AULA 05
INTRODUÇÃO AOS MÉTODOS ESPECTRAIS
PTC 5525 (16/10/2025)

Osvaldo Guimarães

$EDO:$ Solução analítica: $u(x) = x^2 - 5x + 6$

$$u_{xx} - u^2 - 2 + \left(x^2 - 5x + 6\right)^2 = 0$$

$u(-1) = 12$ e $u(1) = 2$

$S:$ system

$$S_m = \underline{\bar{I}}\left[D^2 u - u^2 + \left(x^2 - 5x + 6\right)^2\right]$$

$$J_m = \frac{\partial S_m}{\partial u} = \underline{\bar{I}}\left[D^2 - 2\text{diag}(u)\right]$$

$$BC\begin{cases} u(-1) = 12 \quad J(n,1) = \dfrac{\partial u_0}{\partial u_0} = 1 \\ u(+1) = 2 \quad J(n+1,n+1) = \dfrac{\partial u_n}{\partial u_n} = 1 \end{cases}$$

EDO

$$u_{xx} - u^2 - 2 + \left(x^2 - 5x + 6\right)^2 = 0$$

$$u(-1) = 12 \quad \text{e} \quad u(1) = 2$$

Solução analítica

$$u(x) = x^2 - 5x + 6$$

$S:$ system

$$S_m = \underline{I}\left[D^2 u - u^2 + \left(x^2 - 5x + 6\right)^2 \right]$$

$$J_m = \frac{\partial S_m}{\partial u} = \underline{I}\left[D^2 - 2\mathrm{diag}(u) \right]$$

```
%%
  N = 12; %N even
  bc = [12,2];

  xs = - cos( (0:N).'*pi/N ); DM = poldif(xs,2); D2 = DM(:,:,2);

  Ibb = eye(N+1); Ibb([1,N+1],:) = [];

%% Sistema de equações // dado u

J = zeros(N+1); J(N,1) = 1; J(N+1,N+1) = 1;


%%  Guess
%  uexact = xs.^2-5*xs+6;
  u = zeros(N+1,1);

%%
  change = 1; it = 0;
%%

  while change > 1e-12      % fixed-point iteration
      % Completing the Jacobian

      J(1:N-1,:) = Ibb*D2 - 2*Ibb*diag(u);
      %% Sistema p/ given u
      r = Ibb*D2*u - Ibb*u.^2 - 2 + Ibb*(xs.^2-5*xs+6).^2 ;

      r = [r;u(1)-bc(1);u(N+1)-bc(2)];

      du   = -J\r;
      unew = u + du;

      change = norm(du,inf);
        u = unew; it = it+1;
        disp(int2str(it));
  end

%%
```

Resolva pelo método de Newton no grid de Chebyshev: Solução: $y(x) = e^{-x}$

$y'' \cdot y + y'^2 - 2e^{-2x} = 0 \qquad x \in [-1,1] \subset \mathbb{R}$

$y(-1) = e \quad y(1) = e^{-1}$ Utilize uma expansão com 25 coeficientes e

aproximação inicial $y_{\text{inicial}} = [1,1,...,1]^T$

$$S_m = \underline{I}\left[D^2 y \circ y - (Dy)^2 - 2e^{-2x} \right]$$

$$BC \begin{cases} J(n,1) = \dfrac{\partial u_0}{\partial u_0} = 1 \\ J(n+1,n+1) = \dfrac{\partial u_n}{\partial u_n} = 1 \end{cases}$$

$$J_m = \underline{I}\,\frac{\partial}{\partial y}\left[D^2 y \circ y + (Dy)^2 \right] =$$

$$= \underline{I}\left[\operatorname{diag}(y) D^2 + I \cdot \operatorname{diag}(D^2 y) + 2\operatorname{diag}(Dy) \right]$$

