

DIFERENTES METODOLOGIAS PARA RESOLVER O SISTEMA LINEAR GERADO PELA APLICAÇÃO DO ESQUEMA TRAPEZOIDAL IMPLÍCITO NA EQUAÇÃO DA ADVECÇÃO UNIDIMENSIONAL

Viliam Cardoso da Silveira ¹
Cléa Furtado da Silveira ²

RESUMO

O objetivo desse trabalho é implementar na linguagem de programação Python diferentes metodologias para resolver sistemas lineares, entre elas, método da matriz tridiagonal, decomposição LU, método iterativo de Jacobi, Gauss Seidel e sobre-relaxação sucessiva. Assim, será possível resolver computacionalmente o sistema linear gerado pela aplicação do esquema trapezoidal implícito na equação da advecção unidimensional. Python é uma linguagem de programação de alto nível e por isso foi escolhida nesse trabalho. O desenvolvimento de programas computacionais se torna cada vez mais essencial devido ao avanço da tecnologia. Com base nesse trabalho, os alunos podem ver na prática como resolver um sistema linear, conteúdo que é visto já no ensino fundamental. Para resolver a equação da advecção, foram considerados 400 pontos em x , intervalo de tempo de 0,00025 segundos, velocidades de 8 m/s, 10 m/s e 12 m/s. Inicialmente foi considerado um distúrbio que se propaga ao longo do tempo. Conforme a velocidade é maior ou menor, o distúrbio se desloca mais ou menos rapidamente. São obtidos resultados similares, independente da metodologia utilizada, mostrando para os alunos as diferentes possibilidades que eles podem usar para resolver problemas envolvendo a resolução de sistemas lineares. Muitas vezes, os alunos não têm a possibilidade de ver na prática a aplicação de conceitos vistos em sala de aula, essencial para o processo de aprendizagem.

Palavras-chave: Sistemas lineares, Ensino de matemática, Programação Python.

INTRODUÇÃO

A resolução de equações diferenciais parciais é comum em situações que envolvem o dia a dia de muitas pessoas, sendo que esses tipos de equações estão presente em aplicações da simulação da dispersão de poluentes (Buske et al., 2007), na previsão numérica do tempo (Holton, 2004), entre outras.

Como os alunos se deparam muitas vezes com a resolução de sistemas lineares nos diversos conteúdos que são abordados ao longo do ensino médio e posteriormente no ensino superior, o presente trabalho tem como objetivo mostrar para os alunos que a resolução desse tipo de problema que pode ser resolvido com o auxílio de um computador e de uma linguagem de programação.

¹ Pós-doutorando em Modelagem Matemática na Universidade Federal de Pelotas - UFPel, viliamcardoso2@gmail.com;

² Doutoranda profissional em Educação e Tecnologia no Instituto Federal Sul Rio-grandense - IFSul, cleafurtado@gmail.com;

Foram testadas diferentes metodologias para resolver o sistema linear que é obtido quando a equação da advecção é resolvida pelo esquema trapezoidal implícito. Tal sistema foi resolvido pelo método da matriz tridiagonal (Dumka e Mishra, 2022), decomposição LU (Golub e Loan, 1996), método iterativo de Jacobi, Gauss Seidel e sobre-relaxação sucessiva (Campos Filho, 2007). Ambos os métodos foram implementados na linguagem de programação python (Lutz, 2009), linguagem muito usada no meio acadêmico.

Esse trabalho mostra que os alunos podem usar diferentes métodos para resolver um sistema linear e que os resultados obtidos podem ser similares. Basta dedicação e comprometimento para aplicar os conhecimentos adquiridos em sala de aula, nas mais diversas aplicações possíveis.

METODOLOGIA

A equação da advecção unidimensional é dada por

$$\frac{\partial q}{\partial t} + c \frac{\partial q}{\partial x} = 0$$

sendo c a velocidade, q a amplitude do distúrbio e x variando de $0 \leq x \leq 2 \text{ km}$ com 401 pontos de grade. Considera-se inicialmente que se $abs\left(x - \frac{1}{4}\right) < \frac{1}{8}$, então

$$q = 0,25 \left\{ \cos\left(8\pi \left[x - \frac{1}{4}\right]\right) + 1 \right\}^2$$

ou $q = 0$, caso contrário (Holton, 2004).

