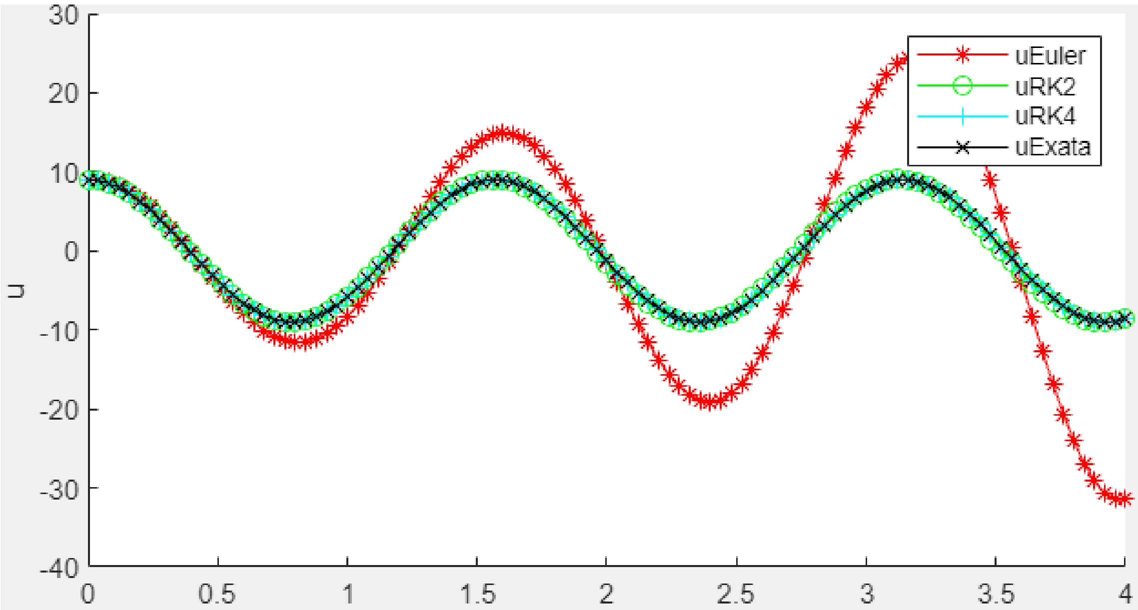
Métodos Numéricos para Sistemas de Equações Diferenciais



Análise Matemática II – Atividade 04 Trabalho

Renato Alexandre Oliveira Craveiro | 2018011392

Ano Letivo 2022/2023 – ISEC – Lic. Engenharia Informática – Regime Pós-Laboral

Índice

1. Introdução

Este trabalho tem como objetivo aplicar Métodos de resolução/aproximação de Sistemas de Equações Diferenciais através de *software* desenvolvido pelos alunos na plataforma MATLAB, por forma a aprofundar os conhecimentos acerca dos diversos métodos e de como desenvolvê-los em MATLAB.

1.1 Sistema de Equações diferenciais: Definição

Um sistema de equações diferenciais é um conjunto de equações diferenciais que envolve duas ou mais funções desconhecidas e as suas respetivas derivadas.

Um sistema de equações diferenciais envolve várias variáveis dependentes e as suas respetivas taxas de variação. Cada equação representa a taxa de variação de uma das funções desconhecidas em relação a alguma variável independente.

Resolver um sistema de equações diferenciais significa encontrar uma expressão analítica para cada uma das funções desconhecidas, para que essas expressões satisfaçam simultaneamente todas as equações do sistema.

Para isso são utilizados métodos analíticos, numéricos ou uma combinação de ambos, dependendo da complexidade do sistema e da natureza do problema em questão.

2. Métodos Numéricos para resolução de PVI

2.1 Método de Euler

O método de Euler é um método simples para resolver problemas de valor inicial (PVI) de equações diferenciais ordinárias (EDO). Ele envolve dividir o intervalo de interesse em um número finito de subintervalos, estimar o valor da solução em cada subintervalo usando a EDO e a largura do subintervalo e repetir esse processo até chegar ao final do intervalo. Embora seja fácil de entender e implementar, ele nem sempre é o método mais preciso ou eficiente.

