Noregs miljø- og biovitskaplege universitet

The Stability of Different Numerical Schemes

Candidate: 15
Faculty of Science and Technology, Norwegian University of Life Sciences, 1432 Ås, Norway

This is a paper done in INF305 (Scientific Computing) at The Norwegian University of Life Sciences. This is a course mostly focusing on Finite Volume Schemes and numerical calculations.

I. INTRODUCTION

Numerical methods are crucial for fields such as fluid dynamics and electromagnetism. The conservation laws often take form of partial-differential equations with no way of finding their exact solution.[1] Rather than abandoning these problems, we turn to numerical approximations by implementing finite differences, finite volumes or finite elements.

In this report, we compare the stability of different finite volume schemes when applied to a simple one-dimensional linear advection equation. We are investigating seven different numerical schemes; Backward-Euler, One-Sided (both left and right), Lax-Friedrichs, Leapfrog, Lax-Wendroff and Beam-Warming. The goal is to figure out when these methods are stable, and what might trigger them to fail.

Unstable numerical schemes can produce errors that grow exponentially, and essentially making our simulations worthless. The opposite would be too dissipative methods, that would smooth out important features in the solution. By comparing these schemes, we should gain some insight in what methods work best.

I'll be using the L2-stability theorem to understand how the CFL condition affects the stability and therefore also the accuracy of each scheme. Given knowledge obtained throughout this course and other literature I expect that the higher-order methods will give more accurate results, but might be prone to instability.

II. LITERATURE REVIEW

This paper is based on both lectures from INF305 at NMBU and "Numerical Methods for Conservation Laws" by Randall J. LeVeque and their chapter on Numerical methods for Linear Equations (chapter 10).

III. THEORY

When analyzing numerical schemes, there are three core concepts; consistency, stability and convergence. Consistency ensures that the scheme accurately approximates the underlying differential equation, while stability ensures that local errors do not grow uncontrollably during computation. Together they make up convergence, which means that the numerical solution approaches the exact solution. This relationship is discussed in the Lax-Richtmyer Equivalence theorem [2], which states that both stability and consistency is needed for convergence.

Finite Volume Scheme

We start from the conservation law in one space dimension.

$$u_t + f(u)_x = 0, \qquad f(u) = a\,u. \tag{1}$$

Here, $u(x,t)$ represents some conserved quantity, for example a fluid. We assign our piecewise-constant approximation the value U_j^n in the (j,n) cell

$$u_h(x,t) \;=\; u_j^n, \qquad (x,t) \in [\,x_{j-\frac{1}{2}},\, x_{j+\frac{1}{2}}) \times [\,t^n,\, t^{n+1})$$

which gives us the integral:

$$\int_{t^n}^{t^{n+1}} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} \big(u_t + f(u)_x\big)\, dx\, dt = 0.$$

By the Fundamental Theorem in t and x:

$$\int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} u(x,t^{n+1})\, dx \;-\; \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} u(x,t^n)\, dx$$
$$+ \int_{t^n}^{t^{n+1}} \Big[f\big(u(x_{j+\frac{1}{2}},t)\big) - f\big(u(x_{j-\frac{1}{2}},t)\big)\Big]\, dt = 0.$$

Define the cell average and the time-averaged fluxes:

$$U_j^n \;:=\; \frac{1}{\Delta x} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} u(x,t^n)\, dx$$

$$F^n_{j+\frac{1}{2}} \;:=\; \frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} f\big(u(x_{j+\frac{1}{2}},t)\big)\, dt.$$

Divide by Δx and rearrange:

$$U_j^{n+1} - U_j^n \;=\; -\frac{\Delta t}{\Delta x}\Big(F^n_{j+\frac{1}{2}} - F^n_{j-\frac{1}{2}}\Big).$$

But since we only know cell-averages U_j^n and U_{j+1}^n we can not compute the integral exactly, but we instead introduce numerical flux g to approximate a time-integral

by a function of two neighboring averages, for example, $F^n_{j+\frac{1}{2}} \approx g(U^n_j, U^n_{j+1})$. From now on we use the notation $\Delta t = k$ and $\Delta x = h$ to be consistent with the book. We then end up with this flux-difference scheme:

$$\boxed{U_j^{n+1} = U_j^n - \frac{k}{h}\Big[g\big(U_j^n,\, U_{j+1}^n\big) - g\big(U_{j-1}^n,\, U_j^n\big)\Big].} \quad (2)$$

Starting from the general flux-difference form (2) we must choose a numerical flux $g(U_L, U_R)$ that is consistent with the physical flux $f(U) = aU$. Different choices of g (upwind, Lax–Friedrichs, Lax–Wendroff, etc.) and of time-stepping (forward vs. backward Euler) then give the different numerical schemes.

