t

Licenciatura em Engenharia Informática

Docente: Arménio Correia

# Índice

[Índice 3](#_Toc70339836)

[Introdução 4](#_Toc70339837)

[Equação diferencial: definição e propriedades 5](#_Toc70339838)

[Definição de PVI 6](#_Toc70339841)

[Métodos Numéricos para resolução de PVI 7](#_Toc70339842)

[Método de Euler 7](#_Toc70339843)

[Método de Euler Melhorado 8](#_Toc70339846)

[Método de RK2 9](#_Toc70339849)

[Método de RK4 11](#_Toc70339852)

[Função ODE45 13](#_Toc70339855)

[Método do Ponto Médio 14](#_Toc70339858)

[Exemplos de aplicação e teste dos métodos 16](#_Toc70339861)

[Exercício 3 do Teste Farol 16](#_Toc70339862)

[Problemas de aplicação do livro 16](#_Toc70339863)

[Manual de Utilização 17](#_Toc70339864)

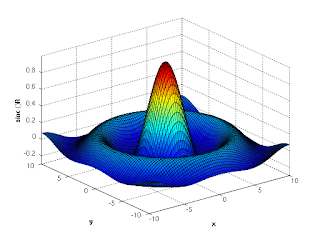
[Conclusão 19](#_Toc70339865)

# Introdução

O primeiro trabalho proposto para a unidade curricular de Análise Matemática 2 consiste no estudo e aplicação de Métodos Numéricos para resolução de Equações Diferenciais Ordinárias (EDO) e Problemas de Valor Inicial (PVI).

Este relatório vai abordar todos os métodos utilizados e alguns pontos chave da criação da interface de texto bem como a interface gráfica em MATLAB.

Os cinco métodos utilizados neste trabalho foram: o método de Euler, o método de Euler Melhorado/Modificado, o método de Runge-Kutta de ordem 2, o método de Runge-Kutta de ordem 4, a função ODE45 do próprio MATLAB e o método do Ponto Médio que foi procurado pelo nosso grupo de trabalho.



## Equação diferencial: definição e propriedades

Uma equação diferencial é uma equação com derivadas de uma ou mais funções. As equações diferenciais podem ser classificadas por tipo, ordem e linearidade.

Uma imagem com texto

Descrição gerada automaticamente

Figura 1 - Exemplo de uma Equação Diferencial

***TIPO:***Uma equação diferencial diz-se ordinária (EDO) quando a função depende apenas de uma única variável independente. Caso dependa de mais do que uma variável então dizemos que se trata de uma equação diferencial parcial (EDP).

Uma imagem com texto, relógio, manómetro

Descrição gerada automaticamente

Figura 2 - Equação Diferencial Ordinária (EDO)

***ORDEM:***A ordem de uma equação diferencial depende do número de derivadas que aparecem na equação. Se na equação aparecerem 2 derivadas então a equação diferencial é de segunda ordem.

**(EDO de ordem 2)**

***LINEARIDADE:***Uma equação diferencial ordinária é linear se tivermos um coeficiente a multiplicar pela derivada, ou por outras palavras, se puder ser escrita nesta forma:

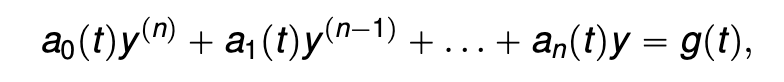


Figura 3 - Outra maneira de escrever uma EDO

## é uma EDO Linear a0 = 1, a1 = -3, a2 = 4.

## não é uma EDO Linear pois temos o uma incógnita (y) a multiplicar pela derivada (y’)

## Definição de PVI

Um **Problema de Valor Inicial**, também conhecido por *PVI*, é uma equação diferencial que é acompanhada do valor da função objetivo em um determinado ponto, chamada de condição inicial.

Equação Diferencial

Condição Inicial

Variável Independente

**PVI**

Figura 4 – Exemplo e solução de um PVI

# Métodos Numéricos para resolução de PVI

## Método de Euler

### ***Fórmulas:***

O **método de Euler** é um procedimento numérico de primeira ordem para solucionar equações diferenciais ordinárias com um valor inicial dado .

