CURSO: ENGENHARIA ELETROTÉCNICA E DE COMPUTADORES

DISCIPLINA: MATEMÁTICA COMPUTACIONAL

PROJETO COMPUTACIONAL

PARTE II

Diogo Martins Alves Nº 86980 (diogo.m.alves@tecnico.ulisboa.pt)

João Santiago Silva Nº 84081 (joao.santiago.s@tecnico.ulisboa.pt)

André Lopes Nº 84004 (andrefplopes@hotmail.com)

Data: 27/12/2017

1.-Resolução

Código do programa feito em Matlab que imprime no ecrã e guarda os valores/vetores $y_0, y_1, \ldots, y_n$ produzidos pelo método de Heun:

```
display('Insira os dados de entrada (f:[a,b]xIR^d->IR^d, y(a) em IR^d, n em IN):');
a = input('Início do intervalo a = ');
b = input('Fim do intervalo b = ');
d = input('Número de equações d = ');
str = (input('f(t, y_1, y_2,..., y_d) = ', 's'));%função a estudar (string) NOTA:↙
todas as variáveis t, y_1, ..., y_d têm que
%aparecer na expressão da função, nem que seja multiplicadas por 0 ex: [-0.001*y_1*y_2↙
+ t*0,  0.001*y_1*y_2-0.3*y_2]
f = inline(str);%converte a string para uma função do matlab
n = input('Número de iterações n = ');
y = zeros(n+1,d); %inicialização do(s) vetor(es) que irão guardar as sucessivas↙
iteradas
y(1,:) = input('Condição inicial y(a) = '); %ex: [799, 1]

h = (b-a)/n; %distância entre os pontos nos quais irá ser aproximada a função

for i = 1:n
    t_i = a + i*h;
    q = num2cell(y(i,:)); % vetor ((y_1)_i, (y_2)_i, ..., (y_d)_i)
    p = num2cell(y(i,:) + h*f(t_i,q{:})); %vetor (...,(y_a)_i + h * f(t_i, (y_a)↙
_i),...) com a=1:d
    y(i+1,:) = y(i,:) + (h/2)*( f(t_i,q{:}) + f(t_i, p{:}) ); %Método de Heun
    %impressão das sucessivas iteradas:
    fprintf('y_%d = ( ', i);
    for j = 1:d
        fprintf(' %.10f, ', y(i,j));
    end
    fprintf(')\n');

end
```

2.(a).-Resolução

Para resolver este problema alterámos algumas linhas de código no programa anterior, como mostra a figura seguinte. Os parâmetros utilizados foram `a = 0`; `b = 100`; `d = 2`; `f(t,y_1,y_2) = [-0.001*y_1*y_2 + t*0, 0.001*y_1*y_2-0.3*y_2]`, em que `y_1` representa S e `y_2` representa I, `n = 10000` e a condição inicial `y(a) = [799, 1]`.

```
display('Insira os dados de entrada (f:[a,b]xIR^d->IR^d, y(a) em IR^d, n em IN):');
a = input('Início do intervalo a = ');
b = input('Fim do intervalo b = ');
d = input('Número de equações d = ');
str = (input('f(t, y_1, y_2,..., y_d) = ', 's'));%função a estudar (string) NOTA:
todas as variáveis t, y_1, ..., y_d têm que
%aparecer na expressão da função, nem que seja multiplicadas por 0 ex: [-0.001*y_1*y_2
+ t*0,  0.001*y_1*y_2-0.3*y_2]
f = inline(str);%converte a string para uma função do matlab
n = input('Número de iterações n = ');
y = zeros(n+1,d); %inicialização do(s) vetor(es) que irão guardar as sucessivas
iteradas
t = zeros(n+1,1); %inicialização do vetor que guarda os valores das abcissas
t(1) = 0;
y(1,:) = input('Condição inicial y(a) = '); %ex: [799, 1]

h = (b-a)/n; %distância entre os pontos nos quais irá ser aproximada a função
for i = 1:n
    t_i = a + i*h;
    t(i+1) = t_i; %atualização do vetor de abcissas
    q = num2cell(y(i,:)); % vetor ((y_1)_i, (y_2)_i, ..., (y_d)_i)
    p = num2cell(y(i,:) + h*f(t_i,q{:})); %vetor (...,(y_a)_i + h * f(t_i, (y_a)
_i),...) com a=1:d
    y(i+1,:) = y(i,:) + (h/2)*( f(t_i,q{:}) + f(t_i, p{:}) ); %Método de Heun
    %impressão das sucessivas iteradas:
    fprintf('y_%d = ( ', i);
    for j = 1:d
        fprintf(' %.10f, ', y(i,j));
    end
    fprintf(')\n');
    %condição de paragem
    if(y(i,2) <= 0.00001)
        break
    end

end
%desenho dos gráficos
figure
plot(t(1:i+1), y(1:i+1,1));
figure
plot(t(1:i+1), y(1:i+1,2));
```

