

Lista 5 (Análise Numérica)

Raphael F. Levy

November 18, 2022

Questão 2:

Uma área importante onde equações parabólicas são usadas é no estudo da evolução espaço-temporal de populações biológicas. Populações tendem a comportar-se como o calor, no sentido de que elas se espalham ou propagam desde áreas com altas densidades até áreas com densidades mais baixas. Além de, obviamente, crescer e morrer. Para modelar a densidade $u(x, t)$ da população no tempo t ($0 \leq t \leq T$) e na posição x ($0 \leq x \leq L$), considere o seguinte modelo dado por uma EDP de reação-difusão:

$$\begin{cases} u_t = cu_{xx} + du, & c, d \in \mathbb{R} \\ u(x, 0) = \sin^2\left(\frac{\pi}{2}x\right) & (0 \leq x \leq L) \\ u(0, t) = 0, & t > 0 \\ u(L, t) = 0, & t > 0 \end{cases}$$

O termo difusivo cu_{xx} causa que a população se espalhe ao longo da direção x . O termo du (reação) contribui com o crescimento da população na razão d . As condições de fronteira representam o fato de que a população vive no espaço $0 \leq x \leq L$. Se a população sobrevive ou segue em direção à extinção vai depender dos valores de c, d e L .

(a)

Prove que o método BTCS, aplicado a essa EDP, é convergente.

Solução:

Primeiramente, vamos definir $U_j^i \approx u(x_j, t_i) \rightarrow x_j = j\Delta x, t_i = i\Delta t$ e $v = c \frac{\Delta t}{\Delta^2 x}$. Agora, por definição de u_t e u_{xx} :

$$\frac{U_j^i - U_j^{i-1}}{\Delta t} = c \frac{U_{j+1}^i - 2U_j^i + U_{j-1}^i}{\Delta^2 x} + dU_j^i$$

$$\begin{aligned}
&\Rightarrow U_j^{i-1} = U_j^i - \Delta t \left(c \frac{(U_{j+1}^i - 2U_j^i + U_{j-1}^i)}{\Delta^2 x} + dU_j^i \right) \\
&= U_j^i(1 - d\Delta t) - v(U_{j+1}^i - 2U_j^i + U_{j-1}^i) = U_j^i(1 - d\Delta t + 2v) - vU_{j+1}^i - vU_{j-1}^i
\end{aligned}$$

Definindo $U^i := (U_1^i, \dots, U_{N_x-1}^i)^T$, chegamos a

$$\begin{bmatrix} 2v+1-d\Delta t & -v & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ -v & 2v+1-d\Delta t & -v & 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & -v & 2v+1-d\Delta t & -v & 0 & \dots & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & 0 & \dots & 0 & -v & 2v+1-d\Delta t \end{bmatrix} U^i = U^{i-1} + \begin{bmatrix} v.a(i\Delta t) \\ 0 \\ \dots \\ 0 \\ v.b(i\Delta t) \end{bmatrix}$$

Chamando a matriz acima de M e zerando o vetor, dado que $a(t) = u(0, t) = 0$ e $b(t) = u(L, t) = 0$, temos $U^i = M^{-1}U^{i-1}$, e o método BTCS é estável se $|\lambda(M^{-1})| < 1$, e é convergente se for diagonalmente dominante.

Para ser diagonalmente dominante, devemos ter

$$|2v+1-d\Delta t| > |-v| + |-v| = 2v$$

Considerando $v > 0$, devemos ter $2v+1-d\Delta t > 2v$ ou $2v+1-d\Delta t < -2v$. Assim, $1-d\Delta t > 0 \Rightarrow \Delta t < \frac{1}{d}$ ou $\Delta t > \frac{4v+1}{d}$.