O esquema trapezoidal implícito (Jacobson, 2005) é obtido considerando diferenças finitas avançadas no tempo

$$\frac{\partial q}{\partial t} = \frac{q_{t+\Delta t} - q_t}{\Delta t} + o(\Delta t)$$

e diferenças centradas no espaço

$$\frac{\partial q}{\partial x} = \frac{q_{x+\Delta x} - q_{x-\Delta x}}{2\Delta x} + o(\Delta x)^2$$

Assim, a equação da advecção unidimensional pode ser escrita como:

$$\frac{q_{x,t+\Delta t} - q_{x,t}}{\Delta t} = -\frac{c}{2} \left[\frac{q_{x+\Delta x,t+\Delta t} - q_{x-\Delta x,t+\Delta t}}{2\Delta x} + \frac{q_{x+\Delta x,t} - q_{x-\Delta x,t}}{2\Delta x} \right]$$

ou de forma similar

$$\frac{q_{x,t+1} - q_{x,t}}{\Delta t} = -\frac{c}{2} \left[\frac{q_{x+1,t+1} - q_{x-1,t+1}}{2\Delta x} + \frac{q_{x+1,t} - q_{x-1,t}}{2\Delta x} \right]$$

$$q_{x,t+1} - q_{x,t} = -\frac{c\Delta t}{4\Delta x} \left[q_{x+1,t+1} - q_{x-1,t+1} + q_{x+1,t} - q_{x-1,t} \right]$$

Considerando

$$\sigma = \frac{c\Delta t}{4\Delta x}$$

então, pode-se escrever

$$q_{x,t+1} - q_{x,t} = -\frac{\sigma}{4}q_{x+1,t+1} + \frac{\sigma}{4}q_{x-1,t+1} - \frac{\sigma}{4}q_{x+1,t} + \frac{\sigma}{4}q_{x-1,t} \\ - \frac{\sigma}{4}q_{x-1,t+1} + q_{x,t+1} + -\frac{\sigma}{4}q_{x+1,t+1} = \frac{\sigma}{4}q_{x-1,t} + q_{x,t} - \frac{\sigma}{4}q_{x+1,t}$$

Aplicando a equação acima em todos os pontos de grade, tem-se o seguinte sistema de equações:

$$\left\{ \begin{array}{l} q_{1,t+1} + \frac{\sigma}{4}q_{2,t+1} = q_{1,t} - \frac{\sigma}{4}q_{2,t} \\ -\frac{\sigma}{4}q_{1,t+1} + q_{2,t+1} + \frac{\sigma}{4}q_{3,t+1} = \frac{\sigma}{4}q_{1,t} + q_{2,t} - \frac{\sigma}{4}q_{3,t} \\ -\frac{\sigma}{4}q_{2,t+1} + q_{3,t+1} + \frac{\sigma}{4}q_{4,t+1} = \frac{\sigma}{4}q_{2,t} + q_{3,t} - \frac{\sigma}{4}q_{4,t} \\ -\frac{\sigma}{4}q_{3,t+1} + q_{4,t+1} + \frac{\sigma}{4}q_{5,t+1} = \frac{\sigma}{4}q_{3,t} + q_{4,t} - \frac{\sigma}{4}q_{5,t} \\ \vdots \\ -\frac{\sigma}{4}q_{n_x-2,t+1} + q_{n_x-1,t+1} + \frac{\sigma}{4}q_{n_x,t+1} = \frac{\sigma}{4}q_{n_x-2,t} + q_{n_x-1,t} - \frac{\sigma}{4}q_{n_x,t} \\ q_{n_x-1,2} + \frac{\sigma}{4}q_{n_x,2} = q_{n_x-1,1} - \frac{\sigma}{4}q_{n_x,1} \end{array} \right.$$

