Stability Analysis of Finite Difference PDE Solvers

Kshiteesh S - EP24BTECH11023

Course - Mathematical Physics

Instructor - Dr. Sangkha Borah

Abstract

Partial Differential Equations govern the mathematical territory of physics. The behaviour of most complex systems can be determined by a PDE / set of PDEs. However, PDEs are notoriously difficult to solve by hand, and some PDEs have solutions that are almost humanely impossible to find analytically. This issue is solved by tackling a PDE numerically by discretizing the domain of analysis (be it space or time or both), discretizing the partial derivative terms (using Taylor expansions) and reducing the problem down to a system of linear equations which can be solved algebraically to approximate the final answer. Such schemes of solving PDEs fall into the large bracket of algorithms known as “Finite Difference Methods” (**FDM**s).

Though, when using FDMs to solve a PDE, one must take into consideration the fact that the "stability" of the FDM is dependent on the way the domain space has been discretized.

This project aims to dive deep into the mathematical analysis of the stability of an FDM.

Contents

1 Finite Difference Methods (FDMs) - an Intuition behind their Design

1.1 Formulating a Finite Difference Method

Suppose that you are given a task to solve the IBVP (Initial Boundary Value Problem - A problem in which we know the initial value $f(x, 0)$ throughout space, and the boundary conditions $f(x_0, t)$ & $f(x_N, t)$ at all times)

$$\mathcal{L}[f(x,t)] = s(x,t)$$

The analytical solution for this could be mentally taxing (or) impossible to find using the calculus approach. The motivation behind implementing Finite Difference methods is to convert a **calculus** problem into a **linear algebra** problem that can be solved analytically to find a 'good enough' approximation of the partial differential equation's solution. We do this by discretizing the domain (creating a mesh), approximating the derivatives using **finite differences**, plugging in the derivative apprximations, and boiling the PDE down to a linear relation.

Discretization is done by taking 'N' grid points $x_0,\ x_1,\ x_2,\ \ldots,\ x_N$ with uniform spacing $h = x_{i+1} - x_i$. We only calculate f at these grid points.

The most common finite difference formulas are:

Forward Difference: $D^+u = \frac{u(x+\Delta x)-u(x)}{\Delta x}$

Backward Difference: $D^-u = \frac{u(x)-u(x-\Delta x)}{\Delta x}$

Central Difference: $D^0u = \frac{u(x+\Delta x)-u(x-\Delta x)}{2\Delta x}$

1.2 An Intuitive Introduction to the Notion of Stability of a Finite Difference Method

Finite Difference Methods are used to find the evolution of the function's value at each grid point,step by step. At each step, there are small errors. Stability means these errors, instead of growing wildly, stay bounded and don't distort your solution. If a method is unstable, even tiny inaccuracies can snowball, making the result unreliable over time.

2 von Neumann Stability Analysis (vNSA)

Suppose you are given a **Linear Differential Equation** $\mathcal{L}[f(x)] = s(x)$ (**with constant coefficients**) and you discretize this Differential Equation using an FDM. The 'error' at each grid point (x_i) at the n^{th} step, ϵ_i^n, will satisfy the discretized PDE. Our agenda is to analyse the growth of the error due to this FDM. This is achieved by **vNSA**.

Steps:

1. Assume the error at each grid point has the form:

$$\epsilon_j^n = \sum_k (G_k)^n e^{ik(j\Delta x)}\, dk \quad \text{(or)} \int_{-\infty}^{\infty} (G_k)^n e^{ik(j\Delta x)}\, dk$$

Where: - ϵ_j^n is the error at grid point j and time step n.
- G_k is the amplification factor for the mode with wavenumber k.
- Δx is the spatial grid spacing.
{The product $(j\Delta x)$ represents distance from initial grid point }

2. Substitute this error representation into the finite difference update equation, and analyse for a single value of k.

3. Solve for the amplification factor G_k (sometimes called the growth factor or eigenvalue) in terms of k. {The error update equation can typically be rewritten so that $e^{ik(j\Delta x)}$ terms cancel, leaving a simplified equation for G_k. if this is not the case, one can implement a Fourier Transform to obtain G_k }

4. Establish the stability condition:

$$|G_k| \leq 1 \; \forall \; k$$

This ensures that no Fourier mode (no error component) grows unbounded as computations proceed.

2.1 Understanding Dissipation and Dispersion Through the Growth Factor

Once the amplification factor G_k has been obtained from the vNSA procedure, its magnitude and phase contain all the necessary information about how the numerical scheme modifies individual Fourier modes of the solution and of the error.

The growth factor can always be written in polar form:

$$G_k = |G_k|\, e^{i\phi_k},$$

where $|G_k|$ controls amplitude change and $\phi_k = \arg(G_k)$ determines the numerical phase shift.

Dissipation

The magnitude $|G_k|$ tells us whether a Fourier mode grows, decays, or is preserved by the numerical method. After n time steps, the amplitude of that mode becomes $|G_k|^n$.

$$\boxed{\text{Dissipation occurs when } |G_k| < 1.}$$

In this case, the numerical method introduces *artificial damping* of error modes. High-frequency components (large k) are typically damped the most, leading to a smoothing effect. If $|G_k| = 1$ for all k, the method is said to be *non-dissipative*.

Dispersion

Dispersion concerns the *phase* of Fourier modes. Even when the amplitude is preserved (i.e. $|G_k| = 1$), the phase angle ϕ_k may differ from the exact PDE's phase.

Writing

$$G_k = |G_k| e^{i\phi_k},$$

the numerical mode advances as

$$e^{i(kj\Delta x - \omega_k n\Delta t)},$$

where the numerical frequency is

$$\omega_k = -\frac{1}{\Delta t}\phi_k.$$

Thus the numerical phase velocity is

$$c_{\text{num}}(k) = \frac{\omega_k}{k} = -\frac{1}{k\Delta t}\phi_k.$$

$$\boxed{\text{Dispersion occurs when } \phi_k \neq 0.}$$

Different wavenumbers then propagate at different (and incorrect) speeds, causing wave packets to distort even in the absence of dissipation.

Combined Interpretation

The growth factor therefore neatly separates the two major spectral effects of a numerical method:

$$\boxed{\begin{aligned} &\text{Dissipation} \iff |G_k| \neq 1, \\ &\text{Dispersion} \iff \arg(G_k) \neq 0. \end{aligned}}$$

A scheme may be:

- purely dissipative ($|G_k| < 1$ and $\arg(G_k) = 0$),
- purely dispersive ($|G_k| = 1$ and $\arg(G_k) \neq 0$),
- both dissipative and dispersive, or
- neither.

Thus, once G_k is obtained through the von Neumann procedure, both stability and qualitative behavior (smoothing, oscillation, phase distortion) follow directly from analysing its magnitude and phase.

3 vNSA of FDM Schemes for the Linear Advection Equation

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

This is the Linear Advection equation, and it is the simplest bi-variate PDE. Various FDMs such as the FTCS Scheme, the Implicit-Euler Scheme, the Lax-Friedrich Scheme, the Lax-Wendroff Scheme, etc. can be used to computationally solve this PDE.

3.1 Choice of Computational Modelling

For all analyses (done in `python` using `numpy` and `matplotlib`) , the linear advection equation

$$u_t + u_x = 0,$$

with wave speed $c = 1$, was used as the model PDE. This isolates the behaviour of numerical schemes on a pure transport problem with no diffusive or reactive effects. The spatial discretization Δx was kept fixed for all schemes, and the total simulation time was held constant (at $T = 2$) so that differences arise solely from the time-stepping methods.

The value of $\lambda = \frac{c\Delta t}{\Delta x}$ was varied by adjusting the number of time steps N_t (and thus Δt) wherever the CFL condition plays a role. This made it possible to observe stable and unstable regimes for each scheme.

Three types of initial conditions were used:

- **Pure sine wave**:

  $$u(x, 0) = \sin(kx), \ (k = 10 \text{ was used})$$

  To observe the behaviour of a single Fourier mode, consistent with the foundations of von Neumann analysis.

- **Gaussian-modulated sine wave**:

  $$u(x, 0) = e^{-(x-x_0)^2/\sigma^2} \sin(kx), \ (k = 10 \text{ was used})$$

  To study how each scheme handles a *localized wave packet*. This initial condition

contains a narrow band of Fourier modes centered around a dominant wavenumber, making it ideal for examining combined dispersion and dissipation effects as the packet propagates.

- **Heaviside-like function**: To examine how schemes behave in the presence of discontinuities. The step contains strong high-frequency Fourier components, so it reveals how aggressively each method damps or distorts sharp gradients and how numerical viscosity influences the resolution of discontinuous features.

