Kuramoto–Sivashinsky Equation

Your work should be presented in the form of a typed report using clear and properly punctuated English. Where appropriate include full program listings and output. If you choose to work in a group of two, please turn in independently prepared reports.

1. Consider the hyper-diffusion equation given by

$$\begin{cases} v_t + \nu v_{xxxx} = 0 & \text{for } (x,t) \in (0,1) \times (0,T) \\ v(0,t) = v(1,t) = 0 & \\ v_{xx}(0,t) = v_{xx}(1,t) = 0 & \text{for } t \in [0,T] \\ v(x,0) = f(x) & \text{for } x \in [0,1] \end{cases}$$

and the finite difference approximation

$$\begin{cases} u_k^{n+1} = u_k^n - \rho\delta^4 u_k^n & \text{for } k = 1, \ldots, K-1 \\ & n = 0, \ldots, N-1 \\ u_0^n = u_K^n = 0 & \\ u_{-1}^n = -u_1^n \text{ and } u_{K+1}^n = -u_{K-1}^n & \text{for } n = 0, \ldots, N \\ u_k^0 = f(k\Delta x) & \text{for } k = 0, \ldots, K \end{cases}$$

where $\rho = \nu\Delta t/\Delta x^4$ with $\Delta t = T/N$ and $\Delta x = 1/K$.

(i) Assume the solutions v are smooth on $[0,1] \times [0,T]$ and show the finite difference approximation is consistent. Find the order of the approximation.

(ii) Find conditions under which this finite difference scheme is stable and use the Lax theorem to show this method is conditionally convergent.

(iii) Write a program which implements this finite difference scheme.

(iv) Use your program to approximate $v(x,t)$ on $[0,1] \times [0,0.25]$ where $\nu = 0.00005$ and $f(x) = -48x^5 + 112x^4 - 64x^3$. Find $v(0.5, 0.25)$ to 3 significant digits.

(v) Write a program that implements the implicit scheme

$$\begin{cases} u_k^{n+1} + \rho\delta^4 u_k^{n+1} = u_k^n & \text{for } k = 1, \ldots, K-1 \\ & n = 0, \ldots, N-1 \\ u_0^n = u_K^n = 0 & \\ u_{-1}^n = -u_1^n \text{ and } u_{K+1}^n = -u_{K-1}^n & \text{for } n = 0, \ldots, N \\ u_k^0 = f(k\Delta x) & \text{for } k = 0, \ldots, K \end{cases}$$

(vi) Use this implicit scheme to approximate $v(0.5, 0.25)$.

(vii) Numerically test whether the implicit scheme is stable for all Δx and Δt.

2. Consider the Kuramoto–Sivashinsky equation given by

$$\begin{cases} v_t + \mu v_{xx} + \nu v_{xxxx} + v v_x = 0 & \text{for } (x,t) \in (0,1) \times (0,T) \\ v(0,t) = v(1,t) = 0 & \\ v_{xx}(0,t) = v_{xx}(1,t) = 0 & \text{for } t \in [0,T] \\ v(x,0) = f(x) & \text{for } x \in [0,1] \end{cases}$$

and the finite difference approximation

$$\begin{cases} u_k^{n+1} = u_k^n - (r\delta^2 + \rho\delta^4)u_k^n - R u_k^n \delta_0 u_k^n & \text{for } k = 1, \dots, K-1 \\ & n = 0, \dots, N-1 \\ u_0^n = u_K^n = 0 & \\ u_{-1}^n = -u_1^n \text{ and } u_{K+1}^n = -u_{K-1}^n & \text{for } n = 0, \dots, N \\ u_k^0 = f(k\Delta x) & \text{for } k = 0, \dots, K \end{cases}$$

where $r = \mu\Delta t/\Delta x^2$, $\rho = \nu\Delta t/\Delta x^4$ and $R = \Delta t/(2\Delta x)$.

(i) Write a program which implements this finite difference scheme.

(ii) Approximate $v(x,t)$ on $[0,1] \times [0,0.25]$ where $\mu = 0.1$, $\nu = 0.00005$ and $f(x) = -48x^5 + 112x^4 - 64x^3$. Find $v(0.5, 0.25)$ to 2 significant digits.

(iii) Write a program that implements the semi-implicit scheme

$$\begin{cases} u_k^{n+1} + (r\delta^2 + \rho\delta^4)u_k^{n+1} = u_k^n - R u_k^n \delta_0 u_k^n & \text{for } k = 1, \dots, K-1 \\ & n = 0, \dots, N-1 \\ u_0^n = u_K^n = 0 & \\ u_{-1}^n = -u_1^n \text{ and } u_{K+1}^n = -u_{K-1}^n & \text{for } n = 0, \dots, N \\ u_k^0 = f(k\Delta x) & \text{for } k = 0, \dots, K \end{cases}$$

(iv) Use this semi-implicit scheme to approximate $v(0.5, 0.25)$.

(v) Write a program that implements the split-implicit scheme

$$\begin{cases} u_k^{n+1} + (r\delta^2 + \rho\delta^4 + R u_k^n \delta_0)u_k^{n+1} = u_k^n & \text{for } k = 1, \dots, K-1 \\ & n = 0, \dots, N-1 \\ u_0^n = u_K^n = 0 & \\ u_{-1}^n = -u_1^n \text{ and } u_{K+1}^n = -u_{K-1}^n & \text{for } n = 0, \dots, N \\ u_k^0 = f(k\Delta x) & \text{for } k = 0, \dots, K \end{cases}$$

(vi) Use this split-implicit scheme to approximate $v(0.5, 0.25)$.

(vii) Compare and contrast the performance of the explicit scheme, the semi-implicit scheme and the split-implicit scheme for values of $K \in \{10, 100, 1000\}$. Which method do you like best? Why?