Esse sistema pode ser escrito na forma matricial da seguinte forma:

$$\begin{bmatrix} 1 & \frac{\sigma}{4} & 0 & 0 & \dots & 0 \\ -\frac{\sigma}{4} & 1 & \frac{\sigma}{4} & 0 & \dots & 0 \\ 0 & -\frac{\sigma}{4} & 1 & \frac{\sigma}{4} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \dots & \vdots \\ 0 & 0 & 0 & -\frac{\sigma}{4} & 1 & \frac{\sigma}{4} \\ 0 & 0 & 0 & 0 & -\frac{\sigma}{4} & 1 \end{bmatrix} \begin{bmatrix} q_{1,t+1} \\ q_{2,t+1} \\ q_{3,t+1} \\ \vdots \\ q_{n_x-1,t+1} \\ q_{n_x,t+1} \end{bmatrix} = \begin{bmatrix} 1 & -\frac{\sigma}{4} & 0 & 0 & \dots & 0 \\ \frac{\sigma}{4} & 1 & -\frac{\sigma}{4} & 0 & \dots & 0 \\ 0 & \frac{\sigma}{4} & 1 & -\frac{\sigma}{4} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \dots & \vdots \\ 0 & 0 & 0 & \frac{\sigma}{4} & 1 & -\frac{\sigma}{4} \\ 0 & 0 & 0 & 0 & \frac{\sigma}{4} & 1 \end{bmatrix} \begin{bmatrix} q_{1,t} \\ q_{2,t} \\ q_{3,t} \\ \vdots \\ q_{n_x-1,t} \\ q_{n_x,t} \end{bmatrix}$$

O sistema linear é resolvido a cada iteração no tempo e foi implementado na linguagem de programação Python, sendo que foram utilizadas diferentes metodologias para resolver o mesmo, entre elas, método da matriz tridiagonal, decomposição LU, método iterativo de Jacobi, Gauss Seidel e sobre-relaxação sucessiva. Foi considerado nas simulações $\Delta x = 0,005$ e $\Delta t = 0,00025$.

RESULTADOS E DISCUSSÃO

Na Figura 1 é mostrada a solução do problema proposto utilizando diferentes metodologias para resolver o sistema linear gerado pela aplicação do esquema trapezoidal implícito na equação da advecção unidimensional, onde é variado o parâmetro velocidade e fixado o tempo, para ver o que acontece com o distúrbio ao longo do domínio considerado.

Como observa-se dessa figura, conforme a velocidade é maior ou menor, o distúrbio se desloca mais ou menos rapidamente.

Na Figura 2 é mostrada a solução do problema proposto variando o parâmetro tempo e mantendo fixa a velocidade. Como observa-se desta figura, conforme o tempo passa, o distúrbio vai se deslocando para a direita.

Observa-se na Figura 3 a variação do distúrbio ao longo do tempo para um x fixado. Como pode-se observar nesta figura, quanto menor a velocidade, mais tempo o distúrbio leva para alcançar uma determinada distância. Para uma velocidade de 8 m/s , o pico do distúrbio leva $0,095 \text{ s}$ para atingir a distância de 1 km , enquanto que para uma velocidade de 12 m/s , o pico do distúrbio leva $0,0635 \text{ s}$ para atingir a distância de 1 km .

Como pode-se observar, todos os métodos utilizados para resolver o sistema linear apresentam praticamente a mesma solução do problema proposto. Isso mostra para os alunos as diferentes possibilidades que eles podem usar para resolver problemas envolvendo situações relacionadas com a resolução de sistemas lineares. Muitas vezes, os alunos não têm a possibilidade de ver na prática a aplicação de conceitos vistos em sala de aula, essencial para o processo de aprendizagem.

CONSIDERAÇÕES FINAIS

A solução do problema proposto foi resolvida de forma satisfatória utilizando diferentes metodologias para resolver o sistema linear gerado pela aplicação do esquema trapezoidal implícito na equação da advecção unidimensional. Esse trabalho mostra para os alunos uma aplicação dos conceitos envolvendo a resolução de sistemas lineares. Muitas vezes os alunos se sentem desmotivados, por não verem uma aplicação prática dos conteúdos abordados em sala de aula.

Os alunos podem abrir o código escrito em python e ir testando cada parâmetro. Podem usar outra malha na direção x , aumentando e diminuindo a quantidade de pontos, podem testar outros valores para a velocidade, aumentar ou diminuir o tempo de simulação. Esse tipo de exercício pode despertar nos alunos o interesse em investigar outros tipos de problemas, o que vai contribuir muito para a formação de cada um.

