Docente: Arménio Correia

RELATÓRIO

Atividade02Trabalho

Métodos Numéricos para SED

Licenciatura em Engenharia Informática

> Um sistema de n equações diferenciais de primeira ordem é um conjunto de n equações diferenciais, com uma variável independente t e n variáveis dependentes.

Trabalho realizado por:

Tomás Gomes Silva - 2020143845

Tomás da Cunha Pinto - 2020144067

Francisco Santos Seabra Mendes - 2020143982

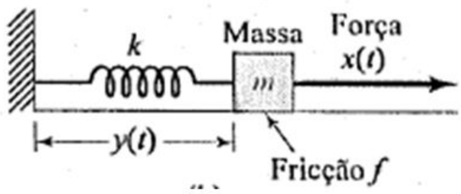
Índice

Introdução

O segundo trabalho proposto para a unidade curricular de Análise Matemática 2 consiste no estudo e aplicação de Métodos Numéricos para resolução de Sistemas de Equações Diferenciais (SED).

Este relatório vai abordar todos os métodos utilizados e alguns pontos chave da criação da aplicação em MATLAB incluindo também a resolução de alguns exercícios.

Os quatro métodos utilizados neste trabalho foram: o método de Euler, o método de Euler Melhorado/Modificado, o método de Runge-Kutta de ordem 2 e o método de Runge-Kutta de ordem 4.



$$m\frac{d^2}{dt^2}y(t)+f\frac{d}{dt}y(t)+ky(t)=x(t)$$

Posição

Velocidade

Aceleração

Sistema de Equações Diferenciais

Um sistema de equações diferencias (SED) é constituída por 2 ou mais equações diferenciais e tantas condições iniciais como equações.
A necessidade de utilizarmos um sistema de equações diferenciais deve-se ao facto de vários problemas da vida real não dependerem apenas de uma variável. Por exemplo, uma população de coelhos pode ser representada por um número só, mas para existe mais por trás "dos panos" que influenciam o número de coelhos numa dada população tais como a quantidade de predadores ou a disponibilidade de alimento.
Por esse motivo, convém-nos ter algo que "interligue" todos os fatores possíveis para que a estimativa do nosso problema seja o mais aproximada possível à realidade.

$$\textbf{PVI} \quad \begin{cases} y' = -2ty & \longleftarrow \text{Equação Diferencial} \\ t \in (0, 1.5) & \longleftarrow \text{Variável Independente} \\ y_0 = 2 & \longleftarrow \text{Condição Inicial} \end{cases}$$

$$\textbf{SED} \quad \begin{cases} \left.\begin{cases} u' = f(t,u,v) \\ v' = g(t,u,v) \end{cases}\right\} & \text{Equações Diferenciais} \\ t \in [a,b] \quad \longleftarrow & \text{Variável Independente} \\ \left.\begin{cases} u(0) = u_o \\ v(0) = v_0 \end{cases}\right\} & \text{Condições Iniciais} \end{cases}$$

Cálculo da solução exata

Há situações em que a solução exata do problema não pode ser calculada, para tratarmos desses casos particulares, colocámos o código que calcula a solução exata do problema dentro de um ***try...catch*** para que essa secção de código tente ser executada e caso falhe sabemos que a solução exata não pôde ser obtida.

Nos casos em que a solução exata pôde ser calculada é mostrado o gráfico da solução exata da **função u** e da **função v**. A tabela é preenchida também com os valores calculados.

```
try
    S = dsolve(diff(u) == f(t,u,v), diff(v) == g(t,u,v), u(0) == u0, v(0) == v0);
    if(isempty(S))
        error(S);
    end
    p = @(t) eval(vectorize(char(S.u)));
    k = @(t) eval(vectorize(char(S.v)));

    h = (b-a)/n;
    t = a:h:b;

    yExataU = p(t);
    yExataV = k(t);

    plot(app.Grafico,yExataU,'-go','LineWidth',1);
    hold(app.Grafico, "on");
    plot(app.Grafico,yExataV,'-mo','LineWidth',1);
    hold(app.Grafico, "on");
    solExata = 1;
    app.ShowErrors.Visible = 'on';
catch
    app.Mensagem.Visible = 'on';
    app.ShowErrors.Visible = 'off';
    solExata = 0;
end
```

Começamos por chamar a função **dsolve** do MATLAB e os parâmetros são:

- **diff(u) == f(t,u,v)** – a derivada da função u é igual à função f
- **diff(v) == g(t,u,v)** – a derivada da função v é igual à função g
- **u(0) == u0** – especificação da condição inicial para a função u
- **v(0) == v0** – especificação da condição inicial para a função u

Após isso, verificamos se foi ou não possível calcular a solução exata utilizando o **if** que avalia se a matriz criada pela função **dsolve** está vazio. Se a matriz estiver vazia isso quer dizer que a solução exata não pôde ser calculada e levanta um erro para que a secção de código na parte **catch** do **try...catch** seja executada.

Supondo que existe uma solução exata, o programa calcula os valores para cada **t** em ambas as funções e apresenta esses resultados no gráfico e na tabela.

Métodos Numéricos para resolução de SED

Método de Euler

Fórmulas:

O **método de Euler** é um procedimento numérico de primeira ordem para solucionar equações diferenciais ordinárias $y' = f(t, y)$ com um valor inicial dado $y(t_0) = y_0$.

Para resolvermos um SED recorrendo ao método de Euler basta adaptarmos a fórmula geral que utilizámos no trabalho prático anterior para podermos resolver também sistemas de equações diferenciais (SED).

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

$$\boldsymbol{v_{i+1} = v_i + hg(t_i\,, u_i, v_i)}$$

(com $i = 0, 1, 2, \dots, n-1$*)*

$\boldsymbol{u_{i+1}}$: Próximo valor aproximado da solução do problema $u(t)$
$\boldsymbol{v_{i+1}}$: Próximo valor aproximado da solução do problema $v(t)$
$\boldsymbol{u_i}$: Valor atual aproximado da solução do problema na abcissa $u(t)$
$\boldsymbol{v_i}$: Valor atual aproximado da solução do problema na abcissa $v(t)$
$\boldsymbol{h}$: Valor de cada subintervalo
$\boldsymbol{f(t_i, u_i, v_i)}$: Valor da função **f** no ponto (t_i, u_i, v_i)
$\boldsymbol{g(t_i, u_i, v_i)}$: Valor da função **g** no ponto (t_i, u_i, v_i)

Algoritmo/Função:

1. Definir o valor do passo do **h**
2. Criar um vetor que contém os valores para **t** a partir de **h**
3. Criar um vetor para as soluções nos diferentes pontos **t** para a função **f**
4. Criar um vetor para as soluções nos diferentes pontos **t** para a função **g**
5. Definir o primeiro elemento do vetor ***u*** para o valor da condição inicial de **u**
6. Definir o primeiro elemento do vetor ***v*** para o valor da condição inicial de **v**
7. Fazer um ciclo iterativo que percorra desde **1** a **n** para calcular a solução aproximada utilizando a fórmula geral para as duas funções

```
h = (b-a) / n; % Amplitude de cada subintervalo
t = a:h:b;
u = zeros(1, n+1); % Alocação de memória para a função u
v = zeros(1, n+1); % Alocação de memória para a função v

u(1) = u0; % Definir o primeiro elemento do array para a função u
v(1) = v0; % Definir o primeiro elemento do array para a função v

for i = 1:n % Aplicar o método de Euler (iteração)
    u(i+1) = u(i) + h * f(t(i), u(i), v(i)); % Cálculo para a função u
    v(i+1) = v(i) + h * g(t(i), u(i), v(i)); % Cálculo para a função v
end
```

Figura 1 - Algoritmo do método de Euler

Método de Euler Melhorado

Fórmulas:

O **método de Euler Melhorado**, ou método de Euler Modificado ou método de Heun é uma melhoria ao método de Euler para solucionar equações diferenciais ordinárias $y' = f(t, y)$ com um valor inicial dado $y(t_0) = y_0$.

Para resolvermos um SED recorrendo ao método de Euler Melhorado basta adaptarmos a fórmula geral que utilizámos no trabalho prático anterior para podermos resolver também sistemas de equações diferenciais (SED).

$$\boldsymbol{u_{i+1} = u_i + \frac{h}{2}(f(t_i, u_i, v_i) + f(t_{i+1}, u_{i+1}, v_{i+1})}$$

$$\boldsymbol{v_{i+1} = u_i + \frac{h}{2}(g(t_i, u_i, v_i) + g(t_{i+1}, u_{i+1}, v_{i+1})}$$

(com $i = 0, 1, 2, \dots, n-1$*)*

$\boldsymbol{u_{i+1}}$: Próximo valor aproximado da solução do problema $u(t)$
$\boldsymbol{v_{i+1}}$: Próximo valor aproximado da solução do problema $v(t)$
$\boldsymbol{u_i}$: Valor atual aproximado da solução do problema na abcissa $u(t)$
$\boldsymbol{v_i}$: Valor atual aproximado da solução do problema na abcissa $v(t)$
$\boldsymbol{h}$: Valor de cada subintervalo
$\boldsymbol{f(t_i, u_i, v_i)}$: Valor da função **f** no ponto (t_i, u_i, v_i)
$\boldsymbol{g(t_i, u_i, v_i)}$: Valor da função **g** no ponto (t_i, u_i, v_i)

Algoritmo/Função:

1. Definir o valor do passo do **h**
2. Criar um vetor que contém os valores para **t** a partir de **h**
3. Criar um vetor para as soluções nos diferentes pontos **t** para a função **f**
4. Criar um vetor para as soluções nos diferentes pontos **t** para a função **g**
5. Definir o primeiro elemento do vetor ***u*** para o valor da condição inicial de **u**
6. Definir o primeiro elemento do vetor ***v*** para o valor da condição inicial de **v**
7. Fazer um ciclo iterativo que percorra desde **1** a **n** para calcular a solução aproximada utilizando a fórmula geral para as duas funções

```
h = (b-a) / n; % Amplitude de cada subintervalo
t = a:h:b; % Criar vetor que vai de "a" a "b" com step de "h"
u = zeros(1, n+1); % Alocação de memória para a função u
v = zeros(1, n+1); % Alocação de memória para a função v

u(1) = u0; % Definir o primeiro elemento do array para a função u
v(1) = v0; % Definir o primeiro elemento do array para a função v

for i = 1:n % Aplicar o método de Euler Melhorado (iteração)
    % Cálculo para a função u
    u(i+1) = u(i) + (h/2) * (f(t(i), u(i), v(i)) + f(t(i+1), u(i+1), v(i+1)));
    % Cálculo para a função v
    v(i+1) = v(i) + (h/2) * (g(t(i), u(i), v(i)) + g(t(i+1), u(i+1), v(i+1)));
end
```

Figura 2 - Algoritmo do método de Euler Melhorado

Método de RK2

Fórmulas:

O método de **Runge-Kutta de ordem 2 (RK2)** é um método bastante simples de se aplicar que permite resolver equações diferenciais ordinárias.
Requer apenas derivadas de primeira ordem e pode fornecer aproximações precisas.

Para resolvermos um SED recorrendo ao método de Runge-Kutta 2 basta adaptarmos a fórmula geral que utilizámos no trabalho prático anterior para podermos resolver também sistemas de equações diferenciais (SED).

$$\boldsymbol{u}_{i+1} = \boldsymbol{u}_i + \frac{\mathbf{1}}{\mathbf{2}}(\boldsymbol{k}_{1u} + \boldsymbol{k}_{2u})$$

$$\boldsymbol{v}_{i+1} = \boldsymbol{v}_i + \frac{\mathbf{1}}{\mathbf{2}}(\boldsymbol{k}_{1v} + \boldsymbol{k}_{2v})$$

(com $i = 0, 1, 2, \dots, n-1$*)*

$\boldsymbol{u}_{i+1}$: Próximo valor aproximado da solução do problema $u(t)$
$\boldsymbol{v}_{i+1}$: Próximo valor aproximado da solução do problema $v(t)$
$\boldsymbol{u}_{i}$: Valor atual aproximado da solução do problema na abcissa $u(t)$
$\boldsymbol{v}_{i}$: Valor atual aproximado da solução do problema na abcissa $v(t)$

> Precisamos também de calcular dois valores, o k_1 e o k_2 para as duas funções:

$\boldsymbol{k}_{1u} = \boldsymbol{h}\boldsymbol{f}(\boldsymbol{t}_i, \boldsymbol{u}_i, \boldsymbol{v}_i)$
$\boldsymbol{k}_{1v} = \boldsymbol{h}\boldsymbol{g}(\boldsymbol{t}_i, \boldsymbol{u}_i, \boldsymbol{v}_i)$