$\mathbf{L^2}$ Stability

We work on a uniform grid $x_j = j\,h$, $t^n = n\,k$. Here we still use the h and k for spatial and time steps. For a grid function $v^n = (v_j^n)_j$, which represents our numerical solution at time step n, the discrete L^2 norm is given by

$$\|v^n\|_2 \;=\; \sqrt{h \sum_j (v_j^n)^2}\,.$$

A one-step linear scheme calculates the next time step, by using the values for the n^{th} timestep.

$$v_j^{n+1} \;=\; \sum_\ell C_\ell\, v_{j+\ell}^n,$$

where the C_ℓ are fixed weights that define how neighboring points influence the update. We say the scheme is L^2–stable if there is a constant M (independent of n, h, k) so that

$$\|v^n\|_2 \;\le\; M\,\|v^0\|_2 \quad \text{for all } n.$$

In practice, this stability condition makes sure that the numerical solution's size remains bounded throughout the calculation, preventing exponential error growth.

To check stability we insert a single Fourier mode

$$v_j^n = A^n e^{ij\xi},$$

where ξ is the wave number and A is the factor by which the amplitude changes in each time step. Plugging into the update gives

$$A^{n+1}e^{ij\xi} = \sum_\ell C_\ell\big(A^n e^{i(j+\ell)\xi}\big) \implies A^{n+1} = h(\xi)\,A^n,$$

This gives us the amplification factor $h(\xi)$

$$h(\xi) \;=\; \sum_\ell C_\ell\, e^{i\ell\xi}.$$

The scheme is stable exactly when

$$|h(\xi)| \;\le\; 1 \quad \text{for every } \xi.$$

Example: For the upwind scheme applied to the advection equation (1), you get

$$v_j^{n+1} = (1-\nu)\,v_j^n + \nu\,v_{j-1}^n, \qquad \nu = \frac{a\,k}{h}.$$

Its amplification factor is

$$h(\xi) = (1-\nu) + \nu\,e^{-i\xi}, \quad |h(\xi)|^2 = 1 - 4\,\nu\,(1-\nu)\,\sin^2\!\big(\tfrac{\xi}{2}\big).$$

Requiring $|h(\xi)| \le 1$ for all ξ gives

$$0 \;\le\; \nu \;\le\; 1.$$

This inequality,

$$\frac{|a|\,k}{h} \le 1, \quad (3)$$

is known as the CFL condition. It means that in one time step information carried by the PDE moves at most one grid cell.

Convergence

If the scheme is consistent (its local truncation error is $O(k^p + h^q)$) and it is L^2–stable, then by the Lax–Richtmyer theorem the numerical solution converges to the true solution as $h, k \to 0$, with global error $O(k^p + h^q)$. [2]

Finite Volume Schemes investigated in this paper

1. Backward-Euler

With backward-Euler in time the general flux–difference scheme becomes

$$U_j^{n+1} = U_j^n - \frac{k}{h}\Big[g\big(U_j^{n+1}, U_{j+1}^{n+1}\big) \;-\; g\big(U_{j-1}^{n+1}, U_j^{n+1}\big)\Big].$$

Choosing the centered flux

$$g(U_L, U_R) = \frac{A}{2}\,(U_L + U_R),$$

we compute

$$\begin{aligned} &g\big(U_j^{n+1}, U_{j+1}^{n+1}\big) - g\big(U_{j-1}^{n+1}, U_j^{n+1}\big) \\ &= \tfrac{A}{2}\big[(U_j^{n+1} + U_{j+1}^{n+1}) - (U_{j-1}^{n+1} + U_j^{n+1})\big] \\ &= \tfrac{A}{2}\,(U_{j+1}^{n+1} - U_{j-1}^{n+1}). \end{aligned}$$