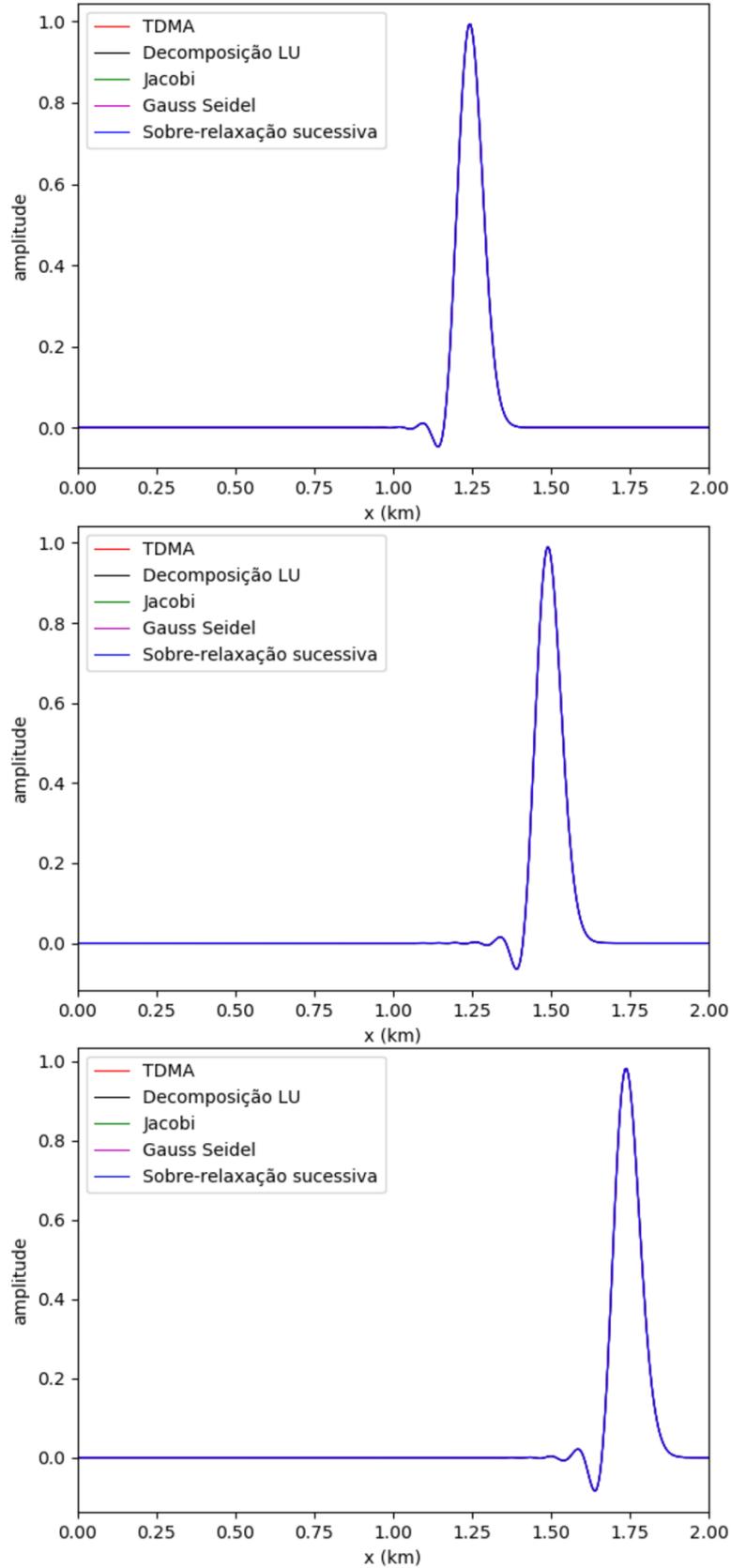


Figura 1: Solução do problema proposto usando diferentes metodologias para resolver o sistema linear, $t = 0,125$ s e velocidades de 8 m/s, 10 m/s e 12 m/s, respectivamente.