Para resolvermos um PVI recorrendo ao método de Euler usamos a fórmula geral:

*(com )*

: Próximo valor aproximado da solução do problema na abcissa

: Próximo valor aproximado da solução do problema na abcissa

: Valor de cada subintervalo

: Valor da equação em e

### ***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 armazenar as soluções nos diferentes pontos **t**
4. Definir o primeiro elemento do vetor ***y*** para o valor da condição inicial
5. Fazer um ciclo iterativo que percorra desde **1 a n** para calcular a solução aproximada utilizando a fórmula geral

Uma imagem com texto

Descrição gerada automaticamente

Figura 4 - 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 com um valor inicial dado .

Para resolvermos um PVI recorrendo ao método de Euler Melhorado usamos a fórmula geral:

*(com )*

: Próximo valor aproximado da solução do problema na abcissa

: Próximo valor aproximado da solução do problema na abcissa

: Valor de cada subintervalo

: Valor da equação em e

### ***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 armazenar as soluções nos diferentes pontos **t**
4. Definir o primeiro elemento do vetor ***y*** para o valor da condição inicial
5. Fazer um ciclo iterativo que percorra desde **1 a n** para calcular a solução aproximada utilizando a fórmula geral

Uma imagem com texto

Descrição gerada automaticamente

Figura 5 - 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 problema de valor inicial (PVI) temos de utilizar a fórmula geral do método de RK2:

*(com )*

: Próximo valor aproximado da solução do problema na abcissa

: Próximo valor aproximado da solução do problema na abcissa

**> Precisamos também de calcular dois valores, o e o :**

: Inclinação no início do intervalo

: Valor de cada subintervalo

: Valor da equação em e

: Inclinação no fim do intervalo

: Valor na abcissa atual

: Valor de cada subintervalo

: Valor aproximado da solução do problema na abcissa atual

: Inclinação no início do intervalo

### ***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 armazenar as soluções nos diferentes pontos **t**
4. Definir o primeiro elemento do vetor ***y*** para o valor da condição inicial
5. Calcular o **k1** e o **k2** para os podermos utilizar na fórmula geral
6. Fazer um ciclo iterativo que percorra desde **1 a n** para calcular a solução aproximada utilizando a fórmula geral

Uma imagem com texto

Descrição gerada automaticamente

Figura 6 - 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 problema de valor inicial (PVI) temos de utilizar a fórmula geral do método de RK4:

*(com )*

: Próximo valor aproximado da solução do problema na abcissa

: Próximo valor aproximado da solução do problema na abcissa

: Valor de cada subintervalo

**> Precisamos também de calcular quatro valores, o , , e o:**

: Inclinação no início do intervalo

: Valor de cada subintervalo

: Valor da equação em e

: Inclinação no ponto médio do intervalo

: Valor na abcissa atual

: Valor de cada subintervalo

: Valor aproximado da solução do problema na abcissa atual

: Inclinação no início do intervalo

: Inclinação no ponto médio do intervalo

: Valor na abcissa atual

: Valor de cada subintervalo

: Valor aproximado da solução do problema na abcissa atual

: Inclinação no ponto médio do intervalo

: Inclinação no final do intervalo

: Valor na abcissa atual

: Valor de cada subintervalo

: Valor aproximado da solução do problema na abcissa atual

: Inclinação no ponto médio do intervalo

### ***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 armazenar as soluções nos diferentes pontos **t**
4. Definir o primeiro elemento do vetor ***y*** para o valor da condição inicial
5. Calcular o **k1**, **k2,** **k3** e o **k4** para os podermos utilizar na fórmula geral
6. Fazer um ciclo iterativo que percorra desde **1 a n** para calcular a solução aproximada utilizando a fórmula geral

Uma imagem com texto

Descrição gerada automaticamente

Figura 7 - Algoritmo do método de RK4

## Função ODE45

### ***Fórmulas:***

A **função ODE45**, ou *Ordinary Differntial Equation 45* é função nativa do MATLAB baseada no método de Runge-Kutta e é utilizada para resolver equações diferenciais ordinárias.