Hence the update is

$$U_j^{n+1} = U_j^n - \frac{A\,k}{2h}\,(U_{j+1}^{n+1} - U_{j-1}^{n+1}),$$

2. Upwind

Starting from the general flux–difference form (2), we then choose a numerical flux that respects the true direction of information propagation in the PDE. For advection, characteristics travel at speed a, so the *upwind* flux simply samples the solution from the upstream side of each interface:

$$g(U_L, U_R) = \begin{cases} a\,U_L, & a > 0, \\ a\,U_R, & a < 0. \end{cases}$$

Hence one obtains the two one-sided schemes:
One-Sided (Left), $a > 0$:

$$U_j^{n+1} = U_j^n - \frac{k}{h}\big(A\,U_j^n - A\,U_{j-1}^n\big) = U_j^n - \frac{A\,k}{h}\big(U_j^n - U_{j-1}^n\big).$$

[6]
One-Sided (Right), $a < 0$:

$$U_j^{n+1} = U_j^n - \frac{k}{h}\big(A\,U_{j+1}^n - A\,U_j^n\big) = U_j^n - \frac{A\,k}{h}\big(U_{j+1}^n - U_j^n\big).$$

[6]

3. Lax–Friedrichs

For Lax–Friedrichs we choose the numerical flux to be the centered average of the physical flux plus a dissipation term:

$$g_{\mathrm{LF}}(U_L, U_R) = \frac{1}{2}\,A\,(U_L + U_R) \;-\; \frac{h}{2k}\,(U_R - U_L).$$

Substitute into (2) and split the algebra:

$$\begin{aligned} &g_{\mathrm{LF}}(U_j, U_{j+1}) - g_{\mathrm{LF}}(U_{j-1}, U_j) \\ &\quad= \underbrace{\frac{A}{2}(U_j + U_{j+1}) - \frac{A}{2}(U_{j-1} + U_j)}_{=\frac{A}{2}(U_{j+1} - U_{j-1})} \\ &\qquad- \underbrace{\frac{h}{2k}(U_{j+1} - U_j) - \frac{h}{2k}(U_j - U_{j-1})}_{=\frac{h}{2k}(U_{j+1} - 2U_j + U_{j-1})} \\ &\quad= \frac{A}{2}(U_{j+1} - U_{j-1}) \;-\; \frac{h}{2k}(U_{j+1} - 2U_j + U_{j-1}). \end{aligned}$$

Hence

$$\begin{aligned} U_j^{n+1} &= U_j^n - \frac{k}{h}\Big[\tfrac{A}{2}(U_{j+1} - U_{j-1}) - \tfrac{h}{2k}(U_{j+1} - 2U_j + U_{j-1})\Big] \\ &= \tfrac{1}{2}\,(U_{j-1}^n + U_{j+1}^n) \;-\; \frac{A\,k}{2h}(U_{j+1}^n - U_{j-1}^n), \end{aligned}$$

[6]

4. Leapfrog

From the central-in-time and central-in-space discretization of $U_t + A\,U_x = 0$,

$$\frac{U_j^{n+1} - U_j^{n-1}}{2k} \;+\; A\,\frac{U_{j+1}^n - U_{j-1}^n}{2h} = 0,$$

we get the two-step leapfrog update

$$U_j^{n+1} = U_j^{n-1} - \frac{k}{2h}\,A\,(U_{j+1}^n - U_{j-1}^n).$$

5. Lax–Wendroff

To regain second-order accuracy, Lax–Wendroff performs a Taylor expansion in time and replaces time derivatives via the PDE:

$$U_j^{n+1} = U_j^n - \frac{A\,k}{2h}(U_{j+1}^n - U_{j-1}^n) + \frac{A^2k^2}{2h^2}(U_{j+1}^n - 2U_j^n + U_{j-1}^n).$$

This centered scheme is non-dissipative and second-order accurate, but may exhibit oscillations near discontinuities. [6]