Em MATLAB:

```
function [t,u,v]= NEulerSED(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
    u(i+1)=u(i)+h*f(t(i),u(i),v(i));
    v(i+1)=v(i)+h*g(t(i),u(i),v(i));
end

end
```

2.2 Método de RK2

O método de Runge-Kutta é um conjunto de algoritmos para resolver problemas de valor inicial (PVI) de equações diferenciais ordinárias (EDO). O método de Runge-Kutta de ordem n (RK-n) envolve o uso de n estimativas intermediárias da derivada da solução em pontos específicos, para calcular uma estimativa da solução no próximo ponto no tempo.

O método de RK2 é um método de segunda ordem que calcula a solução aproximada da EDO usando duas estimativas intermediárias da derivada.

```
function [t, u, v] = NRK2SED(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
        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) + (1/2)*(k1u + k2u);
        v(i+1) = v(i) + (1/2)*(k1v + k2v);
    end
end
```

2.3 Método de RK4

O método de RK4 é o método de quarta ordem do método de Runge-Kutta, que calcula a solução aproximada da EDO usando quatro estimativas intermediárias da derivada.

Em MATLAB:

```
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
        k1u = h * f(t(i), u(i), v(i));
        k1v = h * g(t(i), u(i), v(i));

        k2u = h * f(t(i)+(h/2), u(i)+(k1u/2), v(i)+(k1v/2));
        k2v = h * g(t(i)+(h/2), u(i)+(k1u/2), v(i)+(k1v/2));

        k3u = h * f(t(i)+(h/2), u(i)+(k2u/2), v(i)+(k2v/2));
        k3v = h * g(t(i)+(h/2), u(i)+(k2u/2), v(i)+(k2v/2));

        k4u = h * f(t(i)+h, u(i) + k3u, v(i) + k3v);
        k4v = h * g(t(i)+h, u(i) + k3u, v(i) + k3v);

        u(i+1) = u(i) + (1/6)*(k1u + 2*k2u + 2*k3u + k4u);
        v(i+1) = v(i) + (1/6)*(k1v + 2*k2v + 2*k3v + k4v);
    end
end
```

