



Universidade Federal da Paraíba  
Centro de Ciências Exatas e da Natureza  
Departamento de Matemática  
Mestrado Profissional em Matemática  
em Rede Nacional PROFMAT



# Métodos de Euler e Runge-Kutta: Uma Análise Utilizando o Geogebra<sup>†</sup>

por

**Manoel Wallace Alves Ramos**

sob a orientação do

**Prof. Dr. Flank David Morais Bezerra**

Dissertação apresentada ao Corpo Docente do Mestrado Profissional em Matemática em Rede Nacional PROFMAT-CCEN-UFPA, como requisito parcial para a obtenção do título de Mestre em Matemática.

Junho/2017  
João Pessoa - PB

---

<sup>†</sup> O presente trabalho foi realizado com apoio da CAPES, Coordenação de Aperfeiçoamento de Pessoal de Nível Superior.

R175m Ramos, Manoel Wallace Alves.  
Métodos de Euler e Runge-Kutta: uma análise utilizando o  
Geogebra / Manoel Wallace Alves Ramos. - João Pessoa,  
2017.  
66 f.: il. -  
  
Orientador: Flank David Morais Bezerra.  
Dissertação (Mestrado) - UFPB/ CCEN  
  
1. Equações Diferenciais. 2. Métodos Numéricos. 3. Applet.  
4. Geogebra. 5. Método de Euler. 6. Método de Runge-Kutta.  
I. Título.

UFPB/BC

CDU: 517.9(043)

# Métodos de Euler e Runge-Kutta: Uma Análise Utilizando o Geogebra

por

**Manoel Wallace Alves Ramos**

Dissertação apresentada ao Corpo Docente do Mestrado Profissional em Matemática em Rede Nacional PROFMAT CCEN-UFPB, como requisito parcial para a obtenção do título de Mestre em Matemática.

Área de Concentração: Ensino de Matemática.

Aprovada por:

Flank David Morais Bezerra

Prof. Dr. Flank David Morais Bezerra -UFPB (Orientador)

Elisandra F. gloss de Moraes

Profª. Drª. Elisandra de Fátima Gloss de Moraes - UFPB

Miriam da Silva Pereira

Profª. Drª. Miriam da Silva Pereira - UFPB

Esteban Pereira da Silva

Prof. Dr. Esteban Pereira da Silva - Pós-Doc. UFPB/UFPE

Junho/2017

# Agradecimentos

Primeiramente a Deus, pois sem ele nada é possível.

Aos meu pais, Manoel Cícero Dias Ramos (*in memoriam*) e Marié Alves Ramos, pelo amor incondicional, pelo apoio sem medir esforços, pela confiança e incentivos.

Agradeço a minha esposa, Darlene, pela constante paciência e pelo apoio, amor e carinho durante esta jornada.

Agradeço a minha filha, Manuela, que me motiva e me alegra todos os dias.

Ao meu irmão, Wanderson Alves, por sempre me incentivar.

A toda minha família, por sempre acreditar em mim e pelo grande carinho.

Ao professor Flank Bezerra, por ter me orientado de forma paciente, objetiva e muito competente.

Agradeço aos participantes da banca, Prof. Flank David Moraes Bezerra, Prof<sup>a</sup>. Elisandra Moraes, Prof<sup>a</sup>. Miriam da Silva Pereira, Prof. Esteban Pereira da Silva, pelas importantes sugestões neste trabalho.

Aos professores do PROFMAT, Bruno, Carlos Bocker, Eduardo, Elisandra, Flank, Lenimar, Miriam e Pedro Hinojosa, por todos os ensinamentos.

Sou grato aos amigos, Adim, David, Diego, Eduardo, Erielson, Gustavo, José Carlos, Leônidas, Mailson, Rafael, Ramon e Rômulo, pela ótima convivência ao longo do mestrado e por terem compartilhado comigo conhecimento, dificuldades, mas também muitas risadas.

À CAPES, pelo apoio financeiro.

Agradeço a todos que direta ou indiretamente contribuíram para conclusão deste trabalho.

# Dedicatória

*A minha filha, Manuela Herculano  
Ramos.*

# Resumo

É evidente a importância das equações diferenciais ordinárias na modelagem de problemas em diversas áreas da ciência, bem como o uso de métodos numéricos para resolver tais equações. Os computadores são uma ferramenta extremamente útil no estudo de equações diferenciais, uma vez que através deles é possível executar algoritmos que constroem aproximações numéricas para soluções destas equações. Este trabalho é uma introdução ao estudo de métodos numéricos para equações diferenciais ordinárias. Apresentamos os métodos numéricos de Euler, Euler melhorado e a classe de métodos de Runge-Kutta. Além disso, com o propósito de colaborar com o ensino e aprendizagem de tais métodos, propomos e mostramos a construção de um *applet* criado a partir do uso de ferramentas do software *Geogebra*. O *applet* fornece soluções numéricas aproximadas para um problema de valor inicial, bem como exibe os gráficos das soluções que são obtidas a partir dos métodos numéricos de Euler, Euler melhorado e Runge-Kutta de quarta ordem.

**Palavras-chave:** Método de Euler, Método de Runge-Kutta, Applet, Geogebra, Métodos Numéricos, Equações Diferenciais.

# Abstract

Is evident the importance of ordinary differential equations in modeling problems in several areas of science. Coupled with this, is increasing the use of numerical methods to solve such equations. Computers have become an extremely useful tool in the study of differential equations, since through them it is possible to execute algorithms that construct numerical approximations for solutions of these equations. This work introduces the study of numerical methods for ordinary differential equations presenting the numerical Euler's method, improved Euler's method and the class of Runge-Kutta's methods. In addition, in order to collaborate with the teaching and learning of such methods, we propose and show the construction of an *applet* created from the use of *Geogebra* software tools. The *applet* provides approximate numerical solutions to an initial value problem, as well as displays the graphs of the solutions that are obtained from the numerical Euler's method, improved Euler's method, and fourth-order Runge-Kutta's method.

**Keywords:** Euler's Method, Runge-Kutta's Method, Applet, Geogebra, Numerical Methods, Differential Equations.

# Sumário

|          |                                                                               |           |
|----------|-------------------------------------------------------------------------------|-----------|
| <b>1</b> | <b>Existência e Unicidade de Soluções de Equações Diferenciais Ordinárias</b> | <b>1</b>  |
| 1.1      | Teorema de Picard . . . . .                                                   | 2         |
| <b>2</b> | <b>Método de Euler</b>                                                        | <b>7</b>  |
| 2.1      | O método de Euler . . . . .                                                   | 7         |
| 2.1.1    | Formas alternativas . . . . .                                                 | 11        |
| 2.2      | Erros em aproximações numéricas . . . . .                                     | 13        |
| 2.2.1    | Erro de truncamento global . . . . .                                          | 13        |
| 2.2.2    | Erro de truncamento local . . . . .                                           | 13        |
| 2.2.3    | Erro de arredondamento . . . . .                                              | 13        |
| 2.3      | Erro de truncamento local para o método de Euler . . . . .                    | 14        |
| 2.4      | Método de Euler melhorado (Método de Heun) . . . . .                          | 16        |
| 2.4.1    | Forma alternativa . . . . .                                                   | 19        |
| <b>3</b> | <b>Métodos de Runge-Kutta</b>                                                 | <b>21</b> |
| 3.1      | Runge-Kutta de segunda ordem . . . . .                                        | 22        |
| 3.2      | Runge-Kutta de terceira ordem . . . . .                                       | 23        |
| 3.3      | Runge-Kutta de quarta ordem . . . . .                                         | 24        |
| <b>4</b> | <b>Applet no Geogebra para soluções numéricas de PVI</b>                      | <b>28</b> |
| 4.1      | O <i>Geogebra</i> . . . . .                                                   | 28        |
| 4.2      | Uso do <i>applet</i> . . . . .                                                | 29        |
| 4.3      | Construção do <i>applet</i> . . . . .                                         | 31        |
| 4.3.1    | Passo 1 - Criando ponto inicial e controle deslizante . . . . .               | 31        |
| 4.3.2    | Passo 2 - Obtendo a solução exata do PVI . . . . .                            | 33        |
| 4.3.3    | Passo 3 - Implementando o método de Euler . . . . .                           | 34        |
| 4.3.4    | Passo 4 - Criando a tabela do método de Euler . . . . .                       | 35        |
| 4.3.5    | Passo 5 - Implementando o método de Euler melhorado . . . . .                 | 36        |
| 4.3.6    | Criando a tabela do método de Euler melhorado . . . . .                       | 39        |



|        |                                                                                                        |    |
|--------|--------------------------------------------------------------------------------------------------------|----|
| 4.3.7  | Passo 6 - Implementando o método Runge-Kutta de quarta ordem . . . . .                                 | 39 |
| 4.3.8  | Passo 7 - Criando a tabela do método de Runge-Kutta . . . .                                            | 41 |
| 4.3.9  | Passo 8 - Transferindo as tabelas para a janela de visualização                                        | 42 |
| 4.3.10 | Passo 9 - Abrindo outra janela de visualização e criando textos                                        | 43 |
| 4.3.11 | Passo 10 - Criando campos de entrada e caixas para exibir e esconder as poligonais e tabelas . . . . . | 45 |
| 4.3.12 | Passo 11 - Disponibilizando o applet na web. . . . .                                                   | 49 |

|                                   |           |
|-----------------------------------|-----------|
| <b>Referências Bibliográficas</b> | <b>52</b> |
|-----------------------------------|-----------|

# Lista de Figuras

|      |                                                                                                                                           |    |
|------|-------------------------------------------------------------------------------------------------------------------------------------------|----|
| 2.1  | Método de Euler para determinar soluções próximas da solução exata do PVI (1.1)-(1.2). . . . .                                            | 8  |
| 2.2  | Solução exata do PVI do Exemplo 2.1.1 e soluções aproximadas pelo método de Euler com $h = 0.05$ e $h = 0.1$ . . . . .                    | 10 |
| 2.3  | Dedução integral do método de Euler. . . . .                                                                                              | 12 |
| 2.4  | Método de Euler melhorado para determinar soluções próximas da solução exata do PVI (1.1)-(1.2). . . . .                                  | 17 |
| 2.5  | Solução exata do PVI do Exemplo 2.4.1 e solução aproximada pelo método de Euler melhorado com $h = 0.1$ . . . . .                         | 18 |
| 2.6  | Dedução integral do método de Euler melhorado. . . . .                                                                                    | 20 |
| 3.1  | Solução exata do PVI do Exemplo 3.3.1 e soluções aproximadas pelo método de Runge-Kutta de 4ª ordem com $h = 0.1$ . . . . .               | 26 |
| 4.1  | Applet: Métodos numéricos para EDO. Disponível em <a href="https://geogebra.org/m/xTz9CaRY">https://geogebra.org/m/xTz9CaRY</a> . . . . . | 30 |
| 4.2  | Exibindo a planilha. . . . .                                                                                                              | 31 |
| 4.3  | Criando o ponto inicial. . . . .                                                                                                          | 32 |
| 4.4  | Configurando o controle deslizante. . . . .                                                                                               | 32 |
| 4.5  | Controle deslizante $h$ . . . . .                                                                                                         | 33 |
| 4.6  | Solução exata. . . . .                                                                                                                    | 33 |
| 4.7  | Método de Euler. . . . .                                                                                                                  | 34 |
| 4.8  | Pontos do método de Euler. . . . .                                                                                                        | 34 |
| 4.9  | Poligonal do método de Euler. . . . .                                                                                                     | 35 |
| 4.10 | Tabela do método de Euler. . . . .                                                                                                        | 36 |
| 4.11 | Arredondamento. . . . .                                                                                                                   | 36 |
| 4.12 | Fórmula do método de Euler melhorado. . . . .                                                                                             | 37 |
| 4.13 | Fórmula do método de Euler melhorado atualizada. . . . .                                                                                  | 37 |
| 4.14 | De H4 para H5. . . . .                                                                                                                    | 37 |
| 4.15 | De I4 para I5. . . . .                                                                                                                    | 37 |
| 4.16 | Método de Euler melhorado. . . . .                                                                                                        | 38 |
| 4.17 | Poligonal do método de Euler melhorado. . . . .                                                                                           | 38 |

|      |                                                                                                                                    |    |
|------|------------------------------------------------------------------------------------------------------------------------------------|----|
| 4.18 | Poligonal do método de Euler melhorado. . . . .                                                                                    | 39 |
| 4.19 | Definindo os parâmetros. . . . .                                                                                                   | 40 |
| 4.20 | Fórmula do método de Runge-Kutta de quarta ordem. . . . .                                                                          | 40 |
| 4.21 | Pontos do método de Runge-Kutta de quarta ordem. . . . .                                                                           | 40 |
| 4.22 | Poligonal do método de Runge-Kutta de quarta ordem. . . . .                                                                        | 41 |
| 4.23 | Poligonal do método de Euler melhorado. . . . .                                                                                    | 42 |
| 4.24 | Transferindo a Tabela. . . . .                                                                                                     | 42 |
| 4.25 | Tabelas na janela de visualização. . . . .                                                                                         | 43 |
| 4.26 | Caixa de texto. . . . .                                                                                                            | 44 |
| 4.27 | Textos do <i>applet</i> . . . . .                                                                                                  | 45 |
| 4.28 | Configurando o campo de entrada. . . . .                                                                                           | 46 |
| 4.29 | Campos de entrada. . . . .                                                                                                         | 46 |
| 4.30 | Exibindo/escondendo a poligonal e a tabela do método Euler. . . . .                                                                | 47 |
| 4.31 | Caixa para Exibir/esconder a poligonal e a tabela do método Euler. . . . .                                                         | 47 |
| 4.32 | Exibindo/escondendo a poligonal e a tabela do método Euler melhorado. . . . .                                                      | 48 |
| 4.33 | Exibindo/escondendo a poligonal e a tabela do método de Runge-Kutta de quarta ordem. . . . .                                       | 48 |
| 4.34 | Caixa para Exibir/esconder as poligonais e as tabelas dos métodos de Euler, Euler melhorado e Runge-Kutta de quarta ordem. . . . . | 49 |
| 4.35 | Upload do <i>applet</i> . . . . .                                                                                                  | 49 |

# Lista de Tabelas

|     |                                                                                                         |    |
|-----|---------------------------------------------------------------------------------------------------------|----|
| 2.1 | Resultados do método de Euler com $h = 0.1$ aplicado ao PVI do Exemplo 2.1.1. . . . .                   | 9  |
| 2.2 | Resultados do método de Euler com $h = 0.05$ aplicado ao PVI do Exemplo 2.1.1. . . . .                  | 10 |
| 2.3 | Resultados do método de Euler melhorado com $h = 0.1$ aplicado ao PVI do Exemplo 2.4.1. . . . .         | 18 |
| 3.1 | Resultados do método de Runge-Kutta de 4ª ordem com $h = 0.1$ aplicado ao PVI do Exemplo 3.3.1. . . . . | 26 |

# Introdução

Diversos problemas da física, engenharia, economia, biologia, medicina e em outras áreas da ciência são modelados por Equações Diferenciais Ordinárias (EDOs). Por exemplo, trajetórias balísticas, teoria dos satélites artificiais, estudo de redes elétricas, curvaturas de vigas, estabilidade de aviões, teoria das vibrações, reações químicas e outras aplicações estão relacionadas com equações diferenciais. Muitos matemáticos estudaram a natureza dessas equações por centenas de anos e há muitas técnicas de solução bem desenvolvidas. Os precursores no estudo e desenvolvimento das equações diferenciais foram Isaac Newton (1642-1727) e Gottfried Wilhelm Leibniz (1646-1716), no final do século XVII.

Mesmo que se possa demonstrar que existe uma solução de uma equação diferencial, nem sempre somos capazes de exibi-la de forma explícita. Quando isso ocorre, recorreremos aos **métodos numéricos**<sup>3</sup>.

O primeiro método para calcular uma solução aproximada de uma EDO, a partir de uma condição inicial, foi desenvolvido por Leonhard Euler (1707-1783). O método de Euler, que data de 1768, ainda está “vivo”, não só porque ele desempenha um papel de destaque no ensino e na base metodológica para explicar os métodos mais complicados, mas também por ainda ser usado para obter uma primeira aproximação na solução de EDOs. Generalizações do método de Euler foram desenvolvidas por Carl Runge (1856-1927) em 1895 e 1908, Karl Heun (1859-1929) em 1900 e por Martin Wilhelm Kutta (1867-1944) em 1901. Estes pesquisadores contribuíram para a formulação dos, hoje bastante conhecidos, métodos de Runge-Kutta. Assim, os métodos de Euler e Runge-Kutta formam o bloco dos procedimentos de **passo simples (ou passo único)**, isto é, métodos em que para se obter o valor aproximado  $y_{n+1}$  é necessário apenas conhecer o seu antecessor  $y_n$ .

O primeiro **método de passos múltiplos**<sup>4</sup>, para a resolução de equações diferenciais foi apresentado pelo famoso astrônomo britânico John Couch Adams (1819-1892). Adams, com base no método de Euler e utilizando a equação de Bashforth

---

<sup>3</sup>Os métodos numéricos correspondem a um conjunto de ferramentas ou métodos usados para se obter a solução de problemas matemáticos de forma aproximada. Tais métodos podem ser usados para a obtenção de soluções numéricas para problemas quando, por qualquer razão, não podemos ou não desejamos usar métodos analíticos.

<sup>4</sup>Métodos que utilizam a informação de valores anteriores a  $y_{n+1}$  para estimá-lo

---

(1819-1912), propôs uma nova técnica para encontrar soluções aproximadas para EDOs, conhecida como método de Adams-Bashforth. Em 1925 o astrônomo americano Forest Ray Moulton (1872-1952) propôs uma classe de métodos conhecida por Adams-Moulton. Tais métodos são considerados uma melhoria ao método de Adams-Bashforth. Durante o século XX diversos outros matemáticos contribuíram para o avanço de métodos numéricos para EDOs. Para mais informações sugerimos [6].

Nos últimos anos, o desenvolvimento de computadores com melhor capacidade computacional viabilizou o aumento de problemas que podem ser investigados, de maneira efetiva, por métodos numéricos. Nesse período os métodos numéricos para EDOs começaram a ser utilizados de forma sistemática. Nos dias de hoje, com todo o avanço computacional, utilizar métodos numéricos para encontrar soluções aproximadas de EDOs tem custo computacional baixo e uma elevada precisão. Segundo Arenales [1], embora as soluções analíticas ainda sejam extremamente valiosas, tanto para solução de problemas quanto para fornecer uma visão geral, os métodos numéricos representam alternativas que aumentam enormemente os recursos para confrontar e resolver problemas. Neste trabalho, apresentamos um estudo dos métodos numéricos de Euler e Runge-Kutta. Os métodos são apresentados no contexto o mais simples possível, ou seja, uma única equação de primeira ordem. No entanto, eles podem ser facilmente estendidos para sistemas de equações de primeira ordem, veja [10]. Também propomos um *applet*, criado no software *Geogebra*, que fornece as soluções numéricas de um PVI a partir dos métodos de Euler, Euler melhorado e Runge-Kutta de quarta ordem.

