## **Atividade 02 (2018/2019) » MNEDOPVI**

Unidade Curricular: Análise Matemática II

Docentes: Arménio Correia & Nuno Baeta

Grupo de trabalho: Alexandre Costa Reis - 21280926

Celso André Ferreira Jordão - 21130067

Fabio Capobianchi de Souza - 21280924

[**Atividade 02 (2018/2019) » MNEDOPVI**](#_bu802jg52r8h) **1**

[**Introdução**](#_u6qgp3a8gcf0) **3**

[O que é o MATLAB?](#_tut8w9imvk6a) 3

[Interpretação do problema e definição de PVI](#_41kvjfl1p3d) 3

[**Métodos Numéricos para a resolução de um PVI**](#_q099p3l57bo8) **4**

[Método de Euler](#_6sietjvvtt1) 4

[Fórmulas](#_pmtid8s7p97g) 4

[Algoritmo & Função](#_blt3x76y4xam) 4

[Método de Euler Melhorado](#_29vubgqq591) 5

[Fórmulas](#_tsuet52tnxg6) 5

[Algoritmo & Função](#_k9ph8kvn2icf) 5

[Runge Kutta de Ordem 2](#_caz0vtu2eco) 5

[Fórmulas](#_ujadt6hif1ul) 5

[Algoritmo & Função](#_9ufeeyoq4u9r) 6

[Runge Kutta de Ordem 3](#_8mwqm2qlkkpt) 6

[Fórmulas](#_o2tziv9mivnk) 6

[Algoritmo & Função](#_eppje72m0qge) 6

[Runge Kutta de Ordem 4](#_i8bgk6pn5keo) 7

[Fórmulas](#_w3j7z9h102vq) 7

[Algoritmo & Função](#_m7uzt0ch2z3o) 7

[ODE45](#_nj53omb6c5pi) 8

[Fórmulas](#_wnth7cn8l1ym) 8

[Algoritmo & Função](#_vx2iuq56eq6o) 8

[**Exemplos de aplicação e teste dos métodos**](#_e0joapts8xq1) **9**

[Exercício 4 do um teste A de 2015/2016](#_mtnol09rql21) 9

[Alínea (c)](#_ldnfy9mpspn6) 9

[Alínea (d)](#_ydk0r3jgr6ja) 9

[Exemplos de output - GUI com gráfico e tabela](#_uhts9ros8kcb) 10

[Extração de CSV](#_g44ma13mkpr3) **12**

[**Conclusão**](#_4gy90gwbg1yg) **13**

# Introdução

Esta actividade prática, foi sugerida, pelos professores da UC de AM2, e consiste na realização de uma interface/aplicação que resolva problemas de valor inicial (PVI), recorrendo a métodos distintos (Euler, Runge-Kutta de 2ª e 4ª Ordem e, numa fase mais avançada, a implementação de outros 2 métodos, Euler Modificado e ODE45).

Esta actividade tem como objectivo principal, promover a familiarização com estes métodos numéricos assim como o aprofundamento de conhecimentos a nível de programação, em Matlab.

## O que é o MATLAB?

MATLAB (MATrix LABoratory) é um software interativo de alta performance voltado para o cálculo numérico. O MATLAB integra análise numérica, cálculo com matrizes, processamento de sinais e construção de gráficos num ambiente fácil de usar, onde problemas e soluções são expressos somente como eles são escritos matematicamente, ao contrário da programação tradicional.

O MATLAB é um sistema interativo, cujo elemento básico de informação é uma matriz que não requer dimensões padrão.

Este software permite programar numa linguagem bastante semelhante a linguagem C, permitindo ainda a elaboração de janelas de interface, através da opção **guide.**

## Interpretação do problema e definição de PVI

Uma equação diferencial que satisfaça certas condições adicionais é denominada **Problema de Valor Inicial (PVI).**

Exemplo:

Se forem conhecidas as condições adicionais, podemos obter soluções particulares para a equação diferencial caso contrário, são conhecidas condições adicionais poderemos obter a solução geral.