2.4 Solução Exata

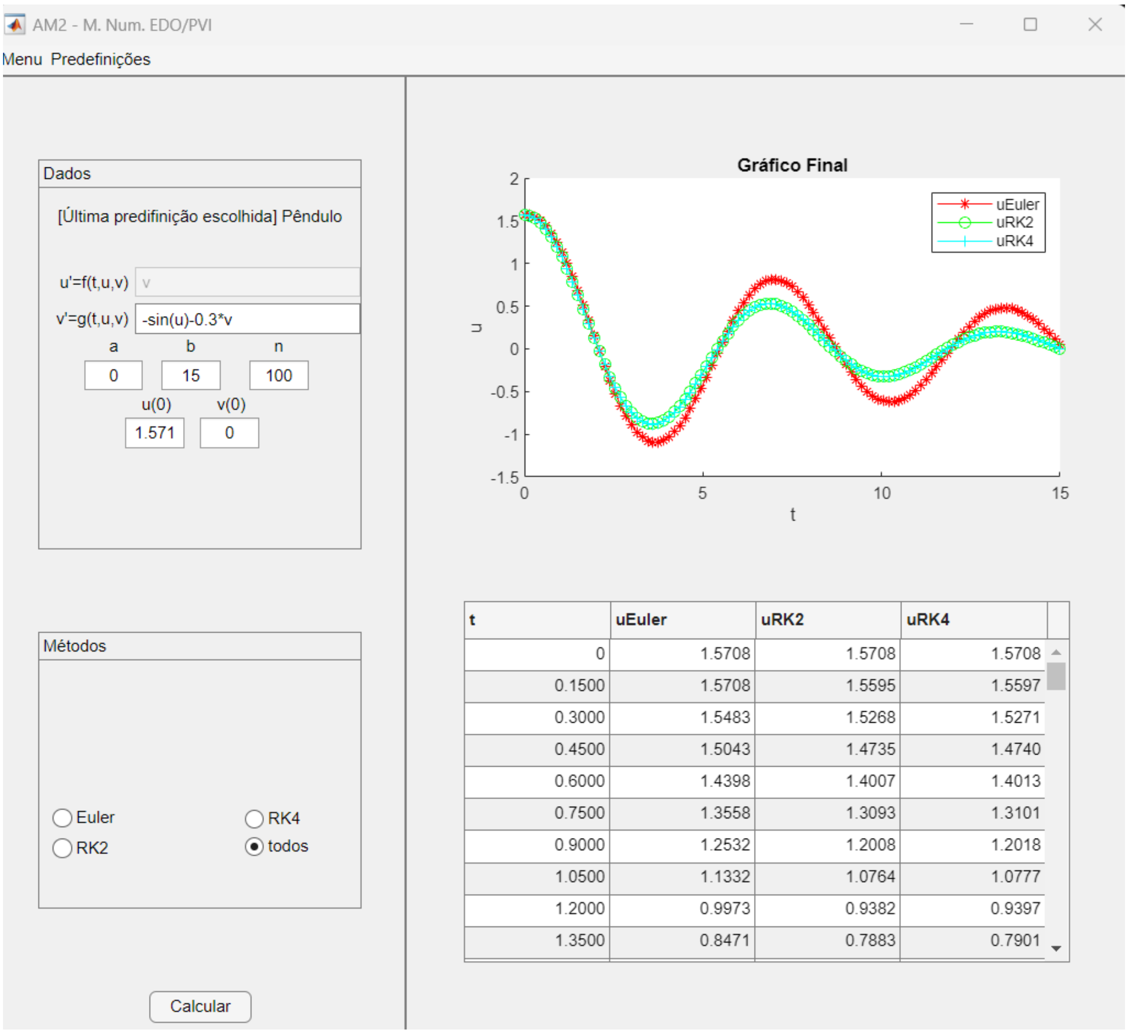
A solução exata calcula o valor exato do problema, sendo utilizada para comparação entre métodos de aproximação utilizados. No entanto, não é possível calcular equações não lineares com uma solução exata.

```
function [t, exata] = SolExata(ODE, a, b, n, u0, v0)
    syms y(t);
    tempExata = dsolve(ODE, ['y(0)=',num2str(u0)], ...
        ['Dy(0)=',num2str(v0)], 't');
    if(~isempty(tempExata))
        ext=@(t) eval(vectorize(char(tempExata)));
        h=(b-a)/n;
        t=a:h:b;
        exata=ext(t);
    else
        exata=[];
end
```

3. Exemplos de aplicação

3.1 Problema do Pêndulo

Este problema apresenta uma equação não linear de 2ª ordem, pelo que não é apresentada uma solução exata.



