Chp 2)Finite Difference Approximations

By

Prof. Dinshaw S. Balsara

2.1) Introduction

We have seen that several PDEs have a *combination of hyperbolic, parabolic and elliptic terms*. Example : Navier-Stokes equations.

Many of the PDEs are also strongly non-linear. The non-linear aspect will be tackled in Chapters 4 and 5.

Even within the context of linear PDEs, *convergence to the physical solution* is not always guaranteed. Obtaining such guarantees is the topic of this Chapter.

We study how to *approximate PDEs on a mesh*. This is known as a Finite Difference Approximation (FDA).

We understand what makes an approximation ***consistent***.

Then we study ***stability*** of stiff source terms, linear parabolic equations and linear hyperbolic equations.

An interesting *deficiency* emerges for *linear hyperbolic equations*, which will only be resolved in the next Chapter.

Problems studied here are initial boundary value problems. They require specification of *boundary conditions*. We begin that study here.

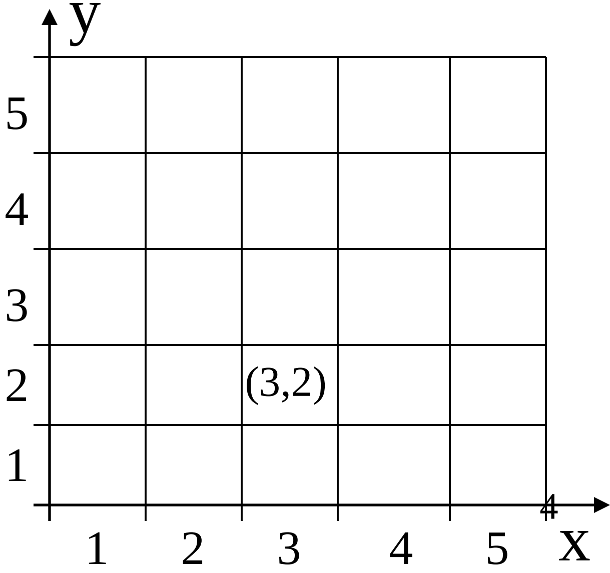
2.2) Meshes and Discretization on a Mesh

To solve problems on a computer we need to represent the physical data. This entails allocating storage, I.e computer memory, I.e. bytes in RAM, to represent the physical variables.

Say we have a rectangular physical problem. We can subdivide our *computational domain*, which covers physical space, to obtain a computational *mesh*. The subdivisions can be labeled to obtain *zones*. For instance, we can talk of the (3,2)th zone of a two-dimensional mesh.

We can then assign physical *data* to each of the zones of the mesh. Eg. For a fluid flow problem we would assign density, momentum density and energy density to each zone.

We expect that as the number of zones is increased by further subdivision of the mesh, the *accuracy* with which we represent the physical problem will increase. Eg., A 10X10 or 100X100 zone mesh would be more accurate than the 5X5 mesh shown in the figure.



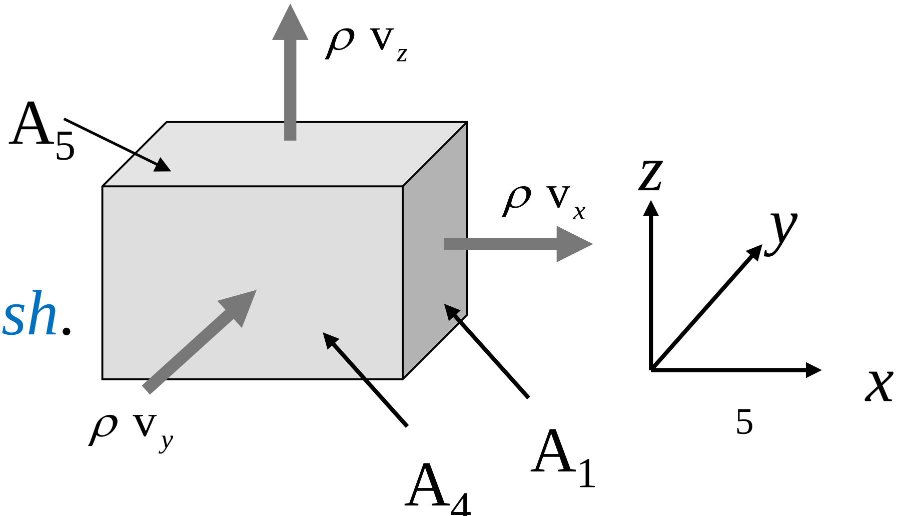
Notice that the fluid variables evolve in time in response to their own spatial gradients. This is often the case with most PDEs.

Question: So what makes the ***conservation form*** so special?
Answer: Gauss' Law.
Let's focus on the continuity equation and the figure below.

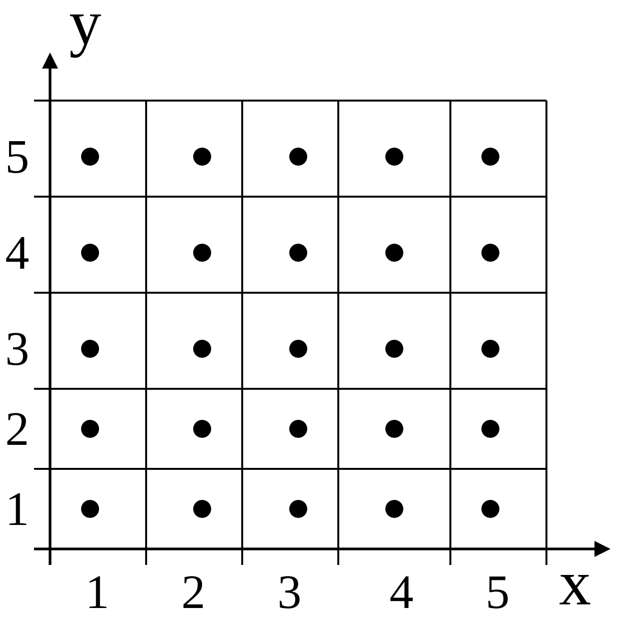
$$\iiint_V \left(\frac{\partial \rho}{\partial t} + \frac{\partial (\rho \, \mathrm{v}_x)}{\partial x} + \frac{\partial (\rho \, \mathrm{v}_y)}{\partial y} + \frac{\partial (\rho \, \mathrm{v}_z)}{\partial z} \right) dx\, dy\, dz = 0 \quad \Rightarrow$$

$$\frac{\partial}{\partial t} \iiint_V \rho \, dx\, dy\, dz + \iint_{A_1} \rho \, \mathrm{v}_x \, dy\, dz - \iint_{A_2} \rho \, \mathrm{v}_x \, dy\, dz + \iint_{A_3} \rho \, \mathrm{v}_y \, dx\, dz - \iint_{A_4} \rho \, \mathrm{v}_y \, dx\, dz$$
$$+ \iint_{A_5} \rho \, \mathrm{v}_z \, dx\, dy - \iint_{A_6} \rho \, \mathrm{v}_z \, dx\, dy = 0$$

