TPC#3
Data de entrega: 18 de Maio de 2023
Autores: Diogo Alexandre Mousinho dos Reis#1 (100%); Pedro Coelho Fróis Silva#2 (100%); Teresa Moral Fernandez#3 (100%)

1. **Considere o problema de valor de fronteira**

$$-\epsilon u'' + u' = 1 \quad em \quad (0,1), \qquad u(0) = u(1) = 0, \tag{1}$$

onde $\epsilon > 0$.

- Verifique que a soluçao exacta é dada por

$$u(x) = x - \frac{e^{\frac{x}{\epsilon}} - 1}{e^{\frac{1}{\epsilon}} - 1}$$

Para verificar se a função $u(x)$ é a solução exacta do problema (1), temos de verificar primeiro se u(x) satisfaz:

$$-\epsilon u'' + u' = 1 \tag{2}$$

Por outro lado, as duas condições iniciais também devem ser satisfeitas: $u(1) = u(0) = 0$. Comecemos por verificar a primeira condição. Para o fazer, derivamos a função u duas vezes e substituímo-la em (2).

$$\begin{aligned} u'(x) = \frac{\mathrm{d}}{\mathrm{d}x}\left[x - \frac{e^{\frac{x}{\epsilon}} - 1}{e^{\frac{1}{\epsilon}} - 1}\right] &= \frac{\mathrm{d}}{\mathrm{d}x}[x] - \frac{1}{e^{\frac{1}{\epsilon}} - 1}\left(\frac{\mathrm{d}}{\mathrm{d}x}\left[e^{\frac{x}{\epsilon}}\right] + \frac{\mathrm{d}}{\mathrm{d}x}[-1]\right) \\ &= 1 - \frac{e^{\frac{x}{\epsilon}} \cdot \frac{1}{\epsilon} \cdot 1}{e^{\frac{1}{\epsilon}} - 1} \\ &= 1 - \frac{e^{\frac{x}{\epsilon}}}{\epsilon \cdot (e^{\frac{1}{\epsilon}} - 1)} \end{aligned} \tag{3}$$

Agora vamos calcular a segunda derivada:

$$\begin{aligned} u''(x) = \frac{\mathrm{d}}{\mathrm{d}x}\left[1 - \frac{e^{\frac{x}{\epsilon}}}{\epsilon \cdot (e^{\frac{1}{\epsilon}} - 1)}\right] &= \frac{\mathrm{d}}{\mathrm{d}x}[1] - \frac{1}{\epsilon \cdot (e^{\frac{1}{\epsilon}} - 1)} \cdot \frac{\mathrm{d}}{\mathrm{d}x}\left[e^{\frac{x}{\epsilon}}\right] \\ &= -\frac{e^{\frac{x}{\epsilon}} \cdot \frac{\mathrm{d}}{\mathrm{d}x}\left[\frac{x}{\epsilon}\right]}{\epsilon \cdot (e^{\frac{1}{\epsilon}} - 1)} = \frac{\frac{1}{\epsilon} \cdot e^{\frac{x}{\epsilon}}}{\epsilon \cdot (e^{\frac{1}{\epsilon}} - 1)} \\ &= -\frac{e^{\frac{x}{\epsilon}}}{\epsilon^2 \cdot (e^{\frac{1}{\epsilon}} - 1)} \end{aligned} \tag{4}$$

Substituir as derivadas em (2) e verificar se:

$$\begin{aligned} -\epsilon u'' + u' = 1 \quad &\iff \quad -\epsilon \cdot \left(-\frac{e^{\frac{x}{\epsilon}}}{\epsilon^2 \cdot (e^{\frac{1}{\epsilon}} - 1)}\right) + \left(1 - \frac{e^{\frac{x}{\epsilon}}}{\epsilon \cdot (e^{\frac{1}{\epsilon}} - 1)}\right) = 1 \\ &\iff \quad \frac{e^{\frac{x}{\epsilon}}}{\cancel{\epsilon \cdot} (e^{\frac{1}{\epsilon}} - 1)} + 1 - \frac{e^{\frac{x}{\epsilon}}}{\cancel{\epsilon \cdot} (e^{\frac{1}{\epsilon}} - 1)} = 1 \end{aligned}$$