Os gráficos de *S* e *I* obtidos foram os seguintes:

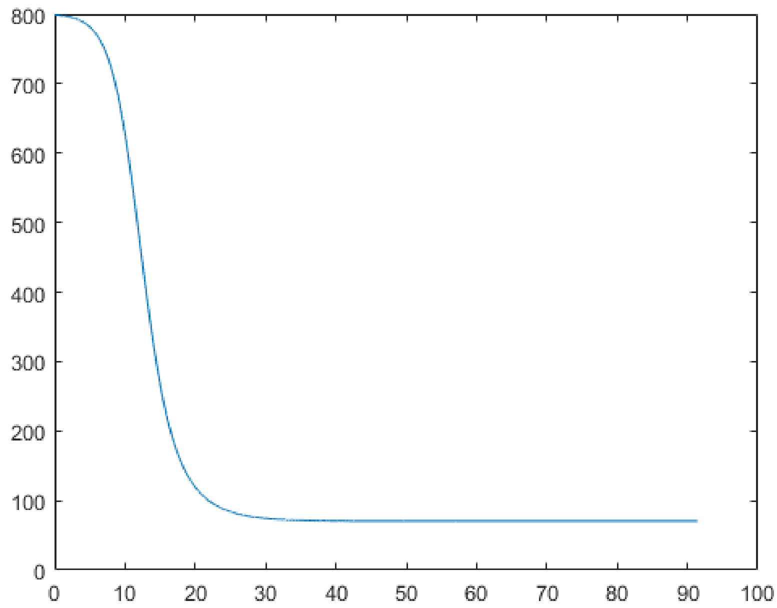

Gráfico de *S(t)*

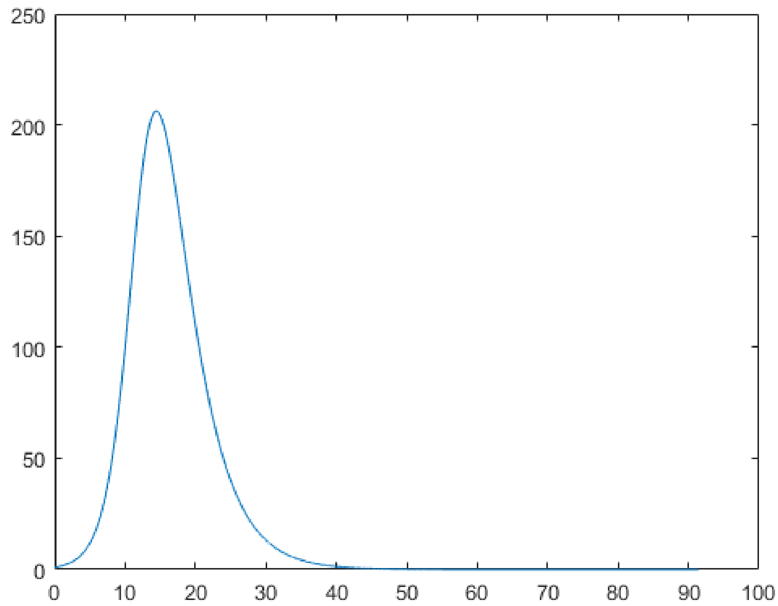

Gráfico de *I(t)*

2.(b).-Resolução

O valor máximo atingido por I é cerca de 206, o que corresponde ao valor máximo que o número de novos casos de infeção atinge no período de tempo estudado.

O limite de S pode ser aproximado pelo último valor desta função que sabemos, que é cerca de 70.