$\boldsymbol{k}_{1u}$: Inclinação no início do intervalo para **f**
$\boldsymbol{k}_{1v}$: Inclinação no início do intervalo para **g**
$\boldsymbol{t}_i$: Valor na abcissa atual
$\boldsymbol{h}$: Valor de cada subintervalo
$\boldsymbol{f}(\boldsymbol{t}_i, \boldsymbol{u}_i, \boldsymbol{v}_i)$: Valor da equação **u** em t_i, u_i e v_i
$\boldsymbol{g}(\boldsymbol{t}_i, \boldsymbol{u}_i, \boldsymbol{v}_i)$: Valor da equação **v** em t_i, u_i e v_i

$$\boldsymbol{k_{2u} = hf(t_{i+1},\ u_i + k_{1u},\ v_i + k_{1v})}$$
$$\boldsymbol{k_{2v} = hg(t_{i+1},\ u_i + k_{1u},\ v_i + k_{1v})}$$

$\boldsymbol{k_{2u}}$: Inclinação no fim do intervalo para **f**
$\boldsymbol{k_{2v}}$: Inclinação no fim do intervalo para **g**
$\boldsymbol{t_{i+1}}$: Valor na abcissa seguinte
$\boldsymbol{h}$: Valor de cada subintervalo
$\boldsymbol{u_i}$: Valor atual aproximado da solução do problema na abcissa $u(t)$
$\boldsymbol{v_i}$: Valor atual aproximado da solução do problema na abcissa $v(t)$
$\boldsymbol{k_{1u}}$: Inclinação no início do intervalo para **f**
$\boldsymbol{k_{1v}}$: Inclinação no início do intervalo para **f**

Algoritmo/Função:

1. Definir o valor do passo do **h**
2. Criar um vetor que contém os valores para **t** a partir de **h**
3. Criar um vetor para as soluções nos diferentes pontos **t** para a função **f**
4. Criar um vetor para as soluções nos diferentes pontos **t** para a função **g**
5. Definir o primeiro elemento do vetor ***u*** para o valor da condição inicial de **u**
6. Definir o primeiro elemento do vetor ***v*** para o valor da condição inicial de **v**
7. Calcular o $\mathbf{k_1}$ e o $\mathbf{k_2}$ para ambas as funções para os utilizarmos na fórmula geral
8. Fazer um ciclo iterativo que percorra desde **1** a **n** para calcular a solução aproximada utilizando a fórmula geral para as duas funções

```
h = (b-a)/n; % Amplitude de cada subintervalo
t = a:h:b; % Criar vetor que vai de "a" a "b" com step de "h"
u = zeros(1,n+1); % Alocação de memória para a função u
v = zeros(1,n+1); % Alocação de memória para a função v
u(1) = u0; % Definir o primeiro elemento do array para a função u
v(1) = v0; % Definir o primeiro elemento do array para a função v

for i = 1:n % Aplicar o método de RK2 (iteração)

    k1u = h*f(t(i), u(i), v(i)); % Cálculo do valor de k1 para a função u
    k1v = h*g(t(i), u(i), v(i)); % Cálculo do valor de k1 para a função v

    k2u = h*f(t(i+1), u(i) + k1u, v(i) + k1v); % Cálculo do valor de k2 para a função u
    k2v = h*g(t(i+1), u(i) + k1u, v(i) + k1v); % Cálculo do valor de k2 para a função v

    u(i+1) = u(i) + (k1u + k2u) / 2; % Cálculo para a função u
    v(i+1) = v(i) + (k1v + k2v) / 2; % Cálculo para a função v
end
end
```

Figura 3 - Algoritmo do método de RK2

Método de RK4

Fórmulas:

Um método fiável e que atinge soluções bem próximas das soluções exatas é precisamente o método de **Runge-Kutta de ordem 4 (RK4)** que nos permite resolver equações diferenciais ordinárias.
O ponto positivo é que não precisamos de calcular nenhuma derivada, mas por outro lado precisamos calcular outra função que é definida avaliando **f** em diferentes pontos.

Para resolvermos um SED recorrendo ao método de Runge-Kutta 4 basta adaptarmos a fórmula geral que utilizámos no trabalho prático anterior para podermos resolver também sistemas de equações diferenciais (SED).

$$\boldsymbol{u_{i+1} = u_i + \frac{h}{6}(k_{1u} + 2k_{2u} + 2k_{3u} + k_{4u})}$$

$$\boldsymbol{v_{i+1} = v_i + \frac{h}{6}(k_{1v} + 2k_{2v} + 2k_{3v} + k_{4v})}$$

(com $i = 0, 1, 2, \dots, n-1$*)*

$\boldsymbol{u_{i+1}}$: Próximo valor aproximado da solução do problema $u(t)$
$\boldsymbol{v_{i+1}}$: Próximo valor aproximado da solução do problema $v(t)$
$\boldsymbol{u_i}$: Valor atual aproximado da solução do problema na abcissa $u(t)$
$\boldsymbol{v_i}$: Valor atual aproximado da solução do problema na abcissa $v(t)$

> Precisamos também de calcular quatro valores, o k_1, k_2, k_3 e o k_4, para as duas funções:

$\boldsymbol{k_{1u} = hf(t_i, u_i, v_i)}$
$\boldsymbol{k_{1v} = hg(t_i, u_i, v_i)}$

$\boldsymbol{k_{1u}}$: Inclinação no início do intervalo para **f**
$\boldsymbol{k_{1v}}$: Inclinação no início do intervalo para **g**
$\boldsymbol{t_i}$: Valor na abcissa atual
$\boldsymbol{h}$: Valor de cada subintervalo
$\boldsymbol{f(t_i, u_i, v_i)}$: Valor da equação **u** em t_i, u_i e v_i
$\boldsymbol{g(t_i, u_i, v_i)}$: Valor da equação **v** em t_i, u_i e v_i

$$\boldsymbol{k_{2u} = hf\left(t_i + \frac{h}{2}, u_i + \frac{1}{2}k_{1u}, v_i + \frac{1}{2}k_{1v}\right)}$$
$$\boldsymbol{k_{2v} = hg\left(t_i + \frac{h}{2}, u_i + \frac{1}{2}k_{1u}, v_i + \frac{1}{2}k_{1v}\right)}$$