# Métodos Numéricos para a resolução de um PVI

Neste capítulo, vamos proceder à explicação da algoritmia e desenvolvimento das funções em causa, isto é:

* Euler
* Euler Melhorado
* Runge Kutta de ordem 2
* Runge Kutta de ordem 3 (Extra)
* Runge Kutta de ordem 4
* ODE 45

## Método de Euler

Este é o mais simples e mais antigo dos métodos numéricos utilizados resolução de um PVI. Foi criado no séc. XVIII pelo matemático Leonhard Euler (1707 – 1783).

### Fórmulas

***ym*+1 *= ym + hf( tm , ym )***

### Algoritmo & Função

**Ler entradas:** f, a, b, n, y0

**Saídas:** y

y(1)=y0

t(1)=a

h=(b-a)/n

**Para i de 1 até n fazer**

y(i+1)=y(i)+h\*f(t(i),y(i))

t(i+1)=t(i)+h

**Fim**

## Método de Euler Melhorado

### Fórmulas



### Algoritmo & Função

**Ler entradas:** f, a, b, n, y0

**Saídas:** y

y=zeros(1,n+1);

y(1)=y0;

t=a:h:b

h=(b-a)/n

**Para i de 1 até n fazer**

y(i+1)=y(i)+h\*f(t(i),y(i));

y(i+1)=y(i)+h/2\*(f(t(i), y(i))+f(t(i+1), y(i+1)));

**Fim**

## Runge Kutta de Ordem 2

### Fórmulas



### 

### Algoritmo & Função

**Ler entradas:** f, a, b, n, y0

**Saídas:** y

h = (b-a)/n;

t(1) = a;

y(1) = y0;

**Para i de 1 até n fazer**

t(i+1) = t(i) + h;

k1 = h\*f(t(i), y(i));

k2 = h\*f(t(i+1), y(i) + k1);

y(i+1) = y(i) + (k1+k2)/2;

**Fim**

## Runge Kutta de Ordem 3

### Fórmulas



### Algoritmo & Função

**Ler entradas:** f, a, b, n, y0

**Saídas:** y

h = (b-a)/n;

t(1) = a;

y(1) = y0;

**Para i de 1 até n fazer**

t(i+1) = t(i) + h;

k1 =h\*f(t(i),y(i));

k2=h\*f(t(i)+ h/2, y(i) + k1/2);

k3=h\*f(t(i)+h,y(i)-k1+2\*k2); y(i+1) = y(i)+(1/6)\*(k1+4\*k2+k3);

**Fim**

## Runge Kutta de Ordem 4

### Fórmulas



### Algoritmo & Função

**Ler entradas:** f, a, b, n, y0

**Saídas:** y

h = (b-a)/n;

t(1) = a;

y(1) = y0;

**Para i de 1 até n fazer**

t(i+1) = t(i) + h;

k1 = h\*f(t(i), y(i));

k2 = h\*f(t(i) + (h/2), y(i) + (k1/2));

k3 = h\*f(t(i) + (h/2), y(i) + (k2/2));

k4 = h\*f(t(i+1), y(i)+k3);

y(i+1) = y(i) + (k1+(2\*k2)+(2\*k3)+k4)/6;

**Fim**

## ODE45

### Fórmulas

***[t,y] = ode45(f,tsp,y0);***

### Algoritmo & Função

**Ler entradas:** f, a, b, n, y0

**Saídas:** y

h=(b-a)/n;

tsp = a:(b-a)/n:b;

y(1) = y0;

**Para i de 1 até n fazer**

[t,y] = ode45(f,tsp,y0);

