



Análise Matemática II **2021/2022**

Atividade 02 - Métodos Numéricos para resolução de **Sistemas de ED**

Inês Filipa dos Reis Moreira - 2019120254

Pedro Emanuel Dinis Serrano - 2016017926

Índice

1 - Introdução	3
2 - Sistemas de Equações Diferenciais (SED)	4
2.1 - Definição de Sistemas de Equações Diferenciais (SED).....	4
2.2 – Métodos para o Sistema de equações Diferenciais (SED)	4
2.2.1 Método de Euler.....	4
2.2.2 Método de Euler Melhorado	6
2.2.3 Metodo de Runge-Kutta de ordem 2 (RK2).....	11
2.2.4 Metodo de Runge-Kutta de ordem 4 (RK4).....	16
2.2.4 Solução Exata	22
3 – Problemas de Aplicação.....	24
3.1 – Problema do Pêndulo	24
3.2 – Problema Mola-Massa com Amortecimento	27
3.3 – Problema Mola-Massa sem Amortecimento	29
3.4 – Problema Vibração Mecânica	30
3.5 – Problema Circuitos em Série	32
4 - Conclusão	34
5 – Bibliografia	35

1 - Introdução

O presente trabalho é sobre sistemas de equação e consiste na implementação em MATLAB de métodos de resolver sistemas de sistemas de equações diferenciais.

O principal objetivo deste trabalho é redefinir e adptar funções anteriormente implementadas na Atividade01 para a resolução de Sistemas de Equações Diferenciais com condições iniciais.

Está organizado em 2 capítulos. No capítulo 1 será abordado as formulas gerais dos métodos implementados. No capitulo 2 abordamos os problemas a resolver com os métodos.

2 - Sistemas de Equações Diferenciais (SED)

2.1 - Definição de Sistemas de Equações Diferenciais (SED)

É um sistema composto por duas ou mais equações contendo derivadas (que podem não ser necessariamente de primeira ordem) de duas ou mais variáveis dependentes relativamente a uma só variável independente.

2.2 – Métodos para o Sistema de equações Diferenciais (SED)

2.2.1 Método de Euler

Definição – O método de Euler é um procedimento de primeira ordem que visa aproximar a solução da equação diferencial $y' = f(t, y)$ que satisfaz a condição inicial $y(t_0) = y_0$. Neste caso iremos usar o método euler para resolver um sistema de equações.

Fórmula Geral (EDs de 1ª ordem):

$$y_{i+1} = y_i + h f(x_i, y_i)$$

onde:

- y_{i+1} – Próximo valor aproximado da solução do problema original
- y_i – Valor aproximado da solução do problema original na abcissa atual
- h - Valor de cada subintervalo (passo)
- $f(t_i, y_i)$ – Valor de f no ponto (t_i, y_i)

Fórmula Geral Modificada para um Sistema de Equações

$$u_{i+1} = u_i + hf(t_i, u_i, v_i)$$

$$v_{i+1} = v_i + hf(t_i, u_i, v_i)$$

- u_{i+1} – Próxima ordenada da solução aproximada $y(t)$
- v_{i+1} – Próxima ordenada da solução aproximada $y'(t)$
- u_i – Ordenada atual da solução aproximada $y(t)$
- v_i – Ordenada da solução aproximada $y'(t)$
- h - Valor de cada subintervalo (passo)
- $f(t_i, u_i, v_i)$ – valor de f no ponto (t_i, u_i, v_i)
- $g(t_i, u_i, v_i)$ – valor de g no ponto (t_i, u_i, v_i)

Algoritmo:

1. Definir o valor de h ;
2. Criar um vetor u
3. Atribuir o primeiro valor de u e de v para guardar as soluções e atribuir
 $u_1 = u_0$ e $v_1 = v_0$;
4. Para i de 1 a n , fazemos o cálculo do método de Euler para i -ésima
iteração no vetor u e no vetor v

```
function y = N_Euler(f,a,b,n,y0)
    h = (b-a)/n;

    t = a:h:b;
    y = zeros(1, n+1);

    y(1) = y0;

    for i=1:n
        y(i+1)=y(i)+h*f(t(i),y(i));
    end
end
```

2.2.2 Método de Euler Melhorado

O Método de Euler Melhorado é semelhante ao Método de Euler Tradicional mas tem uma diferença.

Este método utiliza uma média de inclinações em cada ponto para cada iteração, dando desta forma uma precisão maior.

Fórmula Geral (Eds de 1ª ordem)

$$y_{i+1} = y_i + \frac{h}{2}(k_1 + k_2), i = 0, 1, 2, \dots, n-1$$

- y_{i+1} – Próximo valor aproximado da solução do problema original (na abscissa $t_i + 1$);
- y_i – Valor aproximado da solução do problema original na abscissa atual;
- h - Valor de cada subintervalo (passo);
- k_1 – Inclinação no início do intervalo;
- k_2 – Inclinação no fim do intervalo.

Calculo de k_1

$$k_1 = f(t_i, u_i)$$

- k_1 – Inclinação no início do intervalo;
- $f(t_i, y_i)$ – Valor de f no ponto (t_i, y_i)

Calculo de k_2

$$k_2 = h * f(t_{i+1}, y_i, k_1)$$

- k_2 – Inclinação no fim do intervalo;
- $t_i + 1$ – Proxima abcissa do intervalo escolhido
- h - Tamanho de cada subintervalo (passo);
- y_i – Valor aproximado da solução do problema original na abcissa atual
- k_i – Inclinação no inicio do intervalo

Fórmula Geral Modificada para um Sistema de Equações

$$u_{i+1} = u_i + h * u_k$$

$$v_{i+1} = v_i + h * v_k$$

- u_{i+1} – Aproximação do método de Heun para a i ésima iteração;
- v_{i+1} – Aproximação do método de Heun para a i ésima iteração;
- u_i – Ordenada atual da função aproximada $y(t)$;
- v_i - Ordenada atual da função aproximada $y'(t)$;
- h - Valor de cada subintervalo;
- u_k – Cálculo da média das inclinações;
- v_k – Cálculo da média das inclinações.