No Capítulo 1, apresentamos e demonstramos o Teorema de Picard que trata da existência e unicidade de soluções de EDOs. No Capítulo 2, estudamos os métodos de Euler e Euler melhorado, apresentamos formas alternativas de interpretá-los, além disso, fazemos uma breve análise dos erros associados a esses métodos. No Capítulo 3, abordamos os métodos de Runge-Kutta de 2ª, 3ª e 4ª ordem e mostramos a eficiência e precisão do método de Runge-Kutta de 4ª ordem sobre os métodos de Euler e Euler melhorado. No Capítulo 4, comentamos sobre a importância do uso de *applets* na aprendizagem matemática e propomos um *applet* que gera soluções numéricas de um PVI utilizando os métodos de Euler, Euler melhorado e Runge-Kutta de quarta ordem, além disso, mostramos como construí-lo.

Informamos que todos os cálculos e gráficos do trabalho foram feitos utilizando o software *Geogebra*.

# Capítulo 1

## Existência e Unicidade de Soluções de Equações Diferenciais Ordinárias

Neste trabalho, inicialmente, vamos apresentar e analisar alguns métodos numéricos para soluções de EDOs de primeira ordem na forma

$$y' = f(x, y), \quad (1.1)$$

em que  $y$  é uma função real na variável  $x \in \mathbb{R}$  e  $y' = dy/dx$ . A equação (1.1) pode possuir infinitas soluções, no entanto, estamos interessados em descobrir uma solução particular que satisfaça a seguinte condição inicial: dados dois números reais  $x_0$  e  $y_0$  queremos uma solução para (1.1), tal que

$$y(x_0) = y_0. \quad (1.2)$$

Quando consideramos a equação (1.1) juntamente com a equação (1.2) temos então um **problema de valor inicial (PVI)**.

Duas questões fundamentais surgem ao considerarmos um problema de valor inicial: a solução deste problema existe? Se a solução existir, ela é única? Embora existam muitos critérios alternativos para responder estas perguntas de forma satisfatória, focamos aqui na condição de Lipschitz. Isto é especialmente conveniente porque o mesmo tipo de condição pode ser usado para estudar o comportamento de aproximações.

De forma geral, mesmo que  $f(\cdot, \cdot)$  seja uma função contínua, não há garantia que um PVI possua uma única solução. No entanto, se a função  $f$  estiver sob hipóteses adequadas a existência e unicidade da solução do PVI pode ser assegurada. No Teorema 1.3 (Teorema de Picard<sup>1</sup>) mostraremos esse resultado.

As principais referências bibliográficas para a elaboração deste capítulo foram [16] e [9].

---

<sup>1</sup>Charles Émile Picard (1856-1941), habitualmente referido apenas como Émile Picard, foi um matemático francês, nascido em 24 de Julho de 1856, cujas teorias foram importantes para o avanço da pesquisa em análise, geometria algébrica, e mecânica.

## 1.1 Teorema de Picard

Antes de apresentarmos o Teorema de Picard enunciaremos dois teoremas que auxiliarão na demonstração do mesmo.

**Teorema 1.1.** *Dado  $X \subset \mathbb{R}$ , se uma sequência de funções  $f_n : X \rightarrow \mathbb{R}$  converge uniformemente para  $f : X \rightarrow \mathbb{R}$  e cada  $f_n$  é contínua no ponto  $a \in X$ , então  $f$  é contínua no ponto  $a$ .*

**Teorema 1.2 (Teste de Weierstrass).** *Dada a sequência de funções  $f_n : X \rightarrow \mathbb{R}$ , seja  $\sum a_n$  uma série convergente de números reais  $a_n \geq 0$  tais que  $|f_n(x)| \leq a_n$  para todo  $n \in \mathbb{N}$  e todo  $x \in X$ . Nestas condições, as séries  $\sum |f_n|$  e  $\sum f_n$  são uniformemente convergentes.*

As demonstrações dos Teoremas 1.1 e 1.2 podem ser encontradas em ([12], p. 159 e 163), respectivamente.

**Teorema 1.3 (Teorema de Picard).** *Suponha que  $f(\cdot, \cdot)$  seja uma função contínua em uma região  $U$  do plano  $(x, y)$  que contém o retângulo*

$$R = \{(x, y) : |x - x_0| \leq h, |y - y_0| \leq k\},$$

*em que  $h$  e  $k$  são constantes. Suponha também que existe uma constante positiva  $L$  tal que*

$$|f(x, y_1) - f(x, y_2)| \leq L|y_1 - y_2|, \quad (1.3)$$

*com  $(x, y_1)$  e  $(x, y_2)$  pertencentes ao retângulo  $R$ . Considerando*

$$M = \sup\{|f(x, y)| : (x, y) \in R\},$$

*suponha que  $Mh \leq k$ . Então existe uma única função contínua e diferenciável  $y(x)$ , definida no intervalo  $|x - x_0| \leq h$ , que satisfaz (1.1) e (1.2).*

A condição (1.3) é denominada **condição Lipschitz**<sup>2</sup>, e  $L$  é chamado de **constante de Lipschitz** para  $f$ .

**Demonstração:** Considere  $\psi : I_h \rightarrow J_k$ , em que  $I_h = [x_0 - h, x_0 + h]$  e  $J_k = [y_0 - k, y_0 + k]$ , uma função contínua qualquer (elas existem, uma vez que a função constante  $y_0$  o é). Definimos

$$\psi^*(x) = y_0 + \int_{x_0}^x f(t, \psi(t)) dt \quad \forall x \in I_h.$$

---

<sup>2</sup>Rudolf Otto Sigismund Lipschitz (1832-1903) foi um matemático alemão que trabalhou em quase todos os ramos da matemática pura e aplicada de seu tempo. Ficou bastante conhecido por sua obra intitulada *Lehrbuch der Analysis*.



Mostraremos que  $\psi^*$  é uma função contínua definida de  $I_h$  em  $J_k$ .

De fato,  $\psi^*$  esta bem definida em  $I_h$ , já que  $\psi$  é contínua em  $I_h$ , o que implica que  $f(x, \psi(x))$  esta bem definida e é contínua em  $I_h$  e, portanto, integrável. Além disso, segue da definição de  $\psi^*$  e do Teorema Fundamental do Cálculo que  $\psi^*$  é derivável em  $I_h$  e, portanto, contínua.

Resta mostra que  $\psi^*(x) \in J_k$  qualquer que seja  $x \in I_h$ . De fato, dado  $t \in I_h$  tem-se que  $\psi \in J_k$ , donde  $(t, \psi(t)) \in R$  e então  $|f(t, \psi(t))| \leq M$ . Assim, dado  $x \in I_h$  tem-se que

$$\begin{aligned} |\psi^*(x) - y_0| &= \left| y_0 + \int_{x_0}^x f(t, \psi(t))dt - y_0 \right| = \left| \int_{x_0}^x f(t, \psi(t))dt \right| \\ &\leq \left| \int_{x_0}^x |f(t, \psi(t))|dt \right| \leq \left| \int_{x_0}^x Mdt \right| = M|x - x_0| \\ &\leq Mh \leq k, \end{aligned}$$

o que mostra que  $\psi^*(x) \in J_k$ .

Considere a sequência  $(y_n)$  de funções  $y_n : I_h \rightarrow J_k$ , definida recursivamente e conhecida como *Iteração de Picard*:

$$\begin{aligned} y_0(x) &= y_0, \\ y_n(x) &= y_0 + \int_{x_0}^x f(t, y_{n-1}(t))dt, \quad n = 1, 2, \dots \end{aligned}$$

Observe que  $y_n$  é contínua em  $I_h$ , de acordo com a discussão acima, levando em consideração que  $y_0$  é uma função contínua definida de  $I_h$  em  $J_k$ .

Mostraremos a seguir que a sequência de funções  $(y_n)$  é uniformemente convergente. Note que

$$|y_n(x) - y_{n-1}(x)| \leq \frac{L^{n-1}M}{n!} |x - x_0|^n, \quad \forall x \in I_h \text{ e } n = 1, 2, \dots, \quad (1.4)$$

o que é possível mostrar por indução sobre  $n$ . Para  $n = 1$  tem-se que

$$\begin{aligned} |y_1 - y_0| &= \left| \int_{x_0}^x f(t, y_0)dt \right| \leq \left| \int_{x_0}^x |f(t, y_0)|dt \right| \\ &\leq Mh \leq k. \end{aligned}$$

Supondo por indução que (1.4) é verificada para algum  $n \geq 2$  e usando a condição

Lipschitz (1.3), tem-se

$$\begin{aligned}
 |y_n(x) - y_{n-1}(x)| &= \left| \int_{x_0}^x [f(t, y_{n-1}(t)) - f(t, y_{n-2}(t))] dt \right| \\
 &\leq \int_{x_0}^x |f(t, y_{n-1}(t)) - f(t, y_{n-2}(t))| dt \\
 &\leq \int_{x_0}^x L |y_{n-1}(t) - y_{n-2}(t)| dt = L \int_{x_0}^x |y_{n-1}(t) - y_{n-2}(t)| dt \\
 &\leq L \int_{x_0}^x \frac{L^{n-2} M}{(n-1)!} |t - x_0|^{n-1} dt \\
 &= \frac{L^{n-1} M}{n!} |x - x_0|^n,
 \end{aligned}$$

isto é, (1.4) também é verificada para  $n + 1$ . Isso conclui a prova de que (1.4) se verifica para todo  $n \in \mathbb{N}$ . Mais que isso,

$$|y_n(x) - y_{n-1}(x)| \leq \frac{L^{n-1} h^n M}{n!}, \quad \forall x \in I_h \text{ e } n = 1, 2, \dots$$

Observe agora que a série

$$\sum_{n=1}^{\infty} \frac{L^{n-1} h^n M}{n!}$$

converge para

$$\frac{M}{L} (e^{Lh} - 1).$$

Pelo Teste de Weierstrass (Teorema 1.2), tem-se que a série  $\sum_{n=1}^{\infty} (y_n - y_{n-1})$  converge uniformemente em  $I_h$ . Logo, a sequência  $(y_n)$  também converge uniformemente em  $I_h$ .

Seja

$$y = \lim_{n \rightarrow \infty} y_n.$$

Segue do Teorema 1.1 que  $y$  é uma função contínua. Fazendo  $n \rightarrow \infty$  nos dois lados da equação

$$y_n(x) = y_0 + \int_{x_0}^x f(t, y_{n-1}(t)) dt,$$

tem-se que

$$y(x) = y_0 + \int_{x_0}^x f(t, y(t)) dt.$$

Sendo assim,  $y = y(x)$  satisfaz a equação (1.1) e a condição (1.2). Mostramos que existe uma solução para o PVI (1.1)-(1.2).

Para mostrar a unicidade da solução do PVI (1.1)-(1.2), suponha que  $\varphi : I_h \rightarrow J_k$  seja uma dessas soluções. Ou seja,

$$\varphi(x) = y_0 + \int_{x_0}^x f(t, \varphi(t)) dt.$$

Então, de maneira semelhante ao que foi feito anteriormente, pode-se mostrar por indução que

$$|\varphi(x) - y_n(x)| \leq \frac{L^{n-1} h^n M}{n!}$$

para todo  $n \in \mathbb{N}$ . Como

$$\frac{L^{n-1} h^n M}{n!} \rightarrow 0$$

quando  $n \rightarrow \infty$ , então  $y_n \rightarrow \varphi$ , onde a convergência é evidentemente uniforme. Segue da unicidade do limite de  $(y_n)$  que  $y = \varphi$ . ■

### Exemplo 1.1.1.

Considere a EDO

$$y' = ay,$$

$a \in \mathbb{R}^*$ .

Temos que a função  $f(x, y) = ay$  é contínua e satisfaz a condição de Lipschitz, uma vez que

$$|f(x, y_1) - f(x, y_2)| = |ay_1 - ay_2| = |a(y_1 - y_2)| = |a||y_1 - y_2|.$$

Sendo assim, pelo Teorema 1.3, o PVI formado pela equação  $y' = ay$  e a condição inicial  $y(x_0) = y_0$ , tem solução única em  $-\infty < x < \infty$ . Por métodos elementares encontramos a solução do PVI, que é dada por

$$y(x) = y_0 e^{a(x-x_0)}.$$

### Exemplo 1.1.2.

Considere a EDO

$$y' = y^2.$$

Mostraremos que  $f(x, y) = y^2$  satisfaz a condição de Lipschitz. Observe que

$$|f(x, y_1) - f(x, y_2)| = |y_1^2 - y_2^2| = |(y_1 + y_2)(y_1 - y_2)| = |y_1 + y_2||y_1 - y_2|.$$

Para  $y_1$  e  $y_2$  suficientemente próximos de  $y_0$ , temos

$$y_1, y_2 \in [y_0 - k, y_0 + k] \Rightarrow y_1 + y_2 \in (-2y_0, 2y_0).$$

Dessa forma, temos que  $|y_1 + y_2| \leq 4y_0$ . Sendo assim, para  $L = 4y_0$  e  $y$  suficientemente próximo de  $y_0$  a função  $f(x, y) = y^2$  satisfaz a condição Lipschitz. Consequentemente, pelo Teorema 1.3, concluímos que existe uma única solução  $y(x)$ , em algum intervalo  $(x_0 - h, x_0 + h)$ , satisfazendo  $y' = y^2$ ,  $y(x_0) = y_0$ .

**Exemplo 1.1.3.**

Considere o PVI

$$y' = 3y^{2/3}, \quad y(0) = 0.$$

Claramente  $y(x) = 0$  é uma solução do PVI. Utilizando métodos elementares, obtemos uma segunda solução para o PVI, à saber  $y(x) = x^3$ . Portanto, temos duas soluções para o PVI.

Note que a  $f(x, y) = 3y^{2/3}$  não satisfaz a condição Lipschitz, pois tomando  $y_2 = 0$  temos que

$$|f(x, y_1) - f(x, y_2)| = |f(x, y_1) - f(x, 0)| = 3y_1^{2/3}.$$

Dessa forma,

$$\frac{|3y_1^{2/3} - 0|}{|y_1 - 0|} = \frac{3}{|y_1^{1/3}|}.$$

Para  $y_1 \rightarrow 0$ , tem-se que

$$\frac{3}{|y_1^{1/3}|} \rightarrow \infty.$$

Sendo assim, a relação

$$|f(x, y_1) - f(x, 0)| \leq L|y_1 - 0|$$

não é satisfeita em torno de  $(0, 0)$ .

# Capítulo 2

## Método de Euler

Dentre os métodos numéricos, amplamente divulgados na literatura especializada, para resolução de PVI's, o método mais simples é o método de Euler. O método foi desenvolvido por Euler e publicado em 1768 na obra intitulada *Institutiones Calculi Integralis*. O método de Euler é considerado um dos primeiros métodos numéricos avançados utilizado nos dias de hoje. Neste Capítulo, apresentaremos os métodos de Euler e Euler melhorado.

As principais referências bibliográficas para a elaboração deste capítulo foram [4], [7], [8], [14] e [18].

### 2.1 O método de Euler

No que segue, descrevemos o método: suponha que queremos aproximar a solução do PVI (1.1)-(1.2) em  $x = x_1 = x_0 + h$ , em que  $h$  é pequeno e é chamado de passo. A ideia por trás do método de Euler é usar a reta tangente à curva da solução do PVI através de  $(x_0, y_0)$  para obter tal aproximação. Veja Figura 2.1.

Sabemos que a equação da reta tangente a uma curva em  $(x_0, y_0)$  é dada por

$$y(x) = y_0 + m(x - x_0),$$

em que  $m$  é a inclinação da curva em  $(x_0, y_0)$ . Da equação (1.1), temos que  $m = f(x_0, y_0)$ , então

$$y(x) = y_0 + f(x_0, y_0)(x - x_0). \quad (2.1)$$

Fazendo  $x = x_1$  na equação (2.1) encontramos a aproximação de Euler para a solução exata em  $x_1$ , ou seja,

$$y_1(x_1) = y_0 + f(x_0, y_0)(x_1 - x_0),$$

que pode ser escrita como

$$y_1 = y_0 + hf(x_0, y_0).$$

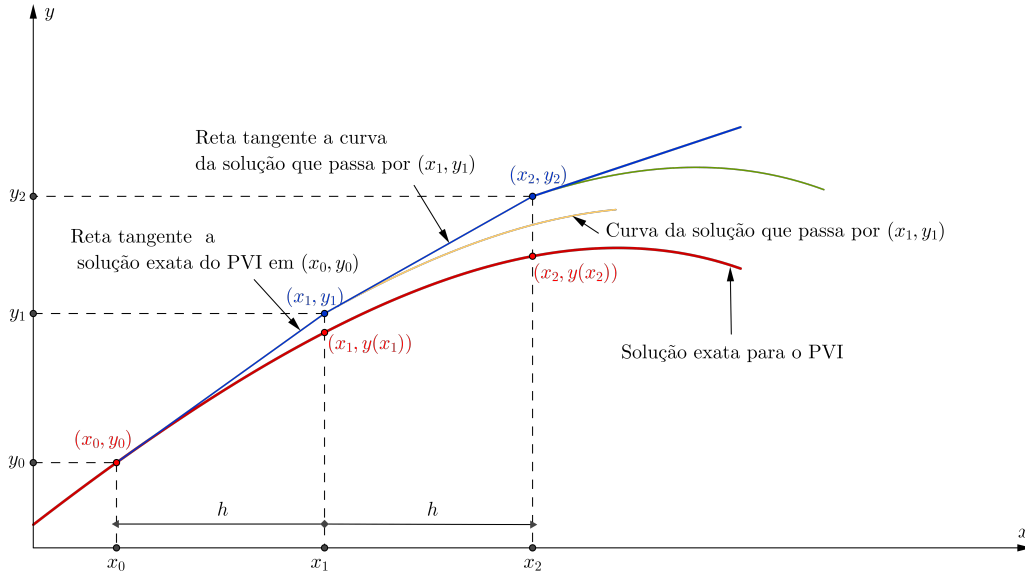


Figura 2.1: Método de Euler para determinar soluções próximas da solução exata do PVI (1.1)-(1.2).

Agora suponha que desejamos encontrar uma aproximação para a solução exata do PVI (1.1)-(1.2) em  $x_2 = x_1 + h$ . Podemos usar a mesma ideia, só que agora tomamos a reta tangente à curva da solução de (1.1) em  $(x_1, y_1)$ . Neste caso, a inclinação da reta tangente é  $f(x_1, y_1)$ , então a equação da reta tangente é

$$y(x) = y_1 + f(x_1, y_1)(x - x_1).$$

Fazendo  $x = x_2$  temos a aproximação desejada

$$y_2 = y_1 + hf(x_1, y_1),$$

em que  $h = x_2 - x_1$ . Continuando este procedimento, determinamos uma sequência de aproximações

$$y_{n+1} = y_n + hf(x_n, y_n), \quad n = 0, 1, \dots$$

para a solução do PVI (1.1)-(1.2) nos pontos  $x_{n+1} = x_n + h$ .