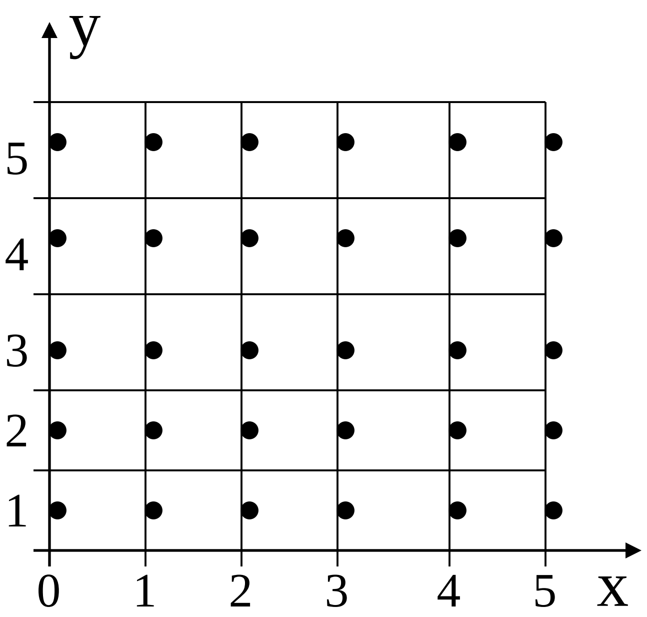
When ***discontinuities/shocks*** are present, we have no hope of predicting the flow structure inside a *zone* in our computational *mesh*. However, the ***conservation form remains valid***!



Data can be allocated (*collocated*) at different locations on the mesh; some examples, along with popular *indexing schemes*, are shown below.

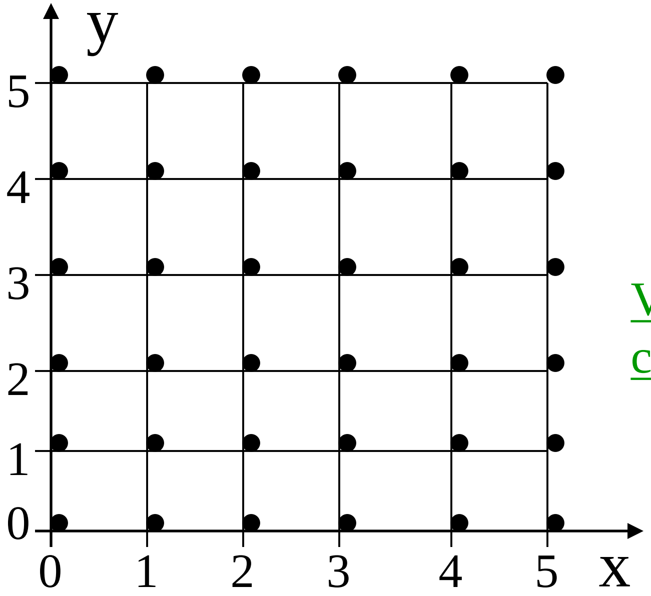


Zone-centered collocation of data



x-Face-centered collocation of data

● Denotes locations where physical data is stored (collocated).



Vertex-centered collocation of data

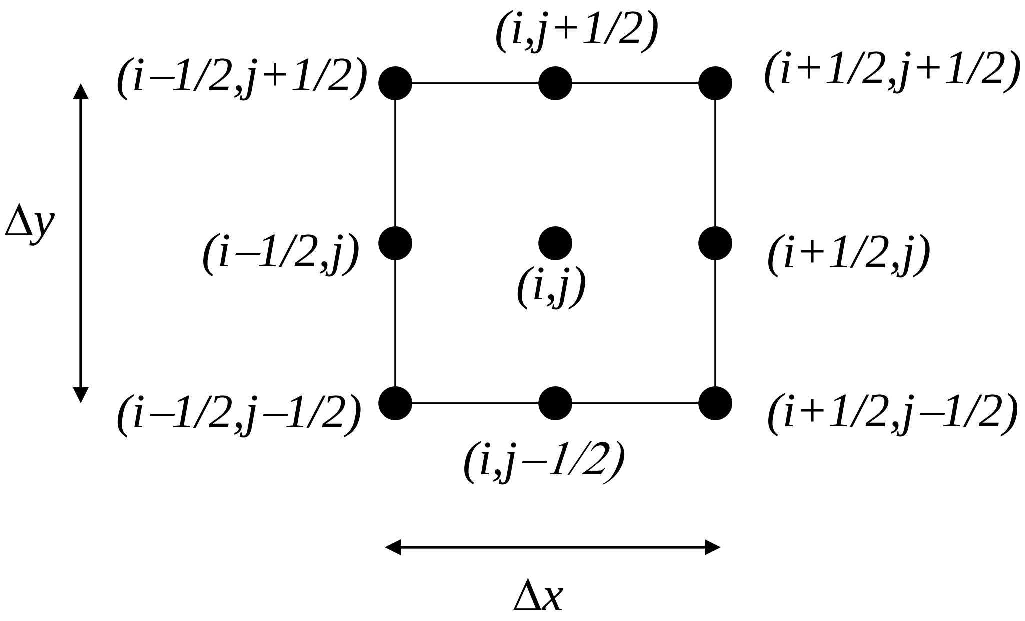
Question: What would the multi-dimensional array declarations be for the meshes shown above?

Question: Which of the collocations above are favored for a) Euler equations, b) MHD, c) Maxwell's equations and d) for the Poisson equation? Give reasons for your answers.

Question: A 3d mesh permits zone-centered, face-centered, edge-centered and vertex-centered collocations. Draw a zone of a 3d mesh and indicate the locations of these collocations. How many different face-centered and edge-centered collocations can you find?

The standard notation for labeling various locations in a zone are shown:

<u>Question</u>: For which different scientific problems would you use these different locations on a mesh?



There are two standard ways of thinking about the solution techniques:

Finite Difference Formulation: Each variable is a point value defined at whatever location it is defined. (*Pros*: Easier, Faster. *Cons*: Not so general; doesn't extend to complex geometries; can't do mesh refinement.)

Finite Volume Formulation : Each variable represents a spatial (or temporal) average over some portion of the zone. (*Pros*: Extends to complex geometries, can do adaptive mesh refinement. *Cons*: Slightly slower)

Example: Obtain the FD formulation for $U_t + F_x + G_y = 0$.