Calculo de u_k

$$u_k = \frac{1}{2} * (u_{k1} + u_{k2})$$

- u_k – Calculo da media das iterações.
- u_{k1} – Inclinação no início do intervalo;

- u_{k2} - Inclinação no fim do intervalo.

Calculo de v_k

$$v_k = \frac{1}{2} * (v_{k1} + v_{k2})$$

- v_k – Calculo da media das iterações.
- v_{k1} – Inclinação no início do intervalo;
- v_{k2} - Inclinação no fim do intervalo.

Calculo de u_{k1}

$$u_{k1} = f(t_i, u_i, v_i)$$

- u_{k1} - Inclinação no inicio do intervalo
- $f(t_i, u_i, v_i)$ – Valor de f no ponto (t_i, u_i, v_i)

Calculo de v_{k1}

$$u_{k1} = f(t_i, u_i, v_i)$$

- v_{k1} - Inclinação no inicio do intervalo
- $g(t_i, u_i, v_i)$ – Valor de g no ponto (t_i, u_i, v_i)

Calculo de u_{k2}

$$u_{k2} = f(t_{i+1}, u_i + u_{k1} * h, v_i + v_{k1} * h)$$

- u_{k2} - Inclinação no fim do intervalo
- t_{i+1} – Proxima abcissa do intervalo escolhido;
- u_{k1} – Inclinação no inicio do intervalo;

- h - Tamanho de cada subintervalo
- u_i – Ordenada atual da solução aproximada $y(t)$
- v_i – Ordenada atual da solução aproximada $y'(t)$
- v_{k1} – Inclinação no início do intervalo
- $f(t_i, u_i, v_i)$ – Valor de f no ponto (t_i, u_i, v_i) .

Calculo de v_{k2}

$$v_{k2} = g(t_{i+1}, u_i + u_{k1} * h, v_i + v_{k1} * h)$$

- v_{k2} - Inclinação no fim do intervalo
- t_{i+1} – Proxima abcissa do intervalo escolhido;
- u_{k1} – Inclinação no inicio do intervalo;
- h - Tamanho de cada subintervalo
- u_i – Ordenada atual da solução aproximada $y(t)$
- v_i – Ordenada atual da solução aproximada $y'(t)$
- v_{k1} – Inclinação no início do intervalo
- $g(t_i, u_i, v_i)$ – Valor de g no ponto (t_i, u_i, v_i)

Algortimo

1. Definir o passo h ;
2. Criar um vetor u e um vetor v para guardar as soluções e
atribuir $u_1 = u_0$ e $v_1 = v_0$
3. Atribuir o primeiro valor de u e de v ;

4. Cálculo da inclinação no início do intervalo;
5. Cálculo da inclinação no fim do intervalo;
6. Cálculo da média das inclinações;
7. Cálculo do valor aproximado para a i ésima iteração.

```
function y = N_EulerM(f,a,b,n,y0)
    h = (b-a)/n;

    t = a:h:b;
    y = zeros(1, n+1);

    y(1) = y0;

    for i=1:n
        k1 = f(t(i),y(i));
        k2 = f(t(i+1), y(i) + k1*h);
        k = 0.5*(k1+k2);
        y(i+1)=y(i)+h*k;
    end
end
```

2.2.3 Metodo de Runge-Kutta de ordem 2 (RK2)

Definição

O RK2 é um metodo simples que requer apenas derivadas de 1ª ordem e pode fornecer aproximações precisas.

Fórmula Geral (Eds 1ª ordem)

$$y_{i+1} = y_i + \frac{1}{2} (k_1 + k_2), i = 0, 1, 2, \dots, n - 1$$

- y_{i+1} – Proximo valor aproximado da solução do problema original
- y_i – Valor aproximado da solução do problema original na abcissa atual
- k_1 – Inclinação no inicio do intervalo
- k_2 – Inclinação no fim do intervalo.

Cálculo de k_1

$$k_1 = h * f(t_i, y_i)$$

- k_1 – Inclinação no inicio do intervalo
- h - Valor de cada subintervalo
- $f(t_i, y_i)$ – Valor de f no ponto (t_i, y_i)

Cálculo de k_2

$$k_2 = h * f(t_{i+1}, y_i + k_1)$$

- k_2 – Inclinação no fim do intervalo;
- t_{i+1} – Abcissa atual do intervalo;
- h - Tamanho de cada subintervalo;
- y_i – Valor aproximado da solução do problema original na abcissa atual;
- k_1 – Inclinação no início do intervalo;
- $f(t_i, y_i)$ – Valor de f no ponto (t_i, y_i) .

Fórmula Geral modificada para um sistema de equações

$$u_{i+1} = u_i + u_k$$

$$v_{i+1} = v_i + v_k$$

- u_{i+1} – Aproximação do método de RK2 para iésima iteração;
- v_{i+1} - Aproximação do método de RK2 para iésima iteração;
- u_i – Ordenada atual da solução aproximada $y(t)$;
- v_i – Ordenada atual da solução aproximada $y'(t)$;
- u_k – Calculo da média das inclinações;
- v_k – Calculo da média das inclinações.
- **Calculo de u_k**

$$u_k = \frac{u_{k1} + u_{k2}}{2}$$

- u_k – Calculo da media das iterações.
- u_{k1} – Inclinação no início do intervalo;
- u_{k2} - Inclinação no fim do intervalo.