3.2 vNSA of the Forward Time–Centered Space (FTCS) Scheme

The FTCS Scheme discretizes the space–time domain by taking a forward difference in time and a centered difference in space. The discretized equation is

$$\frac{f_j^{n+1} - f_j^n}{\Delta t} + c\left(\frac{f_{j+1}^n - f_{j-1}^n}{2\Delta x}\right) = 0,$$

that is,

$$f_j^{n+1} = f_j^n - \frac{c\Delta t}{2\Delta x}\left(f_{j+1}^n - f_{j-1}^n\right) + \mathcal{O}(\Delta t^2 + \Delta x^2).$$

Substituting the error ϵ_j^n gives

$$\epsilon_j^{n+1} = \epsilon_j^n - \frac{c\Delta t}{2\Delta x}\left(\epsilon_{j+1}^n - \epsilon_{j-1}^n\right).$$

For a mode with wavenumber k, assume

$$\epsilon_j^n = (G_k)^n e^{ikj\Delta x}.$$

Substituting into the update equation:

$$(G_k)^{n+1} e^{ikj\Delta x} = (G_k)^n e^{ikj\Delta x} - \frac{c\Delta t}{2\Delta x}\left[(G_k)^n e^{ik(j+1)\Delta x} - (G_k)^n e^{ik(j-1)\Delta x}\right].$$

Dividing by the common factor $(G_k)^n e^{ikj\Delta x}$ gives

$$G_k = 1 - \frac{c\Delta t}{2\Delta x}\left(e^{ik\Delta x} - e^{-ik\Delta x}\right) = 1 - i\frac{c\Delta t}{\Delta x}\sin(k\Delta x).$$

Thus the amplification factor is

$$G_k = 1 - i\lambda \sin(k\Delta x), \qquad \text{where } \lambda = \frac{c\Delta t}{\Delta x}.$$

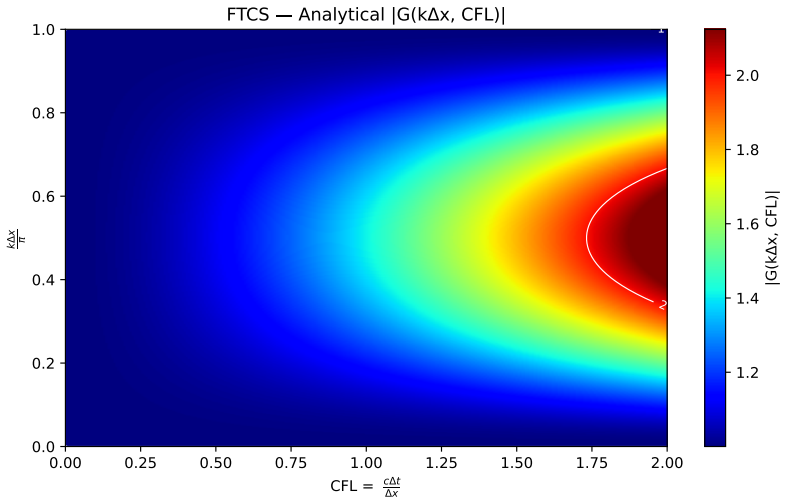


Figure 1: Variation of $|G|$ over different values of $k\Delta x$ and $\lambda = c\frac{\Delta t}{\Delta x}$ for the FTCS Scheme.

Dissipation (Amplitude Error)

The magnitude of G_k is

$$|G_k| = \sqrt{1 + \lambda^2 \sin^2(k\Delta x)}.$$

Since

$$|G_k| > 1 \qquad \forall\, k \neq 0,$$

the FTCS method **amplifies** all nonzero Fourier modes, making the scheme

Unstable and anti–dissipative.

Dispersion (Phase Error)

Write

$$G_k = |G_k| e^{i\phi_k}.$$

The argument (phase) is

$$\phi_k = \arg(G_k) = \arctan\left(-\frac{\lambda \sin(k\Delta x)}{1}\right) = -\arctan(\lambda \sin(k\Delta x))\,.$$

The corresponding numerical frequency is

$$\omega_k = -\frac{\phi_k}{\Delta t},$$

and the numerical phase velocity is

$$c_{\text{num}}(k) = \frac{\omega_k}{k} = \frac{1}{k\Delta t}\arctan(\lambda \sin(k\Delta x))\,.$$

Since $\phi_k \neq$ the exact phase ck for all k, the method introduces a nonzero phase error. Therefore FTCS is

Dispersive (phase-distorting).

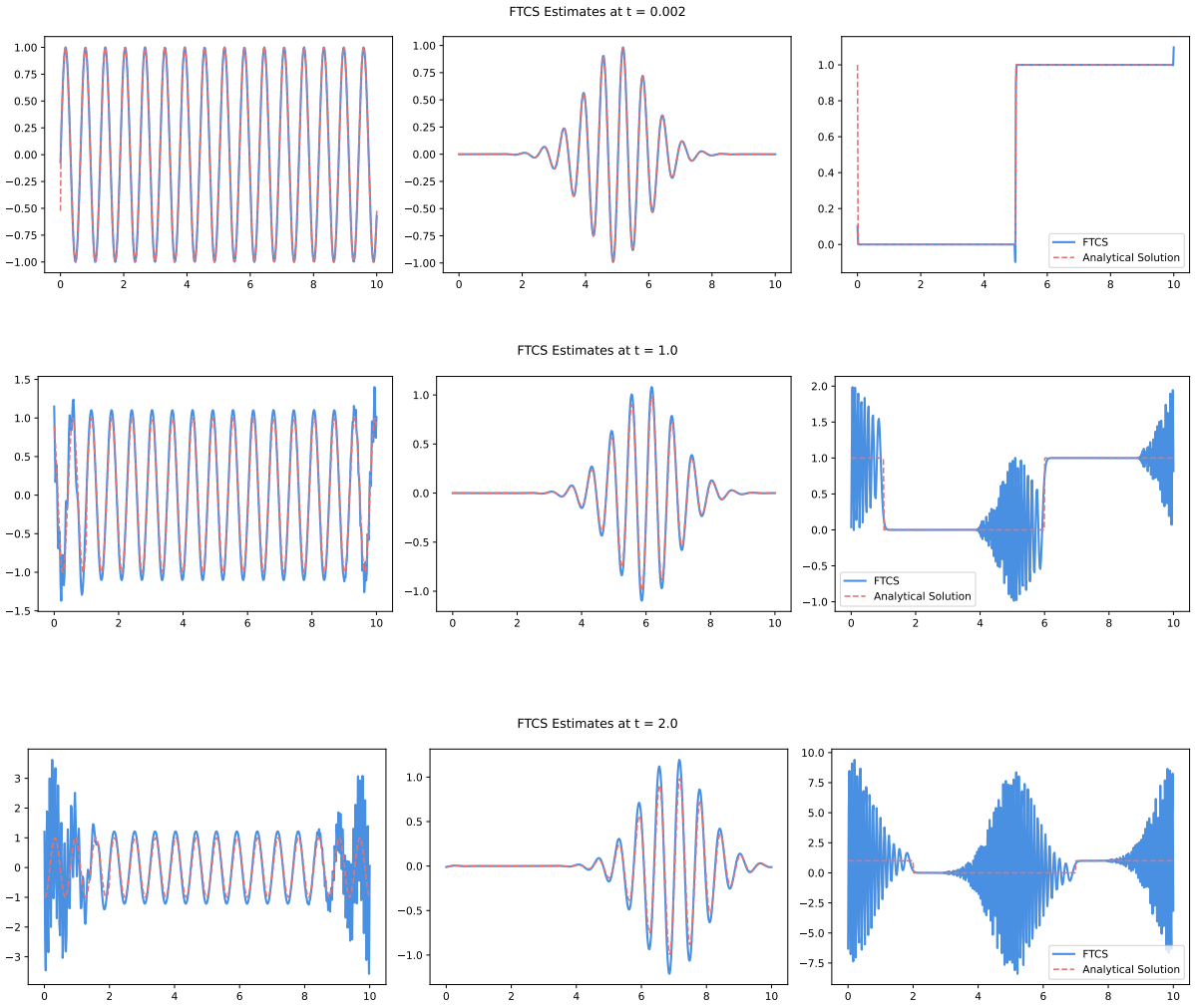


Figure 2: Time Evolution of the FTCS Solutions. Notice that the errors “explode" over time

3.3 vNSA of the Backward Time–Centered Space (BTCS) Scheme

The BTCS Scheme discretizes the space–time domain using a backward difference in time and a centered difference in space (both evaluated at the **next** time step $n+1$). The discretized equation is

$$\frac{f_j^{n+1} - f_j^n}{\Delta t} + c\left(\frac{f_{j+1}^{n+1} - f_{j-1}^{n+1}}{2\Delta x}\right) = 0,$$

that is,

$$f_j^{n+1} + \frac{c\Delta t}{2\Delta x}\left(f_{j+1}^{n+1} - f_{j-1}^{n+1}\right) = f_j^n.$$