We go from time t^n to $t^n + \Delta t$ (see fig. on previous page)

$$\boxed{U_{i,j}^{n+1} = U_{i,j}^{n} - \frac{\Delta t}{\Delta x}\left(F_{i+1/2,j}^{n+1/2} - F_{i-1/2,j}^{n+1/2}\right) - \frac{\Delta t}{\Delta y}\left(G_{i,j+1/2}^{n+1/2} - G_{i,j-1/2}^{n+1/2}\right)}$$

Example: Obtain the FV formulation for the same PDE. Now we have to integrate the PDE over the space-time domain $\left[-\Delta x/2, \Delta x/2\right] \times \left[-\Delta y/2, \Delta y/2\right] \times \left[t^n, t^n + \ \Delta t\right]$

$$\boxed{\overline{U}_{i,j}^{n+1} = \overline{U}_{i,j}^{n} - \frac{\Delta t}{\Delta x}\left(\overline{F}_{i+1/2,j}^{n+1/2} - \overline{F}_{i-1/2,j}^{n+1/2}\right) - \frac{\Delta t}{\Delta y}\left(\overline{G}_{i,j+1/2}^{n+1/2} - \overline{G}_{i,j-1/2}^{n+1/2}\right)}$$

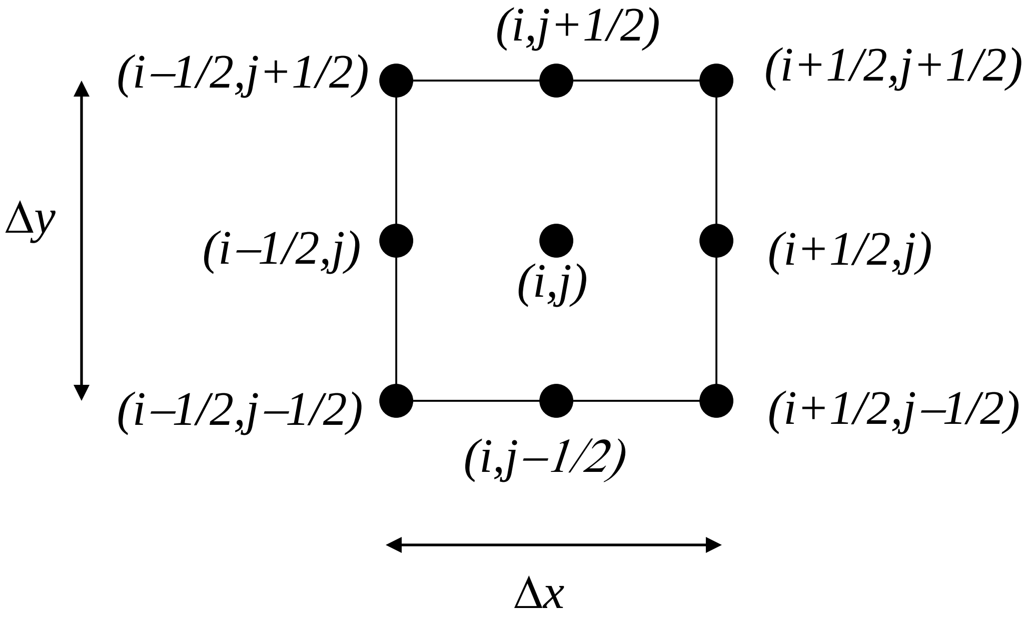
i.e. the solution is only available in terms of averages (Question: Why might that be desirable?):

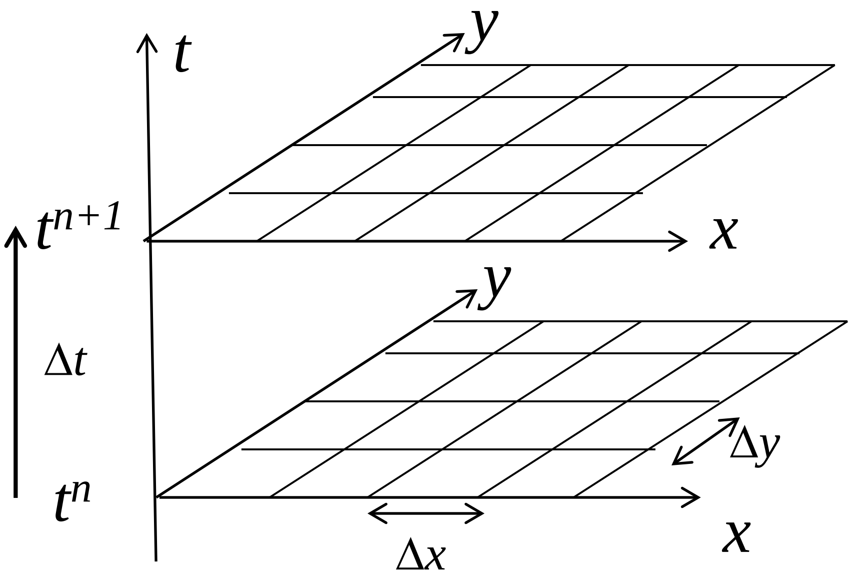
$$\overline{U}_{i,j}^{n} \equiv \frac{1}{\Delta x \ \Delta y} \int_{y=-\Delta y/2}^{y=\Delta y/2} \int_{x=-\Delta x/2}^{x=\Delta x/2} U\left(x, y, t^n\right) \ dx \ dy \quad ;$$

$$\overline{F}_{i+1/2,j}^{n+1/2} \equiv \frac{1}{\Delta t \ \Delta y} \int_{t=t^n}^{t=t^{n+1}} \int_{y=-\Delta y/2}^{y=\Delta y/2} F\left(\Delta x/2, y, t\right) \ dy \ dt \quad ; \quad \overline{F}_{i-1/2,j}^{n+1/2} \equiv \frac{1}{\Delta t \ \Delta y} \int_{t=t^n}^{t=t^{n+1}} \int_{y=-\Delta y/2}^{y=\Delta y/2} F\left(-\Delta x/2, y, t\right) \ dy \ dt \quad ;$$

$$\overline{G}_{i,j+1/2}^{n+1/2} \equiv \frac{1}{\Delta t \ \Delta x} \int_{t=t^n}^{t=t^{n+1}} \int_{x=-\Delta x/2}^{x=\Delta x/2} G\left(x, \Delta y/2, t\right) \ dx \ dt \quad ; \quad \overline{G}_{i,j-1/2}^{n+1/2} \equiv \frac{1}{\Delta t \ \Delta x} \int_{t=t^n}^{t=t^{n+1}} \int_{x=-\Delta x/2}^{x=\Delta x/2} G\left(x, -\Delta y/2, t\right) \ dx \ dt$$

$$\mathrm{U}_t + \mathrm{F}_x + \mathrm{G}_y = 0$$





$$\mathrm{U}_{i,j}^{n+1} = \mathrm{U}_{i,j}^{n} - \frac{\Delta t}{\Delta x}\left(\mathrm{F}_{i+1/2,j}^{n+1/2} - \mathrm{F}_{i-1/2,j}^{n+1/2}\right) - \frac{\Delta t}{\Delta y}\left(\mathrm{G}_{i,j+1/2}^{n+1/2} - \mathrm{G}_{i,j-1/2}^{n+1/2}\right)$$

