, New York, 2003.
- [9] ROMA, A. M., DA SILVA BEVILACQUA, J., AND NÓS, R. L. Métodos para a solução numérica de equações diferenciais ordinárias a valores iniciais, December 27 2023. VERSÃO DE 27 de Dezembro de 2023.
- [10] RUNGE, C. Über empirische funktionen und die interpolation zwischen äquidistanten ordinaten. *Zeitschrift für Mathematik und Physik* 46 (1901), 224–243.
- [11] SZEBEHELY, V., AND PETERS, C. F. Complete solution of a general problem of three bodies. *Astronomical Journal* 72 (September 1967), 876.
- [12] THE VIRGO CONSORTIUM. The Millennium Simulation: A Universe of Galaxies. <https://wwwmpa.mpa-garching.mpg.de/millennium/>, 2006.
- [13] WIKIPEDIA CONTRIBUTORS. N-body problem.