Substituting the error ϵ_j^n gives

$$\epsilon_j^{n+1} + \frac{c\Delta t}{2\Delta x}\left(\epsilon_{j+1}^{n+1} - \epsilon_{j-1}^{n+1}\right) = \epsilon_j^n.$$

For a mode with wavenumber k, assume

$$\epsilon_j^n = (G_k)^n e^{ikj\Delta x}.$$

Substituting into the update equation:

$$(G_k)^{n+1}e^{ikj\Delta x} + \frac{c\Delta t}{2\Delta x}\left[(G_k)^{n+1}e^{ik(j+1)\Delta x} - (G_k)^{n+1}e^{ik(j-1)\Delta x}\right] = (G_k)^n e^{ikj\Delta x}.$$

Dividing by $(G_k)^n e^{ikj\Delta x}$

$$G_k\left[1 + \frac{c\Delta t}{2\Delta x}\left(e^{ik\Delta x} - e^{-ik\Delta x}\right)\right] = 1.$$

Using $e^{i\theta} - e^{-i\theta} = 2i\sin\theta$ gives

$$G_k = \frac{1}{1 + i\lambda\sin(k\Delta x)}, \qquad \lambda = \frac{c\Delta t}{\Delta x}.$$

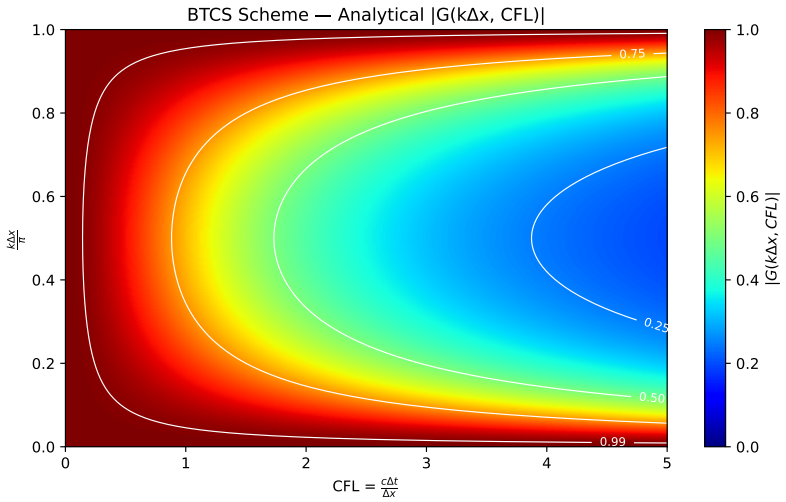


Figure 3: Variation of $|G|$ over different values of $k\Delta x$ and $\lambda = c\frac{\Delta t}{\Delta x}$ for the BTCS Scheme.

Dissipation (Amplitude Error)

The magnitude of the growth factor is

$$|G_k| = \frac{1}{\sqrt{1 + \lambda^2 \sin^2(k\Delta x)}}.$$

Since

$$|G_k| < 1 \qquad \forall\, k \neq 0,$$

all nonzero Fourier modes decay. Therefore BTCS is

Unconditionally stable and dissipative.

The dissipation increases with k, meaning high–frequency modes are damped the most.

Dispersion (Phase Error)

Writing

$$G_k = |G_k| e^{i\phi_k},$$

the argument (phase) is

$$\phi_k = \arg(G_k) = -\arctan(\lambda \sin(k\Delta x))\,.$$

The numerical frequency is

$$\omega_k = -\frac{\phi_k}{\Delta t},$$

and the numerical phase velocity is

$$c_{\text{num}}(k) = \frac{\omega_k}{k} = \frac{1}{k\Delta t}\arctan(\lambda \sin(k\Delta x))\,.$$

Since this phase differs from the exact phase ck, BTCS introduces phase distortion. Therefore it is

Dispersive (phase-distorting).

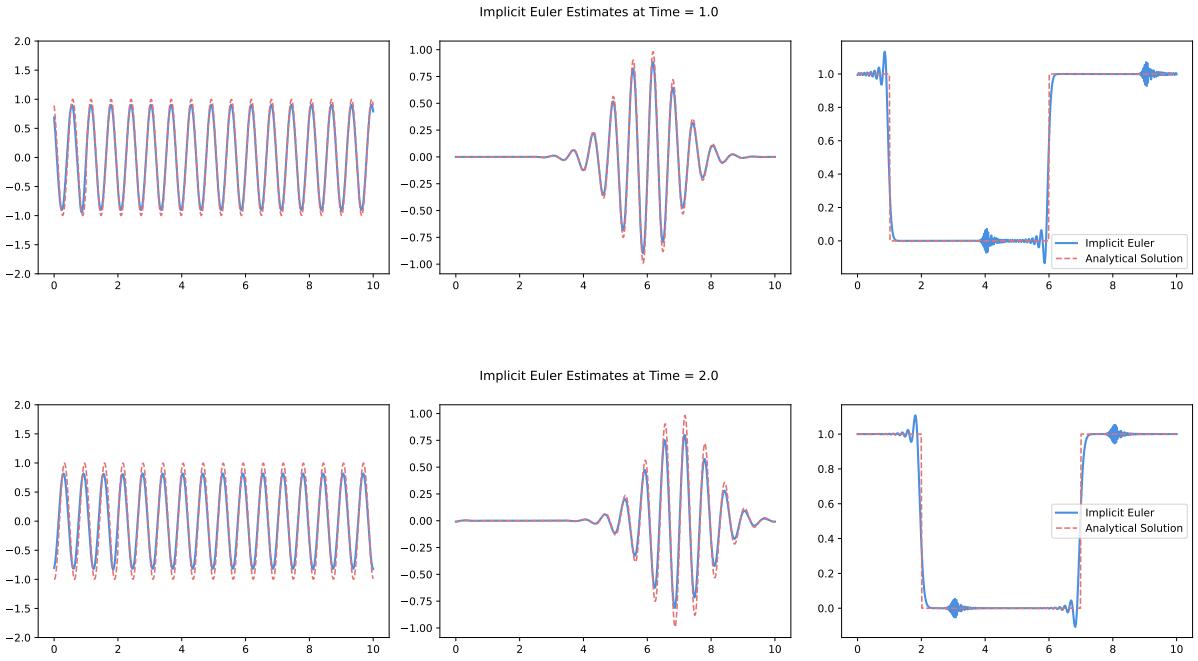


Figure 4: Time Evolution of the BTCS Solutions. The oscillations move away from centre of the step function "symmetrically" due to the phase error.

3.4 vNSA of the Leapfrog Scheme

The Leapfrog method uses a central difference in both space and time:

$$\frac{f_j^{n+1} - f_j^{n-1}}{2\Delta t} + c\left(\frac{f_{j+1}^n - f_{j-1}^n}{2\Delta x}\right) = 0.$$

Rearranging, the update equation becomes

$$f_j^{n+1} = f_j^{n-1} - \lambda\left(f_{j+1}^n - f_{j-1}^n\right), \qquad \lambda = \frac{c\Delta t}{\Delta x}.$$

Substituting error ϵ_j^n gives

$$\epsilon_j^{n+1} = \epsilon_j^{n-1} - \lambda \left(\epsilon_{j+1}^n - \epsilon_{j-1}^n \right).$$

Substituting our fourier mode error, $\epsilon_j^n = (G_k)^n e^{ikj\Delta x}$:

$$(G_k)^{n+1} e^{ikj\Delta x} = (G_k)^{n-1} e^{ikj\Delta x} - \lambda \left[(G_k)^n e^{ik(j+1)\Delta x} - (G_k)^n e^{ik(j-1)\Delta x} \right].$$

Dividing by $(G_k)^n e^{ikj\Delta x}$:

$$G_k = G_k^{-1} - \lambda \left(e^{ik\Delta x} - e^{-ik\Delta x} \right).$$

Using $e^{i\theta} - e^{-i\theta} = 2i \sin\theta$:

$$G_k = G_k^{-1} - i\, 2\lambda \sin(k\Delta x).$$

Multiplying both sides by G_k yields the quadratic amplification equation

$$G_k^2 + i\, 2\lambda \sin(k\Delta x)\, G_k + 1 = 0.$$

Solving for G_k:

$$G_k = -i\lambda \sin(k\Delta x) \;\pm\; \sqrt{1 - \lambda^2 \sin^2(k\Delta x)}.$$

Dissipation (Amplitude Error)

When $\lambda \leq 1$, The term under the square root is positive. Thus, the magnitude of both roots is

$$|G_k| = \sqrt{\lambda^2 \sin^2(k\Delta x) + (1 - \lambda^2 \sin^2(k\Delta x))} = 1.$$

Thus, for $\lambda \leq 1$,

Leapfrog is nondissipative.