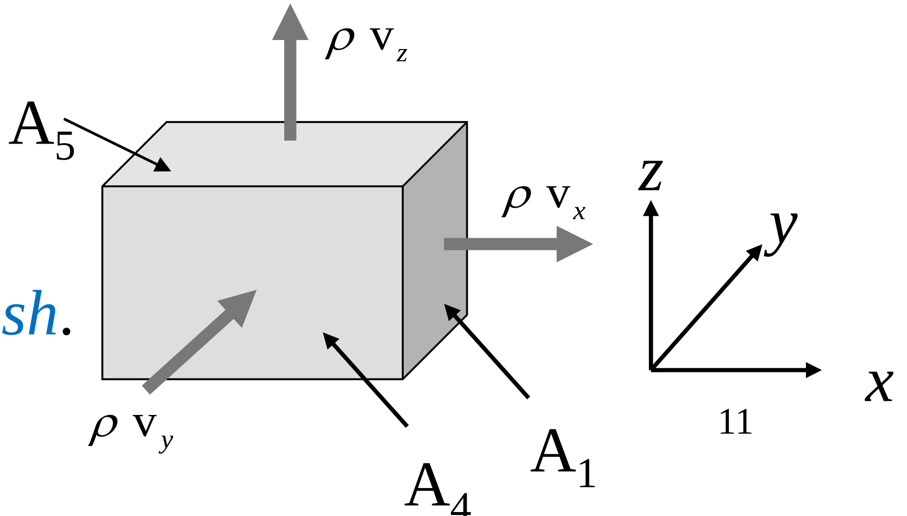
Notice that the fluid variables evolve in time in response to their own spatial gradients. This is often the case with most PDEs.

Question: So what makes the ***conservation form*** so special?
Answer: Gauss' Law.
Let's focus on the continuity equation and the figure below.

$$\iiint_{V} \left(\frac{\partial \rho}{\partial t} + \frac{\partial (\rho \, v_x)}{\partial x} + \frac{\partial (\rho \, v_y)}{\partial y} + \frac{\partial (\rho \, v_z)}{\partial z} \right) dx \, dy \, dz = 0 \qquad \Rightarrow$$

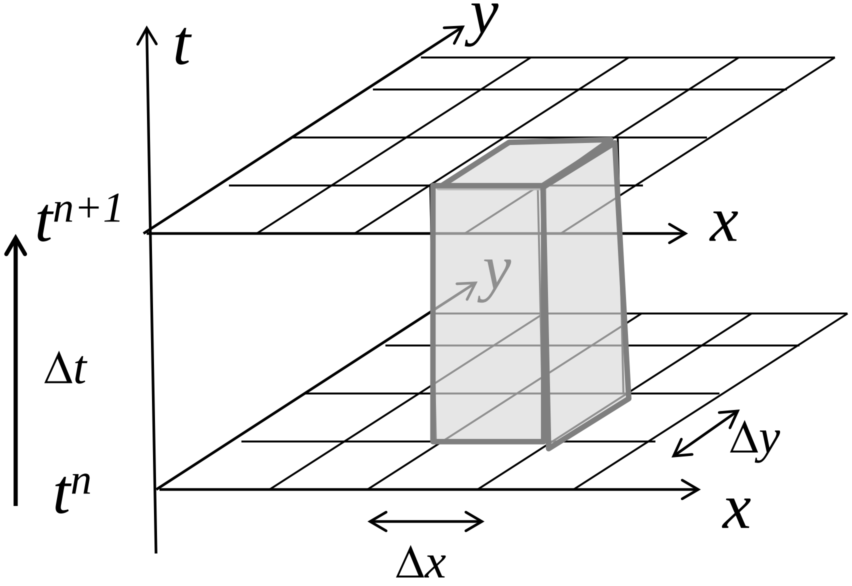
$$\frac{\partial}{\partial t} \iiint_{V} \rho \, dx \, dy \, dz + \iint_{A_1} \rho \, v_x \, dy \, dz - \iint_{A_2} \rho \, v_x \, dy \, dz + \iint_{A_3} \rho \, v_y \, dx \, dz - \iint_{A_4} \rho \, v_y \, dx \, dz$$
$$+ \iint_{A_5} \rho \, v_z \, dx \, dy - \iint_{A_6} \rho \, v_z \, dx \, dy = 0$$

When ***discontinuities/shocks*** are present, we have no hope of predicting the flow structure inside a *zone* in our computational *mesh*. However, the ***conservation form remains valid***!



$$\frac{1}{\Delta x\ \Delta y}\int_{t^n}^{t^{n+1}}\int_{-\Delta y/2}^{\Delta y/2}\int_{-\Delta x/2}^{\Delta x/2}\left(\mathrm{U}_t+\mathrm{F}_x+\mathrm{G}_y\right)dx\ dy\ dt=0$$

$$\frac{1}{\Delta x\ \Delta y}\int_{t^n}^{t^{n+1}}\int_{-\Delta y/2}^{\Delta y/2}\int_{-\Delta x/2}^{\Delta x/2}\frac{\partial\mathrm{U}(x,y,t)}{\partial t}dx\ dy\ dt=$$



$$\overline{\mathrm{U}}_{i,j}^{n+1}=\overline{\mathrm{U}}_{i,j}^{n}-\frac{\Delta t}{\Delta x}\left(\overline{\mathrm{F}}_{i+1/2,j}^{n+1/2}-\overline{\mathrm{F}}_{i-1/2,j}^{n+1/2}\right)-\frac{\Delta t}{\Delta y}\left(\overline{\mathrm{G}}_{i,j+1/2}^{n+1/2}-\overline{\mathrm{G}}_{i,j-1/2}^{n+1/2}\right)$$

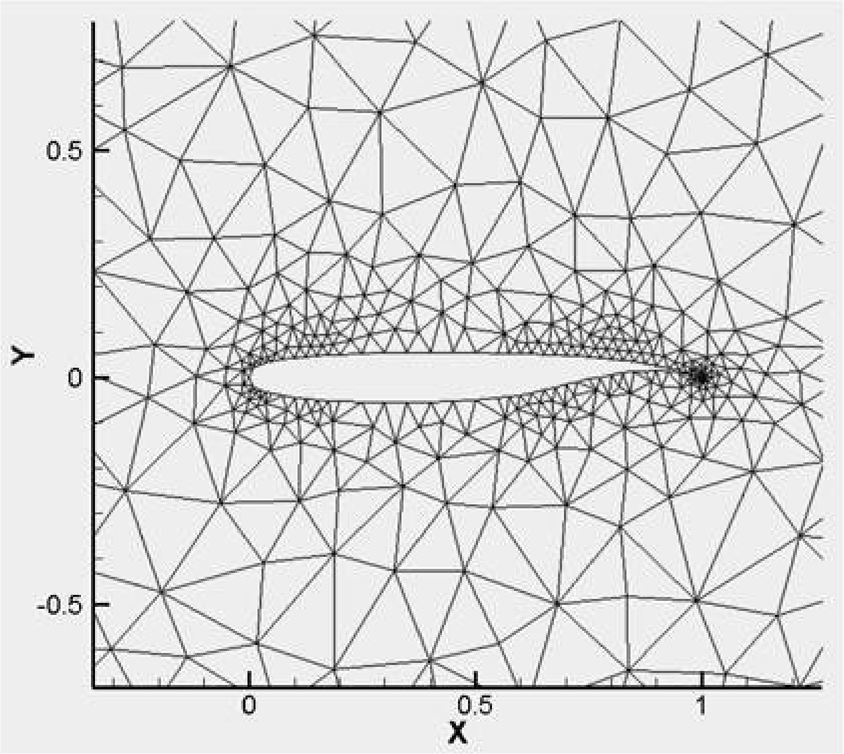
$$\frac{1}{\Delta x\ \Delta y}\int_{t^n}^{t^{n+1}}\int_{-\Delta y/2}^{\Delta y/2}\int_{-\Delta x/2}^{\Delta x/2}\left(\mathrm{U}_t+\mathrm{F}_x+\mathrm{G}_y\right)dx\ dy\ dt=0$$