Cálculo de v_k

$$u_k = \frac{u_{k1} + u_{k2}}{2}$$

- v_k – Calculo da media das iterações.
- v_{k1} – Inclinação no início do intervalo;
- v_{k2} - Inclinação no fim do intervalo.

Cálculo de u_{k1}

$$u_{k1} = h * f(t_i, u_i, v_i)$$

- u_{k1} – Inclinação no inicio do intervalo
- h - Tamanho do subintervalo
- $f(t_i, u_i, v_i)$ – Valor de f no ponto (t_i, u_i, v_i)

Cálculo de v_{k1}

$$v_{k1} = h * g(t_i, u_i, v_i)$$

- v_{k1} – Inclinação no inicio do intervalo
- h - Tamanho do subintervalo
- t_i – Abcissa atual do intervalo escolhido
- u_i – Ordenada atual da solução aproximada $y(t)$
- v_i – Ordenada atual da solução aproximada
- $g(t_i, u_i, v_i)$ – Valor de f no ponto (t_i, u_i, v_i)

Cáculo de u_{k2}

$$u_{k2} = h * f(t_{i+1}, u_i + u_{k1}, v_i + v_{k1})$$

- u_{k2} – Inclinação no fim do intervalo
- h - Tamanho dos subintervalos
- t_{i+1} – Proxima abcissa do intervalo escolhido
- u_{k1} – Inclinação no inicio do intervalo
- u_i – Ordenada atual da solução aproximada $y(t)$
- v_i – Ordenada atual da solução aproximada $y'(t)$
- v_{k1} – Inclinação no inicio do intervalo

Cálculo de v_{k2}

$$v_{k2} = h * g(t_{i+1}, u_i + u_{k1}, v_i + v_{k1})$$

1. v_{k2} – Inclinação no fim do intervalo
2. h - Tamanho dos subintervalos
3. t_{i+1} – Proxima abcissa do intervalo escolhido
4. u_{k1} – Inclinação no inicio do intervalo
5. v_{k1} - Inclinação no inicio do intervalo
6. u_i – Ordenada atual da solução aproximada $y(t)$
7. v_i – Ordenada atual da solução aproximada $y'(t)$

Algoritmo

1. Definir o subintervalo, h ;
2. Criar um vetor u e um vetor v para guardar as soluções e atribuir $u_1 = u_0$
e $v_1 = v_0$;

3. Atribuir o primeiro valor de u e de v ;
4. Cálculo da inclinação no início do intervalo;
5. Cálculo da inclinação no fim do intervalo;
6. Cálculo do valor aproximado para i ésima iteração.

```
function [t,u,v] = NRK2SED(f,g,a,b,n,u0,v0)
%NRK2SED Método de Runge-Kutta de ordem 2 para um Sistema de SED/PVI
% Detailed explanation goes here
% 22/04/2021 Arménio Correia armenioc@isec.pt

h = (b-a)/n;
t = a:h:b;
u = zeros(1,n+1);
v = zeros(1,n+1);
u(1) = u0;
v(1) = v0;
for i = 1:n
    k1u = h*f(t(i),u(i),v(i));
    k1v = h*g(t(i),u(i),v(i));
    k2u = h*f(t(i+1),u(i)+k1u,v(i)+k1v);
    k2v = h*g(t(i+1),u(i)+k1u,v(i)+k1v);
    u(i+1) = u(i)+(k1u+k2u)/2;
    v(i+1) = v(i)+(k1v+k2v)/2;
end
end
```

2.2.4 Metodo de Runge-Kutta de ordem 4 (RK4)

Definição

É o método numerico com maior precisão. A sua fórmula considera para cada iteração 4 valores denominados normalmente por "k" onde o *primeiro* é a inclinação no início do intervalo, o *segundo* é a inclinação no ponto médio usando a primeira inclinação, o *terceiro* é novamente a inclinação no ponto médio do intervalo mas, desta vez, utilizando a segunda inclinação e, finalmente, o *quarto* é a inclinação no final do intervalo. Este método não necessita do cálculo de qualquer derivada de f, mas depende de outra função que é definida avaliando f em diferentes pontos.

Formula Geral (EDs 1ª ordem)

$$y_{i+1} = y_i + \frac{h}{6} (k_1 + 2k_2 + 2k_3 + k_4), i = 0, 1, \dots$$

- y_{i+1} – Aproximação pelo método RK4 de $y(x_n + 1)$
- y_i – Valor de y na iesima iteração;
- h - Valor dos subintervalos.

$$k_1 = f(x_i, y_i)$$

$$k_2 = f\left(x_i + \frac{h}{2}, y_i + \frac{h}{2}k_1\right)$$

$$k_3 = f\left(x_i + \frac{h}{2}, y_i + \frac{h}{2}k_2\right)$$

$$k_4 = f(x_i + h, y_i + h k_3)$$

- k_1 – Inclinação no início do intervalo;
- k_2 – Inclinação no ponto médio do intervalo;
- k_3 – Inclinação no ponto medio do intervalo;
- k_4 – Inclinação no fim do intervalo.

Média ponderada das inclinações: $\frac{k_1 + 2k_2 + 2k_3 + k_4}{6}$

Fórmula Geral modificada para um Sistema de Equações

$$u_{i+1} = u_i + u_k$$

$$v_{i+1} = v_i + v_k$$

- u_{i+1} - Aproximação do método de RK4 para a iésima iteração;
- v_{i+1} – Aproximação do método de RK4 para a iésima iteração;
- u_i – Ordenada atual da solução aproximada $y(t)$;
- v_i – Ordenada atual da solução aproximada $y'(t)$;
- u_k – Cálculo da média das inclinações;
- v_k – Cálculo da média das inclinações.