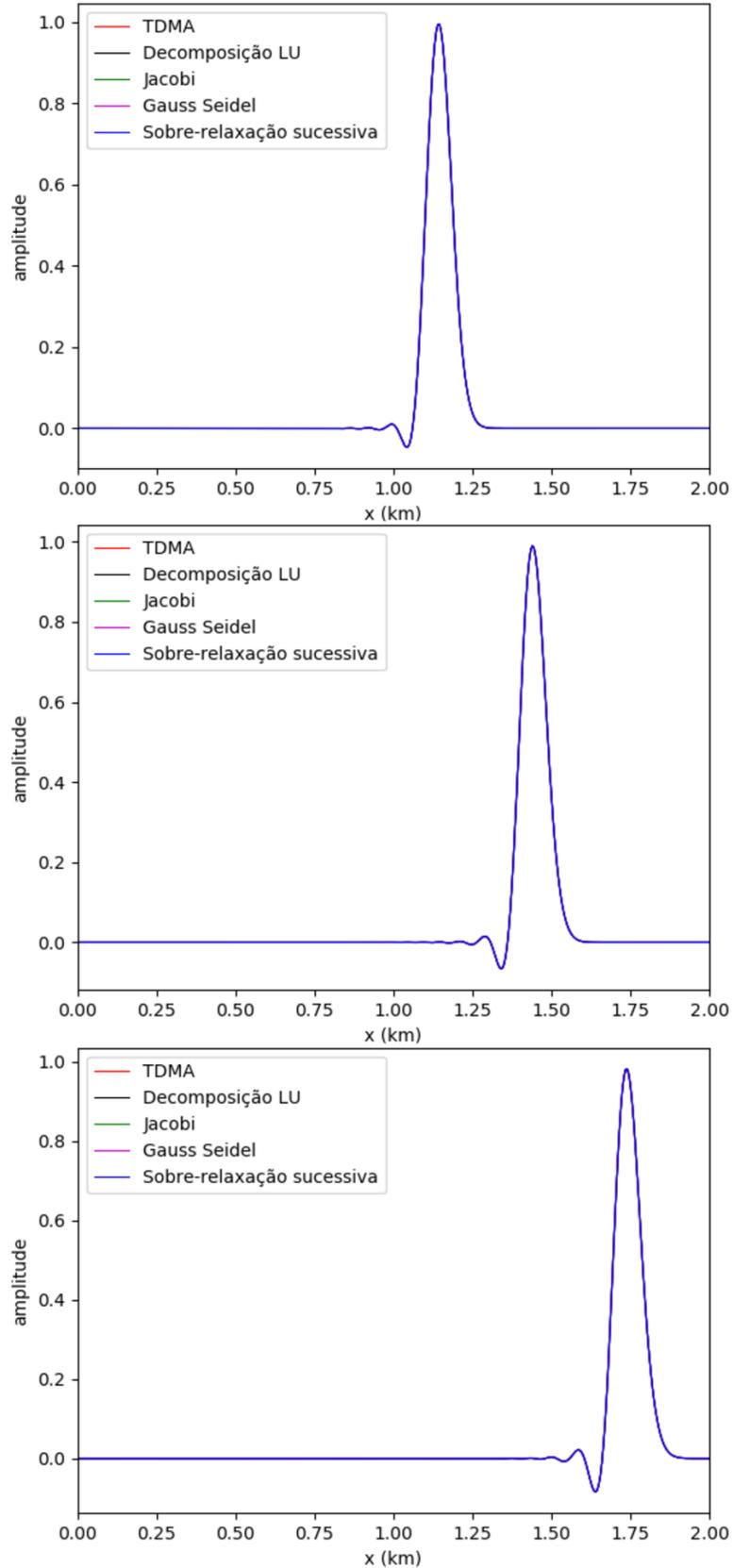


Figura 2: Solução do problema proposto usando diferentes metodologias para resolver o sistema linear, velocidade de 12 m/s e $t = 0,075\text{ s}$, $t = 0,1\text{ s}$ e $t = 0,125\text{ s}$, respectivamente.