When $\lambda > 1$

$$|G_k| = \sqrt{2\lambda^2 \sin^2(k\Delta x) - 1}$$

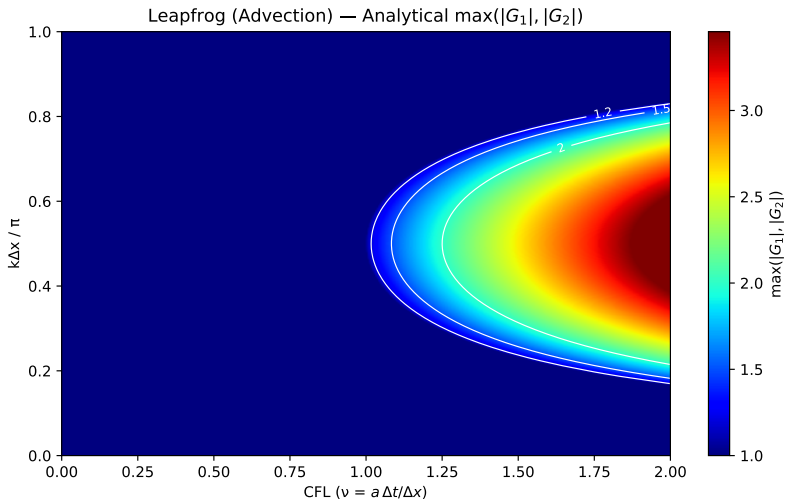


Figure 5: Variation of $\max(|G_1|, |G_2|)$ over different values of $k\Delta x$ and $\lambda = c\frac{\Delta t}{\Delta x}$ for Leapfrog Method.

Dispersion (Phase Error) (when $\lambda \leq 1$

Write $G_k = e^{i\phi_k}$, since $|G_k| = 1$. From the expression for G_k,

$$\phi_k = \arg\left(-i\lambda \sin(k\Delta x) \pm \sqrt{1 - \lambda^2 \sin^2(k\Delta x)}\right).$$

The numerical frequency (calculated taking the +ve root) is:

$$\omega_k = -\frac{\phi_k}{\Delta t}, \qquad c_{\text{num}}(k) = \frac{\omega_k}{k}.$$

Since $\phi_k \neq ck\Delta t$, the numerical phase velocity differs from the true velocity c, making the Leapfrog method:

Dispersive (phase-distorting).

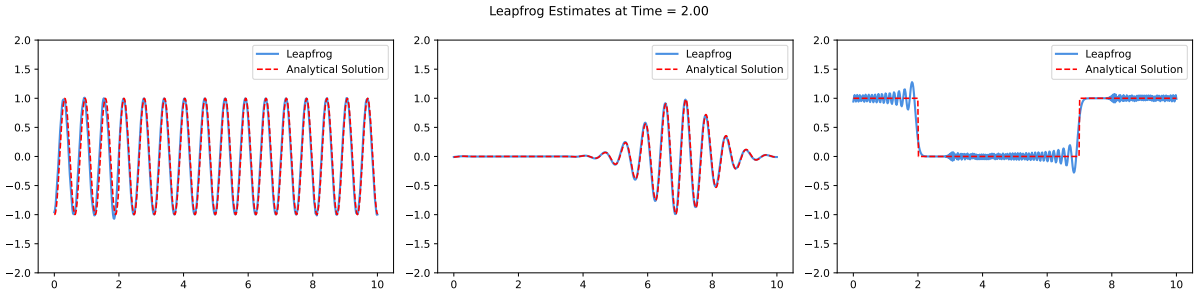


Figure 6: The results of Leapfrog Scheme. The small oscillatory errors in the Step function's solution move away from the center due to phase-distortion.

3.5 vNSA of the Lax-Friedrichs Scheme

The Lax-Friedrichs discretization uses a centered difference in space and a forward difference in time, with an added averaging term:

$$f_j^{n+1} = \tfrac{1}{2}\left(f_{j+1}^n + f_{j-1}^n\right) - \frac{\lambda}{2}\left(f_{j+1}^n - f_{j-1}^n\right), \qquad \lambda = \frac{c\Delta t}{\Delta x}.$$

Substituting the error ϵ_j^n:

$$\epsilon_j^{n+1} = \tfrac{1}{2}\left(\epsilon_{j+1}^n + \epsilon_{j-1}^n\right) - \frac{\lambda}{2}\left(\epsilon_{j+1}^n - \epsilon_{j-1}^n\right).$$

Substituting our Fourier mode error, $\epsilon_j^n = (G_k)^n e^{ikj\Delta x}$:

$$(G_k)^{n+1}e^{ikj\Delta x} = \tfrac{1}{2}(G_k)^n e^{ik(j+1)\Delta x} + \tfrac{1}{2}(G_k)^n e^{ik(j-1)\Delta x} - \frac{\lambda}{2}\left[(G_k)^n e^{ik(j+1)\Delta x} - (G_k)^n e^{ik(j-1)\Delta x}\right].$$

Dividing by $(G_k)^n e^{ikj\Delta x}$:

$$G_k = \tfrac{1}{2}\left(e^{ik\Delta x} + e^{-ik\Delta x}\right) - \frac{\lambda}{2}\left(e^{ik\Delta x} - e^{-ik\Delta x}\right).$$

Using

$$e^{i\theta} + e^{-i\theta} = 2\cos\theta, \qquad e^{i\theta} - e^{-i\theta} = 2i\sin\theta,$$

we obtain the growth factor:

$$G_k = \cos(k\Delta x) - i\,\lambda \sin(k\Delta x).$$

Dissipation (Amplitude Error)

The magnitude of G_k is

$$|G_k| = \sqrt{\cos^2(k\Delta x) + \lambda^2 \sin^2(k\Delta x)}.$$

Note that

$$|G_k| < 1 \quad \text{if } |\lambda| < 1,$$

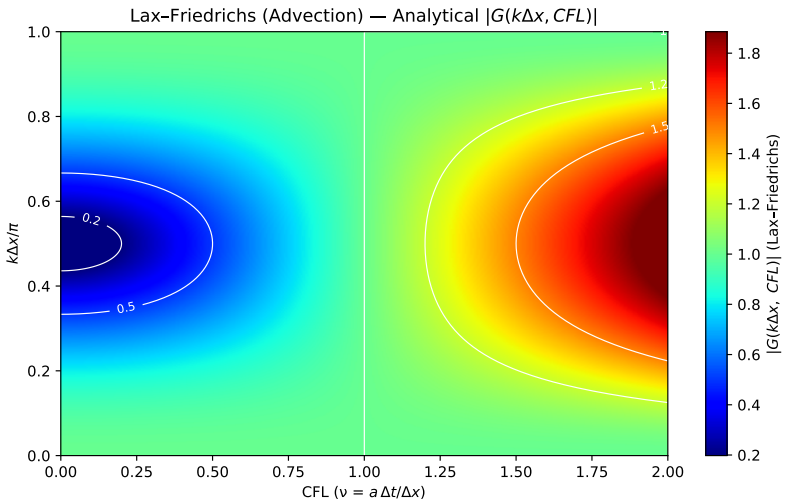


Figure 7: Variation of $|G|$ over different values of $k\Delta x$ and $\lambda = c\frac{\Delta t}{\Delta x}$ for Lax-Friedrich Scheme.

so all nonzero Fourier modes decay.

Thus,

Lax-Friedrichs is dissipative.

The dissipation is strong for high wavenumbers (large $\sin(k\Delta x)$), giving the scheme built-in numerical viscosity.

Dispersion (Phase Error)

Write $G_k = |G_k|e^{i\phi_k}$. The phase is

$$\phi_k = \arg(\cos(k\Delta x) - i\,\lambda \sin(k\Delta x)) = -\arctan\left(\frac{\lambda \sin(k\Delta x)}{\cos(k\Delta x)}\right).$$

Thus the numerical frequency and phase velocity are

$$\omega_k = -\frac{\phi_k}{\Delta t}, \qquad c_{\text{num}}(k) = \frac{\omega_k}{k}.$$

Since $\phi_k \neq ck\Delta t$, the method distorts phase and is dispersive:

Lax–Friedrichs is dispersive.