$$\frac{1}{\Delta x\ \Delta y}\int_{t^n}^{t^{n+1}}\int_{-\Delta y/2}^{\Delta y/2}\int_{-\Delta x/2}^{\Delta x/2}\frac{\partial \mathrm{F}\left(x,y,t\right)}{\partial x}dx\ dy\ dt=$$

$$\frac{1}{\Delta x\ \Delta y}\int_{t^n}^{t^{n+1}}\int_{-\Delta y/2}^{\Delta y/2}\int_{-\Delta x/2}^{\Delta x/2}\frac{\partial \mathrm{G}\left(x,y,t\right)}{\partial y}dx\ dy\ dt=$$

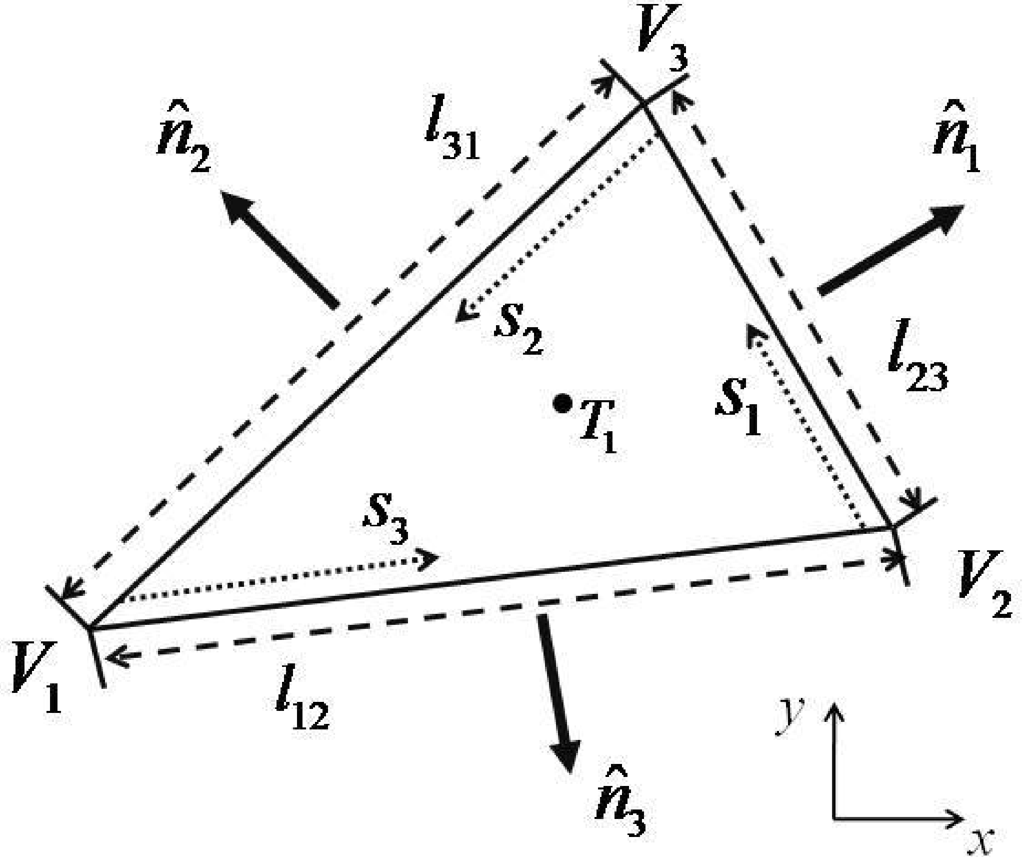
$$\overline{\mathrm{U}}_{i,j}^{n+1}=\overline{\mathrm{U}}_{i,j}^{n}-\frac{\Delta t}{\Delta x}\left(\overline{\mathrm{F}}_{i+1/2,j}^{n+1/2}-\overline{\mathrm{F}}_{i-1/2,j}^{n+1/2}\right)-\frac{\Delta t}{\Delta y}\left(\overline{\mathrm{G}}_{i,j+1/2}^{n+1/2}-\overline{\mathrm{G}}_{i,j-1/2}^{n+1/2}\right)$$

Triangulated meshes in 2d or tetrahedral meshes in 3d are very useful when mapping to complex geometries. This is important when solving configuration-specific problems.



This figure shows a triangulation of the exterior area around a two-dimensional airfoil. It illustrates the value of triangulated meshes and their ability to map complex geometries.

On such meshes, only finite volume methodology works. As before, without specifying how to obtain proper physical fluxes etc., we show how a finite volume discretization of a conservation law may be achieved on an unstructured mesh.



This diagram illustrates the triangle T_1 formed by the vertices V_1, V_2 and V_3. The lengths of the faces that lie opposite to the above vertices are given by l_{23}, l_{31} and l_{12} and are shown by the dashed arrows. The coordinates defined in the faces are denoted by s_1, s_2 and s_3 and are shown by the dotted arrows. The normals to each of the faces are also shown by the thick, solid arrows.

Let the domain of the triangle be denoted by A and let $|A|$ denote its area. A finite volume discretization of $U_t+F_x+G_y=0$ over triangle T_1 would require us to collocate the conserved variables at the center of the triangle and integrate the PDE over the space-time domain $A\times[t^n, t^n+\Delta t]$. As in the case of structured meshes, the time rate of update of the conserved variables is given by the fluxes at the boundaries of the triangle.

The finite volume update equations are given by:

$$\boxed{\overline{\mathrm{U}}_{T_1}^{n+1} = \overline{\mathrm{U}}_{T_1}^{n} - \frac{\Delta t}{|A_1|}\left(\overline{\mathcal{H}}_{23}^{n+1/2} l_{23} + \overline{\mathcal{H}}_{31}^{n+1/2} l_{31} + \overline{\mathcal{H}}_{12}^{n+1/2} l_{12}\right)}$$

Where we have the averages:

$$\overline{\mathrm{U}}_{T_1}^{n} \equiv \frac{1}{|A_1|}\int_{A_1} \mathrm{U}\left(x, y, t^n\right) dx\ dy \quad ; \qquad \overline{\mathcal{H}}_{23}^{n+1/2} \equiv \frac{1}{\Delta t\ l_{23}} \int_{t=t^n}^{t=t^{n+1}} \int_{V_2}^{V_3} \left(n_{1;x}\mathrm{F}\left(s_1, t\right) + n_{1;y}\mathrm{G}\left(s_1, t\right)\right) ds_1\ dt\ ;$$

$$\overline{\mathcal{H}}_{31}^{n+1/2} \equiv \frac{1}{\Delta t\ l_{31}} \int_{t=t^n}^{t=t^{n+1}} \int_{V_3}^{V_1} \left(n_{2;x}\mathrm{F}\left(s_2, t\right) + n_{2;y}\mathrm{G}\left(s_2, t\right)\right) ds_2\ dt\ ;$$