$\boldsymbol{k_{2u}}$: Inclinação no ponto médio do intervalo para **f**
$\boldsymbol{k_{2v}}$: Inclinação no ponto médio do intervalo para **g**
$\boldsymbol{t_i}$: Valor na abcissa atual
$\boldsymbol{h}$: Valor de cada subintervalo
$\boldsymbol{u_i}$: Valor atual aproximado da solução do problema na abcissa $u(t)$
$\boldsymbol{v_i}$: Valor atual aproximado da solução do problema na abcissa $v(t)$
$\boldsymbol{k_{1u}}$: Inclinação no início do intervalo para **f**
$\boldsymbol{k_{1v}}$: Inclinação no início do intervalo para **g**

$$\boldsymbol{k_{3u} = hf\left(t_i + \frac{h}{2}, u_i + \frac{1}{2}k_{2u}, v_i + \frac{1}{2}k_{2v}\right)}$$
$$\boldsymbol{k_{3v} = hg\left(t_i + \frac{h}{2}, u_i + \frac{1}{2}k_{2u}, v_i + \frac{1}{2}k_{2v}\right)}$$

$\boldsymbol{k_{3u}}$: Inclinação no ponto médio do intervalo para **f**
$\boldsymbol{k_{3v}}$: Inclinação no ponto médio do intervalo para **g**
$\boldsymbol{t_i}$: Valor na abcissa atual
$\boldsymbol{h}$: Valor de cada subintervalo
$\boldsymbol{u_i}$: Valor atual aproximado da solução do problema na abcissa $u(t)$
$\boldsymbol{v_i}$: Valor atual aproximado da solução do problema na abcissa $v(t)$
$\boldsymbol{k_{2u}}$: Inclinação no ponto médio para **f**
$\boldsymbol{k_{2v}}$: Inclinação no ponto médio para **g**

$$\boldsymbol{k_{4u} = hf(t_{i+1}, u_i + k_{3u}, v_i + k_{3v})}$$
$$\boldsymbol{k_{4v} = hg(t_{i+1}, u_i + k_{3u}, v_i + k_{3v})}$$

$\boldsymbol{k_{4u}}$: Inclinação no final do intervalo para **f**
$\boldsymbol{k_{4v}}$: Inclinação no final do intervalo para **g**
$\boldsymbol{t_i}$: Valor na abcissa atual
$\boldsymbol{h}$: Valor de cada subintervalo
$\boldsymbol{u_i}$: Valor atual aproximado da solução do problema na abcissa $u(t)$
$\boldsymbol{v_i}$: Valor atual aproximado da solução do problema na abcissa $v(t)$
$\boldsymbol{k_{3u}}$: Inclinação no ponto médio para **f**
$\boldsymbol{k_{3v}}$: Inclinação no ponto médio para **g**

Algoritmo/Função:

1. Definir o valor do passo do **h**
2. Criar um vetor que contém os valores para **t** a partir de **h**
3. Criar um vetor para as soluções nos diferentes pontos **t** para a função **f**
4. Criar um vetor para as soluções nos diferentes pontos **t** para a função **g**
5. Definir o primeiro elemento do vetor ***u*** para o valor da condição inicial de **u**
6. Definir o primeiro elemento do vetor ***v*** para o valor da condição inicial de **v**
7. Calcular o k_1, k_2, k_3 e o k_4 para ambas as funções para utilizar na fórmula geral
8. Fazer um ciclo iterativo que percorra desde **1** a **n** para calcular a solução aproximada utilizando a fórmula geral

```
h = (b-a)/n; % Amplitude de cada subintervalo
t = a:h:b; % Criar vetor que vai de "a" a "b" com step de "h"
u = zeros(1,n+1); % Alocação de memória para a função u
v = zeros(1,n+1); % Alocação de memória para a função v
u(1) = u0; % Definir o primeiro elemento do array para a função u
v(1) = v0; % Definir o primeiro elemento do array para a função v

for i = 1:n % Aplicar o método de RK4 (iteração)
    k1u = h*f(t(i), u(i), v(i)); % Cálculo do valor de k1 para a função u
    k1v = h*g(t(i), u(i), v(i)); % Cálculo do valor de k1 para a função v

    k2u = h*f(t(i) + h/2, u(i) + k1u/2, v(i) + k1v/2); % Cálculo do valor de k2 para a função u
    k2v = h*g(t(i) + h/2, u(i) + k1u/2, v(i) + k1v/2); % Cálculo do valor de k2 para a função v

    k3u = h*f(t(i) + h/2, u(i) + k2u/2, v(i) + k2v/2); % Cálculo do valor de k3 para a função u
    k3v = h*g(t(i) + h/2, u(i) + k2u/2, v(i) + k2v/2); % Cálculo do valor de k3 para a função v

    k4u = h*f(t(i+1), u(i) + k3u, v(i) + k3v); % Cálculo do valor de k4 para a função u
    k4v = h*g(t(i+1), u(i) + k3u, v(i) + k3v); % Cálculo do valor de k4 para a função v

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

Figura 4 - Algoritmo do método de RK4

Exemplos de aplicação e teste dos métodos

Algoritmo de Resolução

Para resolvermos um sistema de equações diferenciais (SED) utilizando métodos numéricos podemos seguir um simples algoritmo de apenas 5 passos:

Supondo que a nossa ED é: $y'' + 20y' - 2\cos(y) = 0,\ \ u_0 = 0,\ \ v_0 = 5,\ \ t \in [0,10]$

Passo 1: Resolver a equação em ordem à derivada de maior grau

$$\begin{gathered} y'' + 20y' + 2\cos(y) = 0 \\ \leftrightarrow y'' = -2\cos(y) + 20y' \end{gathered}$$

Passo 2: Mudança de variável

$$\begin{gathered} u = y, v = y' \\ u' = v, v' = y'' \end{gathered}$$

Passo 3: Derivar <u>u</u> e <u>v</u> e substituir

$$\begin{cases} u' = v \\ v' = -2\cos(y) + 20y' \end{cases}$$

Passo 4: Obter o sistema de equações diferencial (SED)

$$\begin{cases} \begin{cases} u' = v \\ v' = -2\cos(y) + 20y' \end{cases} \\ t \in [0, 10] \\ \begin{cases} u(0) = 0 \\ v(0) = 5 \end{cases} \end{cases}$$