CFL Condition: Meaning and Significance

The stability requirement $|G_k| \leq 1$ gives

$$\cos^2(k\Delta x) + \lambda^2 \sin^2(k\Delta x) \leq 1 \quad \Rightarrow \quad |\lambda| \leq 1.$$

Thus the Lax–Friedrichs CFL condition is

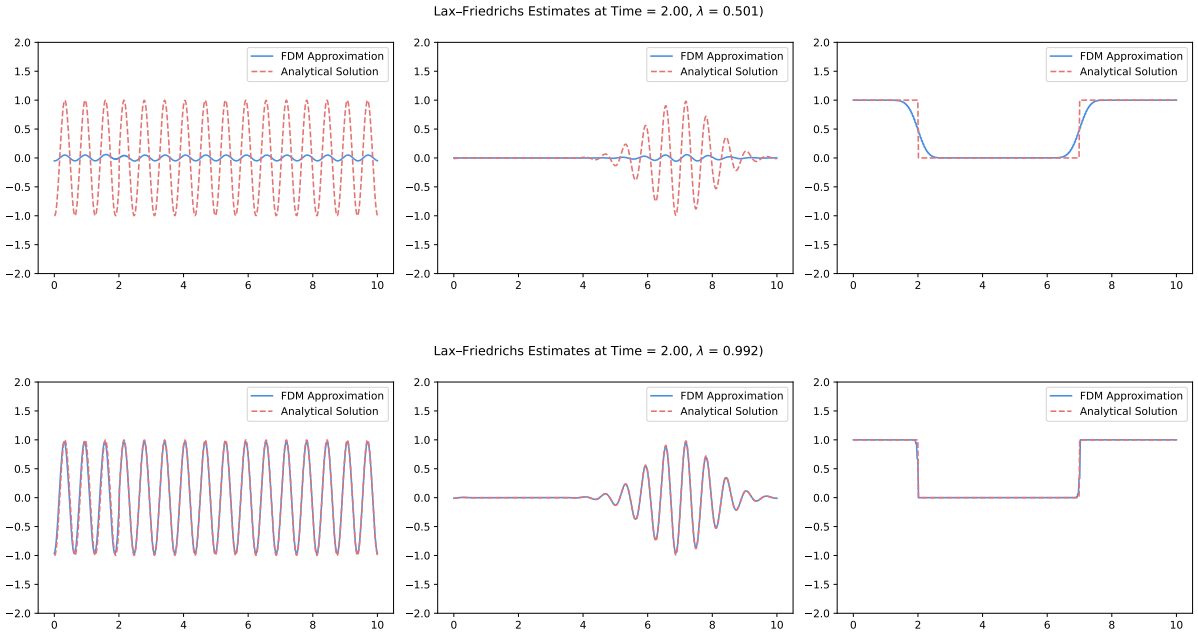
$$\boxed{|c|\frac{\Delta t}{\Delta x} \leq 1.}$$

Meaning. The CFL condition ensures that information carried by the PDE does not move faster than the numerical scheme can propagate it. For advection, c is the wave speed. The grid can only communicate information to neighboring points once every time step, so requiring

$$|c|\Delta t \leq \Delta x$$

means the physical wave travels at most one spatial grid cell in one time step.

Ideal Choice of λ. For the Lax-Friedrich Scheme to run optimally, it is best to choose λ close to 1. If λ exceeds 1, the $|G|$ becomes greater than 1, and the scheme becomes unstable. If we choose $\lambda << 1$, the growth factor reduces significantly, leading to damping.



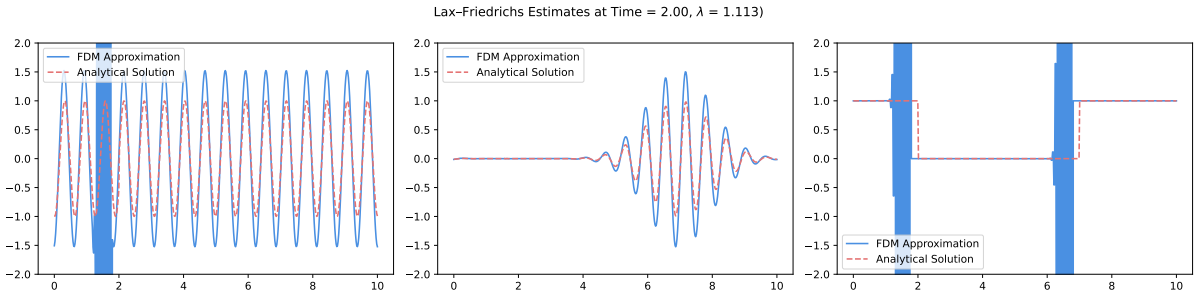


Figure 8: The results of Lax-Friedrich Scheme for $\lambda = 0.5, 1, 1.11$. $\lambda = 0.5$ gives a damped solution (the discontinuity is smoothened), and $\lambda = 1.11$ gives a graph whose errors explode.

3.6 vNSA of the Lax–Wendroff Scheme

The Lax–Wendroff discretization of

$$u_t + c\,u_x = 0$$

is a two–term, second–order scheme obtained by Taylor expansion (or predictor–corrector):

$$f_j^{n+1} = f_j^n - \frac{\lambda}{2}\left(f_{j+1}^n - f_{j-1}^n\right) + \frac{\lambda^2}{2}\left(f_{j+1}^n - 2f_j^n + f_{j-1}^n\right), \qquad \lambda = \frac{c\Delta t}{\Delta x}.$$

Substituting the error ϵ_j^n:

$$\epsilon_j^{n+1} = \epsilon_j^n - \frac{\lambda}{2}\left(\epsilon_{j+1}^n - \epsilon_{j-1}^n\right) + \frac{\lambda^2}{2}\left(\epsilon_{j+1}^n - 2\epsilon_j^n + \epsilon_{j-1}^n\right).$$

Substituting our Fourier mode error, we get: $\epsilon_j^n = (G_k)^n + 1e^{ikj\Delta x}$

$$= (G_k)^n e^{ikj\Delta x} - \frac{\lambda}{2}\left[(G_k)^n e^{ik(j+1)\Delta x} - (G_k)^n e^{ik(j-1)\Delta x}\right] + \frac{\lambda^2}{2}\left[(G_k)^n e^{ik(j+1)\Delta x} - 2(G_k)^n e^{ikj\Delta x} + (G_k)^n e^{ik(j-1)\Delta x}\right].$$

Dividing by $(G_k)^n e^{ikj\Delta x}$:

$$G_k = 1 - \frac{\lambda}{2}\left(e^{ik\Delta x} - e^{-ik\Delta x}\right) + \frac{\lambda^2}{2}\left(e^{ik\Delta x} - 2 + e^{-ik\Delta x}\right).$$

Using trigonometric identities yields the growth factor in closed form:

$$G_k = 1 - i\lambda\sin(k\Delta x) + \lambda^2\left(\cos(k\Delta x) - 1\right).$$

$$|G_k|^2 = \left(1 - 2\lambda^2 \sin^2\left(\frac{k\Delta x}{2}\right)\right)^2 + \lambda^2 \sin^2(k\Delta x).$$

In general $|G_k| \leq 1$ for $\lambda \leq 1$, with equality for special cases (e.g. $\lambda = 1$). In particular, when

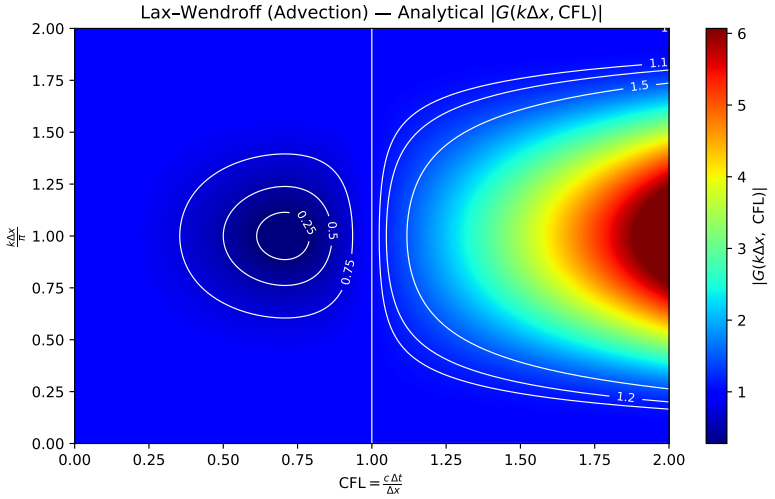


Figure 9: Variation of $|G|$ over different values of $k\Delta x$ and $\lambda = c\frac{\Delta t}{\Delta x}$ for Lax-Wendroff Scheme.

$\lambda = 1$, the growth factor simplifies to $G_k = \cos(k\Delta x) - i\sin(k\Delta x) = e^{-ik\Delta x}$, so $|G_k| = 1$ and the scheme is exactly nondissipative for all modes (this is the CFL– exact case for linear advection).

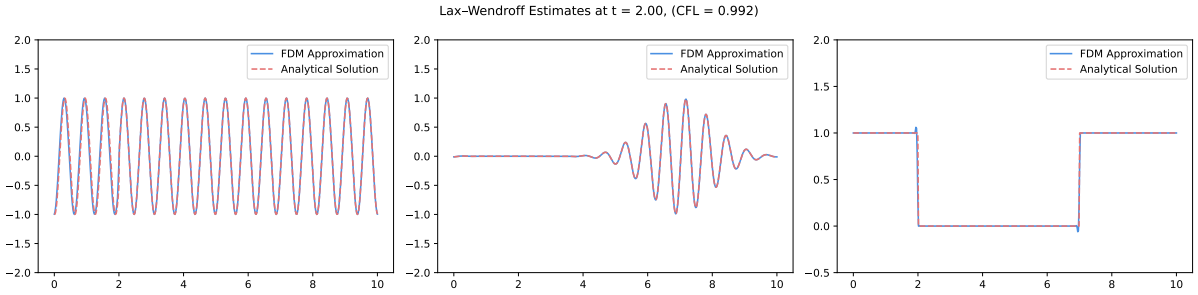


Figure 10: The results of Lax-Wendroff Scheme for $\lambda = 0.99$.