Cálculo de u_k

$$u_k = \frac{(u_{k1} + 2 * u_{k2} + 2 * u_{k3} + u_{k4})}{6}$$

- u_k - Cálculo da média das inclinações
- u_{k1} – Inclinação no início do intervalo
- u_{k2} - Inclinação no ponto médio do intervalo;
- u_{k3} – Inclinação no ponto medio do intervalo;
- u_{k4} – Inclinação no fim do intervalo;

Cálculo de v_k

$$v_k = \frac{(v_{k1} + 2 * v_{k2} + 2 * v_{k3} + v_{k4})}{6}$$

- v_k – Cálculo da média das inclinações;
- v_{k1} – Inclinação no início do intervalo;
- v_{k2} – Inclinação no ponto médio do intervalo;
- v_{k3} – Inclinação no ponto médio do intervalo;
- v_{k4} – Inclinação no fim do intervalo.

Cálculo de u_{k1}

$$u_{k1} = h * f(t_i, u_i, v_i)$$

- u_{k1} – Inclinação no início do intervalo;
- h - Tamanho de cada subintervalo
- $f(t_i, u_i, v_i)$ – Valor de f no ponto (t_i, u_i, v_i) .

Cálculo de v_{k1}

$$v_{k1} = g * f(t_i, u_i, v_i)$$

- v_{k1} – Inclinação no início do intervalo;
- h - Tamanho de cada subintervalo
- $g(t_i, u_i, v_i)$ – Valor de f no ponto (t_i, u_i, v_i) .
-

Cálculo de u_{k2} , u_{k3} , u_{k4}

$$u_{k2} = h * f(t_i + \frac{h}{2}, u_i + 0.5 * u_{k1}, v_i + 0.5 * v_{k1})$$

$$u_{k3} = h * f(t_i + \frac{h}{2}, u_i + 0.5 * u_{k2}, v_i + 0.5 * v_{k2})$$

$$u_{k4} = h * f(t_i + \frac{h}{2}, u_i + 0.5 * u_{k3}, v_i + 0.5 * v_{k3})$$

- u_{k2} – Inclinação no ponto médio do intervalo;
- u_{k3} – Inclinação no ponto médio do intervalo;
- u_{k4} – Inclinação no fim do intervalo;
- h - Tamanho de cada subintervalo;
- t_i – Abcissa atual o intervalo escolhido;
- u_{k1} – Inclinação no início do intervalo;
- u_1 – Ordenada atual da solução aproximada $y(t)$;
- v_i – Ordenada atual da solução aproximada $y'(t)$;
- v_{k1} – Inclinação no início do intervalo.

Cálculo de v_{k2} , v_{k3} , v_{k4}

$$v_{k2} = h * g(t_i + \frac{h}{2}, u_i + 0.5 * u_{k1}, v_i + 0.5 * v_{k1})$$

$$v_{k3} = h * g(t_i + \frac{h}{2}, u_i + 0.5 * u_{k2}, v_i + 0.5 * v_{k2})$$

$$v_{k4} = h * g(t_i + \frac{h}{2}, u_i + 0.5 * u_{k3}, v_i + 0.5 * v_{k3})$$

- v_{k2} – Inclinação no ponto médio do intervalo;
- v_{k3} – Inclinação no ponto médio do intervalo;
- v_{k4} – Inclinação no fim do intervalo;
- h - Tamanho de cada subintervalo;
- t_i – Abcissa atual o intervalo escolhido;
- u_{k1} – Inclinação no início do intervalo;
- v_{k1} – Inclinação no início do intervalo;
- u_1 – Ordenada atual da solução aproximada $y(t)$;
- v_i – Ordenada atual da solução aproximada $y'(t)$.

Algoritmo

1. Definir o subintervalo, h ;
2. Criar um vetor u e um vetor v para guardar as soluções e atribuir $u_1 = u_0$ e $v_1 = v_0$;
3. Atribuir o primeiro valor de u e de v ;
4. Calculo do, k_1 – inclinação no início do intervalo;
5. Calculo do, k_2 – inclinação no ponto médio do intervalo;
6. Calculo do, k_3 – inclinação no ponto médio do intervalo, novamente;
7. Cálculo do, k_4 – inclinação no fim do intervalo.

```

function [t,u,v] = NRK4SED(f,g,a,b,n,u0,v0)
    h = (b-a)/n;

    t = a:h:b;
    u = zeros(1, n+1);
    v = zeros(1, n+1);

    u(1) = u0;
    v(1) = v0;

    for i=1:n
        uK1 = h*f(t(i), u(i), v(i));
        vK1 = h*g(t(i), u(i), v(i));

        uK2 = h*f(t(i) + h/2, u(i) + 0.5*uK1, v(i) + 0.5*vK1);
        vK2 = h*g(t(i) + h/2, u(i) + 0.5*uK1, v(i) + 0.5*vK1);

        uK3 = h*f(t(i) + h/2, u(i) + 0.5*uK2, v(i) + 0.5*vK2);
        vK3 = h*g(t(i) + h/2, u(i) + 0.5*uK2, v(i) + 0.5*vK2);

        uK4 = h*f(t(i+1), u(i) + uK3, v(i) + vK3);
        vK4 = h*g(t(i+1), u(i) + uK3, v(i) + vK3);

        uK = (uK1 + 2*uK2 + 2*uK3 + uK4)/6;
        vK = (vK1 + 2*vK2 + 2*vK3 + vK4)/6;

        u(i + 1) = u(i) + uK;
        v(i + 1) = v(i) + vK;
    end
end