Resolva pelo método de Newton

no grid de Chebyshev:

$$y''\cdot y + y'^2 - 2e^{-2x} = 0 \qquad x \in [-1,1] \subset \mathbb{R}$$

$$y(-1) = e \quad y(1) = e^{-1}$$

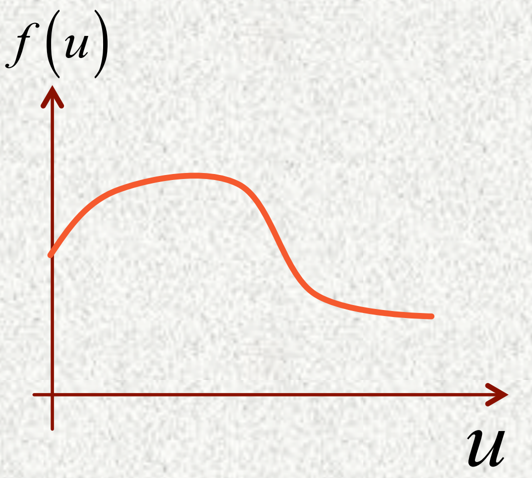
Utilize uma expansão com 25 coeficientes e

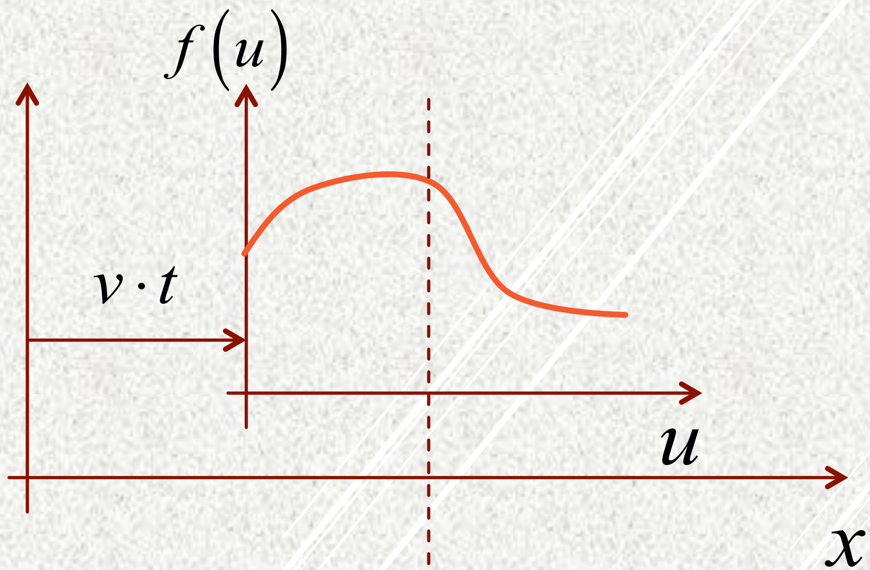
aproximação inicial $y_{\text{inicial}} = [1,1,\ldots,1]^T$

Solução: $y(x) = e^{-x}$

```
%% ODE 2 y(2)*y + (y(1))^2 = 2*exp(-2*x)  Sol: y = exp(-x);
 n = 18;
  bc = [exp(1),exp(-1)];

  xs = - cos( (0:n).'*pi/n );
  D = Generalized_Diff_Mat(xs); D2 = D^2;

  Ibb = eye(n+1); Ibb([1,n+1],:) = [];

%% Sistema de equações // dado u
J = zeros(n+1); J(n,1) = 1; J(n+1,n+1) = 1;
%%  Guess
ca = exp(-1)/2-exp(1)/2;
u = ca*(xs+1) + exp(1);
%%
  change = 1; it = 0;
%%  Loop
  while change > 1e-10      % fixed-point iteration
      % Jacobian :

      J(1:n-1,:) = Ibb*(diag(D2*u)+diag(u)*(D2) ...
                   + 2*diag(D*u) );


      %% System for given u --> F(u)
      r =Ibb*( (D2*u).*u + (D*u).^2 - 2*exp(-2*xs) );

      r = [r;u(1)-bc(1);u(n+1)-bc(2)];

      du  = -J\r; du([1,n+1]) = 0;
      unew = u + du;

      change = norm(du);
        u = unew; it = it+1;
        disp(int2str(it));
  end
```