3.2 Problema da Massa-Mola sem amortecimento

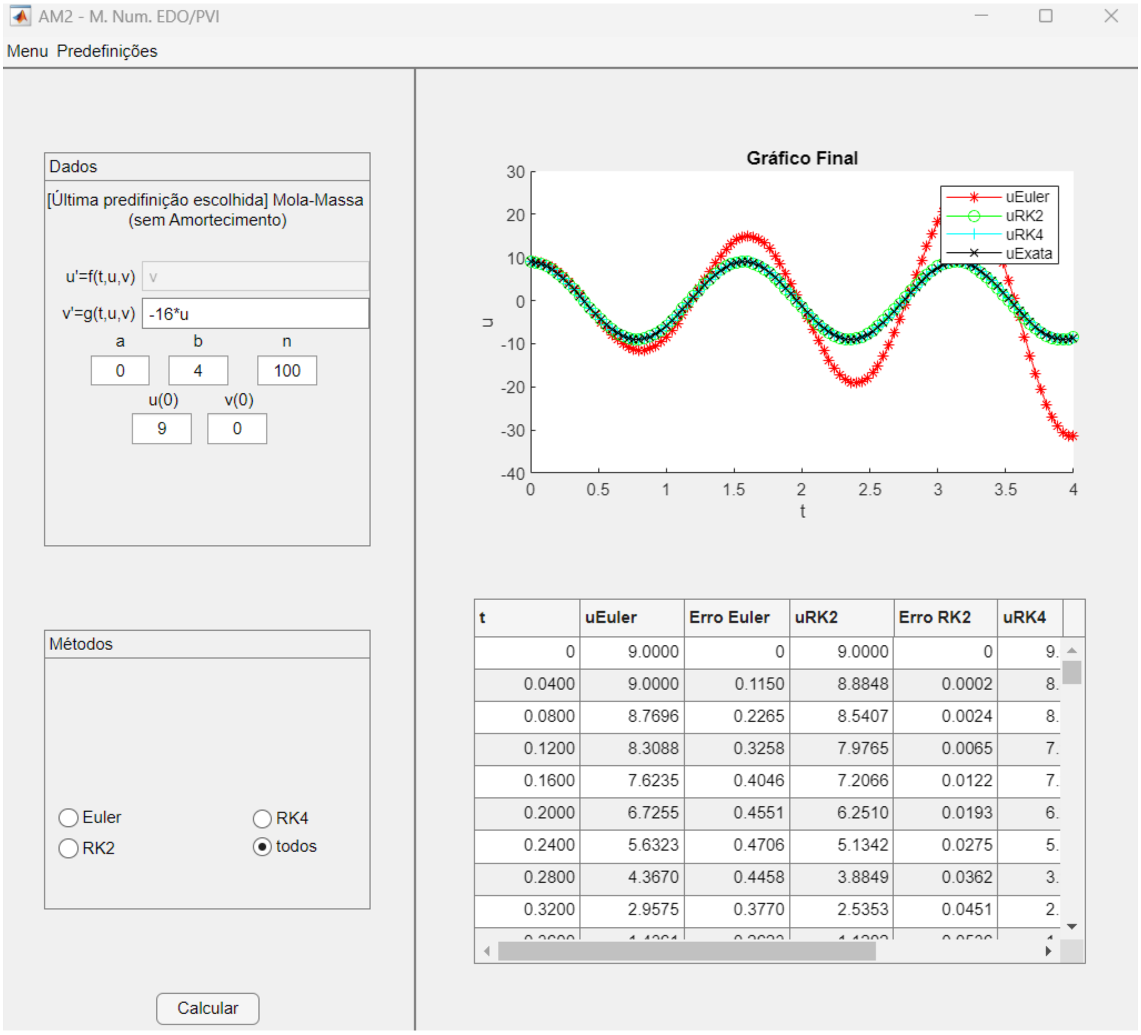
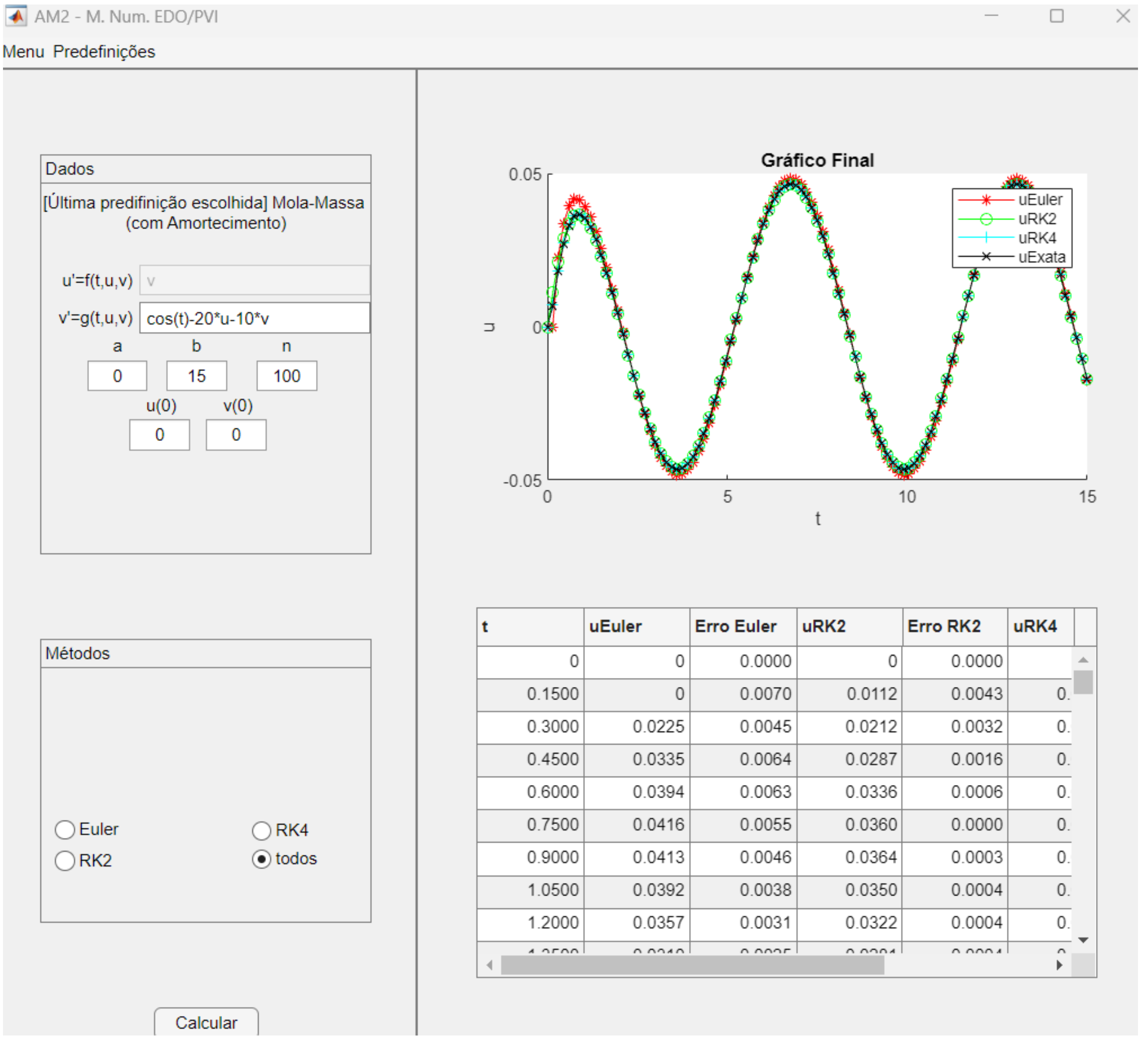