Passo 5: Aplicar os métodos numéricos existentes na APP

Escolher entre os métodos existentes na APP (Euler, Euler Melhorado, Runge-Kutta 2 e Runge-Kutta 4.

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.)

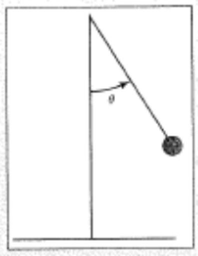


FIGURE 13.1a Simple pendulum.

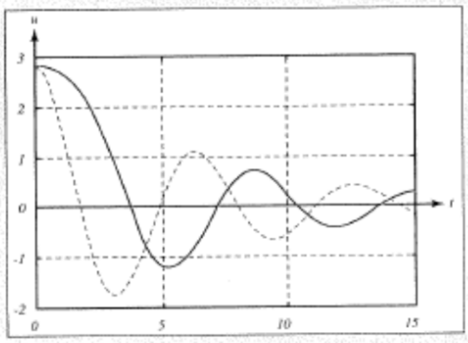


FIGURE 13.1b The motion of a pendulum given by ODE above (solid line) and linearized ODE (dashed line).

$$\begin{cases} \begin{cases} u' = v \\ v' = -\sin(u) - 0.3v \end{cases} \\ t \in [0, 15] \\ \begin{cases} u(0) = \dfrac{\pi}{2} \\ v(0) = 0 \end{cases} \end{cases} \qquad \begin{cases} y'' + 0.3y' + \sin(y) = 0 \\ y(0) = \dfrac{\pi}{2} \\ y'(0) = 0 \end{cases}$$

Para transformar uma Equação Diferencial Ordinária (EDO) de segunda ordem num sistema de duas Equações Diferenciais Ordinárias Lineares de primeira ordem vamos recorrer a um simples algoritmo de 3 passos:

Passo 1: Isolar a derivada de maior ordem

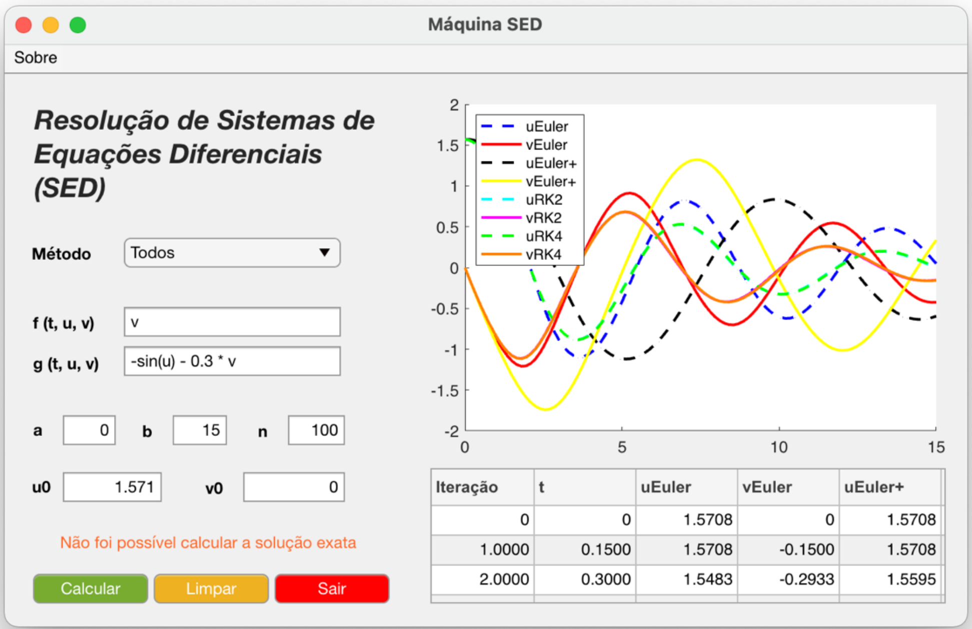
$$y'' + 0.3y' + \sin(y) = 0$$
$$\leftrightarrow y'' = -\sin(y) - 0.3y'$$

Passo 2: Mudança de Variável

$$u = y, v = y'$$
$$u' = v, v' = y''$$

Passo 3: Substituir

$$\begin{cases} u' = v \\ v' = -\sin(u) - 0.3v \end{cases}$$

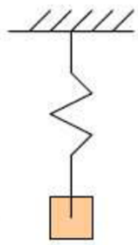


Mola-Massa com Amortecimento

a) $x'' + 2x' + 2x = 4\cos t + 2\sin t, \quad x(0) = 0 \;\; x'(0) = 3$

$\Rightarrow x(t) = e^{-t}\sin t + 2\sin t$

b) A equação $mx'' + kx = 0$ descreve o movimento harmónico simples, ou movimento livre não amortecido, e está sujeita às condições iniciais $x(0) = a$ e $x'(0) = b$ representando, respectivamente, a medida do deslocamento inicial e a velocidade inicial.



Use este conhecimento para dar uma interpretação física do problema de *Cauchy*

$$x'' + 16x = 0 \quad x(0) = 9 \quad x'(0) = 0$$

e resolva-o

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 $k = \dfrac{6.4}{1.28} \Leftrightarrow k = 5 \;\; lb/ft$. Como $W = mg$, tem-se $m = \dfrac{6.4}{32} \Leftrightarrow m = 0.2$

A equação que descreve 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 $0.2x'' = -5x - 2x'$

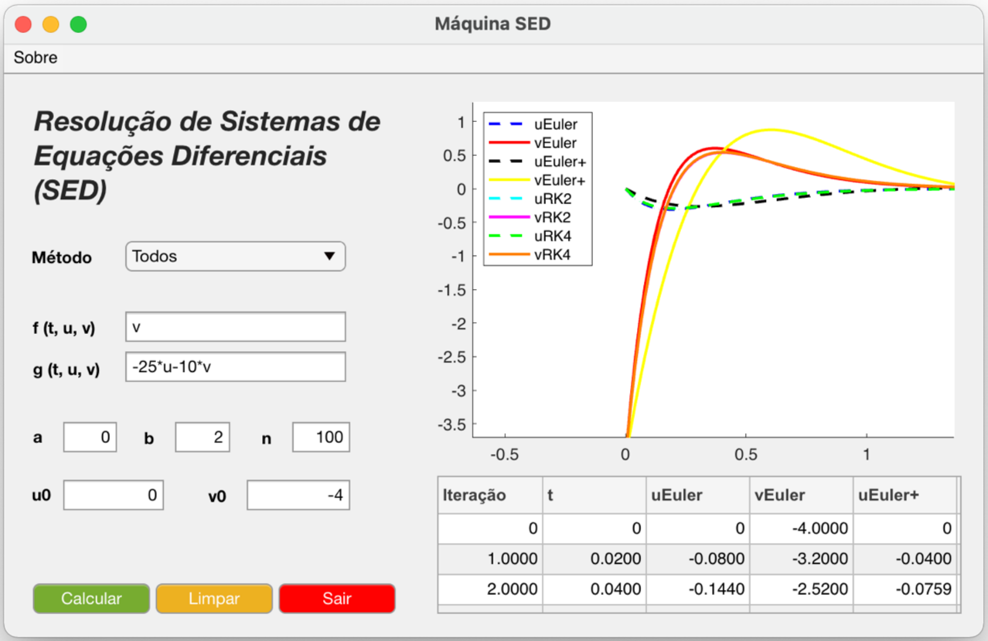
$\Leftrightarrow x'' + 10x' + 25x = 0$ com $x(0) = 0$ e $x'(0) = -4$

$$y'' + 10y' + 25y = 0$$
$$\leftrightarrow y'' = -25y - 10y'$$

$$\begin{cases} y(0) = 0 \\ y'(0) = -4 \end{cases}$$

Desta forma, é possível obter o seguinte sistema de equações

$$\begin{cases} \begin{cases} u' = v \\ v' = -25u - 10v \end{cases} \\ t \in [0, 2] \\ \begin{cases} u(0) = 0 \\ v(0) = -4 \end{cases} \end{cases}$$


Máquina SED
Sobre
Resolução de Sistemas de Equações Diferenciais (SED)
Método
Todos
f (t, u, v)
v
g (t, u, v)
-25*u-10*v
a
0
b
2
n
100
u0
0
v0