Assim, o número de crianças que irão evitar contrair varicela (N), que corresponde ao número de crianças que não contrariam a doença neste período de tempo, nem irão voltar a contrair mais tarde nas suas vidas, é dado pela diferença entre o número de crianças estudadas e a soma do número de crianças infetadas com o número de crianças ainda suscetíveis de contrair a doença no fim da epidemia ser erradicada. Portanto, tem-se:

$$N=800-\mathrm{Max}(I)-\mathrm{Lim}(S)=524$$

2.(c).-Resolução

Os gráficos obtidos foram os seguintes:

Para $v = 0.3$:

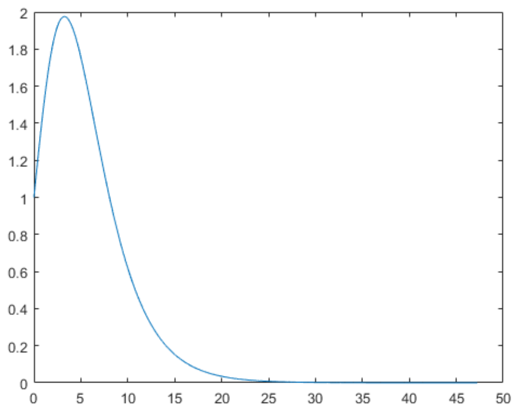

Gráfico de $I(t)$

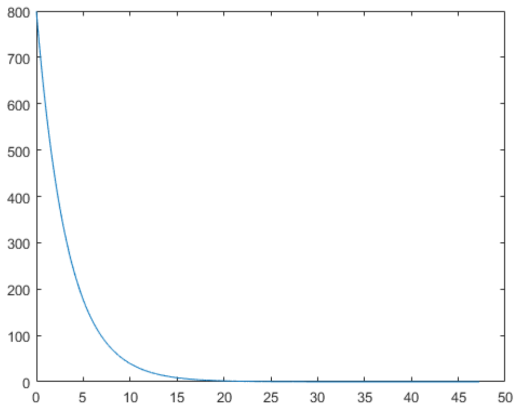

Gráfico de $S(t)$

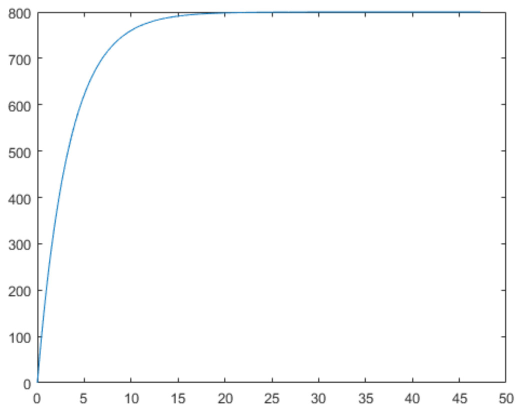

Gráfico de *R(t)*

Para *v* = 0.5:

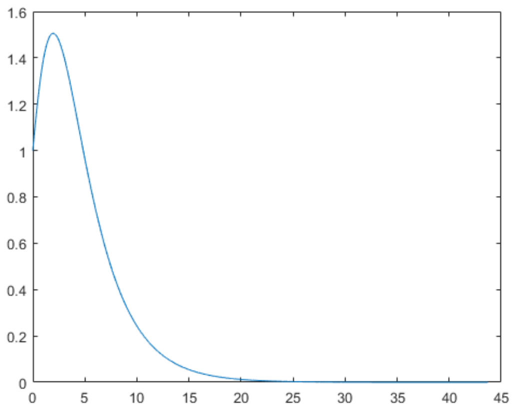

Gráfico de *I(t)*

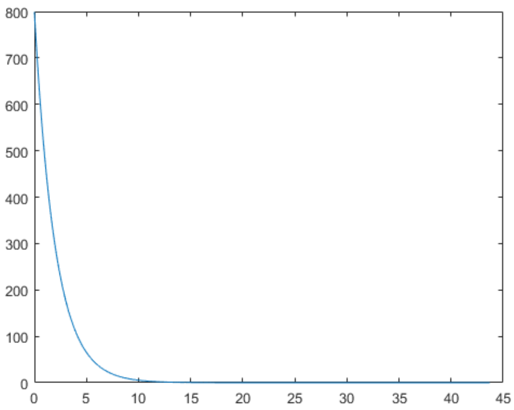

Gráfico de *S(t)*

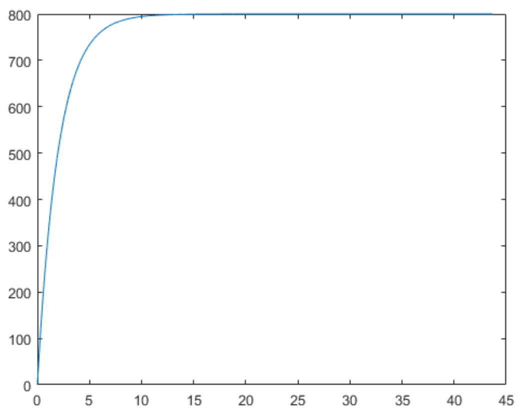

Gráfico de *R(t)*

Para *v* = 0.9:

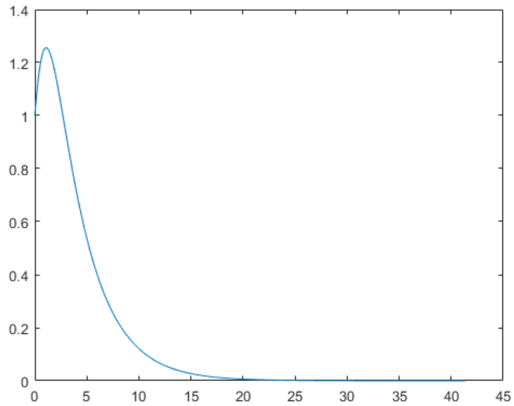

Gráfico de *I(t)*

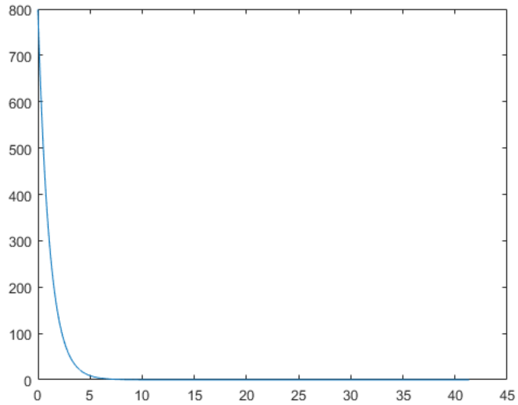

Gráfico de *S(t)*

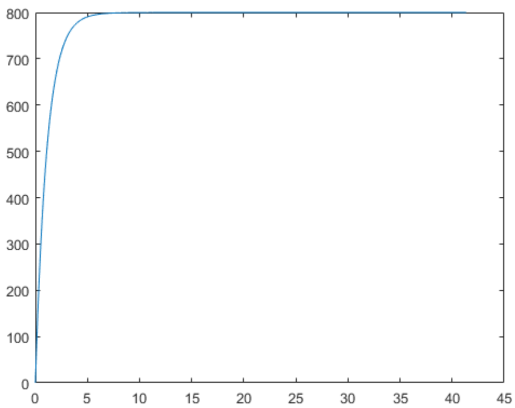

Gráfico de *R(t)*

Nos gráficos acima pode verificar-se de imediato que quanto maior é a taxa de vacinação *v*, menor será o número máximo de infetados, ocorrendo esse pico num instante de tempo menor quanto menor o valor máximo de infetados.

Relativamente ao número de crianças suscetíveis a apanharem varicela, observa-se que o aumento da taxa de vacinação provoca um decréscimo mais rápido e acentuado no valor de *S(t)*, pelo que o número de crianças suscetíveis a apanharem varicela diminui mais rapidamente e, para os valores de v estudados, este número tende para 0, o que não acontecia no caso em que não existia vacinação. Já o número de recuperados evolui com maior declive, até estabilizar, quanto maior a taxa de vacinação. O que significa que mais rapidamente a população fica imune á varicela.

Assim, conclui-se que a vacinação é crucial no controlo do crescimento da varicela na população estudada, uma vez que reduz o número máximo de pessoas infetadas bem como o número de crianças suscetíveis decresce mais rapidamente e a população fica imune também mais rapidamente.