6. Beam-Warming

Start from the Taylor expansion in time:

$$U_j^{n+1} = U_j^n + k\,U_t + \tfrac{k^2}{2}\,U_{tt} + \mathcal{O}(k^3).$$

Use $U_t = -A\,U_x$ and $U_{tt} = A^2\,U_{xx}$:

$$U_j^{n+1} = U_j^n + k\big(-A\,U_x\big) + \tfrac{k^2}{2}\big(A^2\,U_{xx}\big) + \mathcal{O}(k^3).$$

Approximate the derivatives with an upwind bias:

$$U_x \approx \frac{3U_j^n - 4U_{j-1}^n + U_{j-2}^n}{2h}, \qquad U_{xx} \approx \frac{U_j^n - 2U_{j-1}^n + U_{j-2}^n}{h^2}.$$

Putting it all together,

$$\begin{aligned} U_j^{n+1} \approx U_j^n &- \frac{k}{2h}\,A\,(3U_j^n - 4U_{j-1}^n + U_{j-2}^n) \\ &+ \frac{k^2}{2h^2}\,A^2\,(U_j^n - 2U_{j-1}^n + U_{j-2}^n), \end{aligned}$$

IV. METHODS

Given the CFL condition (3) we want to investigate how the different schemes behave with CFL values around 1. To check this we have chosen $CFL = 0.9, 1.0, 1.1$ and will plot their approximations. I have chosen a Gaussian Pulse to approximate, because of its known analytical solution. It should also demonstrate both diffusion and difficulty with sharp pulses well.

Name	Difference equation	Stencil
Backward Euler	$U_j^{n+1} = U_j^n - \frac{k}{2h}A(U_{j+1}^n - U_{j-1}^n)$	$\top$
One-sided (left)	$U_j^{n+1} = U_j^n - \frac{k}{h}A(U_j^n - U_{j-1}^n)$	$\vdash$
One-sided (right)	$U_j^{n+1} = U_j^n - \frac{k}{h}A(U_{j+1}^n - U_j^n)$	$\dashv$
Lax–Friedrichs	$U_j^{n+1} = \frac{1}{2}(U_{j-1}^n + U_{j+1}^n) - \frac{k}{2h}A(U_{j+1}^n - U_{j-1}^n)$	$\wedge$
Leapfrog	$U_j^{n+1} = U_j^{n-1} - \frac{k}{2h}A(U_{j+1}^n - U_{j-1}^n)$	$\Diamond$
Lax–Wendroff	$U_j^{n+1} = U_j^n - \frac{k}{2h}A(U_{j+1}^n - U_{j-1}^n) + \frac{k^2}{2h^2}A^2(U_{j+1}^n - 2U_j^n + U_{j-1}^n)$	$\vdash$
Beam–Warming	$U_j^{n+1} = U_j^n - \frac{k}{2h}A(3U_j^n - 4U_{j-1}^n + U_{j-2}^n) + \frac{k^2}{2h^2}A^2(U_j^n - 2U_{j-1}^n + U_{j-2}^n)$	$\vdash$

TABLE I. Table 10.1 from "Numerical Methods for Conservation Laws" by Randall J. LeVeque [5]

V. RESULTS

A. Backward–Euler

Backward-Euler is unconditionally stable, even for $CFL >= 1.0$. However, the trade-off is a lot a diffusion as you can see visualized. The Gaussian pulse widens and the peak height drops as the CFL increases.

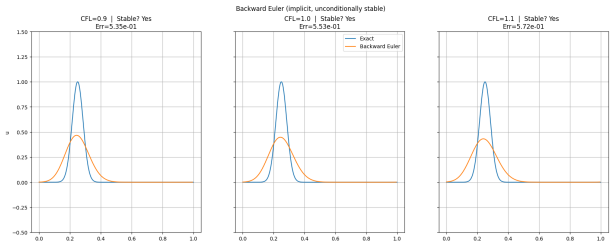

FIG. 1. Backward-Euler at different CFL values

B. One-Sided (Left)

The one-sided left (downwind) is unstable for any nonzero CFL. This is because it samples from the "wrong" side of each cell and violate the CFL restriction for advection.