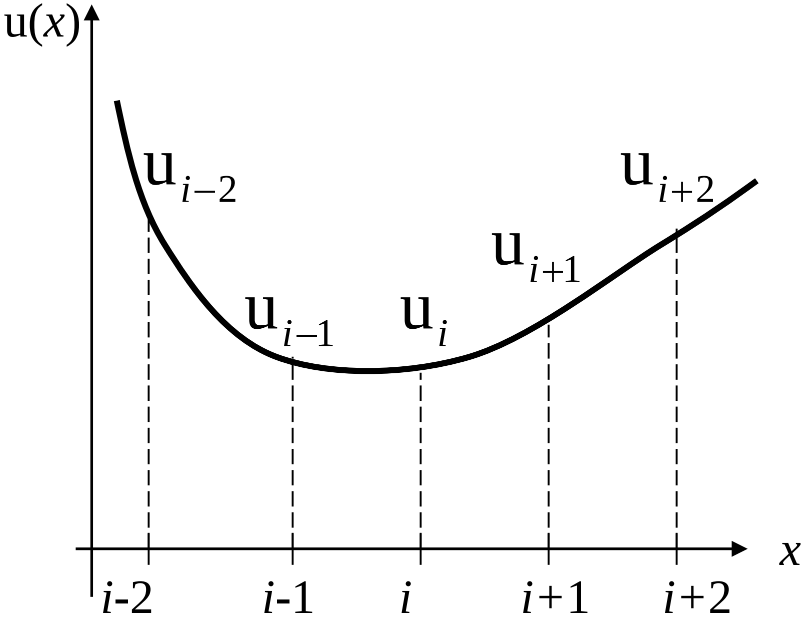
$$\overline{\mathcal{H}}_{12}^{n+1/2} \equiv \frac{1}{\Delta t\ l_{12}} \int_{t=t^n}^{t=t^{n+1}} \int_{V_1}^{V_2} \left(n_{3;x}\mathrm{F}\left(s_3, t\right) + n_{3;y}\mathrm{G}\left(s_3, t\right)\right) ds_3\ dt$$

2.3) Taylor Series and Accuracy of Discretizations. Truncation Error.

We have seen that as we subdivide the mesh we expect the solution to become more and more accurate. But we want to quantify this notion of accuracy. We expect *accuracy* to depend on the size "Δx" of the zones that make up a mesh.

The concept of *Taylor series* gives us a way to make that quantification.

The solution is available as a *mesh function* at discrete locations. Say, for simplicity, that those locations are evenly spaced.



Say we know a function "u" and its derivatives at the origin.

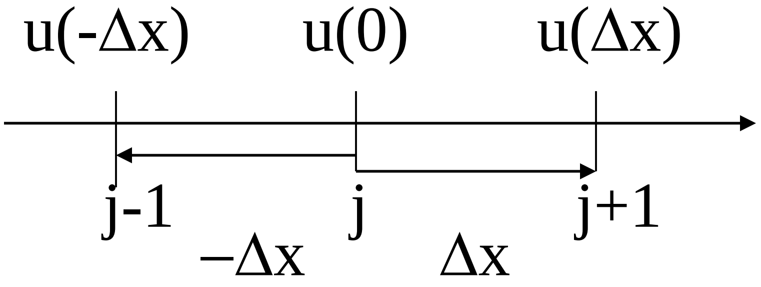
Thus, say we have: $u(0)$; $u_x(0)$; $u_{xx}(0)$; $u_{xxx}(0)$; $u_{xxxx}(0)$

As we increase the number of such derivative terms, we increase the accuracy with which we can predict u (h), a distance "h" from the origin:

$$u(h) = u(0) + u_x(0)\, h + \frac{1}{2}\, u_{xx}(0)\, h^2 + \frac{1}{6}\, u_{xxx}(0)\, h^3 + \frac{1}{24}\, u_{xxxx}(0)\, h^4 + ..$$

We know from calculus that as the terms of the Taylor series are extended, our predicted solution also becomes more accurate. We want to carry that concept over to our discrete numerical representation.

Let us, therefore, take the origin at the jth mesh point of a 1d mesh. The (j+1) th mesh point is located at “Δx” ; the (j-1) th mesh point is located at “-Δx”. At each of those *discrete* locations, we have a *mesh function* : $u_j = u(0)$, $u_{j+1} = u(\Delta x)$ and $u_{j-1} = u(-\Delta x)$, see figure.



Using our formula for Taylor series (truncate at 4th order) we get:

$$u_{j+1} = u(\Delta x) = u(0) + u_x(0)\Delta x + \frac{1}{2}u_{xx}(0)\Delta x^2 + \frac{1}{6}u_{xxx}(0)\Delta x^3 + \frac{1}{24}u_{xxxx}(0)\Delta x^4$$

$$u_j = u(0)$$

$$u_{j-1} = u(-\Delta x) = u(0) - u_x(0)\Delta x + \frac{1}{2}u_{xx}(0)\Delta x^2 - \frac{1}{6}\, u_{xxx}(0)\, \Delta x^3 + \frac{1}{24}u_{xxxx}(0)\Delta x^4$$

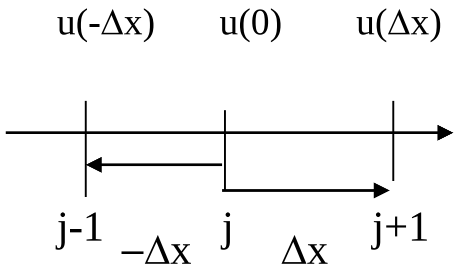
Subtract 3rd equation from 1st to get: $u_{j+1} - u_{j-1} = u_x(0)(2\Delta x) + \frac{1}{3}u_{xxx}(0)\Delta x^3$

or $\quad u_x(0) = \dfrac{u_{j+1} - u_{j-1}}{2\Delta x} - \dfrac{1}{6}u_{xxx}(0)\Delta x^2$

$$u_{j+1} = u(\Delta x) = u(0)+u_x(0)\Delta x+\frac{1}{2}u_{xx}(0)\Delta x^2+\frac{1}{6}u_{xxx}(0)\Delta x^3+\frac{1}{24}u_{xxxx}(0)\Delta x^4$$

$$u_j = u(0)$$

$$u_{j-1} = u(-\Delta x) = u(0)-u_x(0)\Delta x+\frac{1}{2}u_{xx}(0)\Delta x^2-\frac{1}{6}\ u_{xxx}(0)\ \Delta x^3+\frac{1}{24}u_{xxxx}(0)\Delta x^4$$