**Fim**

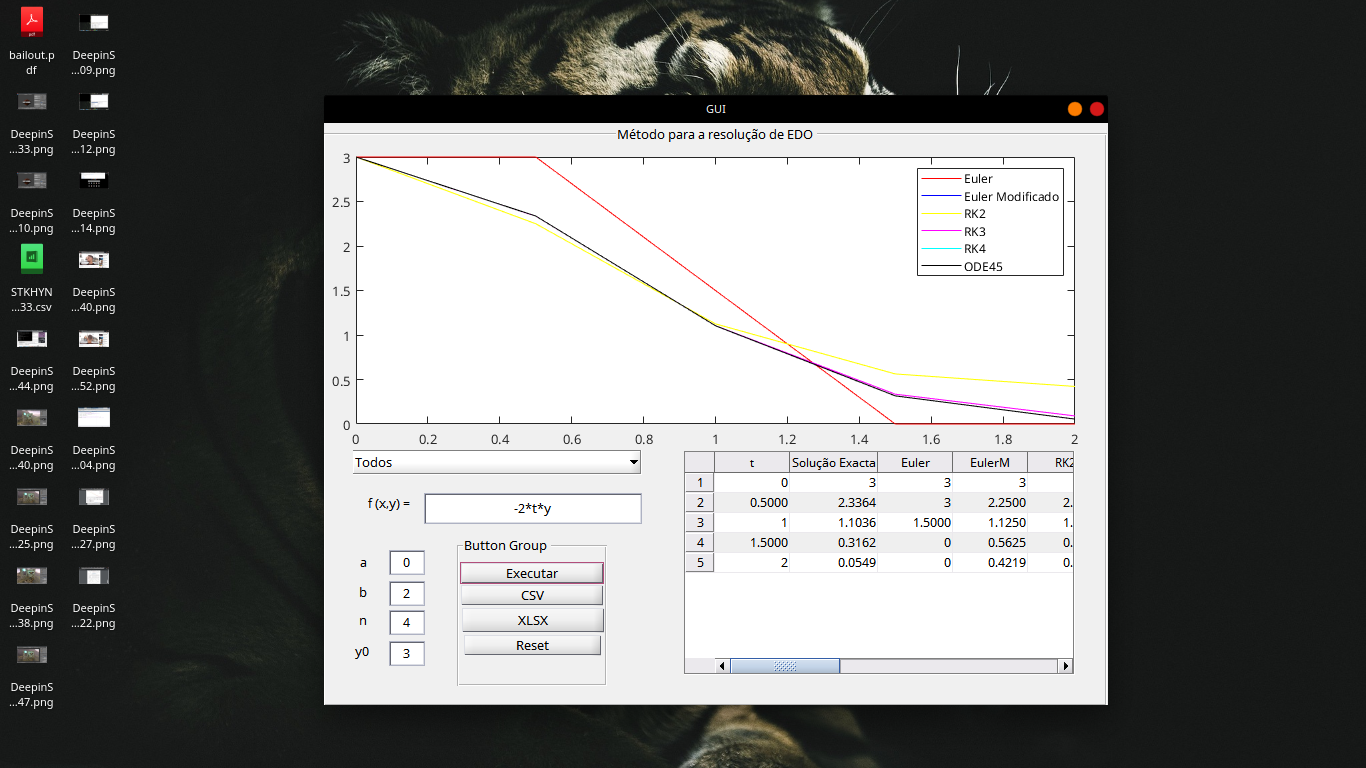
# Exemplos de aplicação e teste dos métodos