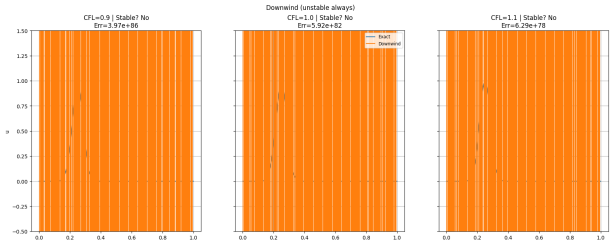

FIG. 2. One-Sided Left at different CFL values

C. One-Sided (Right)

The one-sided right (upwind) is stable as long as $CFL <= 1.0$. At $CFL = 0.9$ and 1.0 it approximates the solution quite good, but for $CFL >= 1.0$ it becomes unstable and the solution quickly degrades.

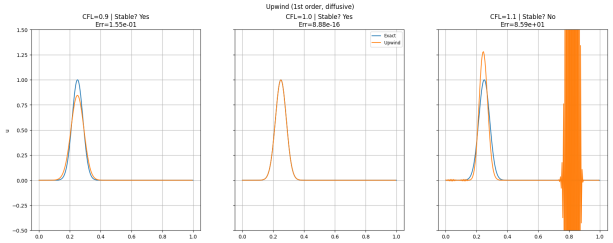

FIG. 3. One-sided right (upwind) at different CFL values

D. Lax–Friedrichs

Lax-Friedrichs is stable up to $CFL = 1.0$, but has more diffusion then the other numerical methods. This leads to the Gaussian pulse being significantly smaller than the exact solution. Beyond $CFL = 1.0$ it also breaks down and produces large errors. This is because it uses an averaging step which then acts like a large second-derivative term.

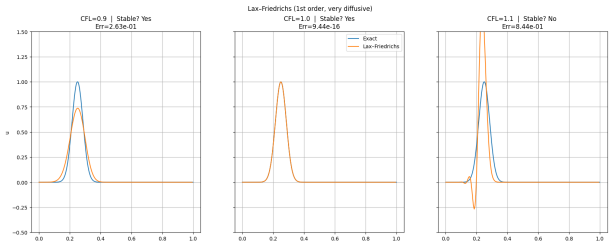

FIG. 4. Lax–Friedrichs scheme at different CFL values

E. Leapfrog

Leapfrog's update function use information from two time intervals in a centered way so that they never smooth out noise. Any high-frequency error stays and shows up as a zigzag shape alongside the exact solution. It fails at $CFL = 1$ because the zigzag mode also has unit amplification. This is because it is the only multi-step method, which means it computes the new solution value using several past time levels, U^n and U^{n-1}. See the section on the CFL condition and amplification value (3).

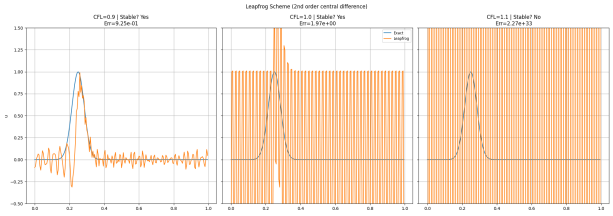

FIG. 5. Leapfrog scheme at different CFL values

F. Lax–Wendroff

Lax-Wendroff adds a second-order correction with a Taylor expansion, but has no real damping. The scheme is good for smooth waves, but because there is no dissipation we get oscillations (overshoots and undershoots) near sharp changes. This is called the Gibbs phenomena.[3][4] These ripples grow if you push the CFL limit.

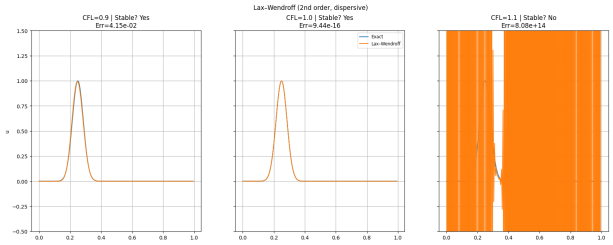

FIG. 6. Lax–Wendroff scheme at different CFL values

G. Beam–Warming

Beam-Warming's update consists of two pieces; a three-point upwind difference (the first term) that gives you a second-order approximation without the oscillations of a centered stencil, and a small Taylor expansion (second term) to get a more exact solution. Together this performs well, even for $CFL > 1.0$.