-4
Calcular
Limpar
Sair
uEuler
vEuler
uEuler+
vEuler+
uRK2
vRK2
uRK4
vRK4
Iteração
t
uEuler
vEuler
uEuler+
0 0 0 -4.0000 0
1.0000 0.0200 -0.0800 -3.2000 -0.0400
2.0000 0.0400 -0.1440 -2.5200 -0.0759

Modelos Vibratórios Mecânicos

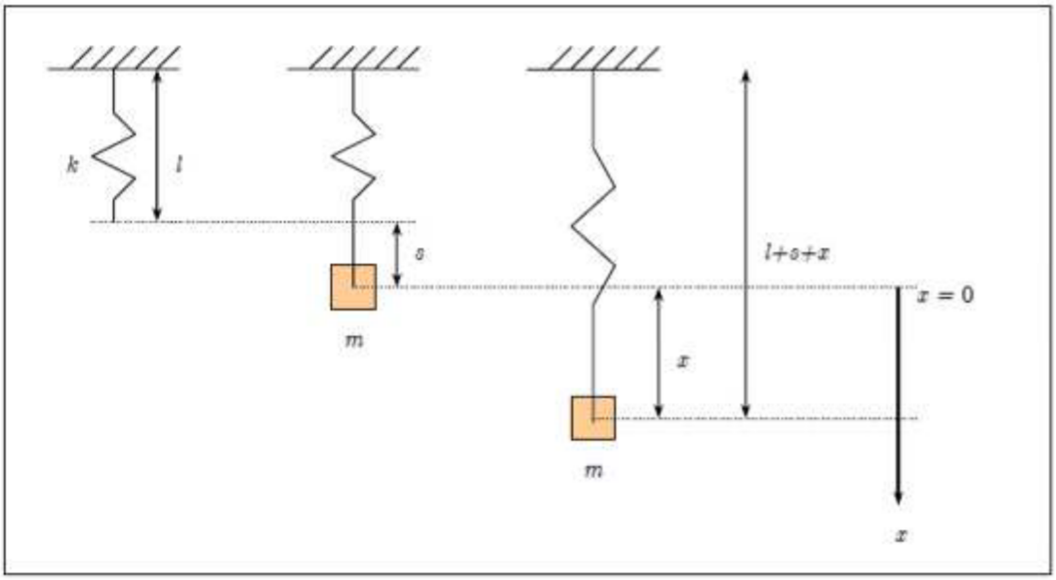
Nestes sistemas, o deslocamento x obedece à equação diferencial linear de 2ª ordem

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

onde:

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

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



a) $x'' + 2x' + 2x = 4\cos t + 2\sin t, \quad x(0) = 0 \quad x'(0) = 3$

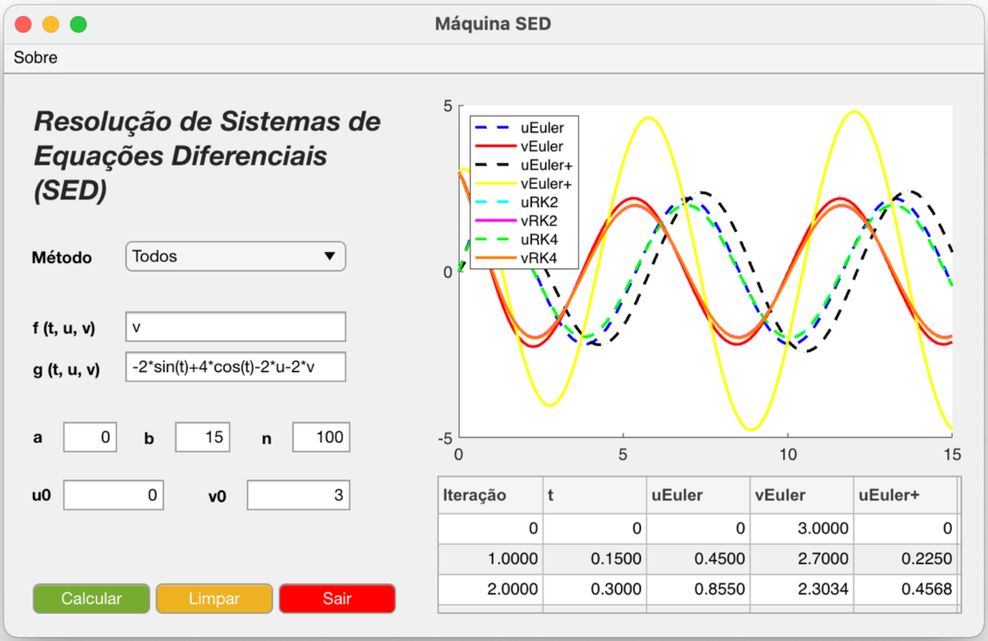
$\Rightarrow x(t) = e^{-t}\sin t + 2\sin t$

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

$$\begin{cases} y(0) = 0 \\ y'(0) = 3 \end{cases}$$

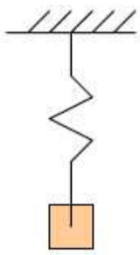
Desta forma, é possível obter o seguinte sistema de equações

$$\begin{cases} \begin{cases} u' = v \\ v' = -2\sin(t) + 4\cos(t) - 2u - 2v \end{cases} \\ t \in [0, 15] \\ \begin{cases} u(0) = 0 \\ v(0) = 3 \end{cases} \end{cases}$$


Máquina SED
Sobre
Resolução de Sistemas de Equações Diferenciais (SED)
Método
Todos
f (t, u, v)
v
g (t, u, v)
-2*sin(t)+4*cos(t)-2*u-2*v
a
0
b
15
n
100
u0
0
v0
3
Calcular
Limpar
Sair
uEuler
vEuler
uEuler+
vEuler+
uRK2
vRK2
uRK4
vRK4
Iteração
t
uEuler
vEuler
uEuler+
0
0
0
3.0000
0
1.0000
0.1500
0.4500
2.7000
0.2250
2.0000
0.3000
0.8550
2.3034
0.4568

Movimento Harmónico Simples

b) A equação $mx'' + kx = 0$ descreve o movimento harmónico simples, ou movimento livre não amortecido, e está sujeita às condições iniciais $x(0) = a$ e $x'(0) = b$ representando, respectivamente, a medida do deslocamento inicial e a velocidade inicial.



Use este conhecimento para dar uma interpretação física do problema de *Cauchy*

$$x'' + 16x = 0 \quad x(0) = 9 \quad x'(0) = 0$$

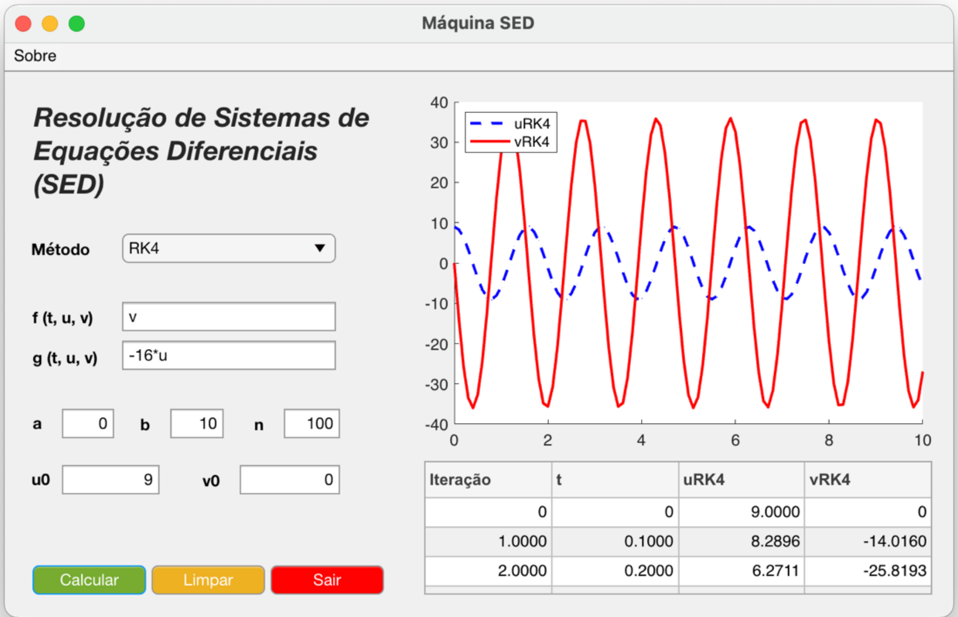
e resolva-o

$$y'' + 16y = 0$$
$$\leftrightarrow y'' = -16y$$

$$\begin{cases} y(0) = 9 \\ y'(0) = 0 \end{cases}$$

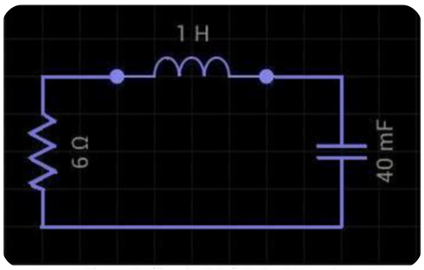
Desta forma, é possível obter o seguinte sistema de equações

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



Circuitos Elétricos

Considere o circuito RLC apresentado abaixo. Com base nos dados determine a expressão que representa a tensão no condensador em função do tempo para um tempo t > 0. Note que $I_L(0) = 4A$ e que $V_C(0) = -4V$.

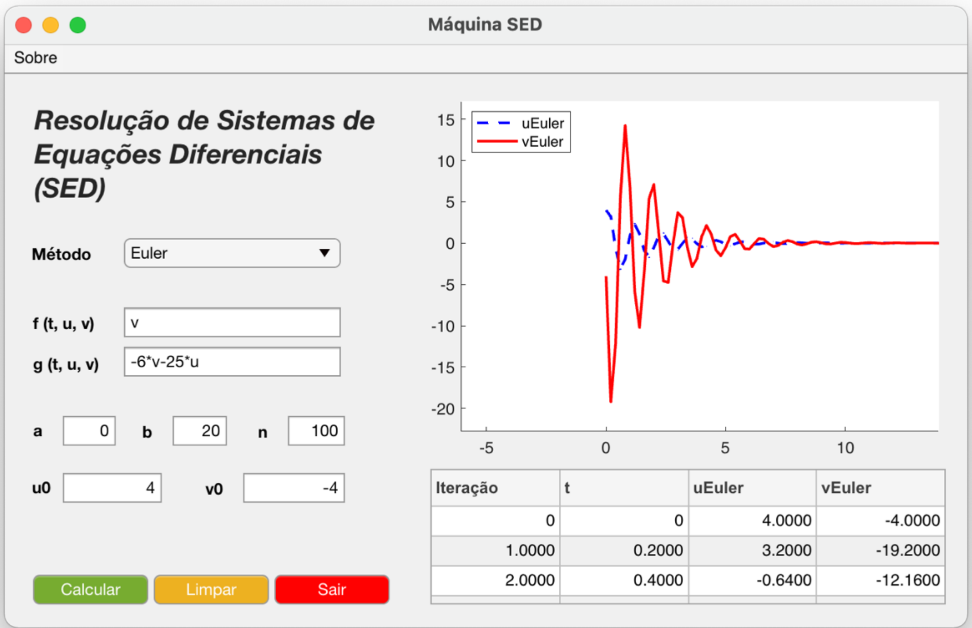


$$y'' + 6y' + 25y = 0$$
$$\leftrightarrow y'' = -6y' - 25y$$

$$\begin{cases} y(0) = 4 \\ y'(0) = -4 \end{cases}$$

Desta forma, é possível obter o seguinte sistema de equações

$$\begin{cases} \begin{cases} u' = v \\ v' = -6v - 25u \end{cases} \\ t \in [0, 20] \\ \begin{cases} u(0) = 4 \\ v(0) = -4 \end{cases} \end{cases}$$