Dessa forma, o método de Euler para uma aproximação da solução do PVI (1.1)-(1.2) através dos pontos  $x_{n+1} = x_0 + nh$  ( $n = 0, 1, \dots$ ) é dado por

$$y_{n+1} = y_n + hf(x_n, y_n), \quad n = 0, 1, \dots \quad (2.2)$$

**Exemplo 2.1.1.**

Considere o problema de valor inicial

$$y' = x - y - 1, \quad y(0) = 1.$$

Usando o método de Euler com  $h = 0.1$  e  $h = 0.05$ , obtemos uma solução aproximada para o PVI em  $[0, 1]$  com 5 casas decimais. Os valores da solução aproximada estão apresentados nas Tabelas 2.1 e 2.2, respectivamente. Ainda nas Tabelas 2.1 e 2.2, apresentamos a solução exata do PVI que é dada por

$$y(x) = 3e^{-x} + x - 2$$

e o **erro absoluto** que é definido por  $|y(x_n) - y_n|$ .

Tabela 2.1: Resultados do método de Euler com  $h = 0.1$  aplicado ao PVI do Exemplo 2.1.1.

| $n$ | $x_n$ | $y_n$   | Solução exata | Erro absoluto |
|-----|-------|---------|---------------|---------------|
| 1   | 0.1   | 0.8     | 0.81451       | 0.01451       |
| 2   | 0.2   | 0.63    | 0.65619       | 0.02619       |
| 3   | 0.3   | 0.487   | 0.52245       | 0.03545       |
| 4   | 0.4   | 0.3683  | 0.41096       | 0.04266       |
| 5   | 0.5   | 0.27147 | 0.31959       | 0.04812       |
| 6   | 0.6   | 0.19432 | 0.24643       | 0.05211       |
| 7   | 0.7   | 0.13489 | 0.18976       | 0.05487       |
| 8   | 0.8   | 0.0914  | 0.14799       | 0.05659       |
| 9   | 0.9   | 0.06226 | 0.11971       | 0.05745       |
| 10  | 1     | 0.04604 | 0.10364       | 0.05760       |

Comparando os valores das aproximações apresentados nas Tabelas 2.1 e 2.2, vemos que o menor passo levou a uma melhor aproximação. Na verdade, quando aplicamos o método de Euler com  $h = 0.05$ , o erro absoluto foi praticamente reduzido a um quarto na comparação do erro com  $h = 0.1$ . Na Figura 2.2 é possível perceber que a solução gerada pelo método de Euler com  $h = 0.05$  é mais próxima da solução exata do PVI do que a solução aproximada com  $h = 0.1$ .

Tabela 2.2: Resultados do método de Euler com  $h = 0.05$  aplicado ao PVI do Exemplo 2.1.1.

| $n$ | $x_n$ | $y_n$   | Solução exata | Erro absoluto |
|-----|-------|---------|---------------|---------------|
| 1   | 0.05  | 0.9     | 0.90369       | 0.00369       |
| 2   | 0.1   | 0.8075  | 0.81451       | 0.00701       |
| 3   | 0.15  | 0.72213 | 0.73212       | 0.01          |
| 4   | 0.2   | 0.64352 | 0.65619       | 0.01267       |
| 5   | 0.25  | 0.57134 | 0.5864        | 0.01506       |
| 6   | 0.3   | 0.50528 | 0.52245       | 0.01718       |
| 7   | 0.35  | 0.44501 | 0.46406       | 0.01905       |
| 8   | 0.4   | 0.39026 | 0.41096       | 0.02070       |
| 9   | 0.45  | 0.34075 | 0.36288       | 0.02214       |
| 10  | 0.5   | 0.29621 | 0.31959       | 0.02338       |
| 11  | 0.55  | 0.2564  | 0.28085       | 0.02445       |
| 12  | 0.6   | 0.22108 | 0.24643       | 0.02535       |
| 13  | 0.65  | 0.19003 | 0.21614       | 0.02611       |
| 14  | 0.7   | 0.16302 | 0.18976       | 0.02673       |
| 15  | 0.75  | 0.13987 | 0.1671        | 0.02723       |
| 16  | 0.8   | 0.12038 | 0.14799       | 0.02761       |
| 17  | 0.85  | 0.10436 | 0.13224       | 0.02788       |
| 18  | 0.9   | 0.09164 | 0.11971       | 0.02807       |
| 19  | 0.95  | 0.08206 | 0.11022       | 0.02816       |
| 20  | 1     | 0.07546 | 0.10364       | 0.02818       |

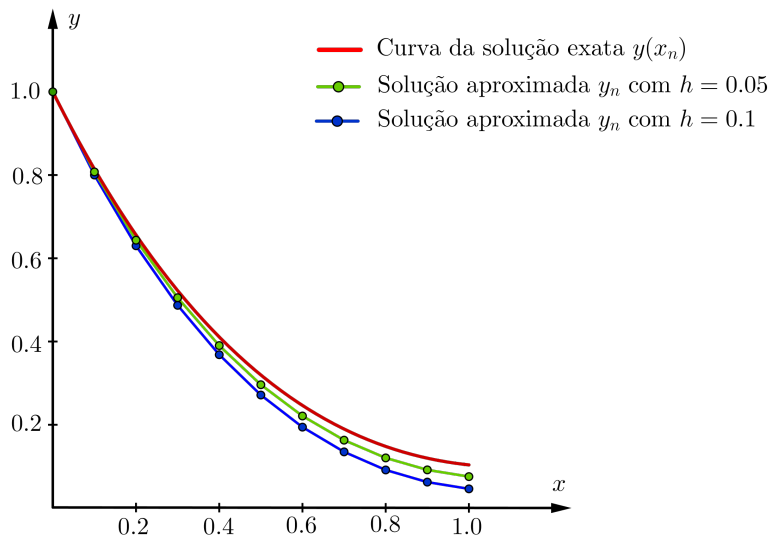


Figura 2.2: Solução exata do PVI do Exemplo 2.1.1 e soluções aproximadas pelo método de Euler com  $h = 0.05$  e  $h = 0.1$ .



No Exemplo 2.1.1 vimos que reduzir o tamanho do passo pela metade ocasionou uma redução considerável do erro absoluto. Podemos continuar reduzindo o tamanho do passo até chegar numa precisão razoavelmente boa, porém a quantidade de cálculos necessários será muito grande. Note que, se tomarmos  $h = 0.001$  no Exemplo 2.1.1, precisamos de 1000 passos para atravessar o intervalo  $[0, 1]$ . Veremos na Seção 2.4 uma derivação do método de Euler, em que é possível obter melhor precisão para o PVI, que a do método de Euler, para o mesmo tamanho de passo.

### 2.1.1 Formas alternativas

Nesta subseção, mencionamos algumas formas alternativas de interpretar o método de Euler. Algumas delas ajudarão na análise do erro e também na construção de métodos mais precisos.

Primeiro, vamos escrever a equação (1.1), em  $x = x_n$ , na forma

$$\frac{d\phi(x_n)}{dx} = f(x_n, \phi(x_n)). \quad (2.3)$$

Dessa forma, aproximando-se a derivada da equação (2.3) pelo quociente de diferenças correspondente, obtém-se

$$\frac{\phi(x_{n+1}) - \phi(x_n)}{x_{n+1} - x_n} \cong f(x_n, \phi(x_n)).$$

Por fim, substituindo  $\phi(x_{n+1})$  e  $\phi(x_n)$  pelos seus respectivos valores aproximados  $y_{n+1}$  e  $y_n$ , e resolvendo-se para  $y_{n+1}$ , obtém-se a fórmula de Euler (2.2).

Outra forma para se chegar na fórmula de Euler é integrar (2.3) entre  $x_n$  e  $x_{n+1}$ . Ou seja,

$$\begin{aligned} \int_{x_n}^{x_{n+1}} \phi'(x) dx &= \int_{x_n}^{x_{n+1}} f(x, \phi(x)) dx \\ \Rightarrow \phi(x_{n+1}) &= \phi(x_n) + \int_{x_n}^{x_{n+1}} f(x, \phi(x)) dx. \end{aligned} \quad (2.4)$$

Note que podemos interpretar geometricamente a integral na equação (2.4) como sendo a área sob a curva na Figura 2.3 entre  $x_n$  e  $x_{n+1}$ . Fazendo uma aproximação do integrando em (2.4) por seu valor em  $x = x_n$ , estaremos aproximando a área real pela área do retângulo sombreado. Sendo assim, temos que

$$\begin{aligned} \phi(x_{n+1}) &\cong \phi(x_n) + f(x_n, \phi(x_n))(x_{n+1} - x_n) \\ &= \phi(x_n) + hf(x_n, \phi(x_n)). \end{aligned} \quad (2.5)$$

Substituindo  $\phi(x_n)$  pelo seu valor aproximado  $y_n$  na equação (2.5), obtemos a aproximação  $y_{n+1}$  para  $\phi(x_{n+1})$ . Desse modo, chegamos na fórmula de Euler  $y_{n+1} = y_n + hf(x_n, y_n)$ . Uma terceira forma de se obter a fórmula (2.2) é supor que

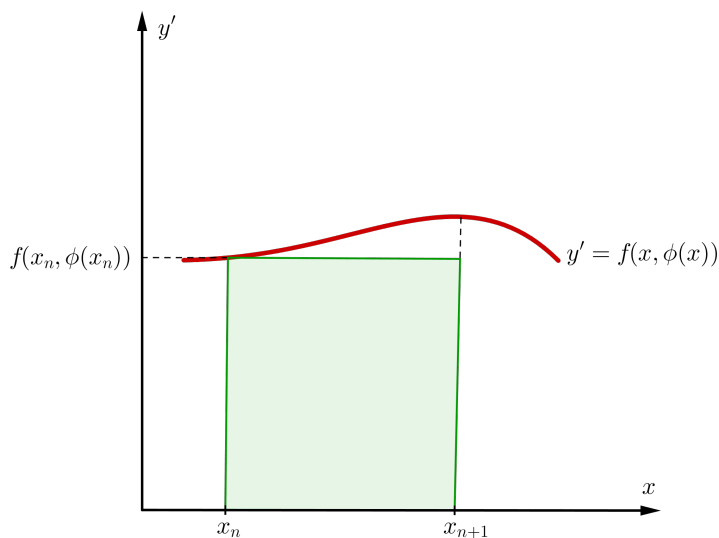


Figura 2.3: Dedução integral do método de Euler.

a solução  $y = \phi(x)$  tem uma série de Taylor em torno do ponto  $x_n$ . Logo,

$$\phi(x_n + h) = \phi(x_n) + \phi'(x_n)h + \phi''(x_n)\frac{h^2}{2!} + \cdots,$$

ou

$$\phi(x_{n+1}) = \phi(x_n) + f(x_n, \phi(x_n))h + \phi''(x_n)\frac{h^2}{2!} + \cdots. \quad (2.6)$$

Truncando-se a série após o termo em  $h$ , substituindo  $\phi(x_{n+1})$  e  $\phi(x_n)$  pelos seus valores aproximados  $y_{n+1}$  e  $y_n$ , respectivamente, obtemos a fórmula (2.2). Se forem usadas mais parcelas na série, obtém-se uma fórmula mais precisa. Além disso, usando uma série de Taylor com resto é possível estimar o tamanho do erro na fórmula. Isso será discutido na próxima seção.

## 2.2 Erros em aproximações numéricas

Ao utilizar métodos numéricos, como o método de Euler, na busca de soluções aproximadas para um problema de valor inicial, devemos estar cientes dos tipos de erros existentes nessa aproximação. Antes de aceitar a solução numérica aproximada como satisfatória, algumas questões devem ser respondidas. Uma dessas questões é quão pequeno deve ser o tamanho do passo  $h$  para que se obtenha precisão desejada sem desprender um grande esforço computacional? A análise dos tipos de erros que surgem no método de Euler nos dará uma ideia de qual é a melhor escolha para o tamanho de  $h$ , afim de que se tenha uma melhor precisão. Dois tipos de erro surgem ao se resolver um problema de valor inicial numericamente, são eles: erro de truncamento e erro de arredondamento.

### 2.2.1 Erro de truncamento global

A diferença  $E_n$  entre a solução exata  $y(x_n)$  do problema de valor inicial (1.1)-(1.2) e seu valor numérico  $y_n$ , com todas as casas decimais, é chamada de **erro de truncamento global**. Temos, pois,

$$E_n = y(x_n) - y_n.$$

Para calcular este erro é necessário conhecer a solução exata e ainda não ter casas decimais arredondadas. Este erro tem duas causas: primeiro, em cada passo usamos uma fórmula aproximada para determinar  $y_{n+1}$ ; segundo, os dados de entrada em cada etapa estão aproximadamente corretos, já que, em geral,  $y(x_n)$  não é igual a  $y_n$ .

### 2.2.2 Erro de truncamento local

A única diferença do **erro de truncamento local** para o global é que tanto  $y(x_n)$  quanto  $y_n$  são usados com valores arredondados com um número finito de casas decimais. Este erro é o mais viável e utilizado em nossas contas quando sabemos a solução exata, assim:

$$e_n = y(x_n) - y_n,$$

em que  $y(x_n)$  e  $y_n$  possuem um número finito de casas decimais arredondadas.

### 2.2.3 Erro de arredondamento

Como em todas as iterações temos um número finito de casas decimais, em certo momento começará a acontecer arredondamentos feitos pelos computadores. Então,

se  $y_n$  for o valor numérico de fato, com todas as casas decimais, e  $Y_n$  for o valor com arredondamento, o erro de arredondamento  $A_n$  será dado por:

$$A_n = y_n - Y_n.$$

O erro de arredondamento é claramente de natureza mais aleatória. Depende do tipo de computador utilizado, da ordem em que os cálculos são efetuados, do método de arredondamento e assim por diante. Por ser mais acessível, vamos usar daqui para a frente o erro de truncamento local como nossa medida principal de precisão de um método numérico e para comparar métodos numéricos diferentes. O erro absoluto definido no Exemplo 2.1.1 e apresentado nas tabelas deste trabalho é um erro de truncamento local. Para saber mais sobre erros decorrentes de métodos numéricos, veja as referências [1] e [14].

## 2.3 Erro de truncamento local para o método de Euler

Para analisarmos o erro de truncamento local do método de Euler vamos utilizar a fórmula de Taylor com resto, dada por

$$y(x) = y(a) + y'(a)\frac{x-a}{1!} + \cdots + y^{(r)}(a)\frac{(x-a)^r}{r!} + y^{(r+1)}(\gamma)\frac{(x-a)^{r+1}}{(r+1)!}, \quad (2.7)$$

em que  $y(x)$  é uma função  $n+1$  vezes derivável no intervalo  $(a, x)$  e  $\gamma$  é algum ponto entre  $a$  e  $x$ . Considerando  $r = 1$ ,  $a = x_n$  e  $x = x_{n+1} = x_n + h$ , obtemos

$$y(x_{n+1}) = y(x_n) + y'(x_n)\frac{h}{1!} + y''(\gamma)\frac{h^2}{2!}.$$

Como  $y'(x_n) = f(x_n, y(x_n))$ , então temos

$$y(x_{n+1}) = y(x_n) + hf(x_n, y(x_n)) + y''(\gamma)\frac{h^2}{2!}. \quad (2.8)$$

Subtraindo (2.2) da equação (2.8), vemos que

$$y(x_{n+1}) - y_{n+1} = [y(x_n) - y_n] + h[f(x_n, y(x_n)) - f(x_n, y_n)] + y''(\gamma)\frac{h^2}{2!}. \quad (2.9)$$

Para calcular o erro de truncamento local, aplicamos a equação (2.9) à solução exata  $y(x)$ , ou seja, tomamos  $y_n$  como sendo  $y(x_n)$ . Dessa forma, vemos de imediato, a partir da equação (2.9) que o erro de truncamento local do método de Euler é

$$e_{n+1} = y(x_{n+1}) - y_{n+1} = y''(\gamma)\frac{h^2}{2}, \quad (2.10)$$

### 2.3. ERRO DE TRUNCAMENTO LOCAL PARA O MÉTODO DE EULER

---

em que  $x_n < \gamma < x_{n+1}$ . Note que o erro de truncamento local para o método de Euler é proporcional ao quadrado do tamanho do passo  $h$ , e o fator de proporcionalidade depende da derivada segunda da solução exata  $y(x)$ .

Normalmente o valor de  $\gamma$  em (2.10) é desconhecido, então o erro *exato* não pode ser calculado, mas podemos determinar um limite superior para o módulo do erro. Sabemos que  $y(x)$  é duas vezes derivável em  $[x_n, x_{n+1}]$ , logo existe o máximo de  $y''(x)$  neste intervalo e denotaremos esse máximo por

$$M = \max|y''(x)|, \quad x \in [x_n, x_{n+1}].$$

Dessa forma, se estivermos calculando o valor da solução  $y(x)$  em um intervalo limitado, teremos

$$|e_{n+1}| \leq M \frac{h^2}{2}. \quad (2.11)$$

A partir da equação (2.11) é possível estimar um intervalo de tolerância para  $h$  de maneira que o erro de truncamento local não ultrapasse um nível de tolerância dado. Por exemplo, supondo que o nível de tolerância para o erro seja  $\delta$ , teremos da equação (2.11) que

$$h \leq \sqrt{\frac{2\delta}{M}}. \quad (2.12)$$

Sendo assim, a equação (2.11) pode ser usada para estimar o intervalo de variação do erro de truncamento local para  $x_n < \gamma < x_{n+1}$ . Por exemplo, considerando o problema de valor inicial do Exemplo 2.1.1

$$y' = x - y - 1, \quad y(0) = 1,$$

que tem como solução exata

$$y(x) = 3e^{-x} + x - 2.$$

Vemos que

$$y''(x) = 3e^{-x}.$$

Logo, pela equação (2.10), temos

$$e_{n+1} = \frac{3e^{-\gamma}h^2}{2}, \quad \text{com } x_n < \gamma < x_{n+1}.$$

Então, para  $h = 0.1$ , o erro no primeiro passo é

$$e_1 = \frac{3e^{-\gamma}(0.1)^2}{2} = 0,015e^{-\gamma}, \quad \text{com } 0 < \gamma < 0.1.$$

Note que  $e_1$  é positivo, além disso  $e^{-\gamma} < e^0$ . Então, segue que

$$e_1 < 0,015e^{-\gamma} < 0.015e^0 = 0.015.$$

Temos ainda que  $e^{-0.1} < e^{-\gamma}$ , logo

$$e_1 > 0,015e^{-\gamma} > e^{-0.1} = 0.01357.$$

Assim, o erro de truncamento local no primeiro passo é estimado no intervalo  $0.01357 < e_1 < 0.015$ . De fato, de acordo com a Tabela 2.1 o erro é 0.01451.