```

2.2.4 Solução Exata

Definição

A solução exata, diz concretamente o resultado que um certo exercício retorna, uma vez que é uma solução exata, esta é muitas vezes usada como modo de comparação com os outros métodos para tentar perceber qual deles é mais eficaz e preciso.

```

function u = sExataSED(f,g,u0,v0)

    syms t u(t) v(t)

    ci=[u(0)==u0,v(0)==v0];

    eqns=[diff(u,t)==f(t,u,v),diff(v,t)==g(t,u,v)];

    [u(t), ~] = dsolve(eqns,ci);

    if isempty(u(t))
        u = 0;
    end
end

```

3 – Problemas de Aplicação

3.1 – Problema do Pêndulo

Example 13-A Motion of a Nonlinear Pendulum

The motion of a pendulum of length L , subject to damping can be described by the angular displacement of the pendulum from the vertical, θ , as a function of time. (See Fig. 13.1.) If we let m be the mass of the pendulum, g the gravitational constant, and c the damping coefficient (i.e., the damping force is $F = -c\theta'$), then the ODE initial-value problem describing this motion is

$$\theta'' + \frac{c}{mL} \theta' + \frac{g}{L} \sin \theta = 0.$$

The initial conditions give the angular displacement and velocity at time zero; for example, if $\theta(0) = a$ and $\theta'(0) = 0$, the pendulum has an initial displacement, but is released with 0 initial velocity.

Analytic (closed-form) solutions rely on approximating $\sin \theta$; the exact solutions to this approximated system do not have the characteristics of the physical pendulum, namely, a decreasing amplitude and a decreasing period. (See Greenspan, 1974, for further discussion.)



FIGURE 13.1a Simple pendulum.

Com o programa criado em MATLAB, conseguimos resolver o problema especificado a cima de maneira simples.

Sendo este problema uma equação diferencial de ordem 2 não linear, não irá ser apresentado uma solução exata.

Resolução

Neste problema, resolvido em aula, considerámos o seguinte:

$$\frac{C}{ml} = 0.3 \quad \frac{C}{L} = 1 \quad t \in [0, 15] \quad \theta = y$$

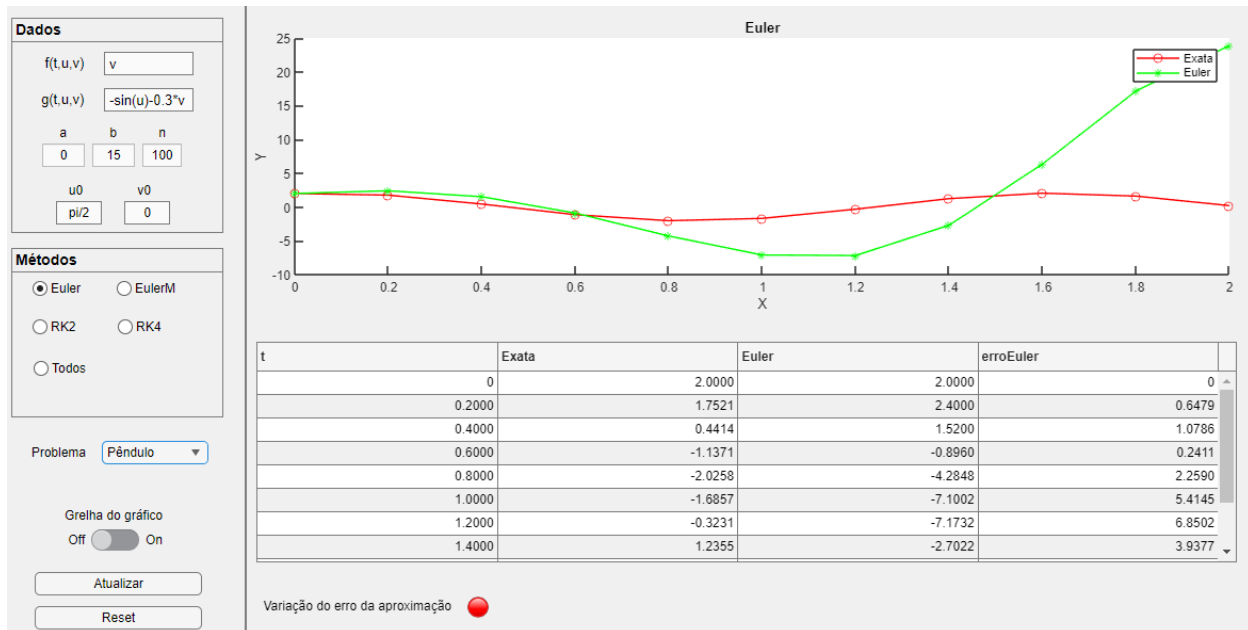
Como o pêndulo é largado com velocidade nula, no ponto em que a corda está perpendicular ao seu suporte, temos as condições iniciais:

$$y(0) = \frac{\pi}{2} \quad y'(0) = 0$$

Desta forma, temos a seguinte equação diferencial: $y'' + 0.3y' + \sin(y) = 0$

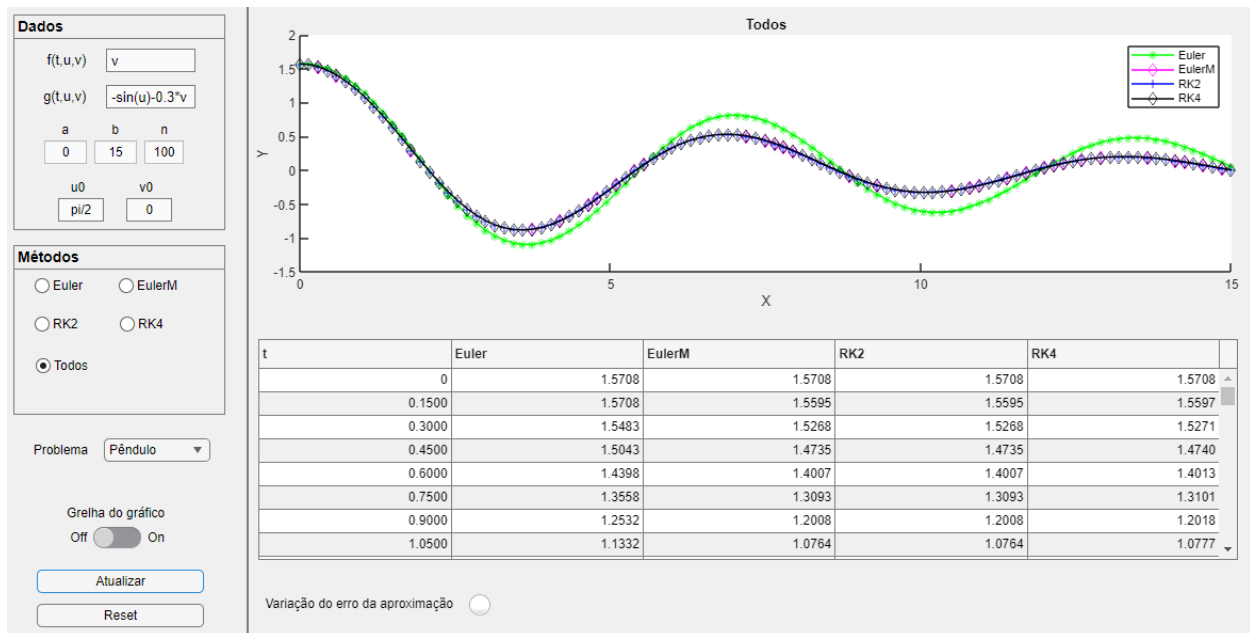
Agora podemos aplicar o algoritmo de resolução e temos o sistema:

$$\left\{ \begin{array}{l} u' = v \\ v' = -\sin(u) - 0.3v \\ t \in [0, 15] \\ u(0) = \frac{\pi}{2} \\ v(0) = 0 \end{array} \right.$$



O programa apresenta a função dada no enunciado, os parâmetros de entrada, os métodos que irão resolver o problema, um gráfico e uma tabela.

Usando todos os métodos, teremos uma comparação entre eles, podemos ver a precisão de cada um para resolver este problema.



3.2 – Problema Mola-Massa com Amortecimento

c) Um peso de 6.4 lb provoca, numa mola, um alongamento de 1.28 ft . O sistema está sujeito à acção duma força amortecedora, numericamente igual ao dobro da sua velocidade instantânea. Determine a equação do movimento do peso, supondo que ele parte da posição de equilíbrio com uma velocidade dirigida para cima de 4 ft/s .

Resolução:

Sabe-se, pela lei de Hooke, que $W = ks$

No caso em estudo $k = \frac{6.4}{1.28} \Leftrightarrow k = 5 \text{ lb/ft}$. Como $W = mg$, tem-se $m = \frac{6.4}{32} \Leftrightarrow m = 0.2$

A equação que decreve o movimento livre amortecido é

$$m \frac{d^2x}{dt^2} = -Kx - b \frac{dx}{dt}$$

onde b é uma constante positiva e o sinal "-" indica que as forças amortecedoras actuam na direcção oposta ao movimento.

Então a equação diferencial de movimento de peso é $0.2x'' = -5x - 2x'$

$\Leftrightarrow x'' + 10x' + 25x = 0$ com $x(0) = 0$ e $x'(0) = -4$

Considerando o enunciado e $x(y) = y(t)$, temos a equação diferencial:

$$y'' + 10y' + 25y = 0$$

E as condições iniciais:

$$y(0) = 0 \quad y'(0) = -4$$

E aplicando os algoritmo de resolução temos o sistema

$$\left\{ \begin{array}{l} u' = v \\ v' = -10v - 25v \\ t \in [0, 2] \\ u(0) = 0 \\ v(0) = -4 \end{array} \right.$$

Obtemos a seguinte informação na GUI:

Dados

$f(t,u,v)$

$g(t,u,v)$

a b n

u0 v0

EulerM

Métodos

☐ Euler ☒ EulerM

☐ RK2 ☐ RK4

☐ Todos

t	Exata	EulerM	erroEulerM
0	0	0	0
0.0200	-0.0724	-0.0720	0.0004
0.0400	-0.1310	-0.1303	0.0007
0.0600	-0.1778	-0.1769	0.0009
0.0800	-0.2145	-0.2135	0.0010
0.1000	-0.2426	-0.2415	0.0011
0.1200	-0.2634	-0.2623	0.0012
0.1400	-0.2781	-0.2769	0.0012

Problema Mola-Mas...