Tabela (t: 0 a 0.4):

t	uEuler	Erro Euler	uRK2	Erro RK2	uRK4	Erro RK4	uExata
0	9.0000	0	9.0000	0	9.0000	0	9.0000
0.0400	9.0000	0.1150	8.8848	0.0002	8.8850	2.0962e-07	8.8850
0.0800	8.7696	0.2265	8.5407	0.0024	8.5431	2.9182e-06	8.5431
0.1200	8.3088	0.3258	7.9765	0.0065	7.9830	8.0139e-06	7.9830
0.1600	7.6235	0.4046	7.2066	0.0122	7.2189	1.5261e-05	7.2189
0.2000	6.7255	0.4551	6.2510	0.0193	6.2704	2.4309e-05	6.2704
0.2400	5.6323	0.4706	5.1342	0.0275	5.1617	3.4705e-05	5.1617
0.2800	4.3670	0.4458	3.8849	0.0362	3.9212	4.5911e-05	3.9211
0.3200	2.9575	0.3770	2.5353	0.0451	2.5805	5.7325e-05	2.5804
0.3600	1.4361	0.2623	1.1202	0.0536	1.1739	6.8307e-05	1.1738
0.4000	-0.1609	0.1019	-0.3240	0.0612	-0.2627	7.8197e-05	-0.2628

3.3 Problema da Massa-Mola com amortecimento



t	uEuler	Erro Euler	uRK2	Erro RK2	uRK4
0	0	0.0000	0	0.0000	
0.1500	0	0.0070	0.0112	0.0043	0.
0.3000	0.0225	0.0045	0.0212	0.0032	0.
0.4500	0.0335	0.0064	0.0287	0.0016	0.
0.6000	0.0394	0.0063	0.0336	0.0006	0.
0.7500	0.0416	0.0055	0.0360	0.0000	0.
0.9000	0.0413	0.0046	0.0364	0.0003	0.
1.0500	0.0392	0.0038	0.0350	0.0004	0.
1.2000	0.0357	0.0031	0.0322	0.0004	0.