Ao discutir os erros decorrentes do uso de métodos numéricos, é comum usar a notação  $O(h^n)$  para indicar a ordem do erro. Dizemos que o erro de truncamento local tem ordem  $n$ , se existe uma constante  $M$  e um número inteiro positivo  $n$  tal que o módulo do erro seja menor ou igual a  $Mh^n$  para  $h$  suficientemente pequeno. Assim, o erro de truncamento local para o método de Euler tem ordem 2, ou seja,  $O(h^2)$ . Dessa forma, se  $h$  for reduzido por um fator de  $1/2$ , então o erro é reduzido por um fator de  $1/4$ , e assim por diante.

## 2.4 Método de Euler melhorado (Método de Heun)

O método de Euler melhorado ou método de Heun<sup>1</sup> é um exemplo do que chamamos de **método preditor-corretor**<sup>2</sup>. A ideia é usar a fórmula do método de Euler para obter uma primeira aproximação para a solução  $y(x_{n+1})$  do PVI (1.1)-(1.2). Denotaremos esta aproximação por  $y_{n+1}^*$ , então

$$y_{n+1}^* = y_n + hf(x_n, y_n).$$

Agora, melhoramos (ou corrigimos) esta aproximação aplicando mais uma vez o método de Euler. Mas desta vez, usamos as médias das inclinações das curvas das soluções através de  $(x_n, y_n)$  e  $(x_{n+1}, y_{n+1}^*)$ . Ou seja,

$$y_{n+1} = y_n + \frac{1}{2}h[f(x_n, y_n) + f(x_{n+1}, y_{n+1}^*)].$$

A Figura 2.4 ilustra como obter a solução aproximada  $y_1$  pelo método de Euler melhorado.

Dessa forma, o método de Euler melhorado para encontrar soluções próximas da solução exata do PVI (1.1)-(1.2) nos pontos  $x_{n+1} = x_0 + nh$  ( $n = 0, 1, \dots$ ) é

$$y_{n+1} = y_n + \frac{1}{2}h[f(x_n, y_n) + f(x_{n+1}, y_{n+1}^*)], \quad (2.13)$$

---

<sup>1</sup>A fórmula tem esse nome em homenagem a Karl Heun (1859-1929).

<sup>2</sup>Um método preditor-corretor é um método multi-passo para calcular a solução numérica de um problema de valor inicial para uma equação diferencial ordinária. Duas fórmulas são usadas em conjunto, primeiro para “predicar” um novo valor da variável dependente, então para “corrigir” a predição.

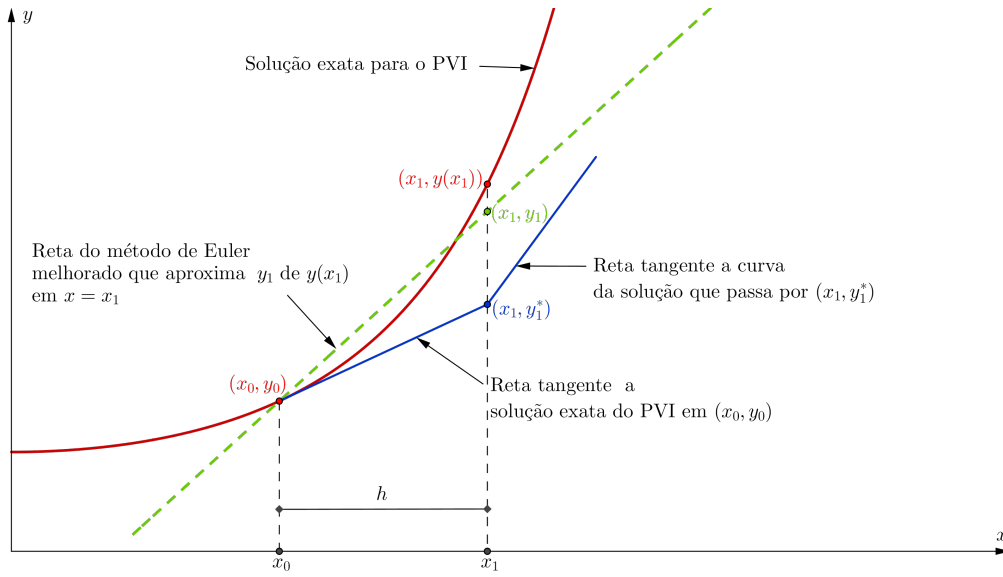


Figura 2.4: Método de Euler melhorado para determinar soluções próximas da solução exata do PVI (1.1)-(1.2).

em que

$$y_{n+1}^* = y_n + hf(x_n, y_n), \quad n = 0, 1, \dots$$

#### Exemplo 2.4.1.

Aplicando o método de Euler melhorado com  $h = 0.1$  para encontrar uma solução aproximada do PVI

$$y' = x - y - 1, \quad y(0) = 1,$$

no intervalo  $[0, 1]$ , obtemos os valores descritos na Tabela 2.3.

Comparando os resultados das Tabelas 2.1 e 2.3, vemos que o método de Euler melhorado tem captado, pelo menos, uma casa decimal de precisão quando  $h = 0.1$ . Isto é um indicativo de que o erro no método de Euler melhorado se comporta com ordem  $h^2$  enquanto que o comportamento do erro no método de Euler tem ordem  $h$ . Na Figura 2.5 esboçamos a solução exata do PVI do Exemplo 2.4.1 e a solução aproximada pelo método de Euler melhorado com  $h = 0.1$ . O erro de truncamento local para o método de Euler melhorado é  $O(h^3)$ . Este resultado é obtido de forma semelhante ao que foi encontrado no erro de truncamento local para o método de Euler.

## 2.4. MÉTODO DE EULER MELHORADO (MÉTODO DE HEUN)

---

Tabela 2.3: Resultados do método de Euler melhorado com  $h = 0.1$  aplicado ao PVI do Exemplo 2.4.1.

| $n$ | $x_n$ | $y_n$   | Solução exata | Erro absoluto |
|-----|-------|---------|---------------|---------------|
| 1   | 0.1   | 0.815   | 0.81451       | 0.00049       |
| 2   | 0.2   | 0.65708 | 0.65619       | 0.00088       |
| 3   | 0.3   | 0.52365 | 0.52245       | 0.00120       |
| 4   | 0.4   | 0.41241 | 0.41096       | 0.00145       |
| 5   | 0.5   | 0.32123 | 0.31959       | 0.00164       |
| 6   | 0.6   | 0.24821 | 0.24643       | 0.00178       |
| 7   | 0.7   | 0.19163 | 0.18976       | 0.00187       |
| 8   | 0.8   | 0.14993 | 0.14799       | 0.00194       |
| 9   | 0.9   | 0.12168 | 0.11971       | 0.00197       |
| 10  | 1     | 0.10562 | 0.10364       | 0.00198       |

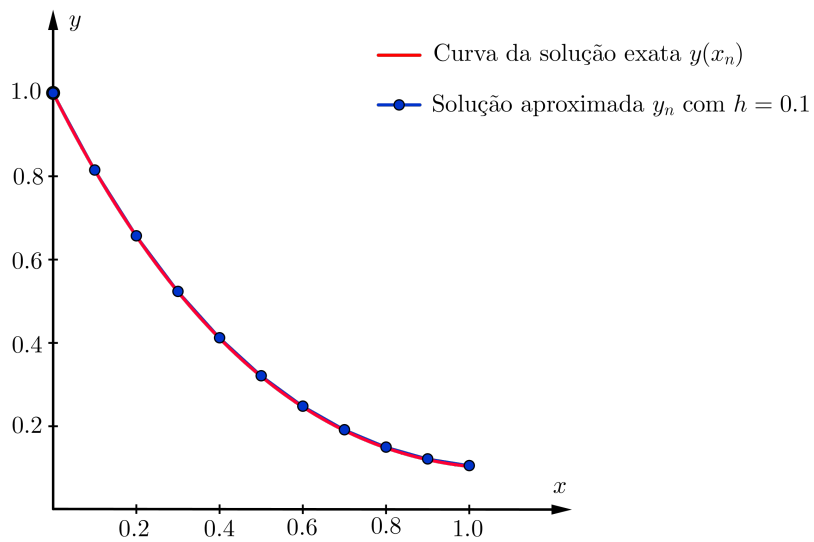


Figura 2.5: Solução exata do PVI do Exemplo 2.4.1 e solução aproximada pelo método de Euler melhorado com  $h = 0.1$ .



### 2.4.1 Forma alternativa

Outra forma de obter a fórmula do método de Euler melhorado é utilizando a equação (2.4). Novamente, vamos considerar que a aproximação de  $y(x_{n+1})$  para o PVI (1.1)-(1.2) pelo método de Euler é

$$y_{n+1}^* = y_n + hf(x_n, y_n).$$

Seja  $y = \phi(x)$  a solução do PVI (1.1)-(1.2). A fórmula do método de Euler melhorado pode ser obtida aproximando o integrando na equação (2.4) pela média de seus valores nas duas extremidades, a saber,

$$\phi(x_{n+1}) = \phi(x_n) + \int_{x_n}^{x_{n+1}} \frac{[f(x_n, \phi(x_n)) + f(x_{n+1}, \phi(x_{n+1}))]}{2} dx.$$

Isso é equivalente a aproximar a área embaixo da curva na Figura (2.6) entre  $x = x_n$  e  $x = x_{n+1}$  pela área do trapézio sombreado. Além disso, se substituindo  $\phi(x_n)$  e  $\phi(x_{n+1})$  pelos seus valores respectivos  $y_n$  e  $y_{n+1}$ , obtemos

$$y_{n+1} = y_n + h \frac{f(x_n, y_n) + f(x_{n+1}, y_{n+1})}{2}. \quad (2.14)$$

Note que a incógnita  $y_{n+1}$  aparece como um dos argumentos de  $f$  na equação (2.14). Sendo assim,  $y_{n+1}$  é definido de forma implícita. Dependendo da natureza da função  $f$  pode ser bem difícil resolver a equação (2.14) para  $y_{n+1}$ . Essa dificuldade pode ser sanada substituindo  $y_{n+1}$ , em  $f(x_{n+1}, y_{n+1})$ , na equação (2.14) por  $y_{n+1}^*$ . Dessa forma, obtemos a equação do método de Euler melhorado

$$y_{n+1} = y_n + \frac{1}{2}h[f(x_n, y_n) + f(x_{n+1}, y_{n+1}^*)].$$

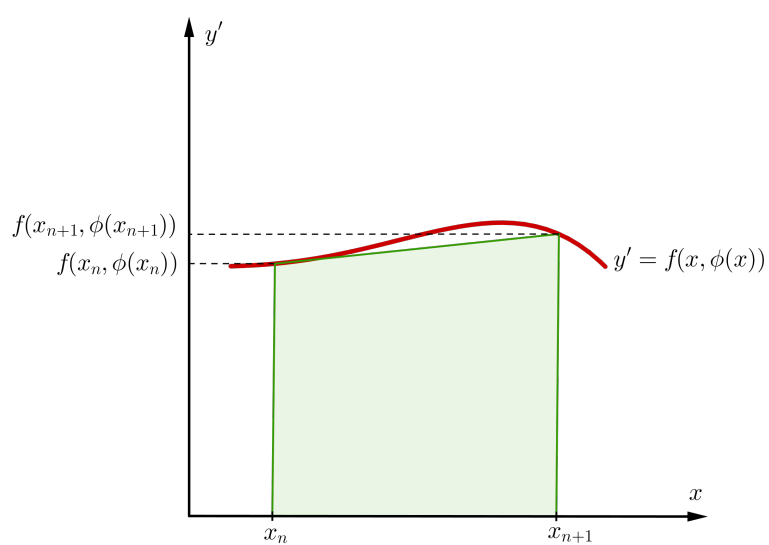


Figura 2.6: Dedução integral do método de Euler melhorado.

## Capítulo 3

# Métodos de Runge-Kutta

Na Seção 2.1.1 mostramos que o método de Euler pode ser obtido a partir da fórmula de Taylor, tomando-se todos os termos até o termo em  $h$ . Perceba que a fórmula de Taylor fornece aproximações arbitrariamente boas, contanto que calculemos as derivadas necessárias. Em outras palavras, os métodos que usam a fórmula de Taylor apresentam o inconveniente computacional de operar simultaneamente com as derivadas da função  $f$ . Aqui, apresentamos os **métodos de Runge-Kutta** de 1ª a 4ª ordem, cuja precisão é equivalente aos métodos da fórmula de Taylor, porém sem o inconveniente de se calcular derivadas.

As principais referências bibliográficas para a elaboração deste capítulo foram [3], [4], [7], [13], [14] e [18].

Basicamente, todos os métodos de Runge-Kutta são generalizações da fórmula de Euler (2.2), em que a função inclinação  $f$  é substituída por uma média ponderada de inclinações ao longo do intervalo  $x_n \leq x \leq x_{n+1}$ . Ou seja,

$$y_{n+1} = y_n + h \sum_{k=1}^m w_k c_k, \quad (3.1)$$

em que os pesos  $w_k$ ,  $k = 1, 2, \dots, m$ , são constantes satisfazendo  $\sum_{k=1}^m w_k = 1$ , e cada  $c_k$ ,  $k = 1, 2, \dots, m$ , é a função  $f$  avaliada em um ponto selecionado  $(x, y)$  tal que  $x_n \leq x \leq x_{n+1}$ . O número  $m$  é chamado de **ordem** do método. A média na equação (3.1) não é formada de maneira aleatória, mas seus parâmetros são escolhidos comparando (3.1) com um polinômio de Taylor conveniente de grau  $m$ .

Note que se tomarmos  $m = 1$ ,  $w_1 = 1$  e  $c_1 = f(x_n, y_n)$  em (3.1), obtemos a fórmula de Euler  $y_{n+1} = y_n + hf(x_n, y_n)$ . Dizemos, portanto, que o método de Euler é um método de **Runge-Kutta de primeira ordem**.

### 3.1 Runge-Kutta de segunda ordem

Fazendo  $m = 2$  em (3.1), temos

$$y_{n+1} = y_n + h(w_1 c_1 + w_2 c_2). \quad (3.2)$$

Considere  $c_1 = f(x_n, y_n)$  e  $c_2 = f(x_n + \alpha h, y_n + \beta h f(x_n, y_n))$ . Sendo assim, devemos encontrar  $w_1, w_2, \alpha$ , e  $\beta$  de modo que (3.2) seja igual ao polinômio de Taylor de  $y$  de grau 2. Para isso, expande-se  $c_2$  numa série de Taylor de duas variáveis, em torno de  $(x_n, y_n)$ , abandonando-se todos os termos a partir do termo em  $h^2$ , ou seja,

$$c_2 = f(x_n, y_n) + \alpha h \frac{\partial f(x_n, y_n)}{\partial x} + \beta h f(x_n, y_n) \frac{\partial f(x_n, y_n)}{\partial y} + O(h^2). \quad (3.3)$$

A partir daqui, vamos desprezar o resto  $O(h^2)$ , pois desejamos que ele seja suficientemente pequeno para desconsiderá-lo. Substituindo (3.3) em (3.2), obtêm-se

$$\begin{aligned} y_{n+1} &= y_n + h[(w_1 + w_2)f(x_n, y_n)] + h^2 \left[ w_2 \alpha \frac{\partial f(x_n, y_n)}{\partial x} \right. \\ &\quad \left. + w_2 \beta f(x_n, y_n) \frac{\partial f(x_n, y_n)}{\partial y} \right]. \end{aligned} \quad (3.4)$$

Agora, expandindo  $y(x)$  em torno de  $x_n$  até o termo em  $h^2$ , temos que

$$y(x_n + h) = y(x_n) + \frac{h}{1!} y'(x_n) + \frac{h^2}{2!} y''(x_n).$$

Como

$$y'(x_n) = f(x_n, y(x_n)) \quad \text{e} \quad y''(x_n) = \frac{\partial f(x_n, y(x_n))}{\partial x} + f(x_n, y(x_n)) \frac{\partial f(x_n, y(x_n))}{\partial y},$$

então

$$y(x_n + h) = y(x_n) + h f(x_n, y(x_n)) + \frac{h^2}{2!} \left[ \frac{\partial f(x_n, y(x_n))}{\partial x} + f(x_n, y(x_n)) \frac{\partial f(x_n, y(x_n))}{\partial y} \right]. \quad (3.5)$$

Como  $x_n + h = x_{n+1}$ , fazendo  $y(x_n) = y_n$  em (3.5) podemos comparar os termos de mesma potência, com relação a  $h$ , das equações (3.5) e (3.4), ou seja,

$$f(x_n, y_n) = (w_1 + w_2) f(x_n, y_n)$$

e

$$\frac{1}{2} \left[ \frac{\partial f(x_n, y_n)}{\partial x} + f(x_n, y_n) \frac{\partial f(x_n, y_n)}{\partial y} \right] = \left[ w_2 \alpha \frac{\partial f(x_n, y_n)}{\partial x} + w_2 \beta f(x_n, y_n) \frac{\partial f(x_n, y_n)}{\partial y} \right].$$

Das igualdades acima, obtém-se o sistema

$$\begin{cases} w_1 + w_2 = 1 \\ w_2\alpha = 1/2 \\ w_2\beta = 1/2 \end{cases} \quad (3.6)$$

O sistema 3.6, claramente possui infinitas soluções, uma vez que é formado por quatro incógnitas e três equações, e uma das incógnitas pode ser tomada arbitrariamente. Escolhendo  $w_2 = 1/2$ , tem-se  $w_1 = 1/2$ ,  $\alpha = 1$  e  $\beta = 1$ . Sendo assim, por (3.2), temos

$$y_{n+1} = y_n + \frac{h}{2}(c_1 + c_2),$$

como  $c_1 = f(x_n, y_n)$ ,  $c_2 = f(x_n + \alpha h, y_n + \beta h f(x_n, y_n))$  e  $x_n + h = x_{n+1}$ , então

$$y_{n+1} = y_n + \frac{h}{2} [f(x_n, y_n) + f(x_{n+1}, y_n + h f(x_n, y_n))].$$

Considerando  $y_{n+1}^* = y_n + h f(x_n, y_n)$ , temos que

$$y_{n+1} = y_n + \frac{h}{2} [f(x_n, y_n) + f(x_{n+1}, y_{n+1}^*)] \quad (3.7)$$

é a fórmula do método de Euler melhorado, visto na Seção 2.4. Dizemos que o método de Euler melhorado é um método de **Runge-Kutta de segunda ordem**. É obvio que o sistema (3.6) possibilita a obtenção de uma infinidade de métodos de Runge-Kutta de segunda ordem. No entanto, todos eles possuem o mesmo erro local de truncamento, ou seja,  $\frac{h^3}{3!} y'''(\gamma)$ ,  $x_n < \gamma < x_{n+1}$  e são equivalentes em precisão ao método de Euler melhorado.