Equação da onda





Para cada abscissa u, temos: $f(u) = f(x - vt)$

$x = u + v \cdot t \Rightarrow u = x - vt$

$$\frac{\partial u}{\partial x} = 1 \quad \text{e} \quad \frac{\partial u}{\partial t} = -v$$

$$\frac{\partial f}{\partial x} = \frac{\partial f}{\partial u} \cdot \frac{\partial u}{\partial x} = \frac{\partial f}{\partial u} \qquad \frac{\partial f}{\partial t} = \frac{\partial f}{\partial u} \cdot \frac{\partial u}{\partial t} = \frac{\partial f}{\partial u}(-v)$$

$$\boxed{\frac{\partial f}{\partial t} = -v \cdot \frac{\partial f}{\partial x}}$$

Portanto:

$$\frac{\partial f}{\partial t} = -v \cdot \frac{\partial f}{\partial x}$$

Onda regressiva: $f = f(x + v \cdot t)$

Em ambos os casos, f é arbitrária.

Para cada abscissa u, temos: $f\left(u\right)=f\left(x-vt\right)$

$$x=u+v\cdot t\Rightarrow u=x-vt$$

$$\frac{\partial u}{\partial x}=1 \quad \text{e} \quad \frac{\partial u}{\partial t}=-v$$

$$\frac{\partial^2 f}{\partial x^2}=\frac{\partial^2 f}{\partial^2 u}$$

$$\frac{\partial f}{\partial t}=\frac{\partial f}{\partial u}\cdot\frac{\partial u}{\partial t}=\frac{\partial f}{\partial u}\left(-v\right) \quad \text{e} \quad \frac{\partial^2 f}{\partial t^2}=\frac{\partial^2 f}{\partial^2 u}\cdot v^2$$

Portanto: $\dfrac{\partial^2 f}{\partial^2 u}=\dfrac{1}{v^2}\dfrac{\partial^2 f}{\partial t^2}=\dfrac{\partial^2 f}{\partial x^2}$

Equação da onda

$$\boxed{\frac{\partial^2 f}{\partial x^2}=\frac{1}{v^2}\frac{\partial^2 f}{\partial t^2}}$$

Método das linhas (MOL)

$$u_t = C_d \cdot \partial_{xx} u$$

$$u_{(0,x)} = \sin(\pi x/2), \; x \in [0,1]$$

$$u(t,0) = 0 \qquad u_x(t,1) = 0$$

Solução analítica: $u = C_d \cdot e^{-\frac{\pi^2}{4}t} \cdot \sin(\pi x/2)$

Forward Euler: (explicit, first order accurate)

$$u_{n+1} = u_n + \Delta t\, f(u_n)\,.$$

Runge–Kutta-4 (RK4): (explicit, fourth order accurate)

$$\begin{aligned} k_1 &= \Delta t\, f(u_n)\,, \\ k_2 &= \Delta t\, f\Big(u_n + \frac{1}{2}\,k_1\Big)\,, \\ k_3 &= \Delta t\, f\Big(u_n + \frac{1}{2}\,k_2\Big)\,, \\ k_4 &= \Delta t\, f(u_n + k_3)\,, \\ u_{n+1} &= u_n + \frac{1}{6}\Big[k_1 + 2\,k_2 + 2\,k_3 + k_4\Big]\,. \end{aligned}$$