É evidente que ambas as fracções se anulam e, portanto, a solução verifica a equação inicial. Vejamos agora que satisfaz as condições iniciais:

$$u(0) = 0 - \frac{e^0 - 1}{e^{\frac{1}{\epsilon}} - 1} = 0 \quad \checkmark$$

$$u(1) = 1 - \cancelto{1}{\frac{e^{\frac{1}{\epsilon}} - 1}{e^{\frac{1}{\epsilon}} - 1}} = 0 \quad \checkmark$$

Por conseguinte, demonstra-se que a solução exacta de (2) é

$$u(x) = x - \frac{e^{\frac{x}{\epsilon}} - 1}{e^{\frac{1}{\epsilon}} - 1}$$

b) Nesta alínea, é pedido que traçemos o gráfico da solução para $\epsilon = 10^{-k}, k = 1, 2, 3$ utilizando uma abordagem inteligente de modo a evitar o overflow para o caso em que $\epsilon = 10^{-3}$.

A abordagem utilizada foi multiplicar e dividir a fração $\frac{e^{\frac{x}{\epsilon}}-1}{e^{\frac{1}{\epsilon}}-1}$ por $e^{-\frac{1}{\epsilon}}$. Obtemos assim que a solução $u(x)$ a utilizar no código é:

$$u(x) = x - \frac{e^{\frac{x-1}{\epsilon}} - e^{-\frac{1}{\epsilon}}}{1 - e^{-\frac{1}{\epsilon}}}$$

O plot do gráfico da solução para os três valores de ϵ é o seguinte:

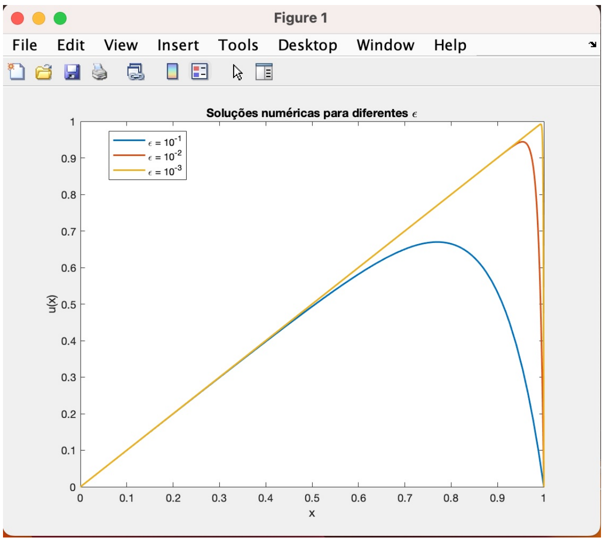

Figura 1: Gráfico com as soluções numéricas para diferentes ϵ

O código utilizado foi o seguinte:

```
for k = 1:3
    eps = 10^(-k);

    u = @(x) x - (exp((x-1)/eps) - exp(-1/eps)) / (1 - exp(-1/eps));

    fplot(u, [0, 1], 'LineWidth', 1.5)
    hold on
end

ylim([0, 1])
xlim([0, 1])
legend('\epsilon = 10^{-1}', '\epsilon = 10^{-2}', '\epsilon = 10^{-3}');
title('Soluções numéricas para diferentes \epsilon')
xlabel('x')
ylabel('u(x)')
```

Figura 2: Código utilizado na alínea b)

c)

Na presente alínea, pretendemos calcular e traçar o gráfico da solução do problema de valor de fronteira através da implementação de um método das diferenças finitas numa rede uniforme de espaçamento $h = \frac{1}{N+1}$.

O método de diferenças finitas discretiza o domínio em pontos internos equidistantes, representados por x = (0:h:1)'. Implementamos um ciclo for que calcula e traça o gráfico da solução exata e da solução do método para cada valor de ϵ. A solução aproximada $u(x)$ é calculada usando a seguinte abordagem:

1. Construção da matriz esparsa A:

   Tivemos de proceder à discretização das derivadas. Usando o desenvolvimento de taylor. Temos que

$$\begin{cases} -u''(x_k) = \frac{1-u'(x_k)}{\epsilon}, \quad k = 1, ..., n \\ u(0) = u(1) = 0 \end{cases}$$

$$\Leftrightarrow \quad \begin{cases} \frac{\epsilon}{h^2}(-u(x_{k+1}) + 2u(x_k) - u(x_{k-1}) + \frac{1}{h}(u(x_{k+1}) - u(x_k)), \\ u(0) = u(1) = 0 \end{cases}$$

$$\Leftrightarrow \quad \begin{cases} \frac{1}{h^2}(u(x_{k+1})(h-\epsilon) + u(x_k)(2\epsilon - h) - u(x_{k-1})(-\epsilon)), \\ u(0) = u(1) = 0 \end{cases}$$

   Os elementos da diagonal principal de A são definidos como: $\frac{2\epsilon}{h^2} - \frac{1}{h}$.