Agora, para achar os autovalores de M :

$$M = \begin{bmatrix} \alpha & \beta & 0 & \dots & \dots \\ \beta & \alpha & \beta & 0 & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \dots & 0 & \beta & \alpha & \beta \\ \dots & \dots & 0 & \beta & \alpha \end{bmatrix} = -\beta \begin{bmatrix} 1 & -1 & 0 & \dots & \dots \\ -1 & 1 & -1 & 0 & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \dots & 0 & -1 & 1 & -1 \\ \dots & \dots & 0 & -1 & 1 \end{bmatrix}_{(=T)} + (\alpha + \beta)I$$

Assim:

$$\begin{aligned}
\lambda(M) &= -\beta\lambda(T) + (\alpha + \beta) = \\
&= -(-v)(1-2\cos(\frac{k\pi}{d+1})) + (2v+1-d\Delta t-v) = v(1-2\cos(\frac{k\pi}{d+1})) + v+1-d\Delta t = \\
&= v - 2v\cos(\frac{k\pi}{d+1}) + v+1-d\Delta t = 2v - 2v\cos(\frac{k\pi}{d+1}) + 1 - d\Delta t = \\
&= v(2 - 2\cos(\frac{k\pi}{d+1})) + 1 - d\Delta t = 2v(1 - \cos(\frac{k\pi}{d+1})) + 1 - d\Delta t,
\end{aligned}$$

que é estável se for maior que 1.

(b)

Utilize o método BTCS para elaborar um programa computacional que tenha como entrada: $c, d, L, T, \Delta t, \Delta x$ e que a saída seja o gráfico da solução $u(x, t)$ para $(x, t) \in [0, L] \times [0, T]$.

Solução:

O método implementado e os gráficos gerados estão no arquivo *Lista5_AN_Raphael_Levy.ipynb*.

(c)

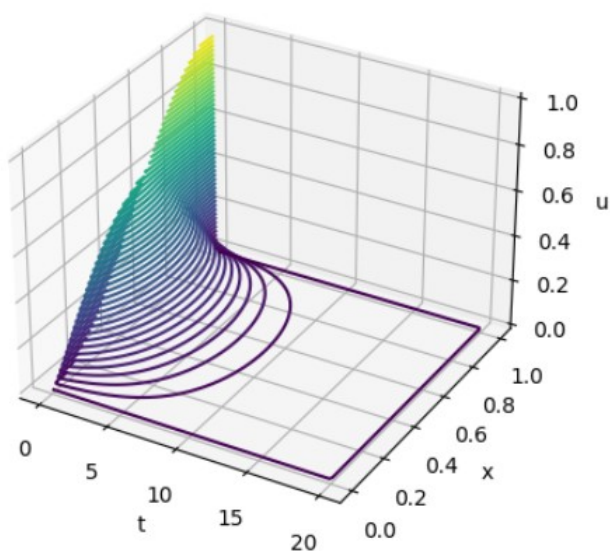
Pode-se provar que para a população sobreviver tem que ser $d > \pi^2 \frac{c}{L^2}$. Comprove computacionalmente esse resultado teórico. Para isto considere $L = 1, c = 1$, e confirme computacionalmente que para $d = 9.5$ a população tende à extinção com o passar do tempo, e que para $d = 10$ a população aumenta no transcorrer do tempo.

Solução:

Observando os gráficos, é fácil de ver que a população irá se extinguir no caso $d = 9.5$ e irá aumentar caso $d = 10$, dado que $\pi^2 \approx 9.87$.

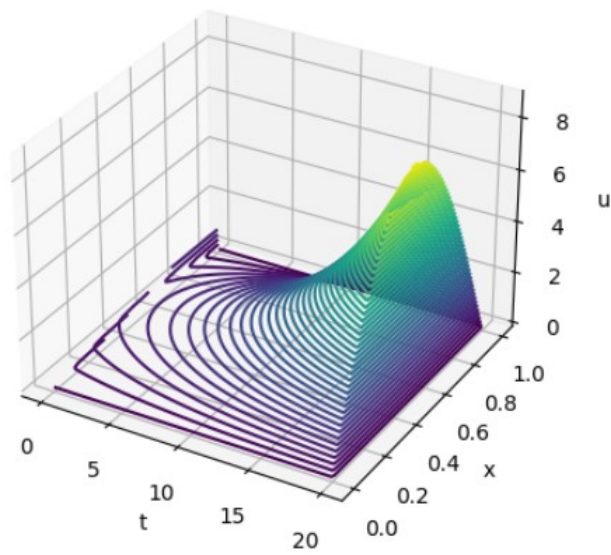
```
BTCS(c=1, d=9.5, L=1, T=20, delta_t=0.1, delta_x=0.01)
```

```
plt.show()
```



```
BTCS(c=1, d=10, L=1, T=20, delta_t=0.1, delta_x=0.01)
```

```
plt.show()
```