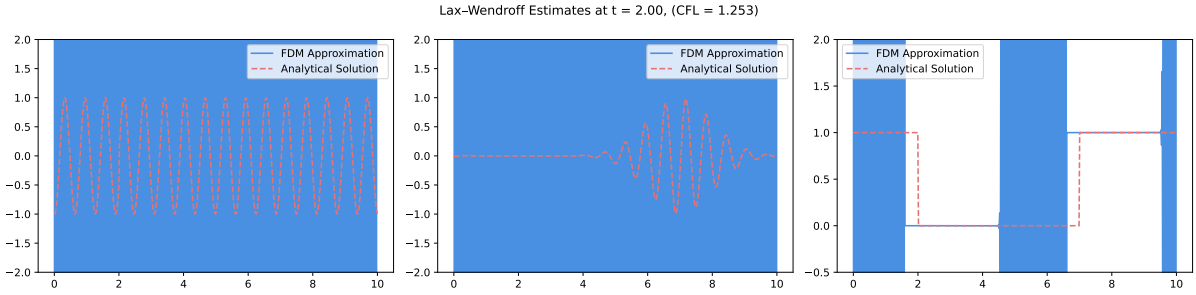


Figure 11: The results of Lax-Wendroff Scheme for $\lambda = 1.25$. The stability condition is not satisfied, so the waveform blows up.

4 vNSA of FDM Schemes for the Heat Equation

$$\frac{\partial f}{\partial t} = \alpha \frac{\partial^2 f}{\partial x^2}$$

This is the one–dimensional Heat Equation, a fundamental parabolic PDE describing the diffusion of heat (or any diffusive quantity) through a medium with diffusivity α.

A variety of finite difference methods (FDMs) can be used to numerically solve this PDE, including the Forward Time–Centered Space (FTCS) Scheme, the Backward Euler (Implicit) Scheme, the Crank–Nicolson Scheme, and the general θ–method.

4.1 Choice of Computational Modelling

For all analyses, the one-dimensional Heat Equation

$$f_t = \alpha f_{xx}$$

was used with diffusivity $\alpha = 1.$ for sake of simplicity. The spatial discretization Δx and total simulation time ($T = 10$) were kept fixed for all schemes. The diffusion stability parameter

$$r = \frac{\alpha \, \Delta t}{\Delta x^2}$$

was varied by adjusting the number of time steps n_t (and therefore Δt).

We used the same initial conditions (**Sine wave, Gaussian modulated Sine Wave, Step Function**) to study the behaviour of the various FDMs

4.2 vNSA of the FTCS Scheme

FTCS discretization uses a forward difference in time and a centered second difference in space:

$$\frac{f_j^{n+1} - f_j^n}{\Delta t} = \alpha \left(\frac{f_{j+1}^n - 2f_j^n + f_{j-1}^n}{\Delta x^2} \right).$$

Rewriting, the update formula is

$$f_j^{n+1} = f_j^n + r \left(f_{j+1}^n - 2f_j^n + f_{j-1}^n \right), \qquad r = \frac{\alpha \Delta t}{\Delta x^2}.$$

Substituting error ϵ_j^n gives

$$\epsilon_j^{n+1} = \epsilon_j^n + r \left(\epsilon_{j+1}^n - 2\epsilon_j^n + \epsilon_{j-1}^n \right).$$

Substituting our Fourier mode error, $\epsilon_j^n = (G_k)^n e^{ikj\Delta x}$:

$$(G_k)^{n+1} e^{ikj\Delta x} = (G_k)^n e^{ikj\Delta x} + r\Big[(G_k)^n e^{ik(j+1)\Delta x} - 2(G_k)^n e^{ikj\Delta x} + (G_k)^n e^{ik(j-1)\Delta x} \Big].$$

Dividing by $(G_k)^n e^{ikj\Delta x}$:

$$G_k = 1 + r \left(e^{ik\Delta x} - 2 + e^{-ik\Delta x} \right).$$

Using $e^{i\theta} + e^{-i\theta} = 2\cos\theta$:

$$G_k = 1 + 2r \left(\cos(k\Delta x) - 1 \right).$$

Dissipation (Amplitude Error)

Let $\theta = k\Delta x$. Then

$$G_k = 1 + 2r(\cos\theta - 1) = 1 - 4r \sin^2 \frac{\theta}{2}.$$

Thus G_k is purely real, and the magnitude is

$$|G_k| = \left| 1 - 4r \sin^2 \frac{\theta}{2} \right|.$$

Since $\sin^2(\theta/2) \in [0, 1]$, stability requires

$$-1 \le 1 - 4r \le 1 \quad \Rightarrow \quad 0 \le r \le \frac{1}{2}. \qquad (\star)$$

For $0 < r < \frac{1}{2}$,

$$|G_k| < 1,$$

so all Fourier modes decay in amplitude. FTCS is therefore dissipative.

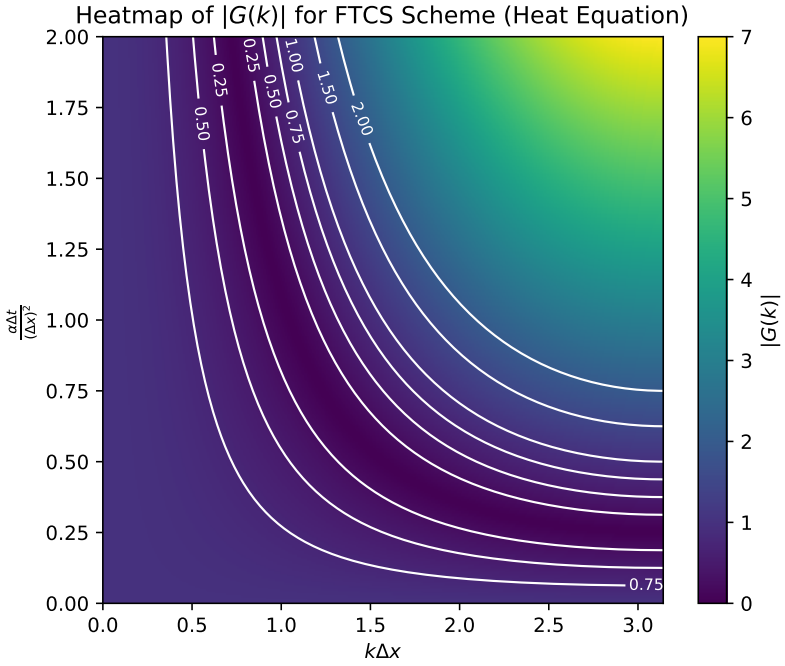


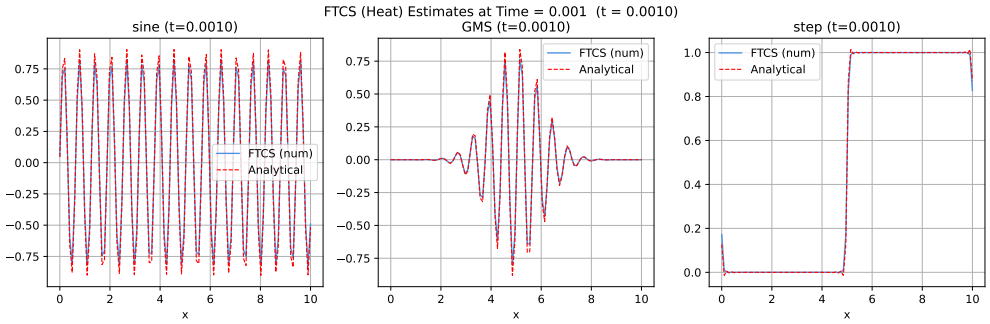
Figure 12: Variation of $|G|$ over different values of $k\Delta x$ and $r = \alpha\frac{\Delta t}{\Delta x^2}$ for FTCS Scheme.

Dispersion (Phase Error)

Because G_k is always real,

$$\arg(G_k) = 0 \quad \text{or} \quad \pi.$$

Thus, FTCS introduces **no phase shift**: it is nondispersive. All errors arise from amplitude decay (diffusion), not from incorrect wave speeds.