## Exercício 4 do um teste A de 2015/2016

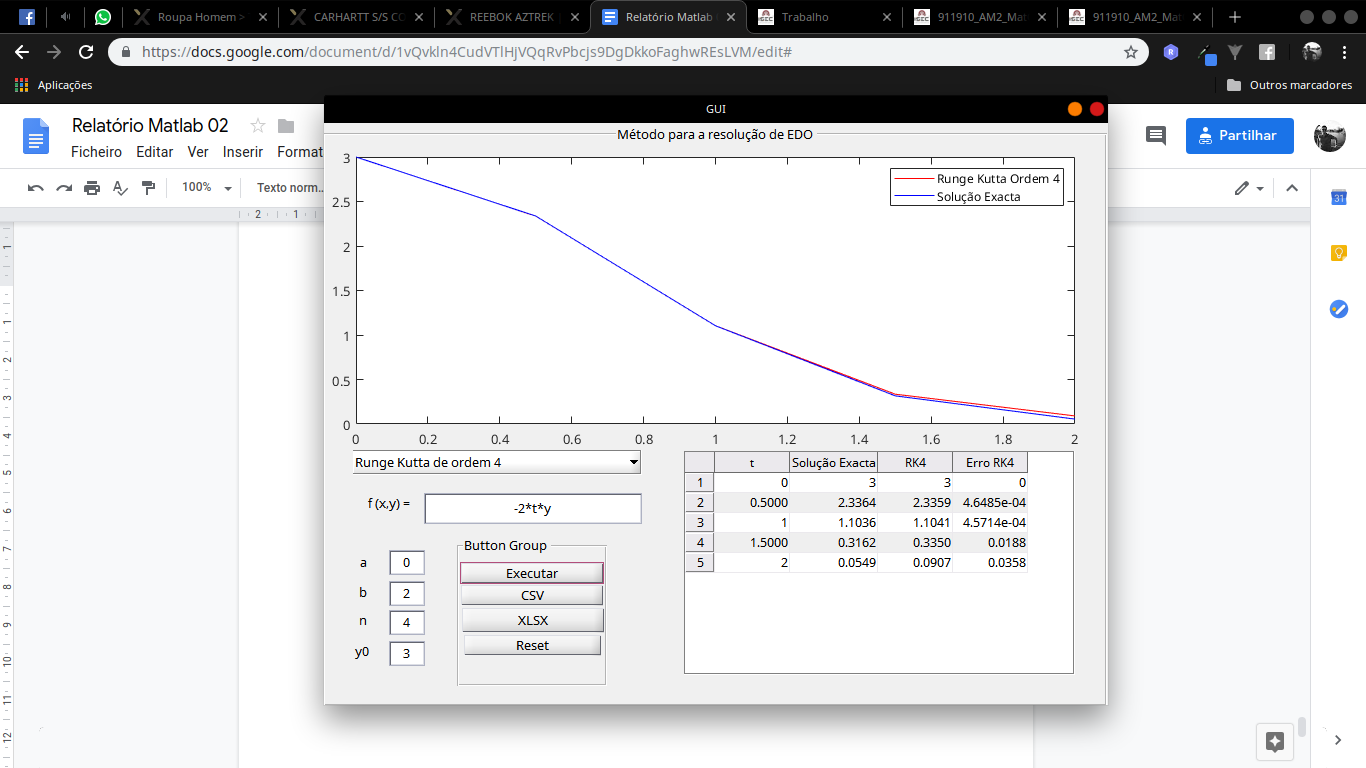
### Alínea (c)