## 3.2 Runge-Kutta de terceira ordem

Os métodos de Runge-Kutta de ordens mais elevadas são obtidos de modo semelhante ao de segunda ordem. Ou seja, para obter os métodos de Runge-Kutta de terceira ordem, façamos  $m = 3$  em (3.1). Dessa forma, obtemos

$$y_{n+1} = y_n + h(w_1 c_1 + w_2 c_2 + w_3 c_3),$$

em que  $c_1$ ,  $c_2$  e  $c_3$  aproximam derivadas em vários pontos no intervalo  $[x_n, x_{n+1}]$ , com

$$\begin{aligned} c_1 &= f(x_n, y_n), \\ c_2 &= f(x_n + \alpha_1 h, y_n + \beta_1 h c_1), \\ c_3 &= f(x_n + \alpha_2 h, y_n + \beta_2 h c_2 + (\alpha_2 - \beta_2) h c_1). \end{aligned}$$

### 3.3. RUNGE-KUTTA DE QUARTA ORDEM

---

A fim de determinar  $w_1, w_2, w_3, \alpha_1, \alpha_2, \beta_1$  e  $\beta_2$ , expandem-se  $c_2$  e  $c_3$  numa série de Taylor de duas variáveis em torno de  $(x_n, y_n)$ . Além disso, expande-se  $y(x)$  numa série de Taylor em torno de  $x_n$ . Procedendo de forma análoga ao que foi feito no desenvolvimento do método de Runge-Kutta de segunda ordem, só que agora, comparando os coeficientes de mesma potência de  $h$  até  $h^3$ , chega-se a um sistema com mais incógnitas do que equações, que também terá infinitas soluções. Não nos atentaremos a resolução de tal sistema de equações. Porém, informamos ao leitor interessado que os cálculos desta resolução podem ser encontrados em [7]. Os métodos de Runge-Kutta de terceira ordem têm erro de truncamento local de  $O(h^4)$  e fornecem resultados exatos se a solução for uma função cúbica.

Os métodos de Runge-Kutta de terceira ordem, mais populares, são:

$$y_{n+1} = y_n + \frac{h}{6}[c_1 + 4c_2 + c_3],$$

em que

$$\begin{aligned} c_1 &= f(x_n, y_n), \\ c_2 &= f\left(x_n + \frac{h}{2}, y_n + \frac{h}{2}c_1\right), \\ c_3 &= f(x_n + h, y_n + 2hc_2 - hc_1), \end{aligned}$$

e

$$y_{n+1} = y_n + \frac{h}{4}\left[c_1 + \frac{3}{2}(c_2 + c_3)\right],$$

onde

$$\begin{aligned} c_1 &= f(x_n, y_n), \\ c_2 &= f\left(x_n + \frac{2}{3}h, y_n + \frac{2}{3}hc_1\right), \\ c_3 &= f\left(x_n + \frac{2}{3}h, y_n + \frac{2}{3}hc_2\right), \end{aligned}$$

que é conhecido como **método de Nystrom**.

### 3.3 Runge-Kutta de quarta ordem

Dentre todos os métodos de Runge-Kutta, o de quarta ordem é o mais conhecido e utilizado. Ele também é conhecido como **método clássico de Runge-Kutta**. Considerando  $m = 4$  em (3.1), o método de Runge-Kutta de quarta ordem consiste em encontrar os parâmetros da fórmula

$$y_{n+1} = y_n + h(w_1c_1 + w_2c_2 + w_3c_3 + w_4c_4),$$

### 3.3. RUNGE-KUTTA DE QUARTA ORDEM

---

com as constantes  $c_1$ ,  $c_2$ ,  $c_3$  e  $c_4$  dadas por:

$$\begin{aligned}c_1 &= f(x_n, y_n), \\c_2 &= f(x_n + \alpha_1 h, y_n + \beta_1 h c_1), \\c_3 &= f(x_n + \alpha_2 h, y_n + \beta_2 h c_1 + \beta_3 h c_2), \\c_4 &= f(x_n + \alpha_3 h, y_n + \beta_4 h c_1 + \beta_5 h c_2 + \beta_6 h c_3).\end{aligned}$$

Ao proceder de forma análoga ao que foi feito para o método de Runge-Kutta de segunda ordem, obtêm-se um sistema com 11 equações e 13 incógnitas e, consequentemente, possui infinitas soluções. A solução mais conhecida deste método, de acordo com Boyce e DiPrima [4], é dada por:

$$y_{n+1} = y_n + \frac{h}{6}[c_1 + 2c_2 + 2c_3 + c_4],$$

em que

$$\begin{aligned}c_1 &= f(x_n, y_n), \\c_2 &= f(x_n + \frac{h}{2}, y_n + \frac{h}{2}c_1), \\c_3 &= f(x_n + \frac{h}{2}, y_n + \frac{h}{2}c_2), \\c_4 &= f(x_n + h, y_n + hc_3).\end{aligned}$$

Este método tem erro de truncamento local proporcional a  $h^5$ . Assim, é duas ordens de grandeza mais preciso do que o método de Euler melhorado e três ordens de grandeza mais preciso que o método de Euler.

#### Exemplo 3.3.1.

Aplicando o método de Runge-Kutta de quarta ordem com  $h = 0.1$  para encontrar uma solução aproximada do PVI

$$y' = x - y - 1, \quad y(0) = 1,$$

no intervalo  $[0, 1]$ , obtemos os valores descritos na Tabela 3.1.

Analisando a Tabela 3.1, descobrimos o motivo do método de Runge-Kutta de quarta ordem ser tão popular. Pois, se desejamos uma solução com precisão de 5 casas decimais, os resultados da Tabela 3.1 nos mostram que não é necessário reduzir o tamanho do passo. A comparação do erro absoluto nas Tabelas 2.1, 2.3 e 3.1, ilustra a eficiência do método de Runge-Kutta sobre os métodos de Euler e Euler melhorado. A Figura 3.1 mostra a poligonal da solução do PVI do Exemplo 3.3.1.

De acordo com Ruggiero [14], os métodos de Runge-Kutta, apesar de serem auto-iniciáveis (pois são de passo simples) e não trabalham com a derivada de  $f(x, y)$ ,

### 3.3. RUNGE-KUTTA DE QUARTA ORDEM

---

Tabela 3.1: Resultados do método de Runge-Kutta de 4ª ordem com  $h = 0.1$  aplicado ao PVI do Exemplo 3.3.1.

| $n$ | $x_n$ | $y_n$   | Solução exata | Erro absoluto |
|-----|-------|---------|---------------|---------------|
| 1   | 0.1   | 0.81451 | 0.81451       | 0.00000       |
| 2   | 0.2   | 0.65619 | 0.65619       | 0.00000       |
| 3   | 0.3   | 0.52246 | 0.52245       | 0.00001       |
| 4   | 0.4   | 0.41096 | 0.41096       | 0.00000       |
| 5   | 0.5   | 0.31959 | 0.31959       | 0.00000       |
| 6   | 0.6   | 0.24644 | 0.24643       | 0.00001       |
| 7   | 0.7   | 0.18976 | 0.18976       | 0.00000       |
| 8   | 0.8   | 0.14799 | 0.14799       | 0.00000       |
| 9   | 0.9   | 0.11971 | 0.11971       | 0.00000       |
| 10  | 1     | 0.10364 | 0.10364       | 0.00000       |

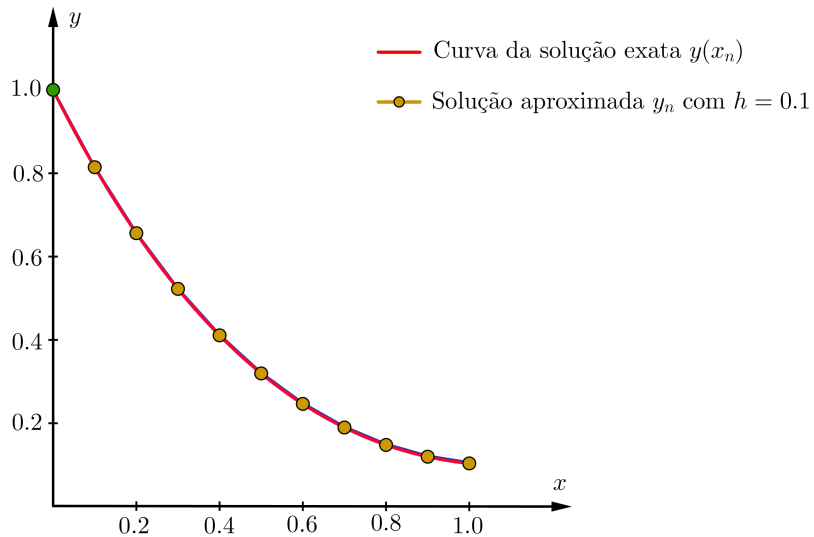


Figura 3.1: Solução exata do PVI do Exemplo 3.3.1 e soluções aproximadas pelo método de Runge-Kutta de 4ª ordem com  $h = 0.1$ .

apresentam a desvantagem de não haver para eles uma estimativa simples para o erro, o que inclusive poderia ajudar na escolha do passo  $h$ . No entanto, existem algumas adaptações dos métodos de Runge-Kutta que são usadas para a estimativa



de erro e controle do tamanho do passo  $h$ . Mais detalhes sobre a estimativa do erro nos métodos de Runge-Kutta podem ser encontrados na referência [7].

Diversos outros conceitos podem ser utilizados no estudo de métodos numéricos para EDO. Por exemplo, conceitos como convergência, estabilidade e consistência são importantes na análise do comportamento do método. Não exploramos tais conceitos neste trabalho, contudo indicamos as referências [7] e [10] ao leitor interessado em aprofundar o conhecimento sobre métodos numéricos para EDO.

## Capítulo 4

# Applet no Geogebra para soluções numéricas de PVI

Realizar cálculos, de forma manual, dos métodos de Euler, Euler melhorado e Runge-Kutta de quarta ordem é tedioso e oneroso. Contudo, estes métodos podem ser implementados, através de rotinas, em diversos softwares, tais como *Maple*, *Matlab*, *Maxima* e *Mathematica*. No entanto, é necessário ter noções de programação para a implementação de tais métodos nos referidos softwares. Neste Capítulo é apresentado um *applet* que de forma interativa e dinâmica fornece a solução exata de um PVI, bem como as soluções numéricas pelo método de Euler, método de Euler melhorado e pelo método de Runge-Kutta de quarta ordem. O *applet* também exibe o erro absoluto e a poligonal da solução aproximada para cada método. Além disso, é possível escolher qual solução se quer exibir ou até mesmo todas elas. O *applet* foi criado no software de geometria dinâmica *Geogebra 5.0* e está disponível em <https://geogebra.org/m/xTz9CaRY>.

### 4.1 O *Geogebra*

O *Geogebra* é um software de Matemática dinâmica, amplamente utilizado em situações de ensino e aprendizagem de matemática em todos os níveis de ensino. Ele possui a grande vantagem de ser gratuito e utilizar múltiplas representações gráficas de objetos matemáticos. Com o *Geogebra* é possível combinar geometria, álgebra, tabelas, gráficos, estatística e cálculo numa única aplicação. O *Geogebra* surgiu em 2001 a partir da Tese de Markus Hohenwarter da Universidade de Salzburgo. Para Download do *Geogebra* acesse <https://geogebra.org>.

Este capítulo não tem por objetivo ensinar a utilizar o *Geogebra* e tampouco servir de manual para utilização do mesmo. O propósito aqui é apresentar um conjunto de comandos e ações via teclado e mouse na interface do *Geogebra* de modo a produzir um material interativo e didático, com uma boa aparência, facilidade de

uso e que apresenta soluções aproximadas de PVIs a partir de três métodos numéricos bastante conhecidos na literatura.

## 4.2 Uso do *applet*

De acordo com Deitel [11], *applet* é um software que é executado no contexto de outro programa. Geralmente, os *applets* apresentam algum tipo de interface para usuário e possuem a vantagem de serem executados diretamente em uma página *web*. Além disso, se caracterizam por:

- Serem programas relativamente pequenos;
- Serem configuráveis;
- Serem interativos;
- Na sua maioria, estarem disponíveis na internet gratuitamente.

De modo geral, *applets* são programas leves, necessitam de pouco tempo para carregar no computador, notebook, smartphone ou tablet e são de fácil utilização.

Diversas pesquisas sobre o uso de *applets* na aprendizagem Matemática, também conhecidos como *Mathlets*<sup>1</sup>, têm sido realizadas ([2], [5], [15], [17]). De acordo com Barcelos et al. [2], o uso correto dos *applets* nas aulas de Matemática permite investigações e experimentações, o que possibilita o estabelecimento de conjecturas sobre determinado conceito e a construção do mesmo, de forma consistente. Ao gerar um *applet* e a sequência didática para uso deste recurso, o professor também constrói conhecimentos, o que contribui para sua prática docente (Santos, [15]). Dessa forma, é importante que o professor de Matemática tenha, por exemplo, oportunidade de aprender a elaborar e modificar *applets*, utilizando ambientes de Matemática dinâmica ou outros recursos.

Ao iniciar os estudos e pesquisas, para a elaboração deste trabalho, referente aos métodos de Euler, Euler melhorado e os métodos de Runge-Kutta, percebemos a necessidade de uma ferramenta didática, interativa, dinâmica e com uma interface agradável, que pudesse colaborar para o ensino e aprendizagem de tais métodos. Foi então que surgiu a ideia de se criar um *applet* no software *Geogebra*, uma vez que o *Geogebra* permite a construção de *applets* dispensando conhecimentos prévios de programação, é de fácil manipulação e configuração e dispõe de uma página *web* onde os *applets* ficam hospedados. Na Seção 4.3 mostraremos como o *applet*

---

<sup>1</sup>Um Mathlet é uma pequena plataforma independente e interativa para o ensino de Matemática. Por esta definição, podemos considerar aplicações desenvolvidas exclusivamente para a internet ou não, em qualquer linguagem de programação, em qualquer plataforma. (SANTOS, 2008, p.16).

## 4.2. USO DO APPLLET

intitulado *Métodos numéricos para EDO* (Figura 4.1) foi construído. A vantagem de conhecer como é feita a construção do *applet* é poder fazer modificações que forem convenientes para quem quer utilizá-lo no ensino de métodos numéricos para EDOs de primeira ordem. Ou seja, do *applet* apresentado neste trabalho podem derivar diversas outras versões mais sofisticadas.

No *applet* da Figura 4.1, um usuário entra com uma EDO de primeira ordem  $y' = f(x, y)$  e uma condição inicial  $(x_0, y_0)$ , então, ao pressionar *Enter* a solução exata do PVI surgirá na janela de visualização. O usuário pode optar por visualizar a solução aproximada do PVI pelo método de Euler, Euler melhorado ou Runge-Kutta de quarta ordem. Além disso, ele pode escolher o tamanho do passo  $h$ . Como resposta, o usuário visualizará a poligonal, com dez passos, do método selecionado e uma tabela contendo as soluções aproximadas, as soluções exatas e os erros absolutos. Escolhemos exibir dez passos, por acreditar que, didaticamente, essa quantidade seja suficiente para o entendimento da relação existente entre os métodos e o tamanho do passo  $h$ . Durante a construção do *applet* é possível alterar a quantidade de passos que se quer exibir.

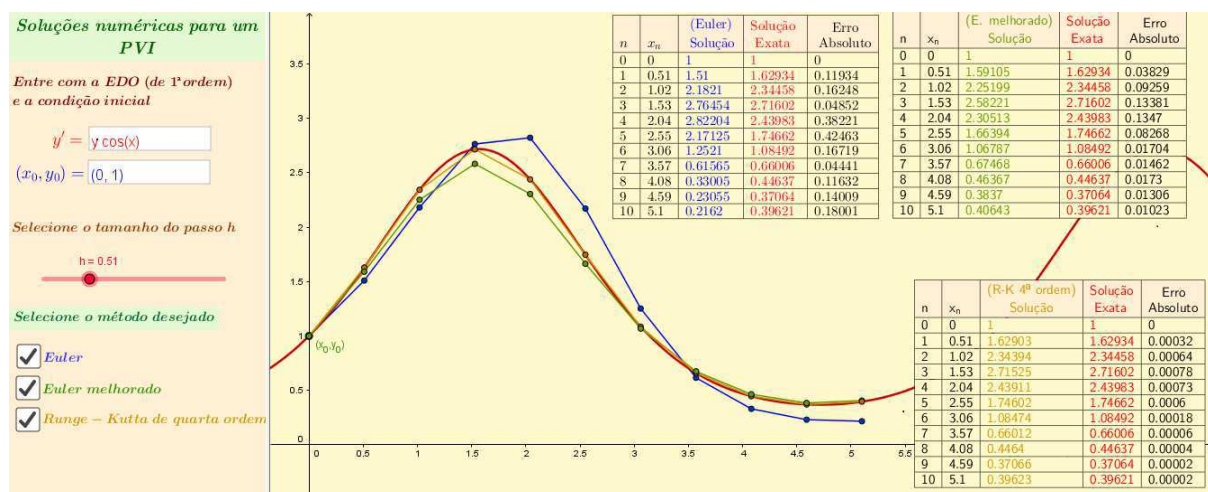


Figura 4.1: Applet: Métodos numéricos para EDO. Disponível em <https://geogebra.org/m/xTz9CaRY>.

Vale ressaltar que, neste trabalho, não apresentamos uma sequência didática que utilize o *applet* no ensino e aprendizagem dos métodos numéricos de passo simples. Contudo, o professor pode criar atividades que permitam o uso do *applet* de forma didática, lúdica e dinâmica. Por exemplo, utilizando o *applet*, podem ser trabalhadas atividades que buscam: relacionar o tamanho do passo e os erros gerados pelos métodos numéricos, facilitar a concepção da representação visual dos métodos de

### 4.3. CONSTRUÇÃO DO APPLET

---

Euler e Euler melhorado, compreender as diferenças entre os métodos e comparar as precisões dos mesmos para um passo de tamanho fixo.

## 4.3 Construção do *applet*

Nesta seção apresentaremos o roteiro para a construção do *applet*.

### 4.3.1 Passo 1 - Criando ponto inicial e controle deslizante

Ao abrir o *Geogebra*, na barra de menus, selecione a opção *Exibir* e em seguida a opção *Planilha* (Figura 4.2).

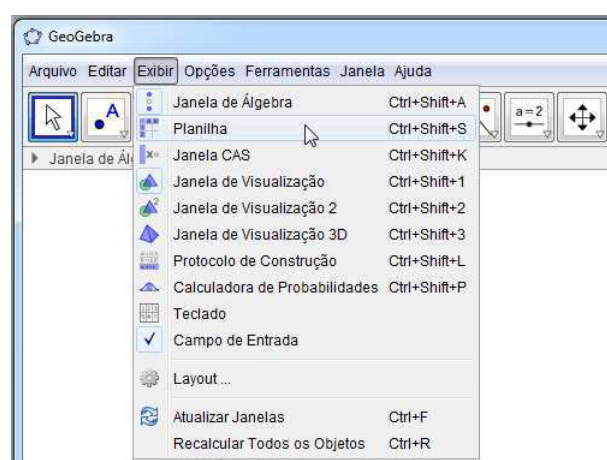


Figura 4.2: Exibindo a planilha.