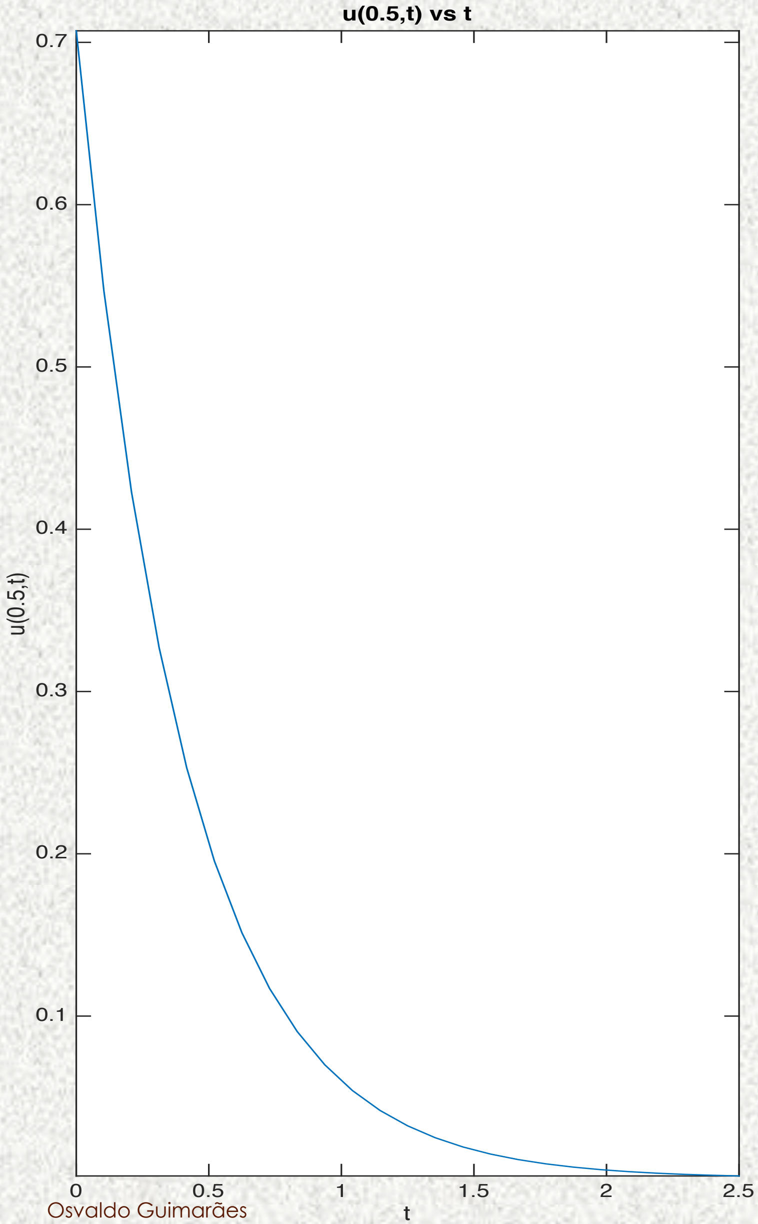
Equação da difusão $$\frac{\partial U}{\partial t} = C_d \cdot \frac{\partial^2 U}{\partial x^2}$$