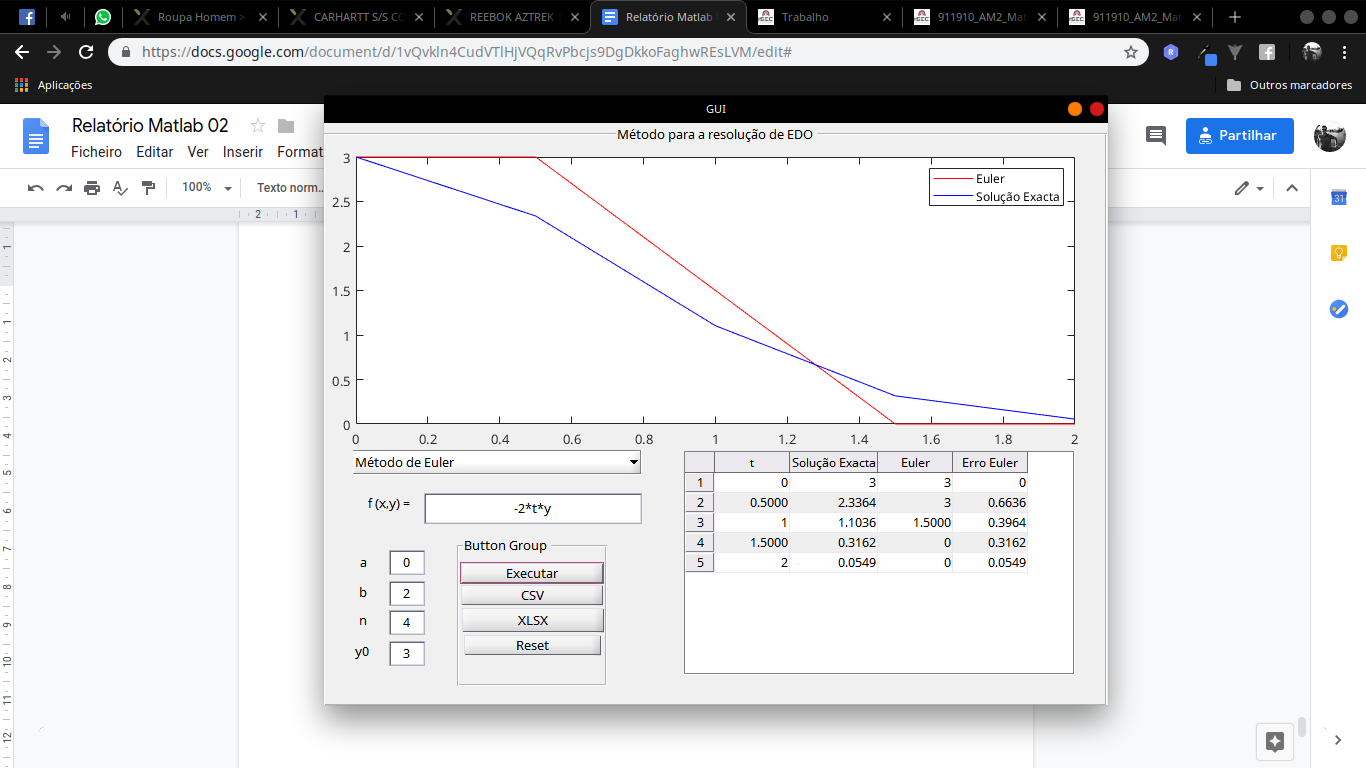
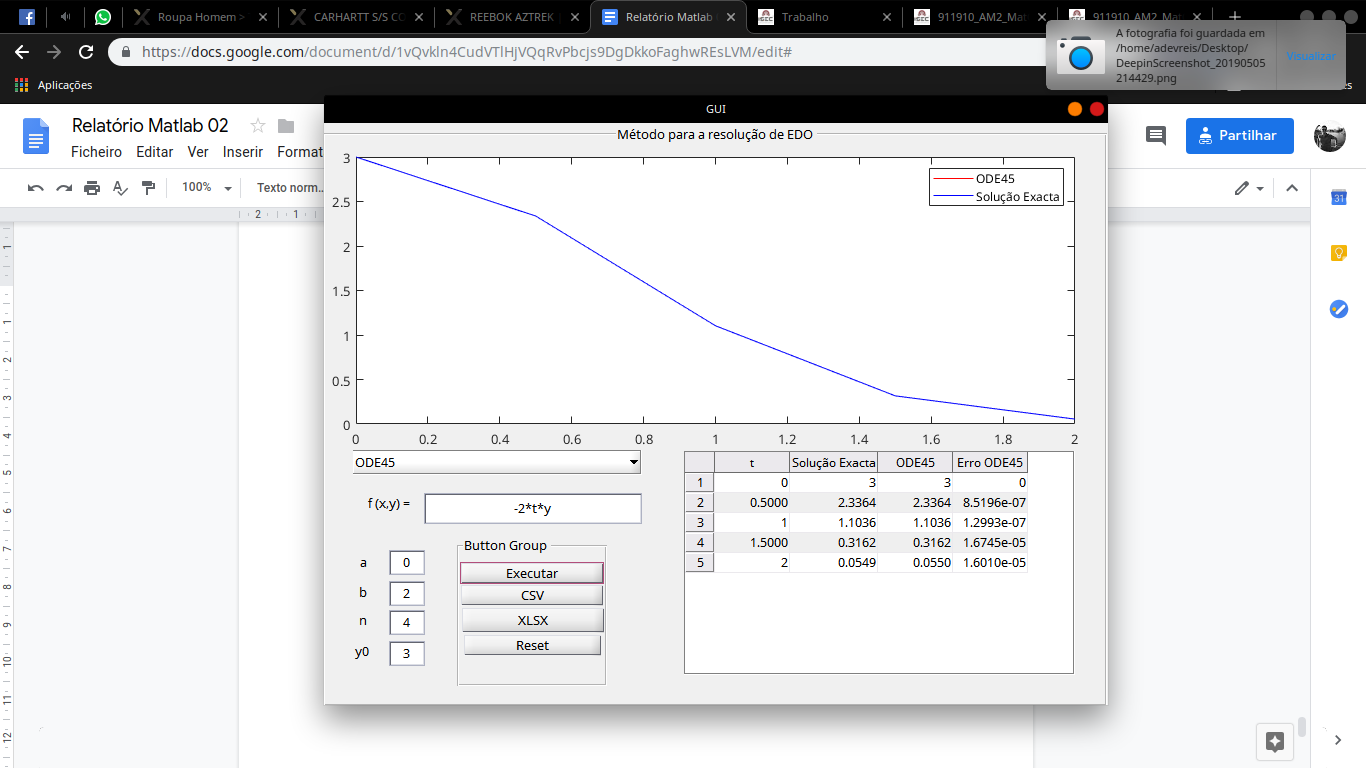
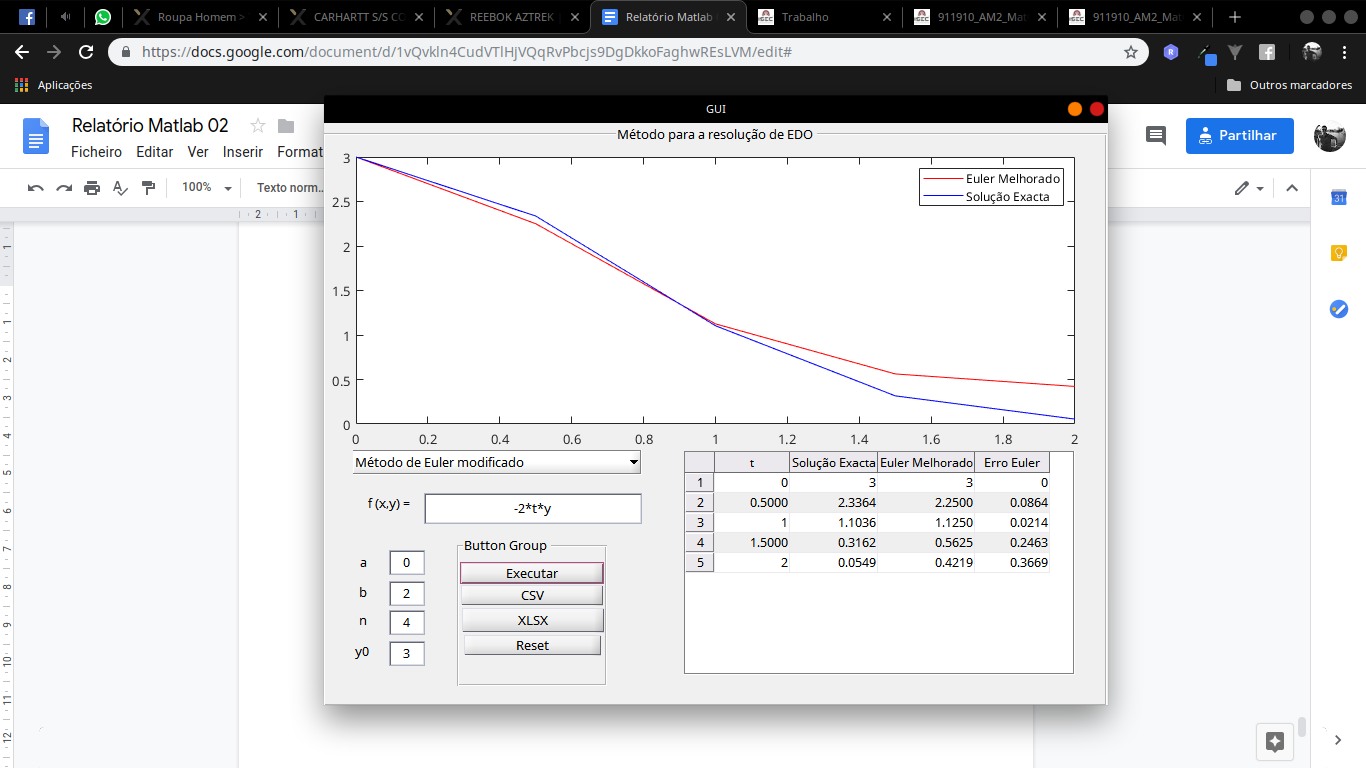
|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| i | ti | Exata | Euler | RK2 | RK4 | Erro Euler | Erro RK4 | Erro RK2 |
| 0 | 0 | 3 | 3 | 3 | 3 | 0 | 0 | 0 |
| 1 | 0.5 | 2.3364 | 3 | 2.2500 | 2.3359 | 0.6636 | 0.0864 | 0.0005 |
| 2 | 1 | 1.1036 | 1.5000 | 1.1250 | 1.1041 | 0.3964 | 0.0214 | 4.5714e-04 |
| 3 | 1.5 | 0.3162 | 0 | 0.5625 | 0.3350 | 0.3162 | 0.2463 | 0.0188 |
| 4 | 2 | 0.0549 | 0 | 0.4219 | 0.0907 | 0.0549 | 0.3669 | 0.0358 |