   Os elementos da diagonal inferior são definidos como: $-\frac{\epsilon}{h^2}$.

   Os elementos da diagonal superior são definidos como $-\frac{\epsilon}{h^2} + \frac{1}{h}$.

   A matriz A é construída usando a função spdiags para aproveitar sua estrutura esparsa.

2. Cálculo do vetor de termos independentes b:

   O vetor b é inicializado com a função $f(x) = 1$ para todos os pontos internos.

3. Resolução do sistema de equações lineares $A * uh = b$:

   O código usa o operador de divisão da matriz, específico do MATLAB, para resolver o sistema de forma eficiente.

   O resultado é armazenado na variável uh, que representa a solução aproximada para o valor atual de ϵ.

4. Cálculo da solução exata:

   A solução exata é dada pela expressão $u(x) = x - \frac{e^{\frac{x-1}{\epsilon}} - e^{-\frac{1}{\epsilon}}}{1-e^{-\frac{1}{\epsilon}}}$, analogamente à alínea anterior, e avalia a solução nos pontos da rede para obter os valores exatos correspondentes.

5. Traçado dos gráficos:

   Finalmente, o código traça os gráficos individuais da solução do método de diferenças finitas (representado pela linha azul) e da solução exata(representada pela linha vermelha a tracejado) para cada valor de ϵ. Dentro do loop, para cada valor de k, um gráfico é criado usando a função plot do MATLAB.

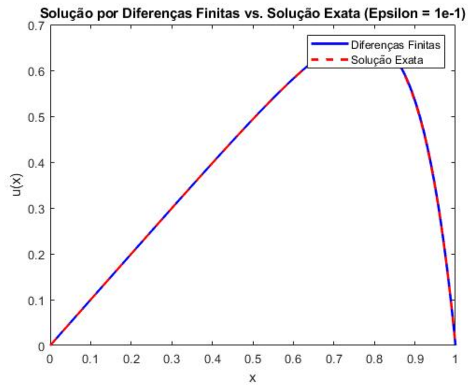

Figura 3: Gráfico com as soluções do método de diferenças finitas vs solução exata para $\epsilon = 0.1$

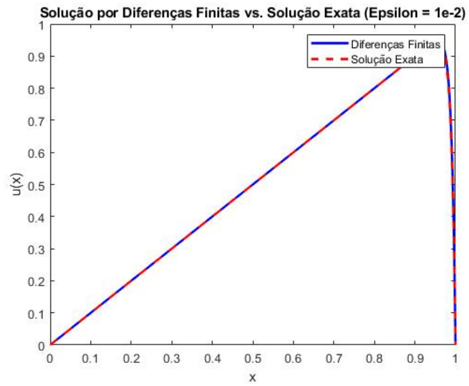

Figura 4: Gráfico com as soluções do método de diferenças finitas vs solução exata para $\epsilon = 0.01$

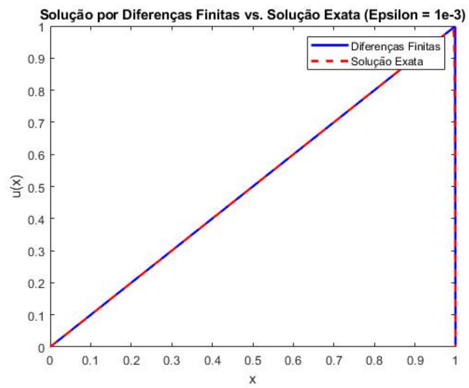

Figura 5: Gráfico com as soluções do método de diferenças finitas vs solução exata para $\epsilon = 0.001$