(d)

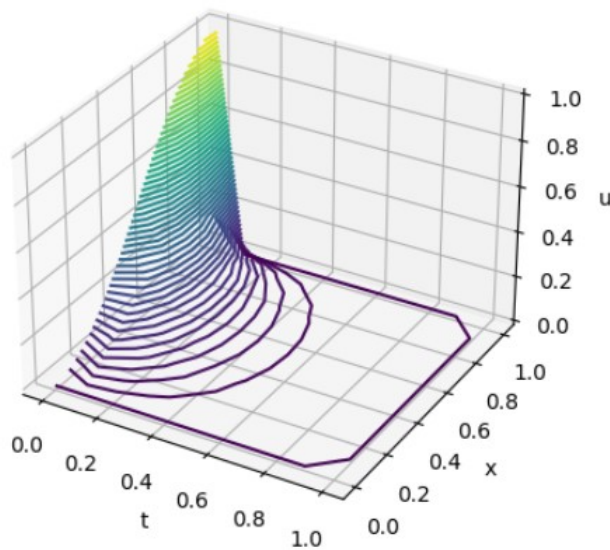
Os resultados computacionais dependem dos valores $\Delta t; \Delta x$ utilizados? Justifique.

Solução:

Dado que o método é convergente, ele não deve depender dos valores passados para Δt e Δx . Observando os gráficos, é possível ver que aumentar a precisão desses valores apenas deixa o gráfico mais definido.

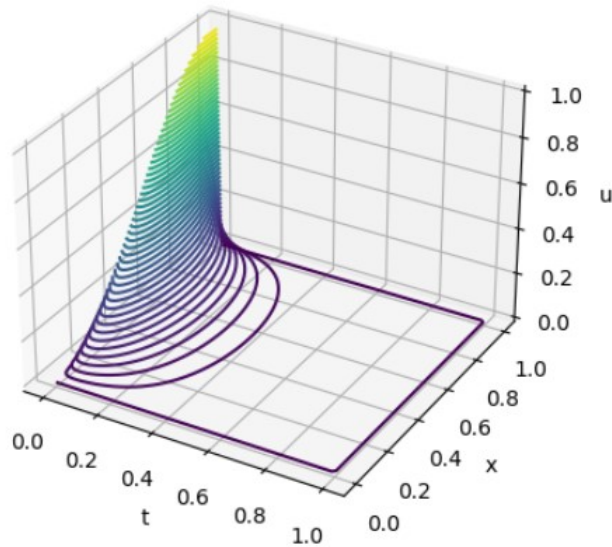
```
BTCS(c=1, d=1, L=1, T=1, delta_t=0.1, delta_x=0.1)
```

```
plt.show()
```



```
BTCS(c=1, d=1, L=1, T=1, delta_t=0.01, delta_x=0.01)
```

```
plt.show()
```



(e)