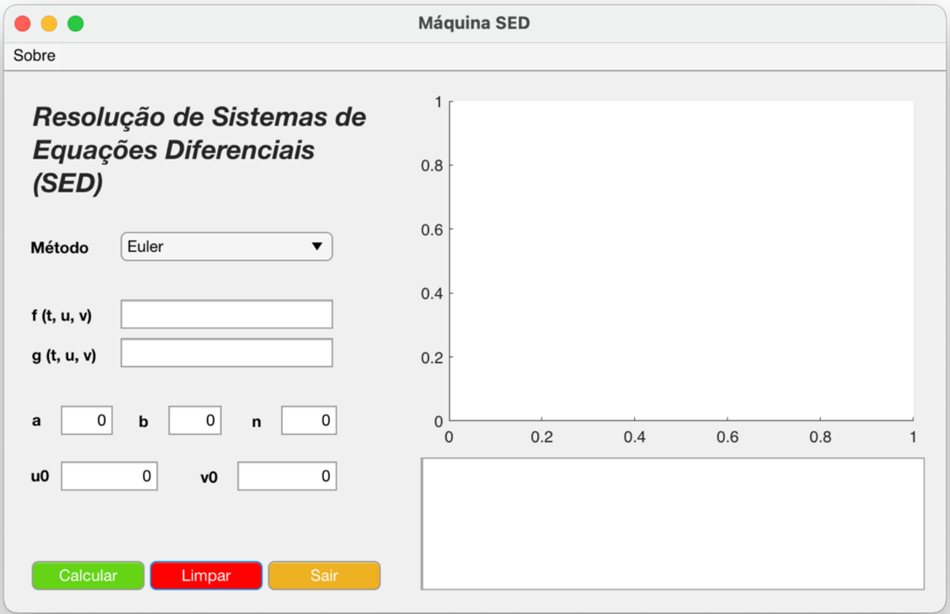


Manual de Utilização

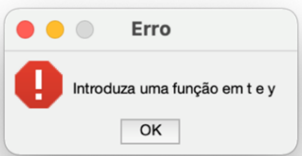
O menu principal tem tudo o que é preciso para resolver os sistemas de equações diferenciais. A escolha do método, as funções **f** e **g** em função de t, u e v, O intervalo de **a** a **b**, o número de **iterações** e as **condições iniciais** para cada uma das equações diferenciais. Do lado direito da aplicação é apresentado os **gráficos** correspondentes à solução exata para as duas equações diferenciais (quando a solução exata puder ser calculada) e a **tabela** que contém todos os valores calculados.

Os três botões que fazem parte da aplicação são os botões: **Calcular**, **Limpar** e **Sair**.

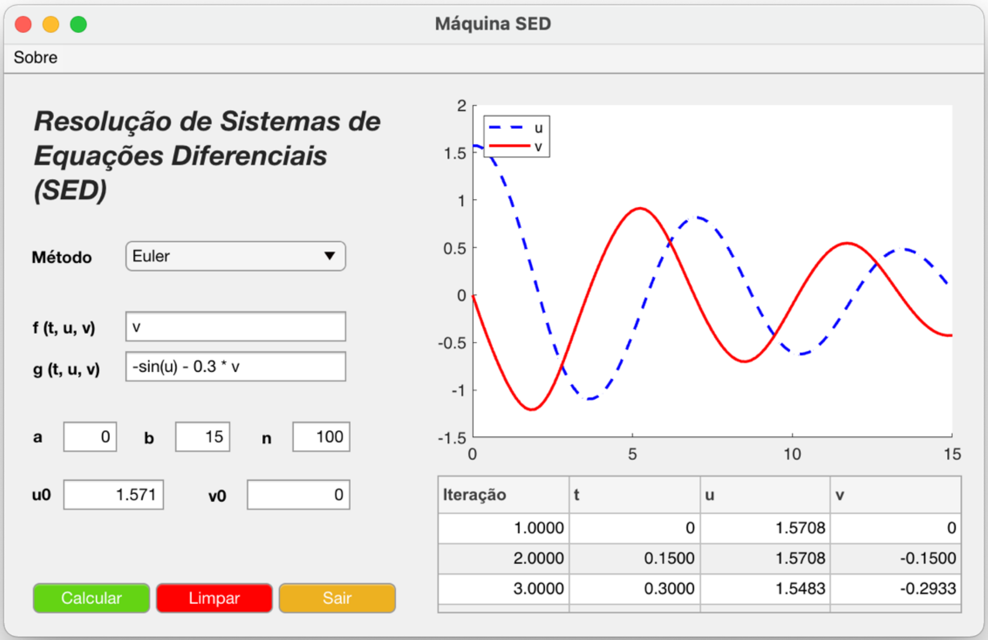
O botão **Limpar** vai limpar todos os campos, gráficos e tabelas que tenham sido preenchidos anteriormente. O botão **Calcular** vai calcular a solução do PVI com base nos valores introduzidos se todos os campos tiverem sido preenchidos corretamente. Naturalmente, o botão **Sair** fará com que o programa termine.



Caso algum campo tenha sido preenchido de maneira incorreta o programa irá apresentar um pop-up a explicar o que é que correu mal.



Pressupondo que o utilizador preencheu todos os campos corretamente, a aplicação deverá mostrar os gráficos das soluções exatas e das aproximações do resultado do sistema de equações diferencial no lado direito. A tabela será também preenchida com os valores calculados utilizando o método escolhido previamente.



Conclusão

A realização deste trabalho deu-nos a oportunidade de construir outra aplicação em MATLAB e veio assim mais uma vez aumentar as nossas competências neste campo.

Os métodos aplicados no trabalho prático anterior provaram-se bastante úteis visto que apenas foi necessário voltar a aplicá-los neste trabalho prático, tendo sido necessário realizar algumas pequenas adaptações para permitir que um sistema de equações diferencial de segunda ordem pudesse ser resolvido com esses mesmos métodos.

As conclusões que tirámos do trabalho anterior, também se mostraram válidas para este trabalho. O método mais fiável é o método de Runge-Kutta 4 (RK4) sendo que este é o método que se aproxima mais da solução exata. Por outro lado, o método de Euler, tal como visto no trabalho anterior, apresenta erros maiores do que todos os outros métodos implementados nesta aplicação.

Algo que nos surpreendeu aquando da realização dos exercícios de aplicação foi o quão úteis os sistemas de equações diferenciais são, não só para a Engenharia como também para várias áreas como a Economia, Comércio, Ciências, comportamento da população humana, entre muitas outras áreas.