Após a planilha aparecer, clique com o botão direito do mouse em qualquer célula, em seguida, clique na opção  **Opções da Planilha ...** e depois desmarque a opção ☐ **Exibir "=" Antes dos Comandos**.

Agora, digite na célula *A2* o ponto  $(0, 1)$ , como na Figura 4.3.

### 4.3. CONSTRUÇÃO DO APLET

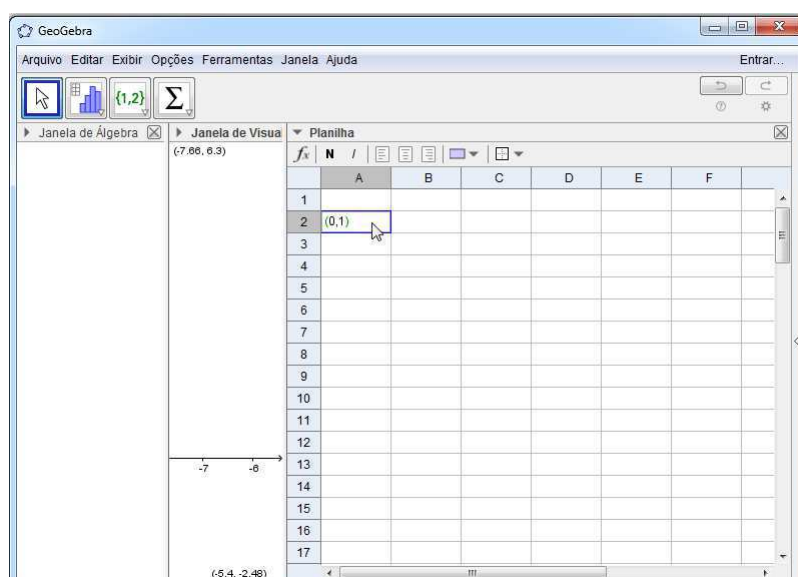


Figura 4.3: Criando o ponto inicial.

Após pressionar *Enter* o ponto  $A_2$  surgirá nas janelas de visualização e de álgebra. Em seguida, na barra de ferramentas, clique na ferramenta  *Controle Deslizante*. Ao clicar em qualquer lugar da janela de visualização surgirá uma caixa para a configuração do controle deslizante. Configure o controle deslizante de acordo com a Figura 4.4. Após pressionar *OK*, o controle deslizante aparecerá na janela de visualização (Figura 4.5).

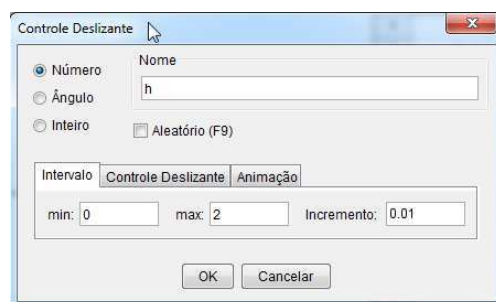


Figura 4.4: Configurando o controle deslizante.

### 4.3. CONSTRUÇÃO DO APLET

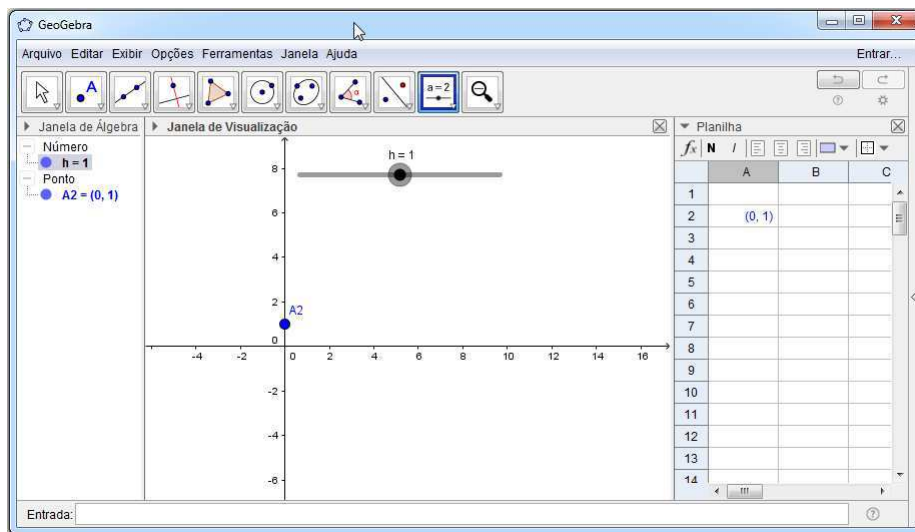


Figura 4.5: Controle deslizante  $h$ .

#### 4.3.2 Passo 2 - Obtendo a solução exata do PVI

No Campo de Entrada, digite a EDO  $y' = y * \cos(x)$ . Em seguida, digite  $g(x) = \text{ResolverEDO}[y', A2]$ . A solução exata  $g(x)$  aparecerá na janela de álgebra e seu gráfico surgirá na janela de visualização, como na Figura 4.6. Escolhemos a EDO  $y' = y * \cos(x)$  por conveniência, contudo na construção pode ser usada qualquer outra EDO de primeira ordem.

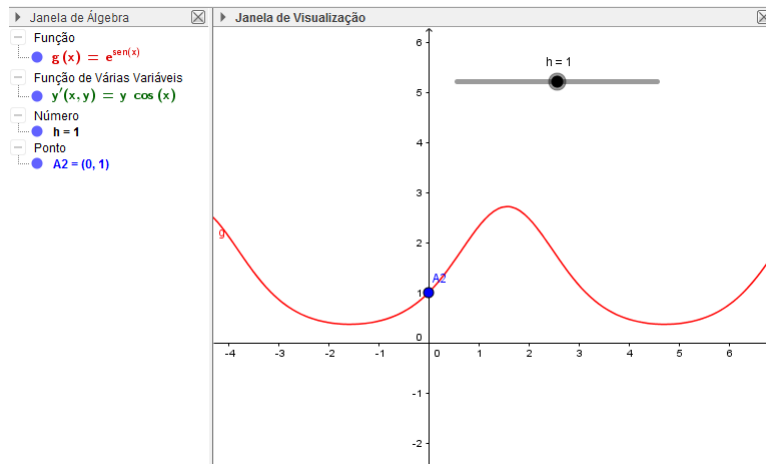
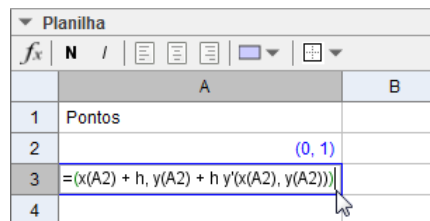


Figura 4.6: Solução exata.

### 4.3.3 Passo 3 - Implementando o método de Euler

Com o controle deslizante fixado em  $h = 0,51$ , digite na célula A1 da planilha a palavra *Pontos*. Na célula A3, entre com a fórmula do método de Euler ( $x(A2) + h, y(A2) + h * y'(x(A2), y(A2))$ ) (Figura 4.7).



|   | A                                        | B      |
|---|------------------------------------------|--------|
| 1 | Pontos                                   |        |
| 2 |                                          | (0, 1) |
| 3 | =x(A2) + h, y(A2) + h * y'(x(A2), y(A2)) |        |
| 4 |                                          |        |

Figura 4.7: Método de Euler.

Agora, clique na alça de preenchimento (canto inferior direito da célula) da célula A3 e arraste até a célula A12. Note que uma sequência de pontos surgirá na janela de visualização (Figura 4.8).

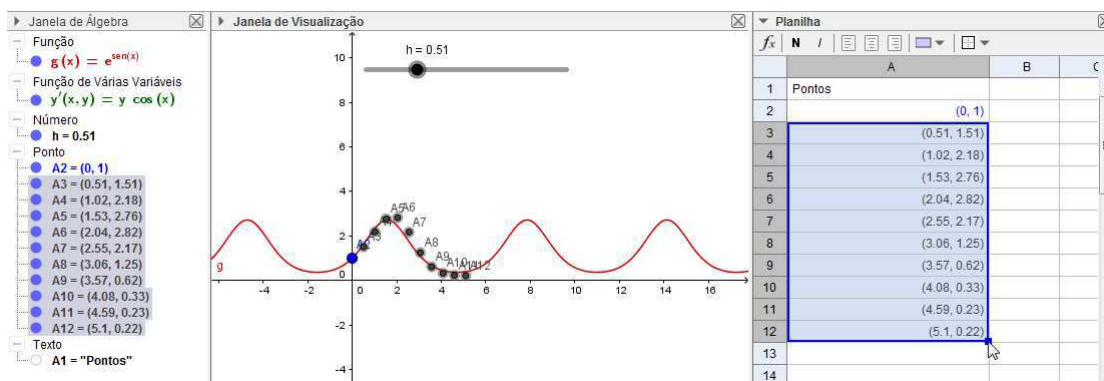


Figura 4.8: Pontos do método de Euler.

Selecione da célula A2 até a célula A12, pressione o botão direito do mouse e clique na opção **Exibir Rótulo**, para que os rótulos dos pontos na janela de visualização desapareçam. Pressionando o botão direito do mouse nas células selecionadas, clicando em *Propriedades* e em seguida na aba *Cor* é possível escolher a cor dos pontos. Os procedimentos para esconder o rótulo e para escolher a cor são realizados de igual forma para qualquer objeto do *Geogebra*. Escolhemos a cor azul para os pontos A2, A3, ..., A12.

Agora, digite no campo de entrada o comando *CaminhoPoligonal*[A2, A3, A4, A5, A6, A7, A8, A9, A10, A11, A12]. Este comando cria uma poligonal dos pontos A2,



### 4.3. CONSTRUÇÃO DO APLET

A3, ..., A12. Depois mude a cor do caminho poligonal para a mesma cor dos pontos, veja Figura 4.9.

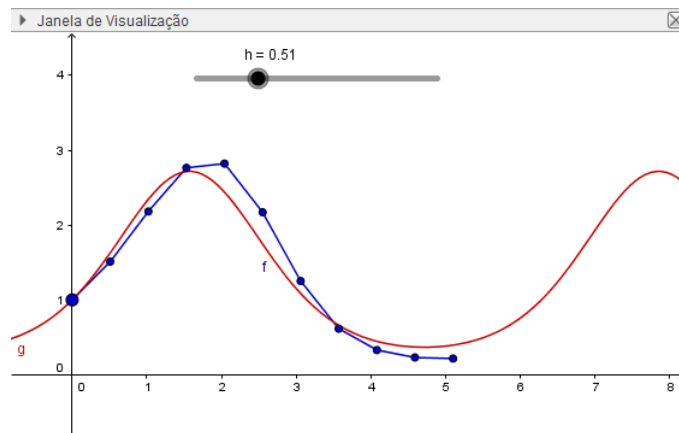


Figura 4.9: Poligonal do método de Euler.

Ao mover o controle deslizante podemos notar que a poligonal de Euler muda conforme os valores de  $h$  se alteram.

#### 4.3.4 Passo 4 - Criando a tabela do método de Euler

Na planilha, nomeie a célula B1 de  $n$  e nas células B2:B12 digite 0, 1, ..., 10, respectivamente. Em seguida, nomeie a célula C1 de  $x_n$  e na célula C2 digite  $x(A2)$ . Logo após, clique na alça de preenchimento da célula C2 e arraste até a célula C12. Nomeie a célula D1 como *Euler solução* e na célula D2 digite  $y(A2)$ . Logo após, clique na alça de preenchimento da célula D2 e arraste até a célula D12. Entre com o texto *Solução exata* na célula E1, na célula E2 digite a fórmula  $g(x(A2))$ , em seguida, estenda a fórmula para as demais células da coluna E até a célula E12. Digite o texto *Erro absoluto* na célula F1, e na célula F2 digite a fórmula  $|D2 - E2|$ . Logo após, estenda a fórmula até a célula F12.

Dessa forma, criamos uma tabela que apresenta a solução numérica do PVI (pelo método de Euler), a solução exata e o erro absoluto para 10 valores de  $x$  definidos a partir do ponto inicial A2 e do tamanho do passo  $h$ , veja Figura 4.10.

### 4.3. CONSTRUÇÃO DO APLET

| $f_x$ | N               | /  |       |               |               |               |
|-------|-----------------|----|-------|---------------|---------------|---------------|
|       | A               | B  | C     | D             | E             | F             |
| 1     | Pontos          | n  | $x_n$ | Euler solução | Solução exata | Erro Absoluto |
| 2     | (0, 1)          | 0  | 0     | 1             | 1             | 0             |
| 3     | (0.51, 1.51)    | 1  | 0.51  | 1.51          | 1.62934       | 0.11934       |
| 4     | (1.02, 2.1821)  | 2  | 1.02  | 2.1821        | 2.34458       | 0.16248       |
| 5     | (1.53, 2.76454) | 3  | 1.53  | 2.76454       | 2.71602       | 0.04852       |
| 6     | (2.04, 2.82204) | 4  | 2.04  | 2.82204       | 2.43983       | 0.38221       |
| 7     | (2.55, 2.17125) | 5  | 2.55  | 2.17125       | 1.74662       | 0.42463       |
| 8     | (3.06, 1.2521)  | 6  | 3.06  | 1.2521        | 1.08492       | 0.16719       |
| 9     | (3.57, 0.61565) | 7  | 3.57  | 0.61565       | 0.66006       | 0.04441       |
| 10    | (4.08, 0.33005) | 8  | 4.08  | 0.33005       | 0.44637       | 0.11632       |
| 11    | (4.59, 0.23055) | 9  | 4.59  | 0.23055       | 0.37064       | 0.14009       |
| 12    | (5.1, 0.2162)   | 10 | 5.1   | 0.2162        | 0.39621       | 0.18001       |
| 13    |                 |    |       |               |               |               |

Figura 4.10: Tabela do método de Euler.

Note que alteramos as cores dos elementos da colunas D e E. Além disso, centralizamos todos os elementos das células preenchidas, selecionando-as e clicando em centralizar . O número de casas decimais dos valores da tabela também podem ser escolhidos, para isso, veja Figura 4.11. Apresentamos os valores com 5 casas decimais.

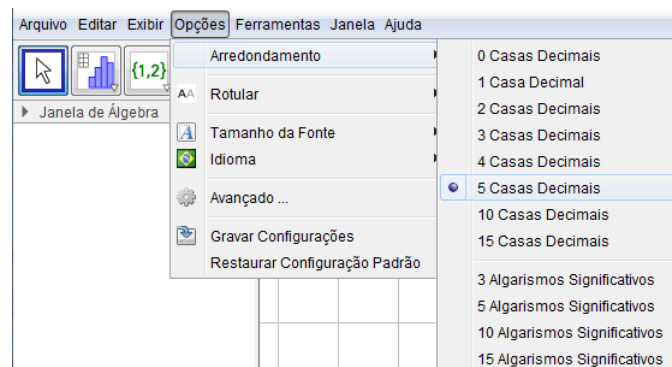


Figura 4.11: Arredondamento.

#### 4.3.5 Passo 5 - Implementando o método de Euler melhorado

O objetivo nesse passo é criar a poligonal do método de Euler melhorado. Na célula H2, digite A2. Na célula H3, entre com a fórmula do método de Euler ( $x(H2) + h, y(H2) + h * y'(x(H2), y(H2))$ ). Em seguida, na célula I1 digite o texto *pontos*, na célula I2 digite A2 e na célula I3 entre com a fórmula do método de Euler melhorado ( $x(H2) + h, y(H2) + h/2 * (y'(x(H2), y(H2)) + y'(x(H3), y(H3)))$ ) (Figura 4.12).

### 4.3. CONSTRUÇÃO DO APLET

| H            | I                                                                  |
|--------------|--------------------------------------------------------------------|
|              | Pontos                                                             |
| (0, 1)       | (0, 1)                                                             |
| (0.51, 1.51) | $(x(H2) + h, y(H2) + h / 2 (y'(x(H2), y(H2)) + y'(x(H3), y(H3))))$ |
|              |                                                                    |
|              |                                                                    |
|              |                                                                    |

Figura 4.12: Fórmula do método de Euler melhorado.

Agora, na célula H4 digite a fórmula de Euler atualizada  $(x(H3) + h, y(I3) + h * y'(x(H3), y(I3)))$  e na célula I4 a fórmula de Euler melhorado atualizada  $(x(H4) + h, y(I3) + h/2 * (y'(x(H3), y(I3)) + y'(x(H4), y(H4))))$  (Figura 4.13).

| H               | I                                                              |
|-----------------|----------------------------------------------------------------|
|                 | Pontos                                                         |
| (0, 1)          | (0, 1)                                                         |
| (0.51, 1.51)    | (0.51, 1.59105)                                                |
| (1.02, 2.29923) | $(x(H4), y(I3) + h / 2 (y'(x(H3), y(I3)) + y'(x(H4), y(H4))))$ |
|                 |                                                                |
|                 |                                                                |

Figura 4.13: Fórmula do método de Euler melhorado atualizada.

Logo após, clique na célula H4 e arraste até a célula H5, faça o mesmo com as células I4 e I5 (Figuras 4.14 e 4.15). Repita esse processo com as células da coluna H e da coluna I, nesta ordem, até as células H12 e I12, respectivamente.

| H               | I               |
|-----------------|-----------------|
|                 | Pontos          |
| (0, 1)          | (0, 1)          |
| (0.51, 1.51)    | (0.51, 1.59105) |
| (1.02, 2.29923) | (1.02, 2.25199) |
| (1.53, 2.85308) |                 |

Figura 4.14: De H4 para H5.

| H               | I               |
|-----------------|-----------------|
|                 | Pontos          |
| (0, 1)          | (0, 1)          |
| (0.51, 1.51)    | (0.51, 1.59105) |
| (1.02, 2.29923) | (1.02, 2.25199) |
| (1.53, 2.85308) | (1.53, 2.58221) |

Figura 4.15: De I4 para I5.

Ao final do procedimento teremos os valores da Figura 4.16.