Tabela (t: 0 a 1.5):

t	uEuler	Erro Euler	uRK2	Erro RK2	uRK4	Erro RK4	uExata
0	0	0.0000	0	0.0000	0	1.0042e-17	0.0000
0.1500	0	0.0070	0.0112	0.0043	0.0073	3.2227e-04	0.0070
0.3000	0.0225	0.0045	0.0212	0.0032	0.0182	2.2160e-04	0.0180
0.4500	0.0335	0.0064	0.0287	0.0016	0.0272	1.1506e-04	0.0271
0.6000	0.0394	0.0063	0.0336	0.0006	0.0331	5.4156e-05	0.0330
0.7500	0.0416	0.0055	0.0360	0.0000	0.0361	2.5095e-05	0.0361
0.9000	0.0413	0.0046	0.0364	0.0003	0.0367	1.2325e-05	0.0367
1.0500	0.0392	0.0038	0.0350	0.0004	0.0354	6.8138e-06	0.0354
1.2000	0.0357	0.0031	0.0322	0.0004	0.0326	4.2197e-06	0.0326
1.3500	0.0310	0.0025	0.0281	0.0004	0.0285	2.6559e-06	0.0285
1.5000	0.0254	0.0020	0.0231	0.0003	0.0234	1.3692e-06	0.0234