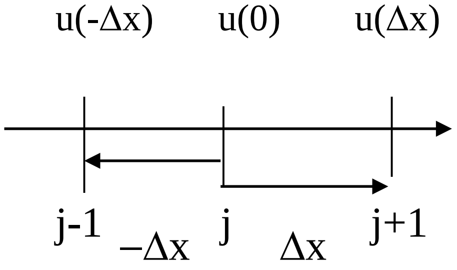


$$u_x(0) = \frac{u_{j+1}-u_{j-1}}{2\Delta x}-\frac{1}{6}u_{xxx}(0)\Delta x^2 \qquad ; \qquad u_{xx}(0) = \frac{u_{j+1}-2u_j\ +u_{j-1}}{\Delta x^2}-\frac{1}{12}u_{xxxx}(0)\Delta x^2$$

$$u_{j+1} = u(\Delta x) = u(0) + u_x(0)\Delta x + \frac{1}{2}u_{xx}(0)\Delta x^2 + \frac{1}{6}u_{xxx}(0)\Delta x^3 + \frac{1}{24}u_{xxxx}(0)\Delta x^4$$

$$u_j = u(0)$$

$$u_{j-1} = u(-\Delta x) = u(0) - u_x(0)\Delta x + \frac{1}{2}u_{xx}(0)\Delta x^2 - \frac{1}{6}\ u_{xxx}(0)\ \Delta x^3 + \frac{1}{24}u_{xxxx}(0)\Delta x^4$$



$$u_x(0) = \frac{u_{j+1} - u_{j-1}}{2\Delta x} - \frac{1}{6}u_{xxx}(0)\Delta x^2 \qquad ; \qquad u_{xx}(0) = \frac{u_{j+1} - 2u_j + u_{j-1}}{\Delta x^2} - \frac{1}{12}u_{xxxx}(0)\Delta x^2$$

$$u_x(0) = \frac{u_{j+1} - u_{j-1}}{2\Delta x} - \frac{1}{6} u_{xxx}(0)\Delta x^2$$

First derivative

Higher order terms. These carry the *truncation error*. The truncation error shows us that our FDA is *second order accurate*.

Finite difference approximation of *first derivative*

We can do a similar one for the *second derivative* to get:

$$u_{xx}(0) = \frac{u_{j+1} - 2u_j + u_{j-1}}{\Delta x^2} - \frac{1}{12} u_{xxxx}(0)\Delta x^2$$

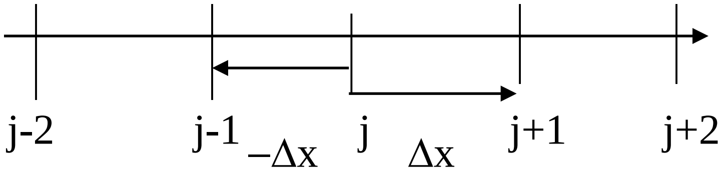
Second derivative

Higher order terms. These carry the truncation error

Finite difference approximation of *second derivative*

Notice: the above discretizations are *second order accurate*.

Question: Show that $u_x(0) = \dfrac{u_{j+1} - u_j}{\Delta x}$ is a first order accurate representation of the first derivative. Find the truncation error.



Question : Show that $u_x(0) = \dfrac{-u_{j+2} + 8\,u_{j+1} - \ 8\,u_{j-1} + u_{j-2}}{12\,\Delta x}$

is a fourth order accurate representation of the first derivative.

Question : Build on the above question to get a 4th order accurate representation of $u_{xx}(0)$. Is it the most compact 4th order representation?

Note: As the accuracy of approximation increases, so does the width of the corresponding ***stencils***. Question: What’s a stencil? Identify three point and five point stencils in the finite difference approximations above.

Question: For implicit problems, is a large stencil, generally speaking, a good thing or a bad thing? How would this limit the accuracy of practical implicit schemes?

Question: How does a larger stencil influence the solution speed on a parallel supercomputer?

2.4) Finite Difference Approximations and their Consistency

Note, therefore, that there is a difference between the *differential form* of an equation and its *finite difference approximation*. The *truncation error* quantifies this difference!

Differential Form : $u_t = \sigma u_{xx}$

Finite Difference Approximation:

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \sigma \left(\frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{\Delta x^2} \right)$$

Question: What is the spatial and temporal order of accuracy of the above finite difference approximation? What is the truncation error for the above finite difference approximation of the parabolic equation?

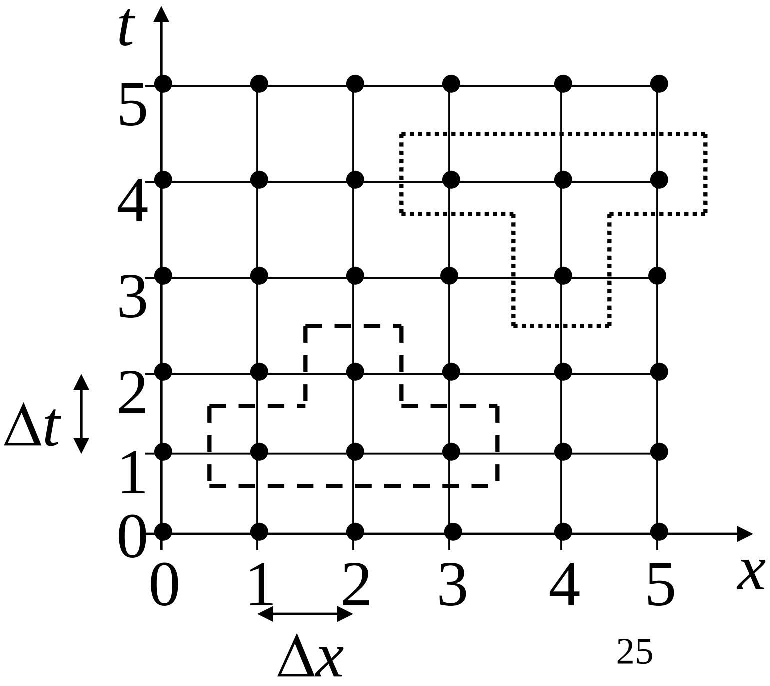
Notice from the above example that truncation errors can be of *different* orders in space and time!

The stencil for the *explicit* heat conduction equation is shown:-

Identify the numerical domain of dependence and range of influence for the FDA.