Ao chamarmos a função é-nos devolvido dois vetores: um vetor com os pontos de testagem e outro vetor com as soluções aproximadas em cada um desses pontos.

**ode45**

: Equação diferencial

: Vetor que contém os pontos onde a função vai ser calculada

: Valor inicial do PVI (condição inicial)

### ***Algoritmo/Função:***

1. Definir o valor do passo do **h**
2. Gerar o vetor **t** a partir do **h** *(vetor a começar em* ***a*** *e a acabar em* ***b*** *com passo* ***h****)*
3. Criar um vetor que contém os valores para **t** a partir de **h**
4. Chamar a função ODE45 com os parâmetros

## Método do Ponto Médio

### ***Fórmulas:***

A **função ODE45**, ou *Ordinary Differntial Equation* é função nativa do MATLAB baseada no método de Runge-Kutta e é utilizada para resolver equações diferenciais ordinárias.

Ao chamarmos a função é-nos devolvido dois vetores: um vetor com os pontos de testagem e outro vetor com as soluções aproximadas em cada um desses pontos.

*Método Explícito*

*Método Implícito*

: Próximo valor aproximado da solução do problema na abcissa

: Próximo valor aproximado da solução do problema na abcissa

: Valor na abcissa atual

: Valor de cada subintervalo

**> Precisamos também de calcular mais um valor adicional, o :**

: Inclinação no início do intervalo

: Valor de cada subintervalo

: Valor da equação em e

### ***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 armazenar as soluções nos diferentes pontos **t**
4. Definir o primeiro elemento do vetor ***y*** para o valor da condição inicial
5. Calcular o **k1** para o podermos utilizar na fórmula geral
6. Fazer um ciclo iterativo que percorra desde **1 a n** para calcular a solução aproximada utilizando o ponto médio explícito e o ponto médio implícito

Uma imagem com texto, interior, captura de ecrã

Descrição gerada automaticamente

Figura 8 - Algoritmo do método do Ponto Médio

# Exemplos de aplicação e teste dos métodos

## Exercício 3 do Teste Farol

Blah blah blah

## Problemas de aplicação do livro

Blah blah blah

# Manual de Utilização

Ao abrir a aplicação ser-lhe-á apresentada a seguinte interface onde poderá encontrar 4 painéis distintos: um painel para a **introdução dos dados do problema de valor inicial (PVI)**, um painel para a **escolha do método** que deseja utilizar para resolver o PVI, um painel com o **plot dos gráficos** e por fim um painel que mostra a **tabela preenchida** com os valores.

Os únicos dois botões que fazem parte da aplicação são os botões: **Limpar** e **Calcular**.

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.

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

Uma imagem com texto

Descrição gerada automaticamente

Partindo do princípio que tudo correu bem, a aplicação mostrará finalmente o gráfico da solução exata assim como o gráfico do método selecionado, e também a tabela que mostra os valores de **t**, a **solução exata** no ponto t, a **solução aproximada** no ponto t do método selecionado e o **erro** que esse método cometeu.



# Conclusão

Este trabalho permitiu-nos observar as diferenças entre os métodos em questão e perceber que existem métodos mais fiáveis do que outros, mas mais importante do que isso, tivemos a oportunidade de ver o quão úteis estes métodos são para a resolução de problemas do mundo real.

De um modo geral, concluímos que a função ODE45 se destacou imenso dos outros métodos tendo tanto uma melhor precisão como também conseguiu chegar à solução exata em muito poucas iterações. O método de Runge-Kutta de ordem 4 também foi um excelente competidor aproximando-se imenso da solução exata do problema. O método de Euler deixou um pouco a desejar sendo que os erros do método eram bastante maiores do que os erros dos outros métodos e a solução dada pelo método de Euler ficava bastantes vezes aquém da solução exata.

Graças ao tempo que despendemos neste trabalho, podemos dizer com toda a certeza que as nossas competências no que toca à programação em MATLAB beneficiaram imenso.