3.4 Problema do Circuito Elétrico

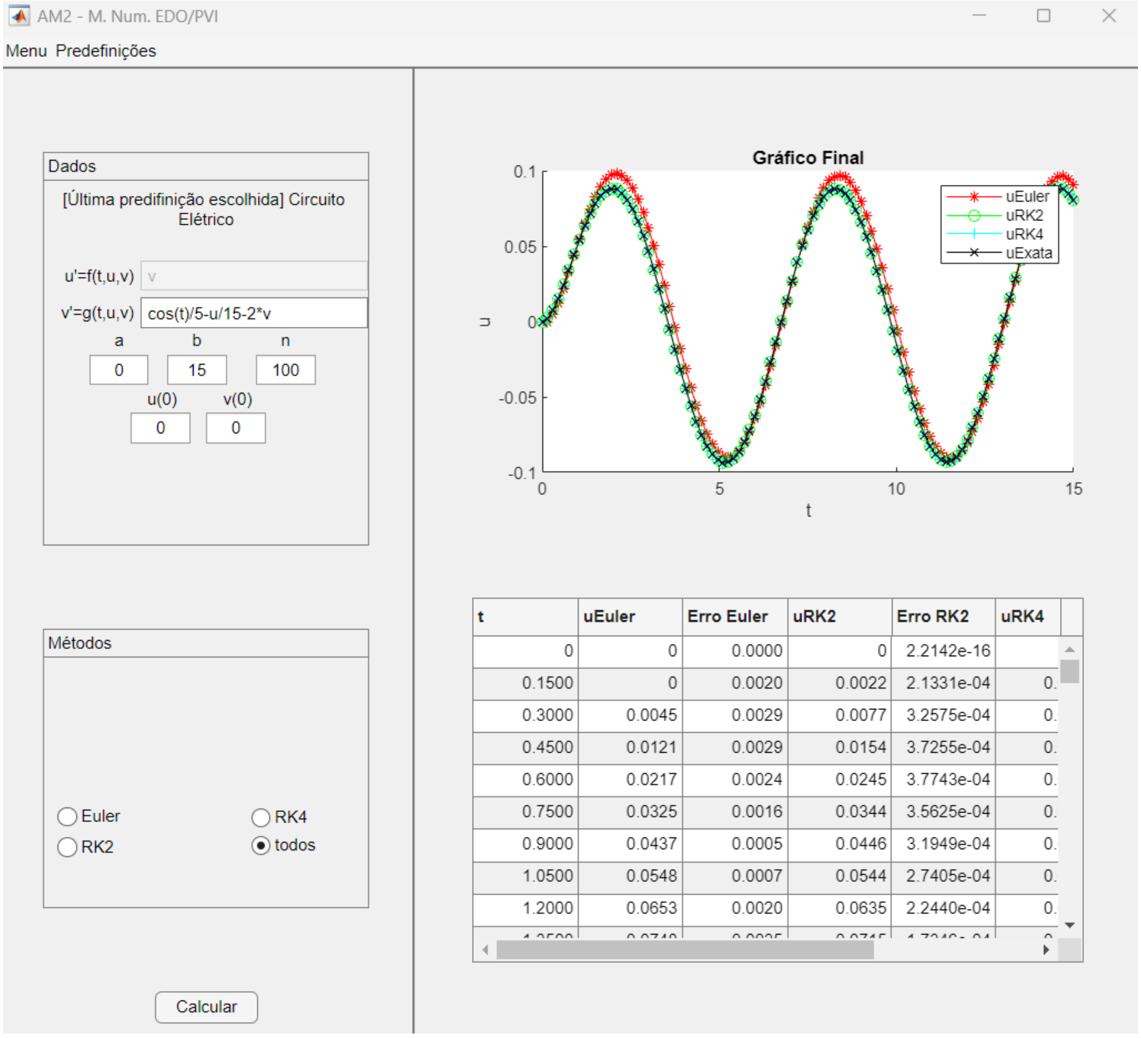


Tabela (t: 0 a 1.5):

t	uEuler	Erro Euler	uRK2	Erro RK2	uRK4	Erro RK4	uExata
0	0	0.0000	0	2.2142e-16	0	2.2142e-16	-0.0000
0.1500	0	0.0020	0.0022	2.1331e-04	0.0020	1.0066e-06	0.0020
0.3000	0.0045	0.0029	0.0077	3.2575e-04	0.0074	1.5758e-06	0.0074
0.4500	0.0121	0.0029	0.0154	3.7255e-04	0.0150	1.8613e-06	0.0150
0.6000	0.0217	0.0024	0.0245	3.7743e-04	0.0242	1.9656e-06	0.0242
0.7500	0.0325	0.0016	0.0344	3.5625e-04	0.0341	1.9554e-06	0.0341
0.9000	0.0437	0.0005	0.0446	3.1949e-04	0.0442	1.8740e-06	0.0442
1.0500	0.0548	0.0007	0.0544	2.7405e-04	0.0541	1.7490e-06	0.0541
1.2000	0.0653	0.0020	0.0635	2.2440e-04	0.0632	1.5974e-06	0.0632
1.3500	0.0748	0.0035	0.0715	1.7346e-04	0.0713	1.4298e-06	0.0713
1.5000	0.0830	0.0049	0.0782	1.2312e-04	0.0781	1.2525e-06	0.0781

4 Conclusão

Em conclusão, os sistemas de equações diferenciais podem aparecer quando existem várias funções e derivadas para o mesmo problema.

Muita das vezes, para casos em que as equações não sejam lineares, não existe forma de calcular a solução exata, tendo de se basear em dados nas aproximações resultantes dos métodos numéricos.

A aplicação desenvolvida permite o cálculo e apresentação gráfica/tabela das equações a que se pretende obter resultados. Nos casos em que a equação linear não tenha solução, não é apresentado um erro nem os resultados da mesma.

5 Bibliografia

Moodle ISEC – AM2 (04/2023):

https://moodle.isec.pt/moodle/course/view.php?id=15679

MATLAB Answers:

https://www.mathworks.com/matlabcentral/answers/index

6 Autoavaliação e heteroavaliação do trabalho submetido numa escala de 0 a 5 valores

Renato Alexandre Oliveira Craveiro – 3.5 valores em 5.