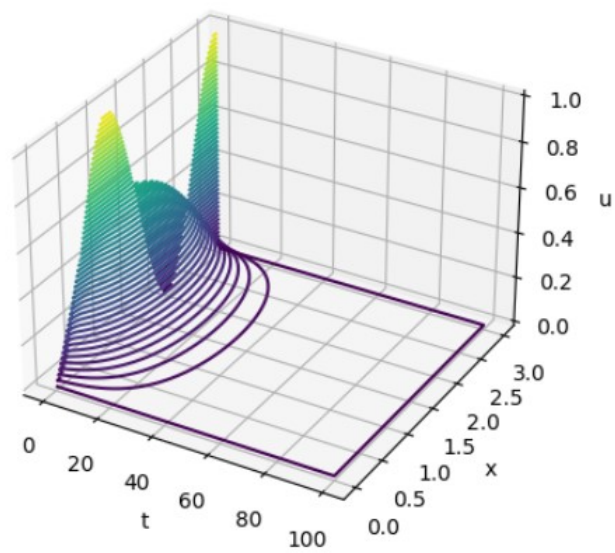
Ecologistas que estudam sobrevivência de espécies frequentemente estão interessados em conhecer o menor valor de L tal que a população não fique extinta. Suponha que sejam conhecidos $c = d = 1$. Determine, usando simulações no computador, esse valor mínimo de L . Compare com o resultado teórico do item c.

Solução:

Utilizando o que sabemos por c), que $d > \pi^2 \frac{c}{L^2}$, devemos ter $dL^2 > \pi^2 c \Rightarrow L^2 > \pi^2 \frac{c}{d}$, logo nesse caso $L^2 > \pi^2 \Rightarrow L > \pi$. Rodando o código usando $L = 1$, $L = 3$, $L = 3.14$, $L = 3.15$ e $L = 4$, é possível ver que se $L < \pi$, a população irá se extinguir, o que não acontece para $L > \pi$.

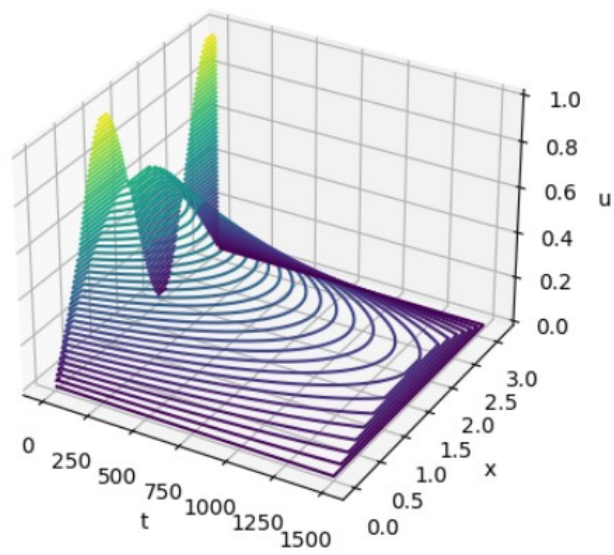
```
BTCS(c=1, d=1, L=3, T=100, delta_t=0.1, delta_x=0.01)
```

```
plt.show()
```



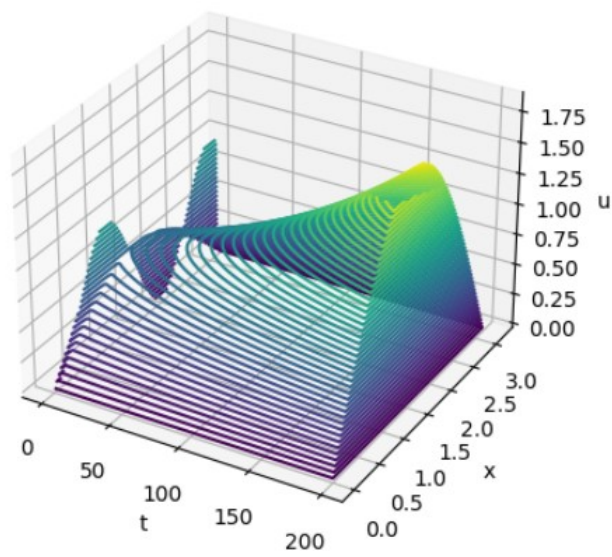
```
BTCS(c=1, d=1, L=3.14, T=1500, delta_t=0.1, delta_x=0.01)
```

```
plt.show()
```




```
BTCS(c=1, d=1, L=3.15, T=200, delta_t=0.1, delta_x=0.01)
```

```
plt.show()
```



```
BTCS(c=1, d=1, L=4, T=20, delta_t=0.1, delta_x=0.01)
```

```
plt.show()
```