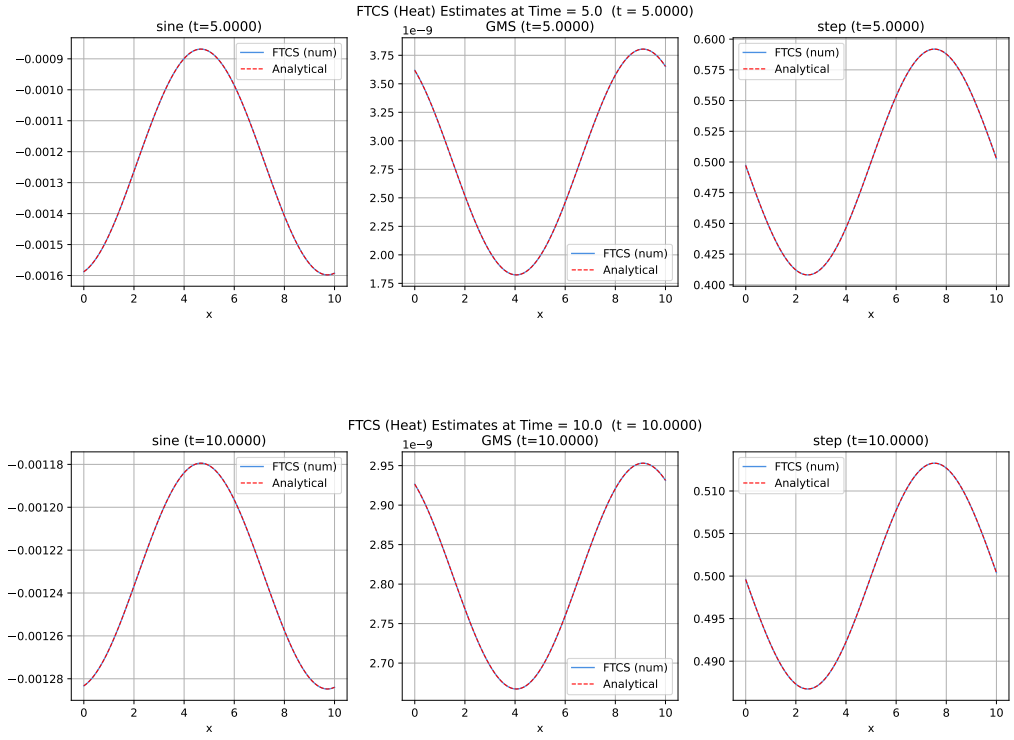


Figure 13: The Evolution of the solution with time using FTCS Scheme ($r = 0.1$)

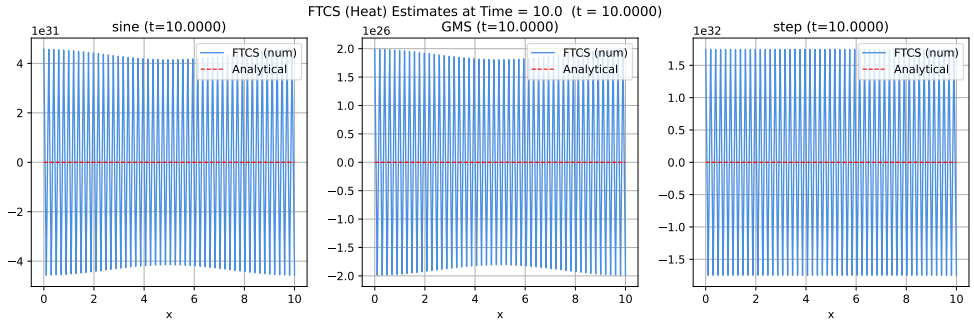


Figure 14: The results of FTCS Scheme with $r = 0.51$ values. The result blows up when $r > 0.5$

4.3 vNSA of the BTCS Scheme

For the Heat Equation

$$f_t = \alpha f_{xx},$$

the Backward Euler discretization uses a backward difference in time and a centered second difference in space, all evaluated at the new time level:

$$\frac{f_j^{n+1} - f_j^n}{\Delta t} = \alpha \left(\frac{f_{j+1}^{n+1} - 2f_j^{n+1} + f_{j-1}^{n+1}}{\Delta x^2} \right).$$

Rewriting, the update equation becomes

$$f_j^{n+1} = f_j^n + r\left(f_{j+1}^{n+1} - 2f_j^{n+1} + f_{j-1}^{n+1}\right), \qquad r = \frac{\alpha \Delta t}{\Delta x^2}.$$

Substituting error ϵ_j^n yields

$$\epsilon_j^{n+1} = \epsilon_j^n + r\left(\epsilon_{j+1}^{n+1} - 2\epsilon_j^{n+1} + \epsilon_{j-1}^{n+1}\right).$$

Substituting our Fourier mode error, $\epsilon_j^n = (G_k)^n e^{ikj\Delta x}$:

$$(G_k)^{n+1} e^{ikj\Delta x} = (G_k)^n e^{ikj\Delta x} + r\Big[(G_k)^{n+1} e^{ik(j+1)\Delta x} - 2(G_k)^{n+1} e^{ikj\Delta x} + (G_k)^{n+1} e^{ik(j-1)\Delta x}\Big].$$

Dividing by $(G_k)^n e^{ikj\Delta x}$:

$$G_k = 1 + r\, G_k\left(e^{ik\Delta x} - 2 + e^{-ik\Delta x}\right).$$

Using $e^{i\theta} + e^{-i\theta} = 2\cos\theta$:

$$G_k = 1 + 2rG_k\left(\cos(k\Delta x) - 1\right).$$

Solving for G_k:

$$G_k = \frac{1}{1 + 2r(1 - \cos(k\Delta x))} = \frac{1}{1 + 4r\sin^2\frac{k\Delta x}{2}}.$$

Dissipation (Amplitude Error)

The growth factor is purely real and satisfies

$$G_k = \frac{1}{1 + 4r\sin^2(\frac{k\Delta x}{2})}, \qquad 0 < G_k \leq 1.$$

Thus,

$$|G_k| = G_k < 1 \quad \text{for all } k \neq 0,$$

so all Fourier modes decay in amplitude. Backward Euler is therefore strongly dissipative, with the highest wavenumbers decaying the fastest.

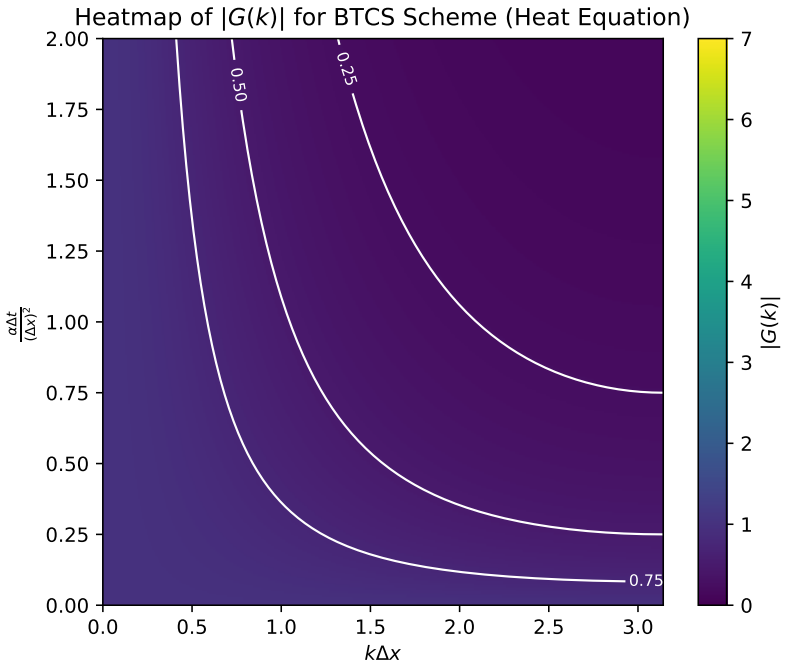


Figure 15: Variation of $|G|$ over different values of $k\Delta x$ and $r = \alpha \frac{\Delta t}{\Delta x^2}$ for BTCS Scheme.

Dispersion (Phase Error)

Since G_k is real and positive,

$$\arg(G_k) = 0,$$

and the scheme introduces no phase shift. Backward Euler is therefore nondispersive: all numerical error arises from excessive diffusion, not from incorrect wave propagation.

Stability

Because

$$0 < G_k \leq 1 \quad \text{for all } r > 0,$$

the scheme is unconditionally stable. There is no restriction on Δt or r for stability:

Implicit Euler is unconditionally stable.