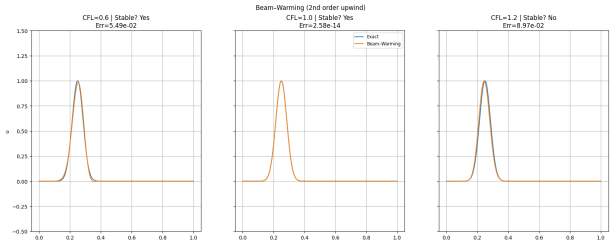

FIG. 7. Beam–Warming scheme at different CFL values

$$L^2 - errors$$

TABLE II. L_2-Error for each scheme at different CFL numbers

Scheme	CFL=0.9	CFL=1.0	CFL=1.1
Euler	5.35×10^{-1}	5.53×10^{-1}	5.72×10^{-1}
Downwind	3.97×10^{86}	5.92×10^{82}	6.29×10^{78}
Upwind	1.55×10^{-1}	8.88×10^{-16}	8.59×10^{1}
Lax–Friedrichs	2.63×10^{-1}	9.44×10^{-16}	8.44×10^{-1}
Leapfrog	9.25×10^{-1}	1.97×10^{0}	2.27×10^{33}
Lax–Wendroff	4.15×10^{-2}	9.44×10^{-16}	8.08×10^{14}
Beam–Warming	1.13×10^{-2}	2.58×10^{-14}	8.85×10^{-2}

In (II) we see that Beam-Warming and Lax-Friedrichs performs the best. Of the two, Beam-Warming performs the absolute best. While Lax-Wendroff performs better than Beam-Warming up until $CFL = 1$ it blows up after this. Even for a $CFL = 2.0$ Beam-Warming gets a L^2-error of $1.99 * 10^-2$, but blows up after this point.

VI. DISCUSSION AND CONCLUSION

In our experiments, we observed results consistent with our hypothesis. They show how the order of the equation, time-discretizations and CFL-number all effect both the stability and the consistency of the solution. The implicit method Backward-Euler is not affected by $CFL > 1.0$, but has a lot of diffusion. This happens because the implicit time discretization uses the unknown future state in the update, it inherently damps all modes and remains stable. This can be beneficial for robust simulation where you primary goal is stability, while sharp amplitudes can be neglected.

For the explicit first-order methods downwind is useless in this case (due to logical error), while Upwind and Lax-Friedrichs needs $CFL < 1.0$ for stability. Upwind fails a little before Lax-Friedrichs due to its dampening, but this does however effect the consistency a little. You should choose first-order methods when the need for robustness and simplicity outweigh accuracy.

The explicit second-order (Leapfrog, Lax-Wendroff, Beam-Warming) gives far better consistency and therefore have smaller global error, but are prone to oscillations (over- and undercompansations) at sharp edges. Here Leapfrog and Lax-Wendroff use different approaches. Leapfrog's centered-in-time-and-space stencil is non-dissipative, so any small checkerboard error remains undamped and leads to blow-up at $CFL = 1.0$. Lax–Wendroff restores second-order accuracy via a Taylor-expansion correction but without damping produces Gibbs oscillations near steep gradients and also fails when $CFL > 1$. Beam–Warming combines a three-point upwind bias with a small curvature correction, yielding the best compromise: sharp peak preservation

and stability for CFL ≤ 1. You should choose second-order methods when you need more accurate methods.

[1] Conservation law. https://en.wikipedia.org/wiki/Conservation_law, 2025.
[2] Lax equivalence theorem, 2025.
[3] David Gottlieb and ChiWang Shu. On the gibbs phenomenon and its resolution. *SIAM Review*, 39:644–668, 1997. "Originally found behind paywall. Used chatgpt to find this.".
[4] University of Oklahoma CFD2003 Lecture notes. Monotonicity of advection schemes, 2003.
[5] Randall J. LeVeque. *Numerical Methods for Conservation Laws*. Birkhäuser Basel, 2 edition, 1992. "Had to use chatGPT to get a link. Link from task description didnt work.".
[6] Randall J. LeVeque. *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press, 2004.