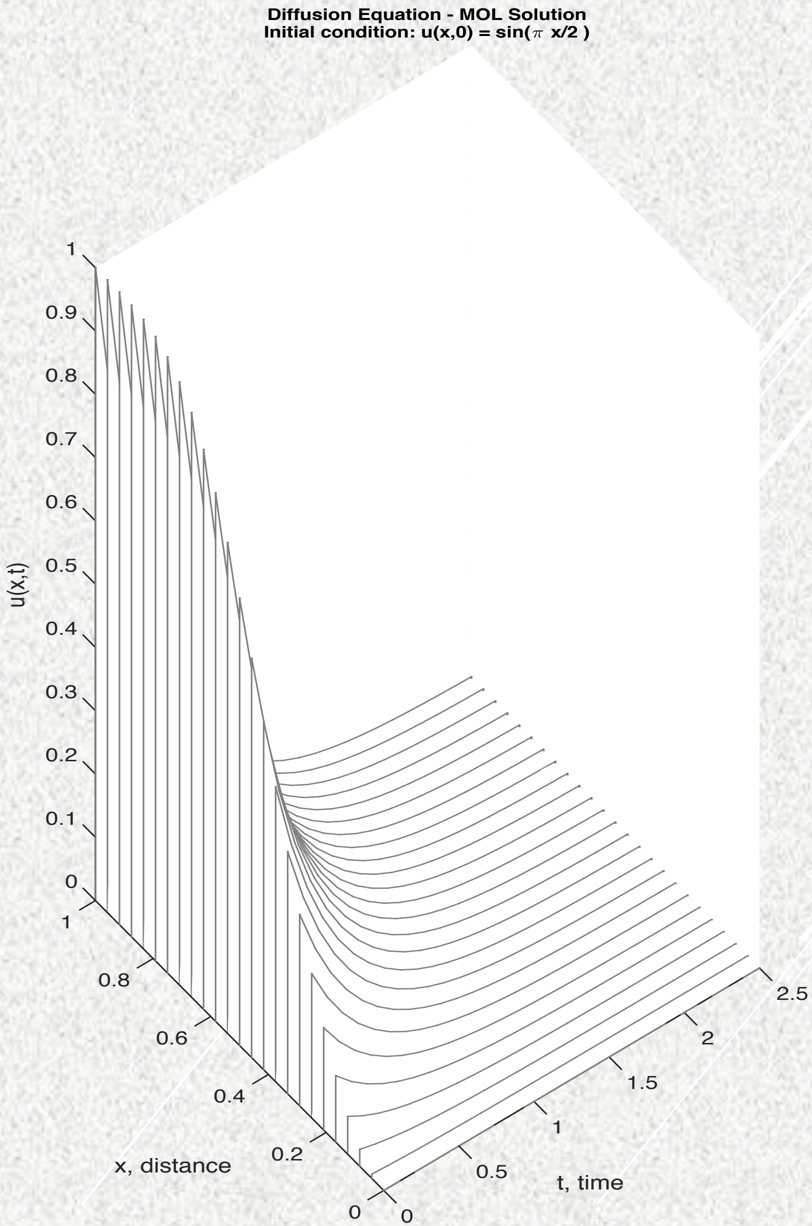
```
% Script to solve time dependent PDE
% Cd is the diffusion coefficient
function PDE_meuCd(Cd)

%% Typical Cd = 0.2 - Observ the behavior with Cd = 1.0
tic
N = 25; n = N-1;% N odd - With N = 21 ~ 100
xL = -cos((0:n)'*pi/n); xs = (xL+1)/2;
u0 = sin(pi*xs/2);
%%
D1 = Generalized_Diff_Mat(xs); D1n = D1; D1n(N,:) = 0;
D2 = D1*D1n;
uprime = @(t,u) Cd*D2*[0;u(2:end)];
%% Time vector
Nt = N;
t0=0.0;   tf=2.5; tout=linspace(t0,tf,Nt); nout = Nt;
tic
reltol=1.0e-11; abstol=1.0e-11;
  options=odeset('RelTol',reltol,'AbsTol',abstol);
  [t,u]=ode45(uprime,tout,u0,options); %Runge-Kutta
toc
```

A página seguinte descreve somente os plots.