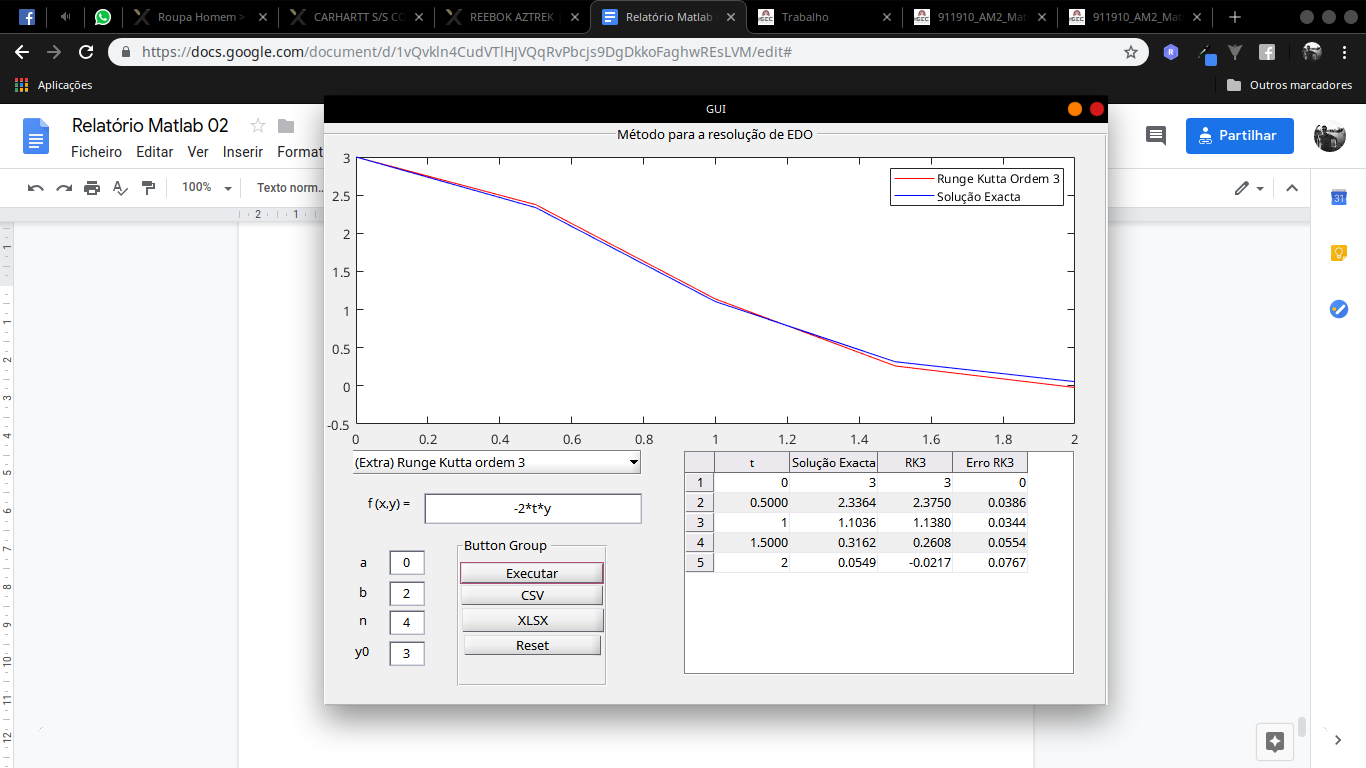
### Alínea (d)