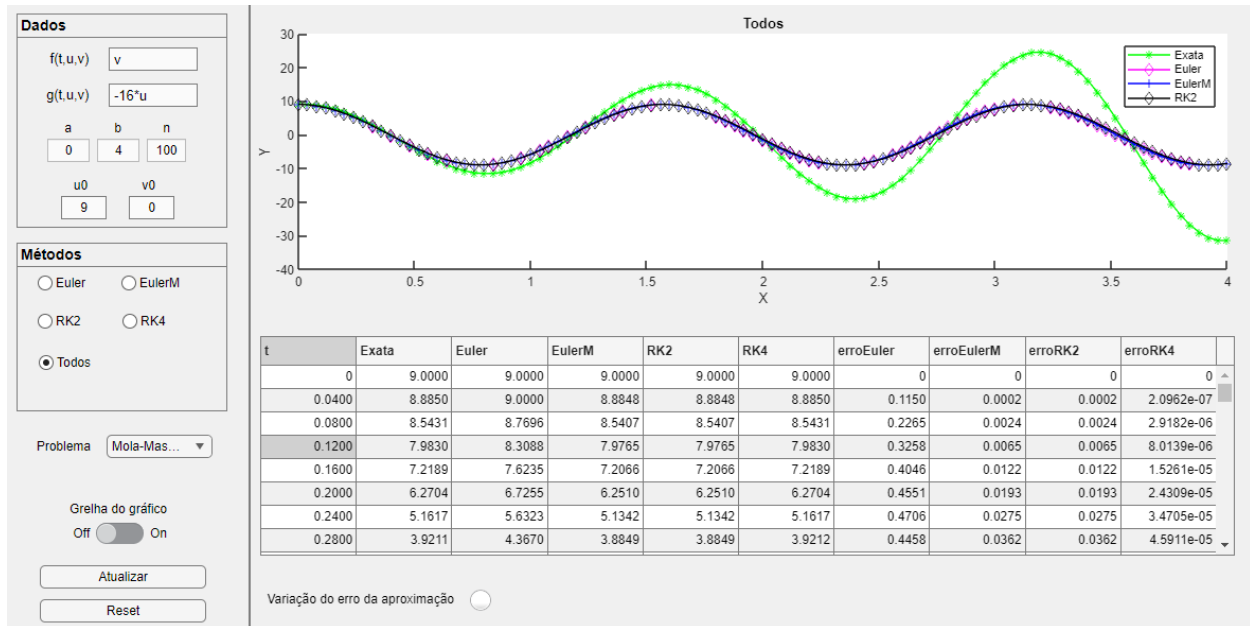
Grelha do gráfico ☐ Off ☒ On

Atualizar Reset

Varição do erro da aproximação ●

3.3 – Problema Mola-Massa sem Amortecimento

Será o mesmo que o anterior mas uma condição importante não existe amortecimento.



3.4 – Problema Vibração Mecânica

Modelos vibratórios mecânicos

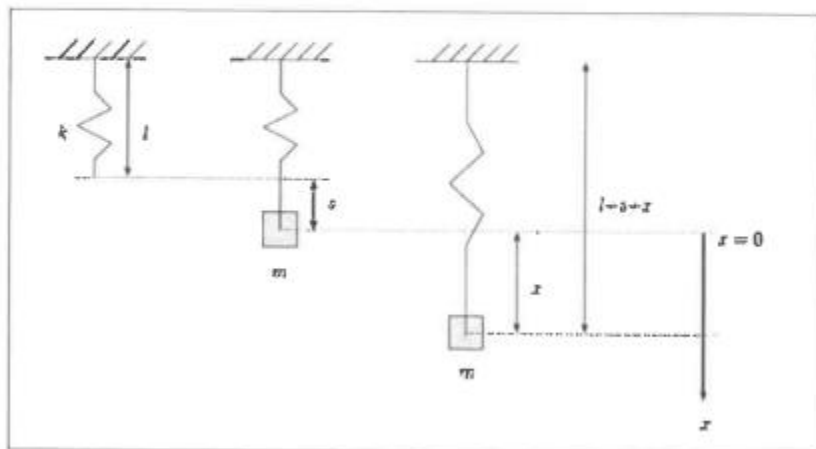
Nestes sistemas, o deslocamento x obedece à equação diferencial linear de 2ª ordem

$$mx'' + bx' + k(x) = f(t)$$

onde:

m = massa; x = deslocamento; b = factor de amortecimento;

k = constante da mola e $f(t)$ = força aplicada



Considerando o enunciado $x(t) = y(t)$, temos a seguinte ED:

$$y'' + 2y' + 2y - 4 \cos(t) + 2 \sin(t) = 0$$

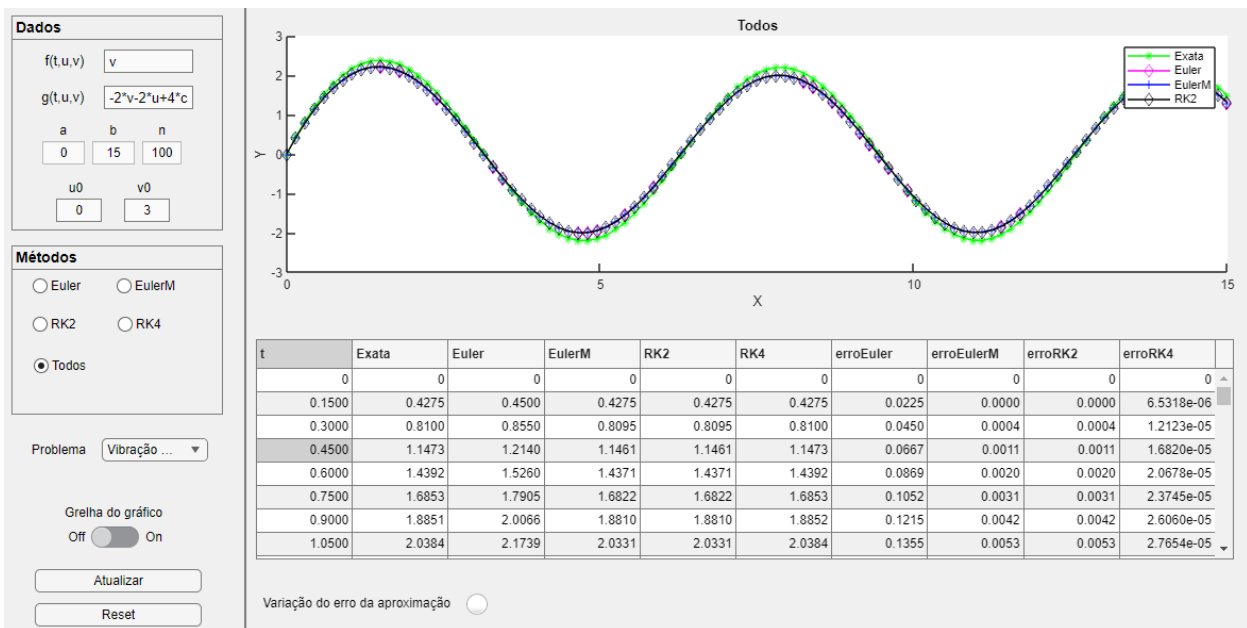
Com condições iniciais:

$$y(0) = 0 \quad y'(0) = 3$$

E com o algoritmo de resolução temos o seguinte sistema de equações:

$$\begin{cases} u' = v \\ v' = -16u \\ t \in [0, 10] \\ u(0) = 9 \\ v(0) = 0 \end{cases}$$

Temos na GUI o seguinte:



3.5 – Problema Circuitos em Série

- Circuito eléctrico em série

$$Lq'' + rq' + \frac{1}{c}q = e(t) \quad (*)$$

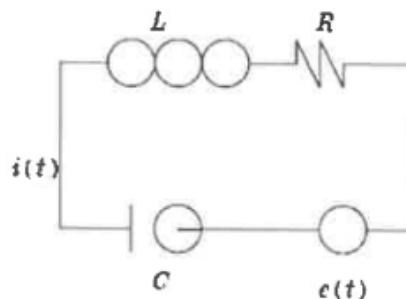
L – Indutância

q – carga

R – Resistência

C – capacidade

$e(t)$ – força electromotriz



Pelas leis de Kirchhoff, num circuito indutivo-resistivo-capacitivo (L - R - C) série, em que a corrente varia com o tempo, a carga q acumulada no condensador é dada pela equação diferencial linear de 2ª Ordem. (*)

Considreando o enunciado temos a seguinte equação diferencial:

$$y'' + 5y' + 4y - 2e^{-\frac{t}{4}} = 0$$

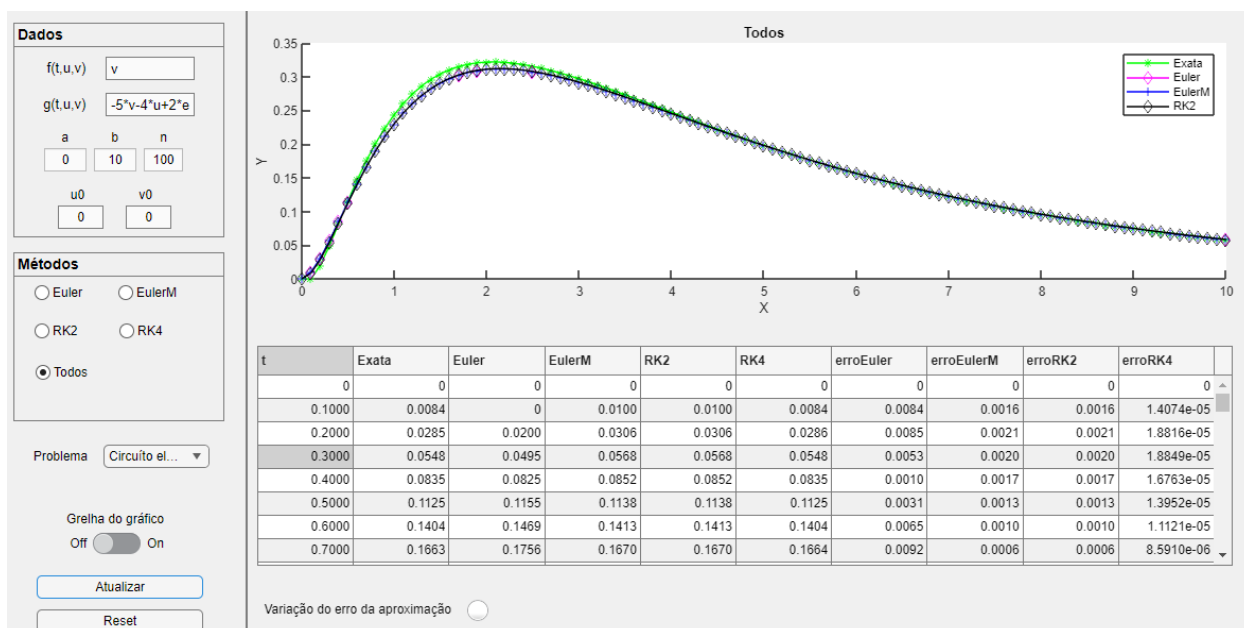
Considerando que o circuito se encontra desligado no instante $t = 0$,

Temos : $y(0) = 0$ $y' = 0$

Temos o seguinte sistema:

$$\left\{ \begin{array}{l} u' = v \\ v' = -5v - 2e^{-\frac{t}{4}} \\ t \in [0, 10] \\ u(0) = 0 \\ v(0) = 0 \end{array} \right.$$

Assim temos no GUI o seguinte:



4 - Conclusão

Neste trabalho abordamos o assunto dos métodos numéricos existentes como por exemplo método de Euler, método de Euler Melhorado, RK2, RK4, e concluímos que estes métodos possibilitam a resolução de problemas com Equações diferenciais de ordem 2.

Este trabalho foi importante para o nosso aprofundamento deste tema visto que permitiu-nos desenvolver por nós próprios uma aplicação em MATLAB e aplicar os métodos numéricos dados em aula.

5 – Bibliografia

https://pt.wikipedia.org/wiki/M%C3%A9todo_de_Euler

https://pt.wikipedia.org/wiki/M%C3%A9todo_de_Runge-Kutta