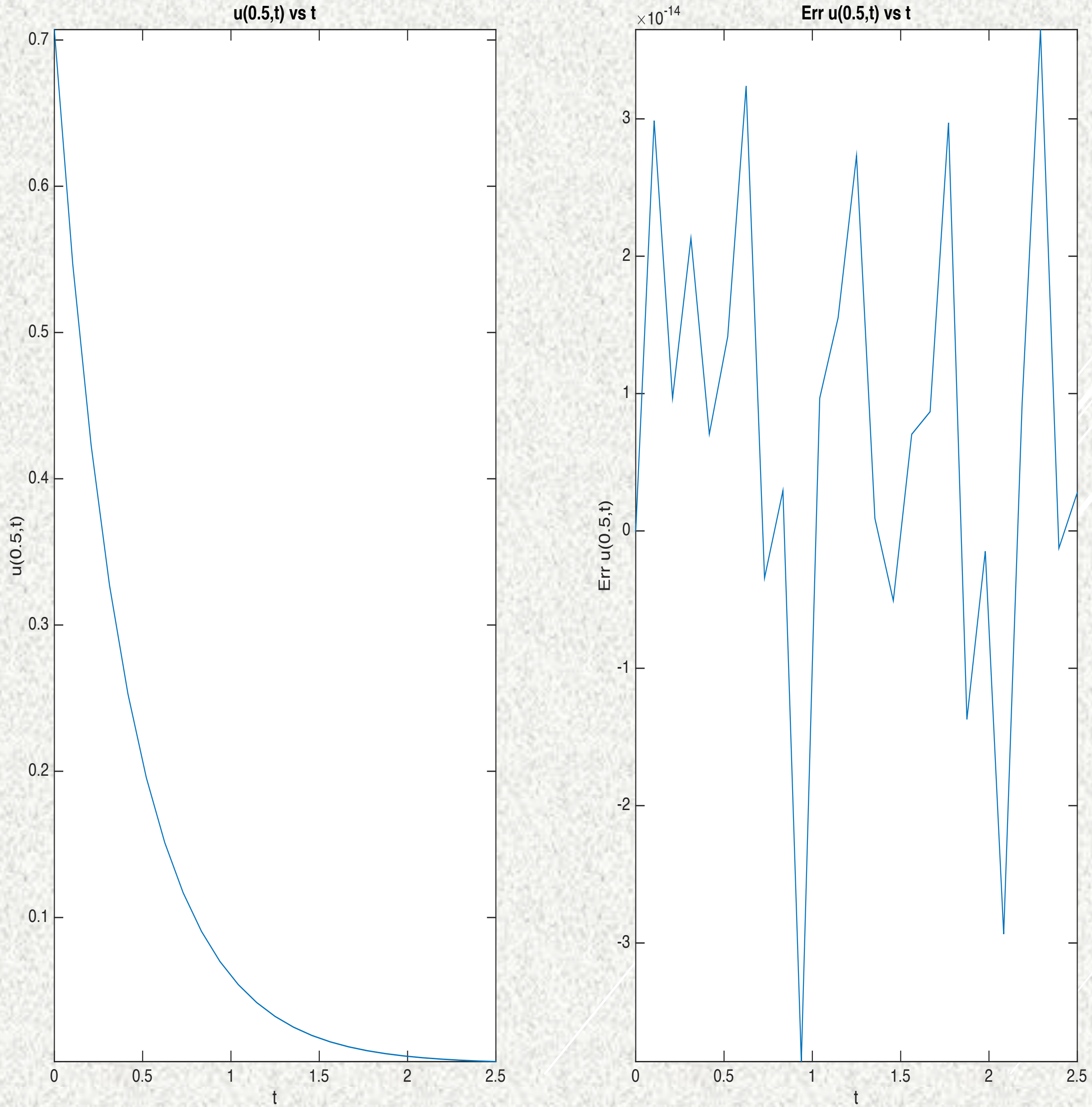
```
%%
[xx,tt] = meshgrid(xs,tout);
u_Exact = @(t,x) exp(-Cd*pi^2/4.0*t).*sin(pi*x/2);
figure (3);
surf(tt,xx,u_Exact(tt,xx) - u);
xlabel t; ylabel x
%%
n2=n/2+1;    sine=sin(pi/2.0*0.5);
for i=1:nout
  u_plot(i)=u(i,n2);
  u_anal(i)=exp(-Cd*pi^2/4.0*t(i))*sine;
  err_plot(i)=u_plot(i)-u_anal(i);
end
% Display selected output
fprintf('\n  abstol = %8.1e   reltol = %8.1e\n',...
        abstol,reltol);
fprintf('\n     t        u(0.5,t)  u_anal(0.5,t)   err u(0.5,t)\n');
for i=1:5:nout
  fprintf('%6.3f%15.6f%15.6f%15.7f\n',...
          t(i),u_plot(i),u_anal(i),err_plot(i));
end
```