A figura 9 é a imagem que melhor representa a solução gráfica do problema PVI, pois o Y0 (ponto inicial) é definido pelo enunciado como 0 e o intervalo é de 0 a 2, o que não se verifica na figura 8.

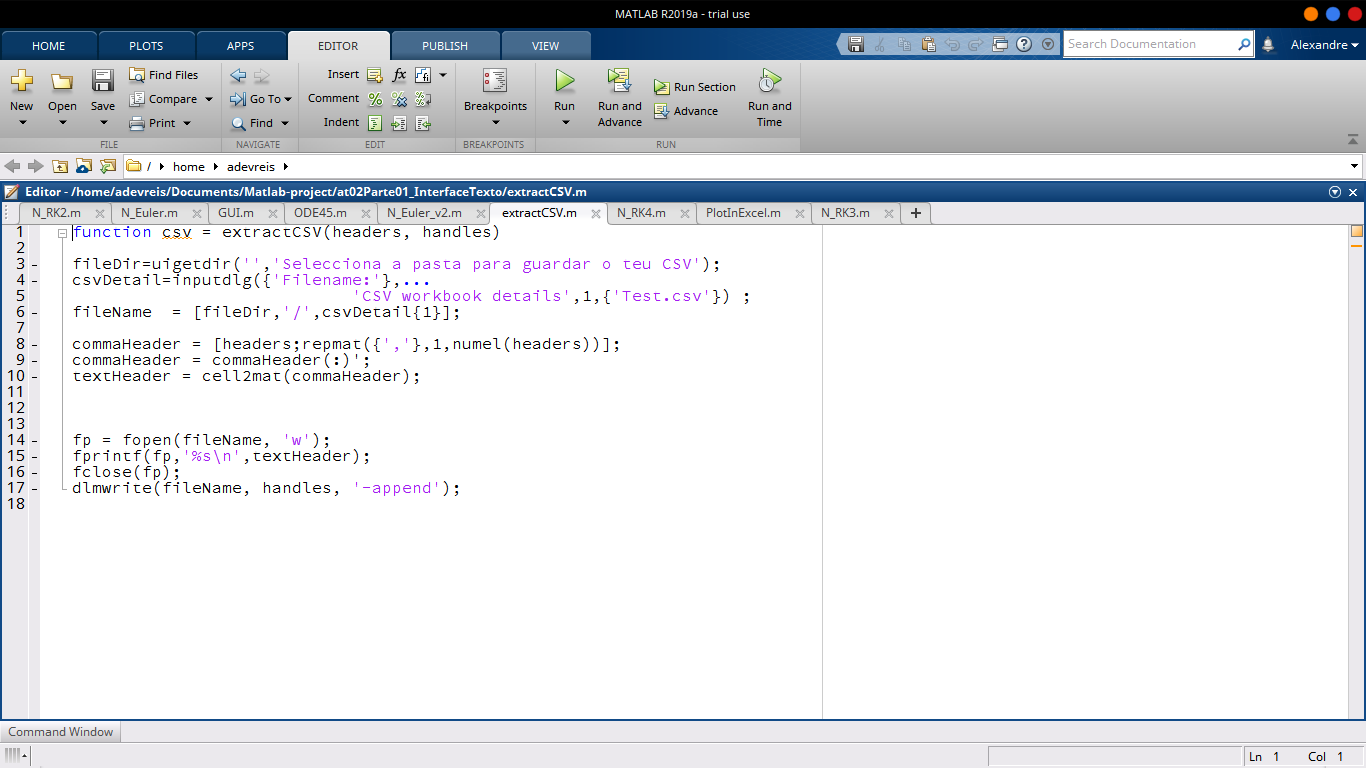
## Exemplos de output - GUI com gráfico e tabela







# Extração de CSV



Esta função pede ao utilizador que especifique onde quer guardar o ficheiro csv e qual o nome do mesmo.

Após isto, a função faz o parse dos headers e de seguida insere-os no ficheiros através da função fprintf. Para inserir a informação no csv, usamos o dlmwrite, com o argumento append, que ao invés de limpar tudo, adiciona ao que foi escrito anteriormente, ou seja, os headers.

# Conclusão

Neste relatório, foram abordados foram abordadas as funções faladas nas lecionadas nas aulas, tratando-os minuciosamente desde as fórmulas utilizadas, algoritmos e alguns exercícios de aplicação para testar o funcionamento da GUI.

Os objectivos da actividade prática foram alcançados, pois a aplicação, possivelmente, não tem erros assinaláveis, o que foi bom para a nossa evolução na programação em MATLAB.

Todas as funções foram desenvolvidas dentro dos padrões definidos pelos professores Arménio Correia e Nuno Baeta, sendo assim útil a nossa perceção de métodos numéricos e de programação.