4.4 vNSA of the Crank–Nicolson Scheme

Crank–Nicolson method averages the spatial second derivative between time levels n and $n+1$:

$$\frac{f_j^{n+1} - f_j^n}{\Delta t} = \alpha \frac{1}{2} \left(\frac{f_{j+1}^n - 2f_j^n + f_{j-1}^n}{\Delta x^2} + \frac{f_{j+1}^{n+1} - 2f_j^{n+1} + f_{j-1}^{n+1}}{\Delta x^2} \right).$$

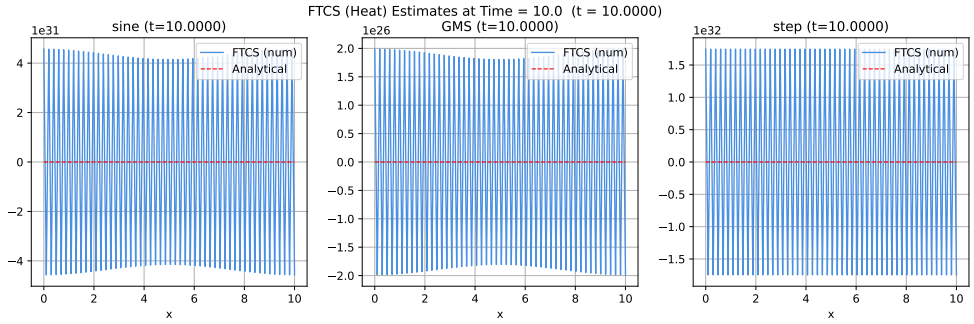


Figure 16: The results of running the BTCS Scheme. Notice the the solution is slightly 'off' due to the dissipative nature of the method.

Rewriting,

$$f_j^{n+1} = f_j^n + \frac{r}{2}\left(f_{j+1}^n - 2f_j^n + f_{j-1}^n\right) + \frac{r}{2}\left(f_{j+1}^{n+1} - 2f_j^{n+1} + f_{j-1}^{n+1}\right), \qquad r = \frac{\alpha\Delta t}{\Delta x^2}.$$

Substituting error ϵ_j^n:

$$\epsilon_j^{n+1} = \epsilon_j^n + \frac{r}{2}\left(\epsilon_{j+1}^n - 2\epsilon_j^n + \epsilon_{j-1}^n\right) + \frac{r}{2}\left(\epsilon_{j+1}^{n+1} - 2\epsilon_j^{n+1} + \epsilon_{j-1}^{n+1}\right).$$

Substituting our Fourier mode error, $\epsilon_j^n = (G_k)^n e^{ikj\Delta x}$:

$$(G_k)^{n+1}e^{ikj\Delta x} = (G_k)^n e^{ikj\Delta x} + \frac{r}{2}\Big[(G_k)^n e^{ik(j+1)\Delta x} - 2(G_k)^n e^{ikj\Delta x} + (G_k)^n e^{ik(j-1)\Delta x}\Big]$$
$$+\frac{r}{2}\Big[(G_k)^{n+1} e^{ik(j+1)\Delta x} - 2(G_k)^{n+1} e^{ikj\Delta x} + (G_k)^{n+1} e^{ik(j-1)\Delta x}\Big].$$

Dividing by $(G_k)^n e^{ikj\Delta x}$:

$$G_k = 1 + \frac{r}{2}\left(e^{ik\Delta x} - 2 + e^{-ik\Delta x}\right) + \frac{r}{2}G_k\left(e^{ik\Delta x} - 2 + e^{-ik\Delta x}\right).$$

Using $e^{i\theta} + e^{-i\theta} = 2\cos\theta$:

$$G_k = 1 + r(\cos(k\Delta x) - 1) + rG_k(\cos(k\Delta x) - 1).$$

Solving for G_k:

$$G_k\big[\,1 - r(\cos\theta - 1)\,\big] = 1 + r(\cos\theta - 1), \qquad \theta = k\Delta x.$$

Hence,

$$G_k = \frac{1 + r(\cos\theta - 1)}{1 - r(\cos\theta - 1)}.$$

Using $\cos\theta - 1 = -2\sin^2(\theta/2)$ gives the standard form:

$$G_k = \frac{1 - 2r\sin^2(\frac{\theta}{2})}{1 + 2r\sin^2(\frac{\theta}{2})}.$$

Dissipation (Amplitude Error)

Since both numerator and denominator are real,

$$|G_k| = \left|\frac{1 - 2r\sin^2(\frac{\theta}{2})}{1 + 2r\sin^2(\frac{\theta}{2})}\right|.$$

For all $r > 0$ and all θ,

$$|G_k| \leq 1.$$

Low-frequency modes (small θ) satisfy

$$|G_k| \approx 1 - 4r\sin^2(\tfrac{\theta}{2}) + \mathcal{O}(r^2),$$

so dissipation is weak. High frequencies are smoothly damped but not as aggressively as in Backward Euler.

Thus Crank–Nicolson is mildly dissipative.

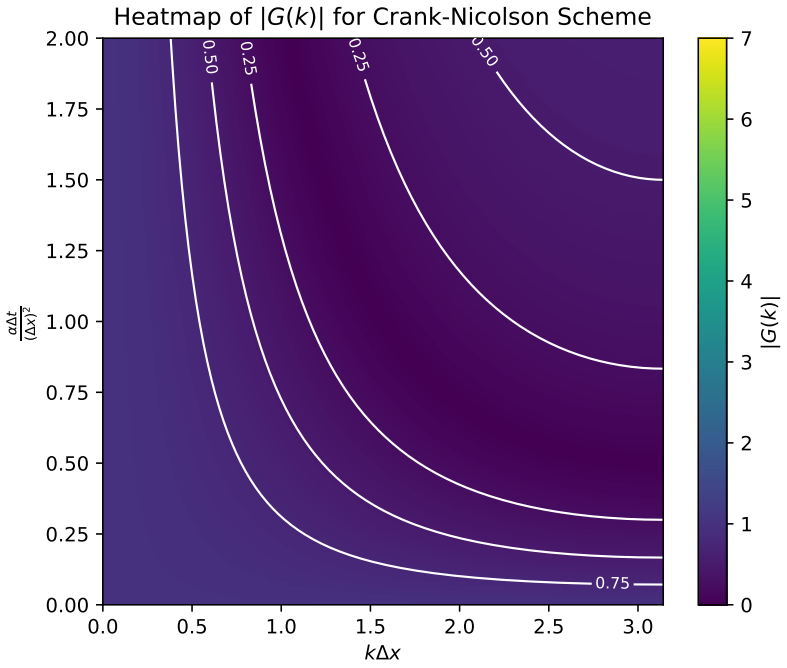


Figure 17: Variation of $|G|$ over different values of $k\Delta x$ and $r = \alpha\frac{\Delta t}{\Delta x^2}$ for Crank-Nicolson Scheme. Note that the value never crosses 1.

Dispersion (Phase Error)

Because G_k is real,

$$\arg(G_k) = 0 \quad \text{or} \quad \pi,$$

so the scheme is nondispersive. No phase shift is introduced; diffusion error is purely amplitude-based.

Stability

$$|G_k| = \left| \frac{1 - 2r\sin^2(\frac{\theta}{2})}{1 + 2r\sin^2(\frac{\theta}{2})} \right| \leq 1 \quad \forall\, r > 0,$$

Crank–Nicolson is unconditionally stable.

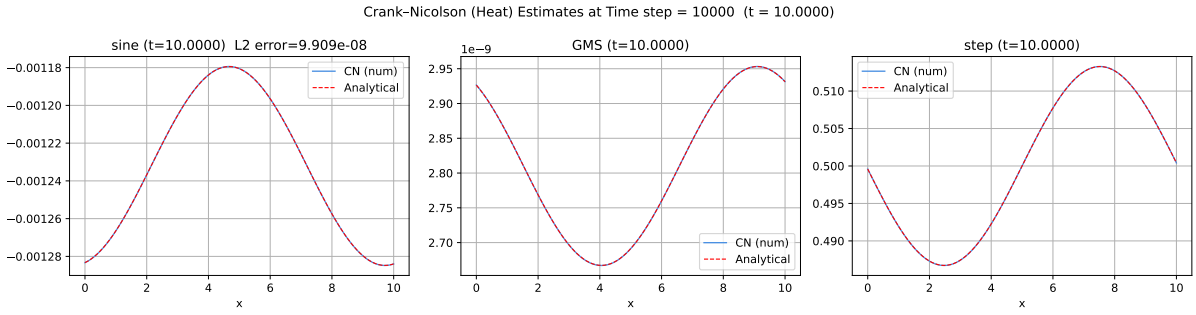


Figure 18: The results obtained from Crank-Nicolson Scheme

5 Conclusion

Using von Neumann Stability Analysis, we examined the behaviour of several finite difference schemes applied to the Linear Advection and Heat Equations. Analysis of the corresponding amplification factors provided clear insight into the stability criteria, as well as the dissipative and dispersive properties characteristic of each method.

6 References

R. J. LeVeque, *Finite Difference Methods for Ordinary and Partial Differential Equations*

R. R. Rosales, https://math.mit.edu/classes/18.300/Notes/Notes_vNSA.pdf

K. K. Makwana, *Time Dependent PDEs - Computational Physics*

6.1 GitHub

Here's my repository on GitHub. The code used to implement the following methods, along with all the plots, will be found in here.

https://github.com/Skiddy05/von-Neumann-Analysis-of-FDMs-for-PDEs

(The report is also a part of the repo, so don't enter an infinte recursion)