### 4.3. CONSTRUÇÃO DO APLET

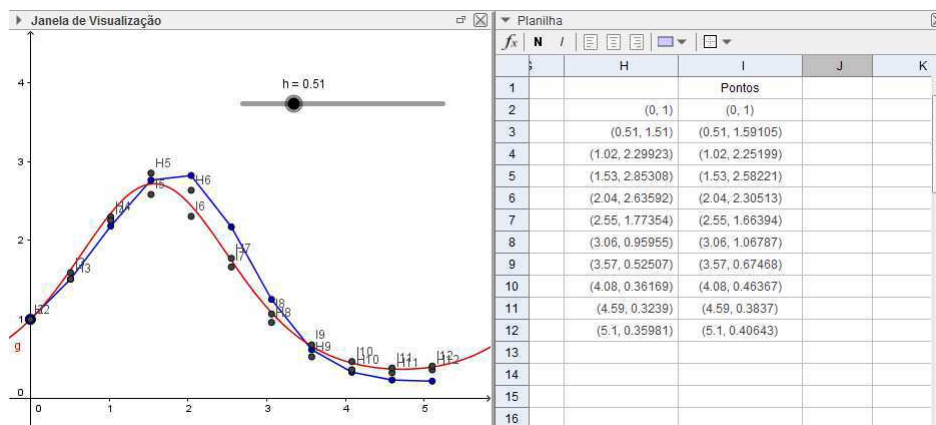


Figura 4.16: Método de Euler melhorado.

Note que os pontos H2:H12 também surgem na janela de visualização. No entanto, só queremos que os pontos I2:I12 apareçam, uma vez que eles são os pontos do método de Euler melhorado. Sendo assim, selecione da célula H2 até H12, clique com o botão direito do mouse na seleção e clique na opção  **Exibir Objeto**, para que os pontos H2:H12 desapareçam da janela de visualização. Em seguida, esconda o rótulo dos pontos I2:I12, assim como fizemos com os pontos A2:A12. Em seguida, mude a cor dos pontos I2:I12 para a cor verde.

Para criar a poligonal do método de Euler melhorado, digite no campo de entrada *CaminhoPoligonal*[I2, I3, I4, I5, I6, I7, I8, I9, I10, I11, I12]. Logo após, esconda o rótulo do caminho poligonal e altere sua cor para verde. Com isso, teremos a construção da Figura 4.17.

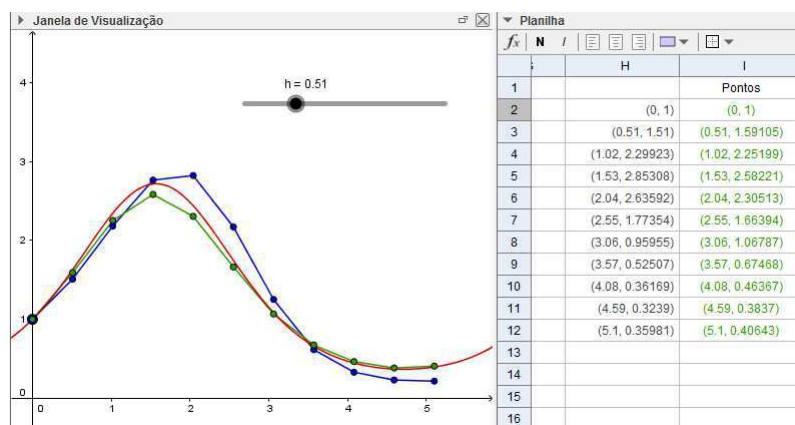


Figura 4.17: Poligonal do método de Euler melhorado.

### 4.3.6 Criando a tabela do método de Euler melhorado

Na planilha, nomeie a célula J1 de  $n$  e nas células J2:J12 digite  $0, 1, \dots, 10$ , respectivamente. Em seguida, nomeie a célula K1 de  $x_n$  e na célula K2 digite  $x(I2)$ . Logo após, clique na alça de preenchimento da célula K2 e arraste até a célula K12. Nomeie a célula L1 como *Euler melhorado solução* e na célula L2 digite  $y(I2)$ . Logo após clique na alça de preenchimento da célula L2 e arraste até a célula L12. Entre com o texto *Solução exata* na célula M1, na célula M2 digite a fórmula  $g(x(I2))$ , em seguida, estenda a fórmula para as demais células da coluna M até a célula M12. Digite o texto *Erro absoluto* na célula N1, e na célula N2 digite a fórmula  $|L2 - M2|$ . Em seguida, estenda a fórmula até a célula N12.

Dessa forma, criamos uma tabela que apresenta a solução numérica do PVI (pelo método de Euler melhorado), a solução exata e o erro absoluto (Figura 4.18).

|    | I               | J   | K     | L                       | M             | N             |
|----|-----------------|-----|-------|-------------------------|---------------|---------------|
| 1  | Pontos          | $n$ | $x_n$ | Euler melhorado solução | Solução exata | Erro absoluto |
| 2  | (0, 1)          | 0   | 0     | 1                       | 1             | 0             |
| 3  | (0.51, 1.59105) | 1   | 0.51  | 1.59105                 | 1.62934       | 0.03829       |
| 4  | (1.02, 2.25199) | 2   | 1.02  | 2.25199                 | 2.34458       | 0.09259       |
| 5  | (1.53, 2.58221) | 3   | 1.53  | 2.58221                 | 2.71602       | 0.13381       |
| 6  | (2.04, 2.30513) | 4   | 2.04  | 2.30513                 | 2.43983       | 0.1347        |
| 7  | (2.55, 1.66394) | 5   | 2.55  | 1.66394                 | 1.74662       | 0.08268       |
| 8  | (3.06, 1.06787) | 6   | 3.06  | 1.06787                 | 1.08492       | 0.01704       |
| 9  | (3.57, 0.67468) | 7   | 3.57  | 0.67468                 | 0.66006       | 0.01462       |
| 10 | (4.08, 0.46367) | 8   | 4.08  | 0.46367                 | 0.44637       | 0.0173        |
| 11 | (4.59, 0.3837)  | 9   | 4.59  | 0.3837                  | 0.37064       | 0.01306       |
| 12 | (5.1, 0.40643)  | 10  | 5.1   | 0.40643                 | 0.39621       | 0.01023       |
| 13 |                 |     |       |                         |               |               |

Figura 4.18: Poligonal do método de Euler melhorado.

Note que alteramos as cores dos elementos da colunas L e M.

### 4.3.7 Passo 6 - Implementando o método Runge-Kutta de quarta ordem

Com o objetivo de criar a poligonal do método de Runge-Kutta de quarta ordem, nomeie as células P1, Q1, R1, S1 e T1 de *Pontos*,  $k_1$ ,  $k_2$ ,  $k_3$ ,  $k_4$ , respectivamente. Na célula P2 digite  $A2$  e nas células Q2, R2, S2, T2 digite as fórmulas  $y'(x(P2), y(P2))$ ,  $y'(x(P2)+h/2, y(P2)+Q2*h/2)$ ,  $y'(x(P2)+h/2, y(P2)+R2*h/2)$  e  $y'(x(P2)+h/2, y(P2)+S2*h)$ , respectivamente, como na Figura 4.19.

### 4.3. CONSTRUÇÃO DO APLET

| P      | Q               | R                                 | S                                 | T                             |
|--------|-----------------|-----------------------------------|-----------------------------------|-------------------------------|
| Pontos | $k_1$           | $k_2$                             | $k_3$                             | $k_4$                         |
| A2     | $y'(P2), y(P2)$ | $y'(x(P2) + h/2, y(P2) + Q2 h/2)$ | $y'(x(P2) + h/2, y(P2) + R2 h/2)$ | $y'(x(P2) + h, y(P2) + S2 h)$ |

Figura 4.19: Definindo os parâmetros.

Em seguida, na célula P3 digite a fórmula do método de Runge-Kutta de quarta ordem  $(x(P2) + h, y(P2) + (h/6) * (Q2 + 2 * R2 + 2 * S2 + T2))$  (Figura 4.20).

| P                                                |
|--------------------------------------------------|
| Pontos                                           |
| (0, 1)                                           |
| $(x(P2)+h, y(P2) + h / 6 (Q2 + 2R2 + 2S2 + T2))$ |

Figura 4.20: Fórmula do método de Runge-Kutta de quarta ordem.

Para obter os parâmetros  $k_1, k_2, k_3, k_4$  da linha 3, clique na célula Q2 e arraste até a célula Q3, faça o mesmo para as células R2, S2, e T2. Depois que os valores de  $k_1, k_2, k_3, k_4$  da linha 3 forem obtidos, faremos o mesmo procedimento para encontrar os valores das células P4, Q4, R4, S4 e T4, ou seja, clique na célula da linha de cima e arraste. Realizando esse procedimento em todas as linhas até a linha 12, na ordem P, Q, R, S e T, obtêm-se os valores da Figura 4.21.

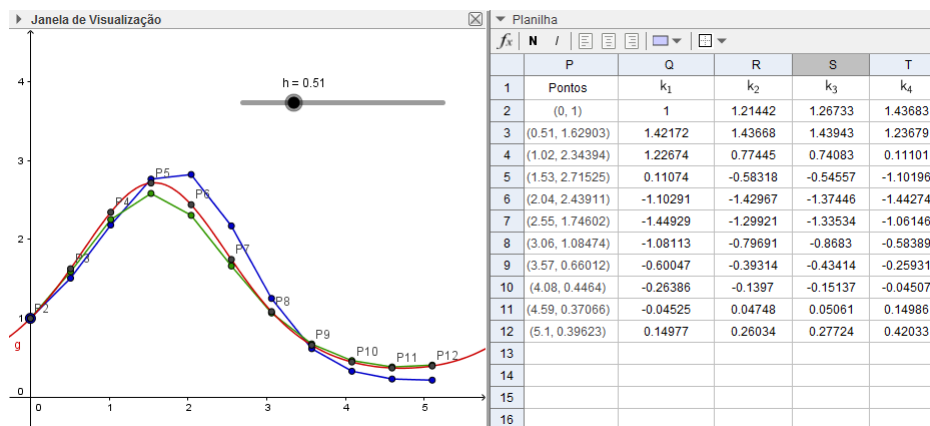


Figura 4.21: Pontos do método de Runge-Kutta de quarta ordem.

Agora, assim como fizemos no método de Euler e no método de Euler melhorado, esconda os rótulos dos pontos P2:P12 e altere sua cor para marrom.

Para criar a poligonal do método de Runge-Kutta de quarta ordem, digite no campo de entrada *CaminhoPoligonal*[P2, P3, P4, P5, P6, P7, P8, P9, P10, P11, P12].

### 4.3. CONSTRUÇÃO DO APLET

Logo após, esconda o rótulo do caminho poligonal e altere sua cor para marrom. Assim, teremos a construção da Figura 4.22.

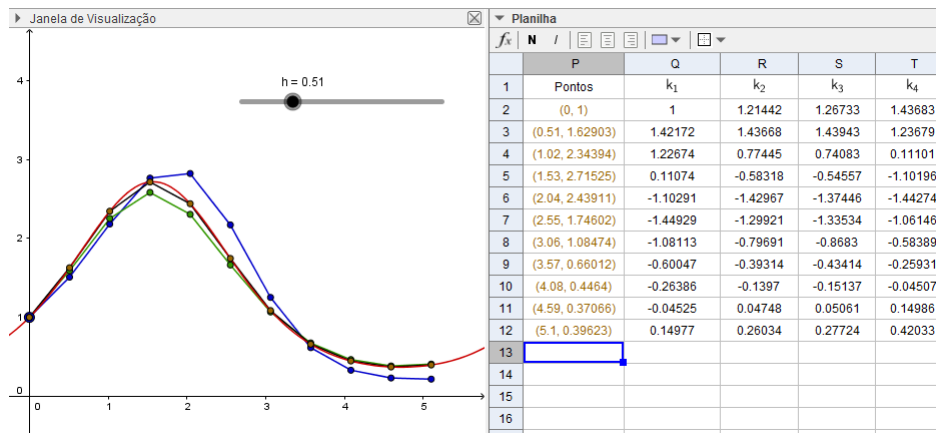


Figura 4.22: Poligonal do método de Runge-Kutta de quarta ordem.

#### 4.3.8 Passo 7 - Criando a tabela do método de Runge-Kutta

De forma análoga a construção das tabelas do método de Euler e do método de Euler melhorado, nomeie a célula U1 de  $n$  e nas células U2:U12 digite 0, 1, ..., 10, respectivamente. Em seguida, nomeie a célula V1 de  $x_n$  e na célula V2 digite  $x(P2)$ . Logo após, clique na alça de preenchimento da célula V2 e arraste até a célula V12. Nomeie a célula W1 como  $R-K$  solução e na célula W2 digite  $y(P2)$ . Logo após, clique na alça de preenchimento da célula W2 e arraste até a célula W12. Entre com o texto *Solução exata* na célula X1, na célula X2 digite a fórmula  $g(x(P2))$ , em seguida, estenda a fórmula para as demais células da coluna X até a célula X12. Digite o texto *Erro absoluto* na célula Y1, e na célula Y2 digite a fórmula  $|W2 - X2|$ . Logo após, estenda a fórmula até a célula Y12. Assim, criamos uma tabela que apresenta a solução numérica do PVI (pelo método de Runge-Kutta de quarta ordem), a solução exata e o erro absoluto (Figura 4.23).

### 4.3. CONSTRUÇÃO DO APLET

| $f_x$ | N               | I        |          |          |          |     |       |              |               |               |
|-------|-----------------|----------|----------|----------|----------|-----|-------|--------------|---------------|---------------|
|       | P               | Q        | R        | S        | T        | U   | V     | W            | X             | Y             |
| 1     | Pontos          | $k_1$    | $k_2$    | $k_3$    | $k_4$    | $n$ | $x_n$ | R-K1 solução | Solução exata | Erro absoluto |
| 2     | (0, 1)          | 1        | 1.21442  | 1.26733  | 1.43683  | 0   | 0     | 1            | 1             | 0             |
| 3     | (0.51, 1.62903) | 1.42172  | 1.43668  | 1.43943  | 1.23679  | 1   | 0.51  | 1.62903      | 1.62934       | 0.00032       |
| 4     | (1.02, 2.34394) | 1.22674  | 0.77445  | 0.74083  | 0.11101  | 2   | 1.02  | 2.34394      | 2.34458       | 0.00064       |
| 5     | (1.53, 2.71525) | 0.11074  | -0.58318 | -0.54557 | -1.10196 | 3   | 1.53  | 2.71525      | 2.71602       | 0.00078       |
| 6     | (2.04, 2.43911) | -1.10291 | -1.42967 | -1.37446 | -1.44274 | 4   | 2.04  | 2.43911      | 2.43983       | 0.00073       |
| 7     | (2.55, 1.74602) | -1.44929 | -1.29921 | -1.33534 | -1.06146 | 5   | 2.55  | 1.74602      | 1.74662       | 0.0006        |
| 8     | (3.06, 1.08474) | -1.08113 | -0.79691 | -0.8683  | -0.58389 | 6   | 3.06  | 1.08474      | 1.08492       | 0.00018       |
| 9     | (3.57, 0.66012) | -0.60047 | -0.39314 | -0.43414 | -0.25931 | 7   | 3.57  | 0.66012      | 0.66006       | 0.00006       |
| 10    | (4.08, 0.4464)  | -0.26386 | -0.1397  | -0.15137 | -0.04507 | 8   | 4.08  | 0.4464       | 0.44637       | 0.00004       |
| 11    | (4.59, 0.37066) | -0.04525 | 0.04748  | 0.05061  | 0.14986  | 9   | 4.59  | 0.37066      | 0.37064       | 0.00002       |
| 12    | (5.1, 0.39623)  | 0.14977  | 0.26034  | 0.27724  | 0.42033  | 10  | 5.1   | 0.39623      | 0.39621       | 0.00002       |
| 13    |                 |          |          |          |          |     |       |              |               |               |

Figura 4.23: Poligonal do método de Euler melhorado.

Note que alteramos as cores dos elementos da colunas W e X.

#### 4.3.9 Passo 8 - Transferindo as tabelas para a janela de visualização

Neste passo, vamos transferir as tabelas da planilha para a janela de visualização. Selecione da célula B1 até a célula F12, em seguida, clique com o botão direito do mouse e selecione a opção *Criar*, depois clique na opção *Tabela* (Figura 4.24). Dessa forma, a tabela selecionada aparecerá na janela de visualização.

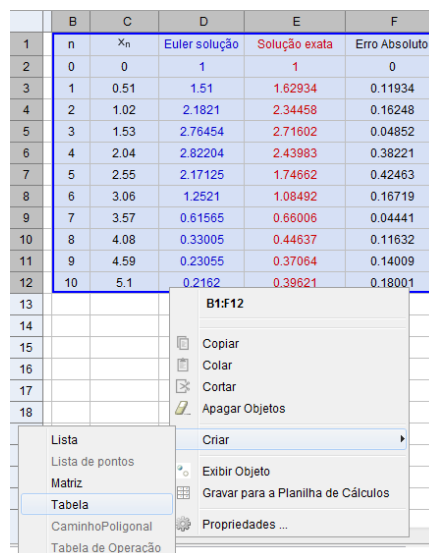


Figura 4.24: Transferindo a Tabela.



### 4.3. CONSTRUÇÃO DO APLET

Repita o procedimento para as tabelas que vão da célula J1 até a célula N12 e da célula U1 até a célula Y12. Dessa forma, as tabelas do método de Euler, do método de Euler melhorado e do método de Runge-Kutta de quarta ordem irão surgir na janela de visualização, como na Figura 4.25.

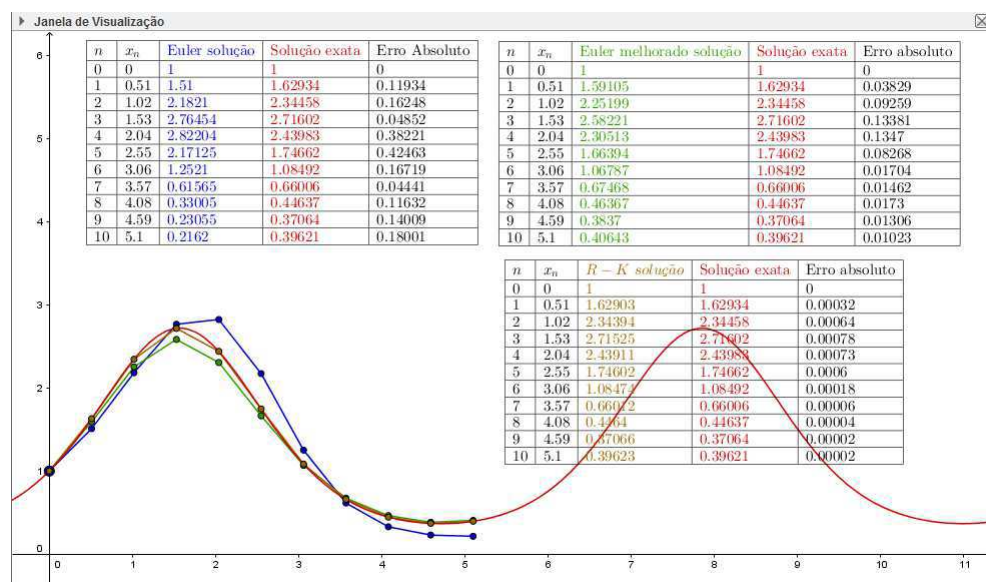


Figura 4.25: Tabelas na janela de visualização.

#### 4.3.10 Passo 9 - Abrindo outra janela de visualização e criando textos

Nesta etapa da construção abriremos uma outra janela de visualização e nela inseriremos alguns textos.