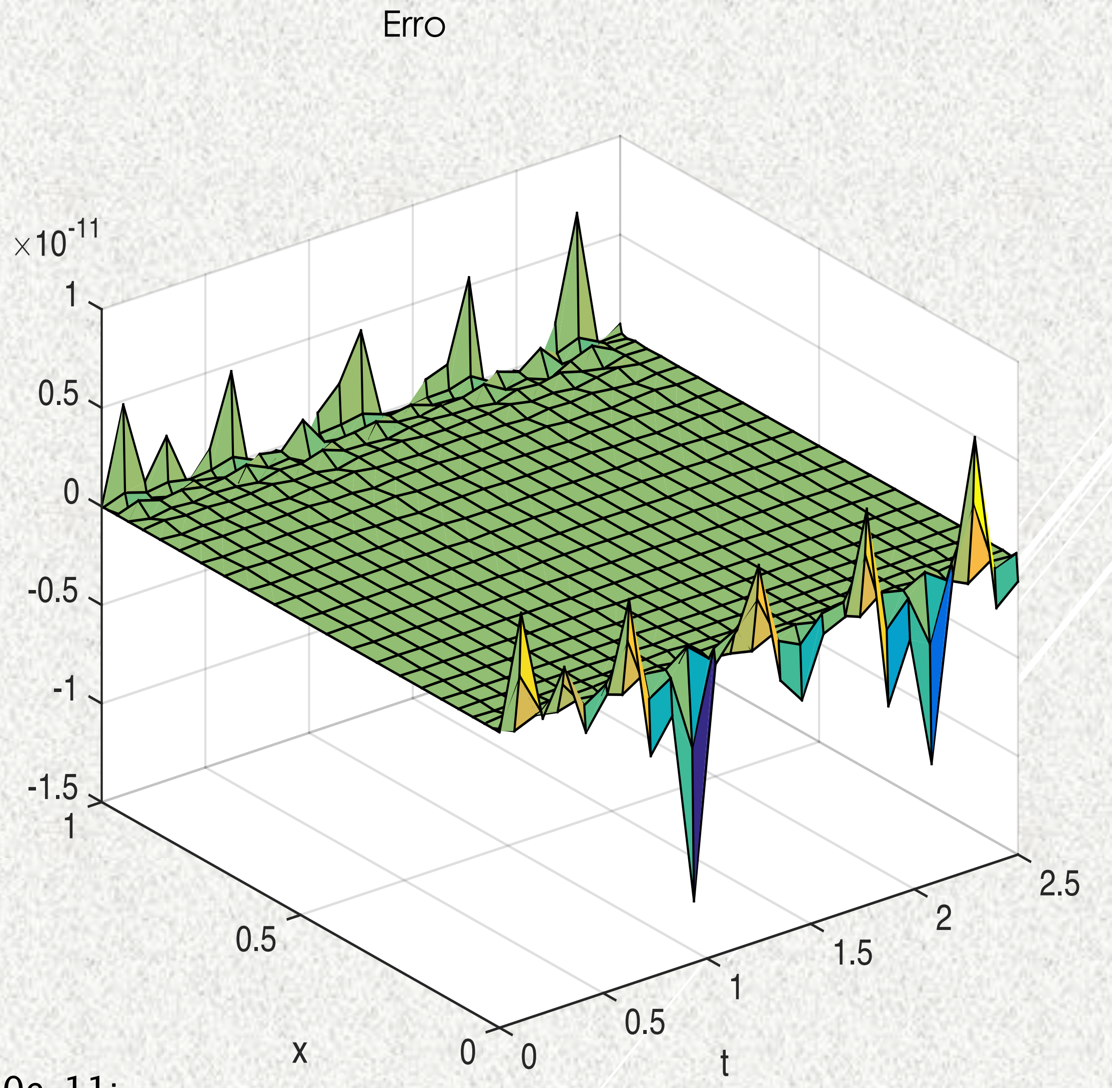
```
toc
% Plot numerical solution and errors at x = 1/2
  figure (1);
  subplot(1,2,1)
  plot(t,u_plot); axis tight
  title('u(0.5,t) vs t'); xlabel('t'); ylabel('u(0.5,t)')

  subplot(1,2,2)
  plot(t,err_plot); axis tight
  title('Err u(0.5,t) vs t'); xlabel('t'); ylabel('Err u(0.5,t)')
%% Plot numerical solution in 3D perspective
  figure(2);
  colormap('Gray');
  C=ones(N,Nt);
  g=linspace(0,1,N); % For distance x
  waterfall(t,g,u',C);
  axis('tight');
  grid off
  xlabel('t, time')
  ylabel('x, distance')
  zlabel('u(x,t)')
  s1 = sprintf('Diffusion Equation - MOL Solution');
  sTmp = sprintf('u(x,0) = sin(\\pi x/2 )');
  s2 = sprintf('Initial condition: %s', sTmp);
  title([{s1}, {s2}], 'fontsize', 12);
```





Osvaldo Guimarães


u(0.5,t) vs t
u(0.5,t)
t
Err u(0.5,t) vs t
×10^-14
Err u(0.5,t)
t



Tarefa:
código Matlab
desta solução c/
ode45 ou ode113.

reltol=1.0e-11; abstol=1.0e-11;
options=odeset('RelTol',reltol,'AbsTol',abstol);

TAREFA

Implementar em Julia e comentar as ODEs não lineares apresentadas nos slides 2 a 5 desta aula.
Gráficos e margens de erro.