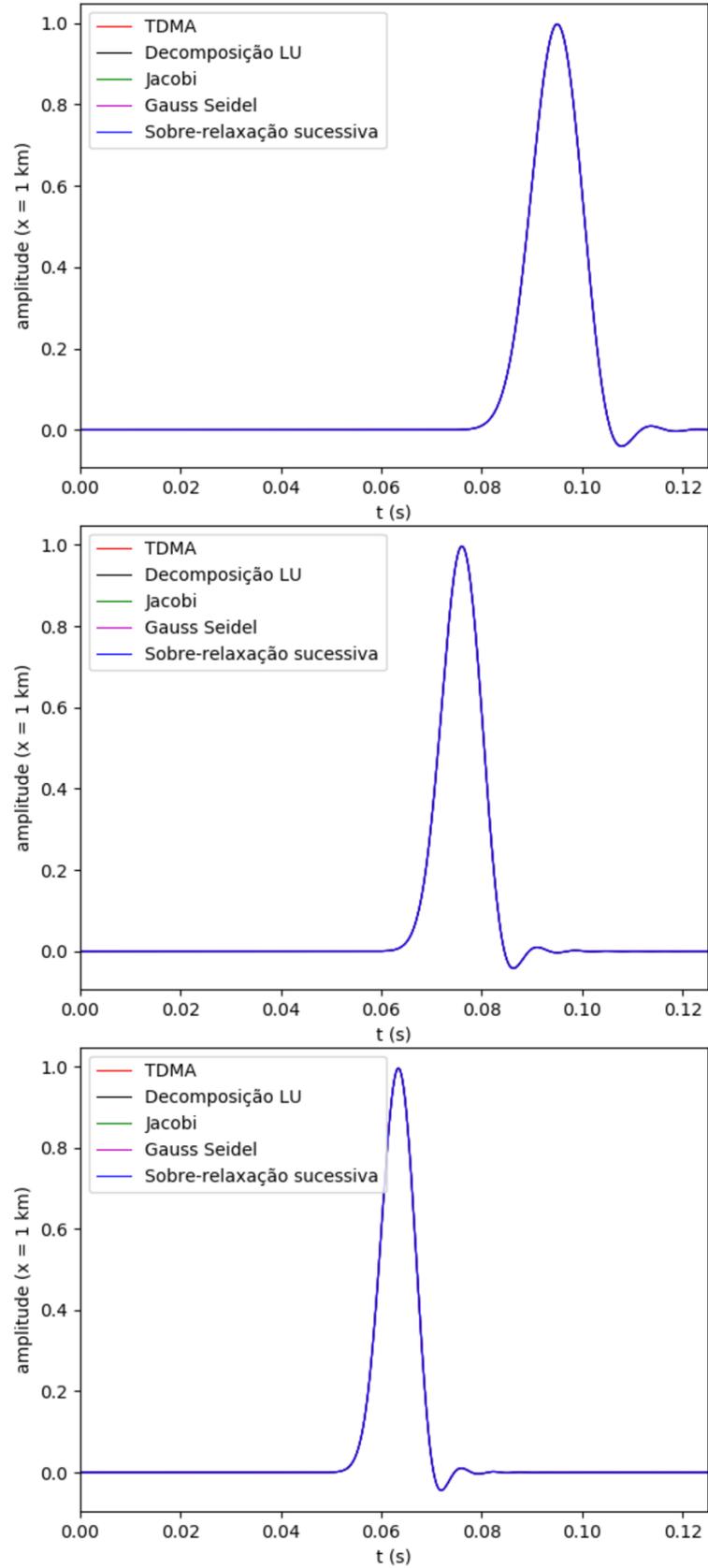


Figura 3: Solução do problema proposto usando diferentes metodologias para resolver o sistema linear, $x = 1 \text{ km}$ e velocidades de 8 m/s , 10 m/s e 12 m/s , respectivamente.

REFERÊNCIAS

BURDEN, R. L. & FAIRES, J. D.. Numerical analysis. Canada, Brooks/Cole Cengage Learning (2010).

BUSKE, D.; VILHENA, M. T.; MOREIRA, D. M.; TIRABASSI, T.. Simulation of pollutant dispersion for low wind conditions in stable and convective planetary boundary layer. **Atmospheric Environment**, V. 41, P. 5496-5501, 2007.

CAMPOS FILHO, F. F.. Algoritmos Numéricos. Belo Horizonte, LTC Editora (2007).

DUMKA, P. & MISHRA, D. R.. Understanding the TDMA/Thomas algorithm and its Implementation in Python, **International Journal of All Research Education and Scientific Method**, V. 10, P. 998-1002, 2022.

GOLUB, G. H.; LOAN, C. F. V.. Matrix computations. London, The Johns Hopkins University Press (1996).

JACOBSON, M. Z.. Fundamentals of Atmospheric Modeling. New York, Cambridge University Press (2005).

HOLTON, J. R.. An Introduction to Dynamic Meteorology. FOURTH EDITION, San Diego, California USA, Elsevier Academic Press (2004).

LUTZ, M.. Learning Python. FOURTH EDITION. Sebastopol, CA, O'Reilly (2009).