Na barra de menus, clique em *Exibir* e depois na opção  **Janela de Visualização 2**. Uma nova janela de visualização surgirá. Em seguida, clique no controle deslizante com o botão direito do mouse, logo depois, clique em *Propriedades*, clique na aba *Avançado*, desmarque a opção *Janela de Visualização* e marque a opção *Janela de visualização 2*. Seguindo esses passos o controle deslizante desaparece da janela de visualização e aparece na janela de visualização 2.

Para criar textos, selecione a ferramenta  **Texto** na barra de ferramentas. Em seguida, clique na janela de visualização 2. A caixa da Figura 4.26 surgirá. Digite o texto da Figura 4.26, marque a opção ☒ **Fórmula LaTeX** e logo depois clique em *OK*. O texto *Soluções numéricas para um PVI* aparecerá na janela de visualização 2. Marcar a opção *Fórmula LaTeX*, na caixa de texto, permite que textos e fórmulas

### 4.3. CONSTRUÇÃO DO APPLET

---

sejam digitados usando os mesmos comandos e a mesma sintaxe do Latex. Para configurar o texto, alterando cor, tamanho e tipo da fonte, clique com o botão direito do mouse em cima do texto e selecione a opção *Propriedades*.

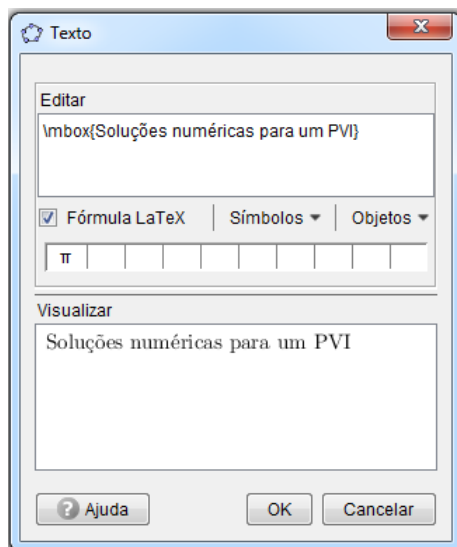


Figura 4.26: Caixa de texto.

Todos os textos do *applet*, já configurados e posicionados convenientemente, foram criados da mesma forma e estão na Figura 4.27.

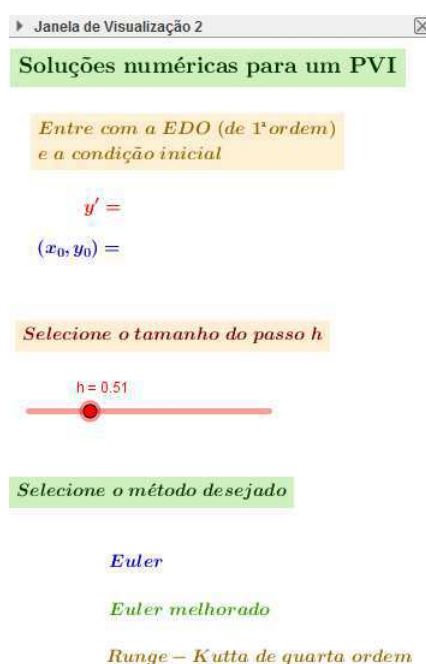


Figura 4.27: Textos do *applet*.

Vale lembrar que em todos os textos são marcados a opção *Posição absoluta na tela*, clicando sobre os mesmos com o botão direito do mouse e selecionando esta opção.

#### 4.3.11 Passo 10 - Criando campos de entrada e caixas para exibir e esconder as poligonais e tabelas

Vamos criar dois campos de entrada: um para a EDO e outro para a condição inicial. Na barra de ferramentas, selecione a ferramenta  Campo de Entrada. Em seguida, clique em qualquer local da janela de visualização 2. A caixa da Figura 4.28 aparecerá. Digite  $y'$  como legenda para o campo de entrada e vincule o objeto  $y'(x, y) = y \cos(x)$ , como mostra a Figura 4.28

### 4.3. CONSTRUÇÃO DO APLET

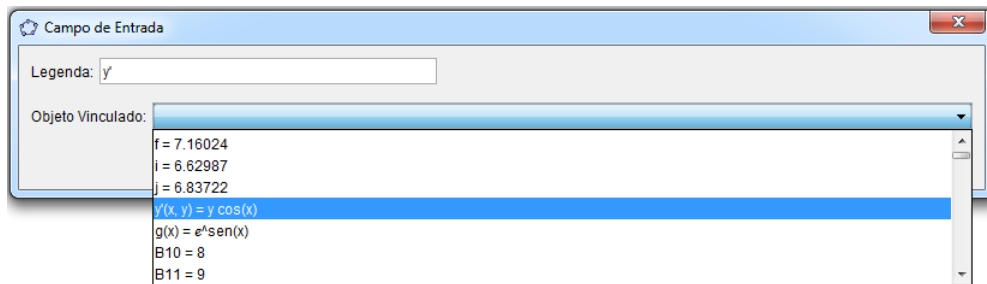


Figura 4.28: Configurando o campo de entrada.

Ao criar o campo de entrada na janela de visualização, clique nele com o botão direito do mouse e selecione a opção *Propriedades*, na aba *Básico*, desmarque a opção *Exibir Rótulo* e na aba *Estilo*, digite 10 para o comprimento do campo de texto. Também é possível configurar a cor da fonte e a cor de fundo do campo de entrada na aba *Cor*.

Crie outro Campo de entrada, da mesma forma que o anterior, com a legenda  $(x_0, y_0)$  e vinculado ao objeto A2. Aplique as mesmas configurações do campo de entrada anterior a esse segundo campo. Em seguida, posicione-os conforme a Figura 4.29.

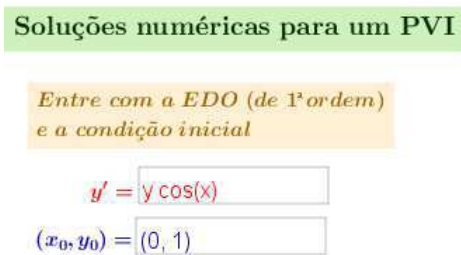


Figura 4.29: Campos de entrada.

Agora, para criar as caixas que permitem a exibição (ou não) das poligonais e tabelas, selecione a ferramenta  Caixa para Exibir / Esconder Objetos e clique em qualquer local da janela de visualização 2. Ao clicar na janela de visualização 2, surgirá uma caixa na qual pode-se escolher os objetos que se deseja exibir/esconder. De acordo com a Figura 4.30, digite *Euler* para a legenda e Selecione os objetos: *CaminhoPoligonal f*, os pontos A2:A12 e o texto2 que corresponde a tabela do método de Euler.

### 4.3. CONSTRUÇÃO DO APLET

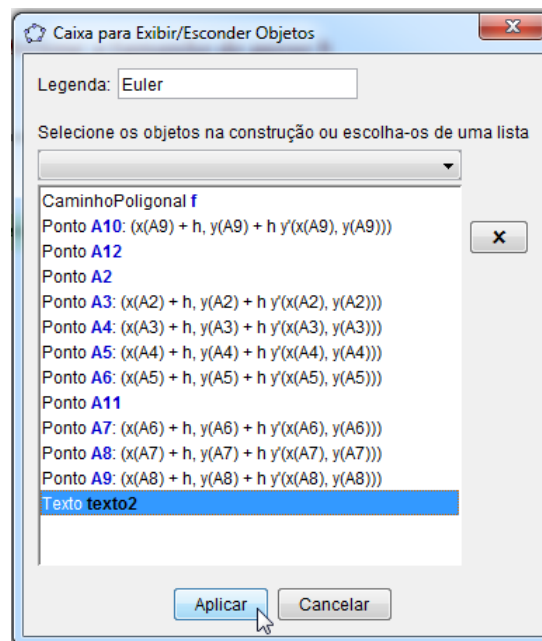


Figura 4.30: Exibindo/escondendo a poligonal e a tabela do método Euler.

Ao clicar em *Aplicar* a caixa que Exibe/esconde a poligonal e a tabela do método de Euler aparecerá na janela de visualização 2. Em seguida, esconda o rótulo da caixa e a posicione de acordo com a Figura 4.31.

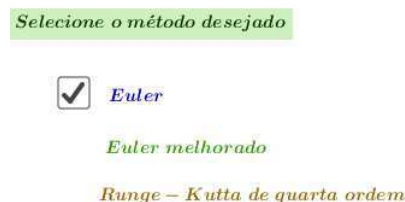


Figura 4.31: Caixa para Exibir/esconder a poligonal e a tabela do método Euler.

Quando a caixa estiver marcada a poligonal e a tabela do método de Euler serão exibidas na janela de visualização. Crie mais duas caixas de acordo com as Figuras 4.32 e 4.33, em seguida, oculte os rótulos destas caixas e as posicione de acordo com a Figura 4.34.

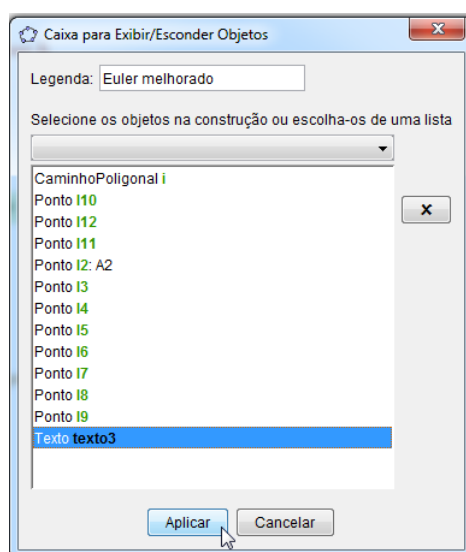


Figura 4.32: Exibindo/escondendo a poligonal e a tabela do método Euler melhorado.

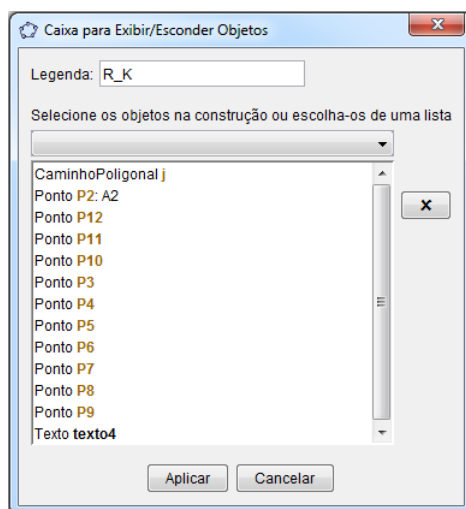


Figura 4.33: Exibindo/escondendo a poligonal e a tabela do método de Runge-Kutta de quarta ordem.

### 4.3. CONSTRUÇÃO DO APPLET

---

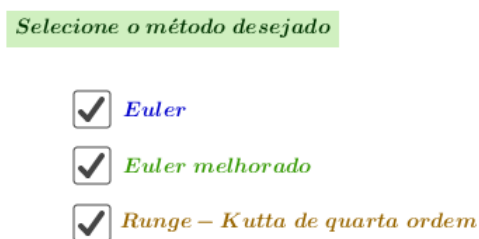


Figura 4.34: Caixa para Exibir/esconder as poligonais e as tabelas dos métodos de Euler, Euler melhorado e Runge-Kutta de quarta ordem.

Por fim, clicando com o botão direito do mouse em qualquer local da janela de visualização e selecionando a opção *Propriedades* é possível alterar a cor de fundo da janela. Da mesma forma, pode-se alterar, cor, fonte, tamanho, localização, etc ... de qualquer objeto das janelas de visualização. Sendo assim, é possível configurar a interface do *applet* de diversas formas. A versão final, do *applet* apresentado neste trabalho, é a da Figura 4.1.

#### 4.3.12 Passo 11 - Disponibilizando o applet na web.

Após a finalização do *applet*, para que seja possível disponibilizá-lo na *web*, é necessário criar uma conta no site oficial do *Geogebra* em <https://accounts.geogebra.org/user/create/expiration/129600/clientinfo/website>. Logo após, na barra de menus do arquivo da construção do *applet*, clique em *Arquivo*, em seguida, selecione a opção *Exporta* e clique em  *Planilha Dinâmica como Página WEB (html) ...* Ctrl+Shift+W. Ao abrir a janela da Figura 4.35, edite o título e as descrições do *applet*.

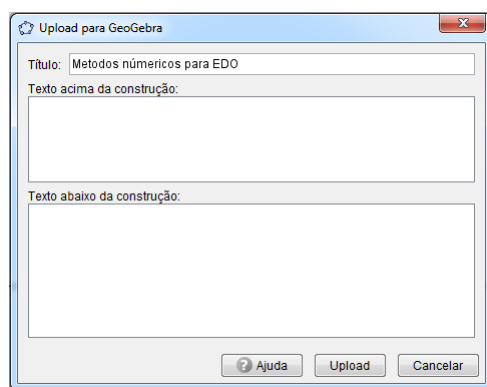


Figura 4.35: Upload do *applet*.

### 4.3. CONSTRUÇÃO DO APPLET

---

Ao clicar em *Upload* o *applet* será, de fato, criado e estará disponível na *web*, no site de materiais do *Geogebra*. Para fazer modificações no *applet* criado, basta abrir a construção (arquivo .ggb), fazer as alterações necessárias e salvá-las. Estas modificações aparecerão, automaticamente, quando o *applet* for aberto, uma vez que o mesmo está associado à construção feita no *Geogebra*. No *applet* é possível realizar ações diversas, mas, ao fechá-lo esse retornará ao seu estado original, ou seja, não é possível salvar modificações, diretamente, no próprio *applet*. Contudo, qualquer usuário pode fazer o download do arquivo .ggb, do *applet*, e assim, modificá-lo ou melhorá-lo.



## Considerações finais

Neste trabalho fizemos uma introdução ao estudo de métodos numéricos de passo simples que são utilizados na resolução de problemas de valor inicial. Apresentamos o método de Euler de aproximação por meio da reta tangente à curva da solução exata da equação diferencial ordinária, por integrais e, até mesmo, através de uma série de Taylor truncada. Utilizando a série de Taylor, analisamos o erro de truncamento local do método de Euler. Vimos que o erro está diretamente relacionado ao tamanho do passo. Mostramos que, realizando um pequeno ajuste na fórmula do método de Euler, se obtém o método de Euler melhorado. Através da análise do erro de truncamento local do método de Euler melhorado, concluimos que esse método é mais preciso, comparado ao método de Euler para um mesmo tamanho de passo. Apresentamos os métodos de Runge-Kutta de segunda, terceira e quarta ordem. Mostramos que o método de Runge-Kutta de segunda ordem equivale ao método de Euler melhorado. Através de um exemplo, comparamos os métodos de Euler, Euler melhorado e Runge-Kutta de quarta ordem e percebemos o quão mais eficiente é o método de Runge-Kutta comparado com os demais. Propomos um *applet*, construído no *Geogebra* e disponibilizado na web, que mostra as soluções numéricas de um problema de valor inicial utilizando os métodos de Euler, Euler melhorado e Runge-Kutta de quarta ordem. Mostramos passo a passo a construção do *applet*. Esperamos que o *applet* seja uma ferramenta útil no ensino e aprendizagem de métodos numéricos e que este trabalho possa motivar professores a criar *applets*, ou melhorar o aqui apresentado, de forma conveniente.

# Referências Bibliográficas

- [1] ARENALES, S.; DAREZZO, A. *Cálculo Numérico: Aprendizagem com Apoio de Software*. São Paulo: Thomson Learning, 2008.
- [2] BARCELOS, G. T.; BATISTA, S. C. F.; BEHAR, P. A.; Passerino, L. M. Applets em Ambientes de Geometria Dinâmica: Ações Para a Formação de Professores de Matemática. *Renote*. v 7, n. 3, p. 559-569, 2009.
- [3] BARROSO, L. C.; BARROSO, M. M. A.; FILHO, F. F. C.; CARVALHO, M. L. B.; MAIA, M. L. *Cálculo Numérico com Aplicações*. 2. ed. São Paulo: Harbra, 1987.
- [4] BOYCE, W. E.; DIPRIMA, R. C. *Equações Diferenciais Elementares e Problemas de Valores de Contorno*. 9. ed. Rio de Janeiro: LTC, 2010.
- [5] BRANDÃO, L. O.; ISOTANI, S.; MOURA, J. G. Imergindo a Geometria Dinâmica em Sistemas de Educação a Distância: iGeom e SAW. *Revista Brasileira de Informática na Educação*. v. 14, n. 1, p. 41-49, 2006.
- [6] BUTCHER, J. C. Numerical Methods for Ordinary Differential Equations in the 20th Century. *Journal of Computational and Applied Mathematics*. v 125, n.1-2, p. 1-29, 2000.
- [7] BUTCHER, J. C. *Numerical Methods for Ordinary Differential Equations*. 2. ed. Chichester: John Wiley & Sons, 2008.
- [8] CHAPRA, S. C.; CANALE R. P. *Métodos Numéricos para Engenharia*. 5. ed. Porto Alegre: AMGH, 2011.
- [9] COLLINS, Peter. J. *Differential and Integral Equations*. New York: Oxford University Press, 2006.
- [10] DAVIS, M. E. *Numerical Methods and Modeling for Chemical Engineers*. New York: John Wiley & Sons, 1984.
- [11] DEITEL, H. M.; DEITEL, P. J. *Java, como programar*. 8. ed. São Paulo: Pearson, 2010.

- [12] LIMA, Elon Lages. *Análise Real*. 12. ed. Coleção Matemática Universitária. Rio de Janeiro: IMPA, 2016. v.1.
- [13] MAIOLI, G. *Métodos Numéricos para Equações Diferenciais Ordinárias*. Dissertação de Mestrado em Matemática. Universidade Estadual Paulista, Rio Claro. 2015.
- [14] RUGGIERO, M. A. G.; LOPES, V. L. R. *Cálculo Numérico: Aspectos Teóricos e Computacionais*. 2. ed. São Paulo: Pearson, 1996.
- [15] SANTOS V. C. P. *Mathlets: Possibilidades e Potencialidades para uma Abordagem Dinâmica e Questionadora no Ensino de Matemática*. Dissertação de Mestrado em Ensino de Matemática. Universidade Federal do Rio de Janeiro, Rio de Janeiro. 2008.
- [16] SOTOMAYOR, Jorge. *Lições de Equações Diferenciais Ordinárias*. Projeto Euclides. Rio de Janeiro: IMPA, 1979.
- [17] UNDERWOOD, J.; HOADLEY, C.; LEE, H. S.; HOLLEBRANDS, K. F.; DIGIANO, C.,; RENNINGER, K. A. Idea: Identifying Design Principles in Educational Applets. *Educational Technology Research and Development*. v. 53, n. 2, p. 99-112, 2005.
- [18] ZILL, D. G. *A First Course in Differential Equations with Modeling Applications*. 9. ed. Stanford, Cengage Learning, 2008.