Por fim, apresentamos o código utilizado

```
%Dados e partição
N =1000;
h = 1/(N+1);
x = (0:h:1)';
f = ones(N, 1);
e = ones(N,1);

num_eps = 3; % Número de valores de epsilon
uh_values = cell(num_eps, 1); % Célula para armazenar as soluções
u_exata_values = cell(num_eps, 1); % Célula para armazenar as soluções exatas

for k = 1:num_eps
    eps = 10^(-k);

    % Criar a matriz de rigidez esparsa
    diagonal = (2*eps/(h^2)) - (1/h);
    diagonalinf = -eps/(h^2);
    diagonalsup = -eps/(h^2) + (1/h);

    A = spdiags([diagonalinf*e, diagonal*e, diagonalsup*e], -1:1, N, N);

    % Vetor de carga b
    b = f;

    % Resolução do sistema de equações lineares A*uh = b
    uh = A\b;
    uh = [0; uh; 0];
    uh_values{k} = uh; % Armazenar a solução para o valor atual de epsilon

    % Cálculo da solução exata
    u_exata = @(x) x - (exp((x-1)/eps) - exp(-1/eps)) / (1 - exp(-1/eps));
    u_exata_vals = u_exata(x); % Avaliar a função nos pontos da rede
    u_exata_values{k} = u_exata_vals; % Armazenar a solução exata para o valor atual de epsilon

end

% Plot dos gráficos individuais
colors = {'b', 'b', 'b'};
for k = 1:num_eps
    figure;
plot(x, uh_values{k}, colors{k}, 'LineWidth', 2);
hold on
plot(x, u_exata_values{k}, 'r--', 'LineWidth', 2);
hold off
legend('Diferenças Finitas', 'Solução Exata');
xlabel('x');
ylabel('u(x)');
title(sprintf('Solução por Diferenças Finitas vs. Solução Exata (Epsilon = 1e-%d)', k));
end
```

Figura 6: Código utilizado na alínea c)

d) Nesta alínea investigamos a convergência do método numérico quando h decresce, ou seja, quando N aumenta, utilizando a norma máxima e a norma

$$||u||_p = (h\sum_{j=1}^{N}|u(x_j)|^p)^{\frac{1}{p}}$$

quando $p = 1$ e $p = 2$.

Calculamos os erros para seis valores de N diferentes: $100, 500, 1000, 1000, 10000$ e 1000000. Os resultados podem ser observados nas seguintes tabelas:

N	$\|\|u\|\|_\infty$	$\|\|u\|\|_1$	$\|\|u\|\|_2$
100	0.018995	0.005013	0.008057
500	0.003700	0.000997	0.001585
1000	0.001884	0.000498	0.000791
10000	0.000184	0.000050	0.000079
100000	0.000018	0.00005	0.000008
1000000	0.000002	0.000000	0.000001

Tabela 1: Erros para $\epsilon = 0.1$

N	$\|\|u\|\|_\infty$	$\|\|u\|\|_1$	$\|\|u\|\|_2$
100	0.361639	0.005794	0.038904
500	0.040116	0.001031	0.005314
1000	0.019181	0.000508	0.002574
10000	0.001847	0.000050	0.000251
100000	0.000184	0.000005	0.000025
1000000	0.000018	0.000001	0.000002

Tabela 2: Erros para $\epsilon = 0.01$

N	$\|\|u\|\|_\infty$	$\|\|u\|\|_1$	$\|\|u\|\|_2$
100	1.112347	0.991099	0.996105
500	0.985672	0.379697	0.451357
1000	0.367248	0.000581	0.012490
10000	0.019199	0.000051	0.000815
100000	0.001847	0.000005	0.000079
1000000	0.000184	0.000001	0.000008

Tabela 3: Erros para $\epsilon = 0.001$

À medida que N aumenta, como este $\to \infty$ então $h = \frac{1}{1+N} \to 0$, o que é possível de confirmar com os resultados obtidos nos gráficos: conseguimos observar que há medida que aumentamos os valores de N os valores dos erros diminuiem, aproximando-se de 0.

Adicionamos o seguinte código dentro do ciclo for utilizado na alínea c):

```
% Cálculo do erro
erro = norm(uh-u_exata_vals, inf);
erro1 = h*norm(uh-u_exata_vals, 1);
erro2 = (h^(1/2))*norm(uh-u_exata_vals, 2);

fprintf('Erro para epsilon = 1e-%d:\n', k);
fprintf('Erro infinito: %.6f\n', erro);
fprintf('Erro 1: %.6f\n', erro1);
fprintf('Erro 2: %.6f\n', erro2);
fprintf('\n');
```

Figura 7: Código utilizado na alínea d)