Compare contrast with the same for the differential form of the PDE.

$$u_t = \sigma\, u_{xx} \qquad \text{v/s} \qquad \frac{u_i^{n+1} - u_i^n}{\Delta t} = \sigma \left(\frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{\Delta x^2} \right)$$



Question: Draw stencils for each of the the following Finite Difference Approximations:

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \sigma \left(\frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{\Delta x^2} \right) \quad \leftarrow \text{ Fully Explicit}$$

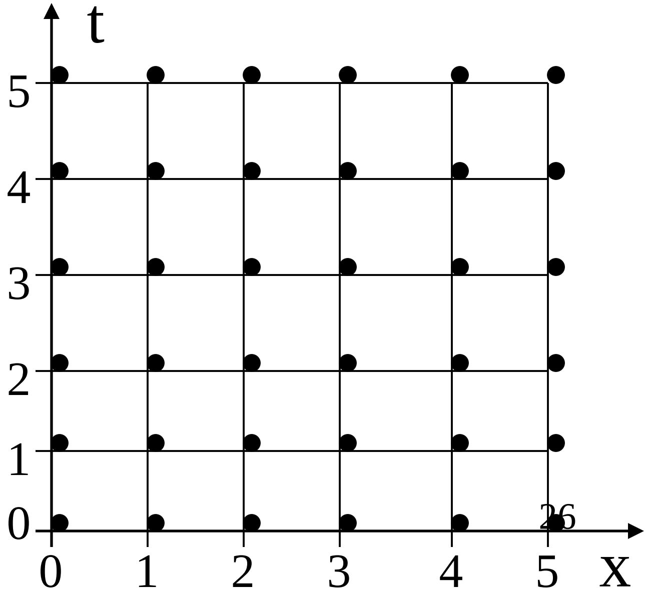
$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \sigma \left(\frac{u_{i+1}^{n+1} - 2u_i^{n+1} + u_{i-1}^{n+1}}{\Delta x^2} \right) \quad \leftarrow \text{ Fully Implicit}$$

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \alpha\ \sigma \left(\frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{\Delta x^2} \right) + (1-\alpha)\ \sigma \left(\frac{u_{i+1}^{n+1} - 2u_i^{n+1} + u_{i-1}^{n+1}}{\Delta x^2} \right)$$

$$(\ 0 \le \alpha \le 1\ )$$

of the Differential Form : $u_t = \sigma\, u_{xx}$

Question: What are the *stencils*, *domains of dependence* and *ranges of influence* of the above schemes? (Hint: Look at the stencils and ask which points will influence the solution at which other points?)



Consistency of a numerical scheme.

In the previous sections we have seen how we can use the concept of “discretization error” to produce “finite difference approximation” (FDA) to a “partial differential equation” (PDE) that is “good enough”.

But what really determines “good enough”? Certainly, we want the computed solution from a finite difference approximation to approximate the solution of the PDE up to some specified (and specifiable) discretization error.

Formally speaking, we say that the finite difference approximation provides a ***consistent*** approximation to the PDE if the finite difference approximation tends to the PDE in the limit where $\Delta t \to 0$ and $\Delta x \to 0$.

Question: Is the finite difference approximation

$$\frac{u_i^{n+1} - 2u_i^n}{\Delta t} = \sigma\left(\frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{\Delta x^2}\right)$$

a consistent approximation of the PDE : $u_t = \sigma\, u_{xx}$?

We realize, therefore, that an accurate enough finite difference approximation will produce a consistent approximation of the PDE. But will the physics always be correctly represented if all we have is a consistent approximation? In other words, is a consistent finite difference approximation ***sufficient*** for correctly representing the physics? ***Answer : NO! It is possible to have consistent approximations to a PDE that will not represent the physics correctly!***

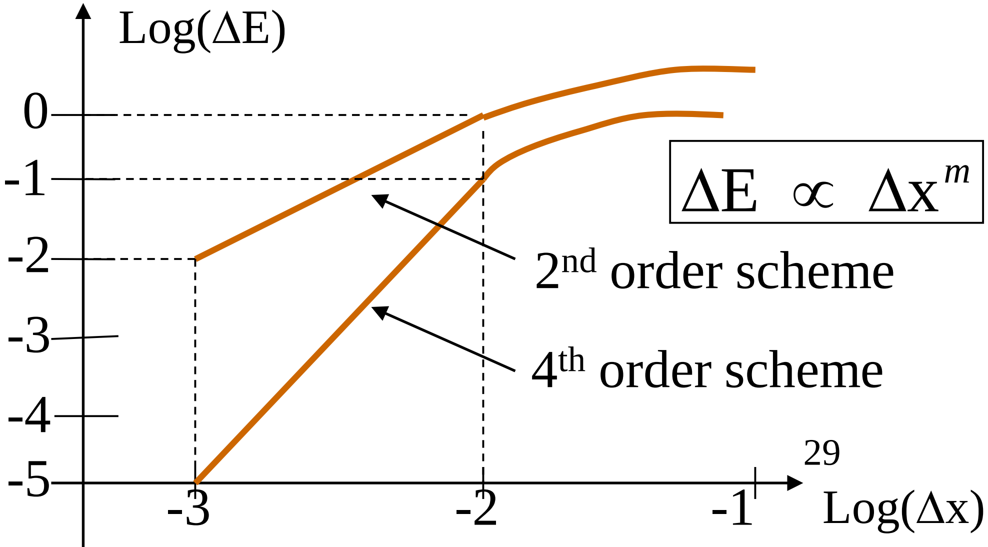
We need one more thing at a very minimum. That thing is ***stability***!

In other words, we can usually have *multiple*, consistent approximations to a PDE. Stability provides a further criterion by which we can winnow out several useless finite difference approximations. We only wish to retain the few consistent finite difference approximations to a PDE that pass the further test of stability.

Our expectation is that the solution of a more accurate scheme will approach the exact solution of the PDE faster than the corresponding solution of a less accurate scheme.

Practical Considerations Regarding Accuracy Analysis

In practice we can solve the same problem with the same scheme on meshes with different resolutions, Δx_1 , Δx_2 , Δx_3 , … (where $\Delta x_1 > \Delta x_2 > \Delta x_3 > \ldots$). We can compute the corresponding errors in the solution ΔE_1 , ΔE_2 , ΔE_3 , … on the meshes. We know that asymptotically we should have $\Delta E \propto \Delta x^m$ when Δx becomes small enough. Thus plotting log (ΔE) v/s log (Δx) allows us to "read off" the order of accuracy "*m*" of the scheme. The plot below, comparing a second order scheme to a 4th order scheme, gives the general idea.



So, higher order schemes may have a higher computational cost, but they also have an increased computational benefit.

If the problem has an analytic solution, we can obtain ΔE on the meshes by comparing the analytic solution to the numerical one that is obtained on each of the meshes.

If the problem does not have an analytic solution, we take the finest mesh in our set of meshes and obtain ΔE on the rest of the meshes by comparing their solution to the solution on the finest mesh. This may be a dicey thing to do because the finest mesh too could have a spurious component in its solution. If a finer mesh starts showing different solutions it could always be physical (take turbulence as an example) but one has to approach such situations with a modicum of circumspection.

The above plot shows that one may sometimes have to go to very large meshes (very small Δx) before the scheme reaches its asymptotic accuracy. Most schemes reach their design accuracy from below, I.e. on smaller meshes they are less accurate than their design accuracy. This often limits the accuracy one can safely advertise for one’s simulation of a physical problem.

How do we quantify error ΔE? There are various "error norms" to choose from. The situation dictates which one we choose. Say $\{ u_i \mid i = 1, \dots N\}$ gives the solution on "N" mesh points $\{ x_i \mid i = 1, \dots N\}$. Say $u^{exact}(x)$ is the exact solution.

L_2 norm (useful for parabolic/elliptic equn's):

$$\Delta E_2 = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(u_i - u^{exact}(x_i)\right)^2}$$

L_1 norm (useful for hyperbolic equn's):

$$\Delta E_1 = \frac{1}{N}\sum_{i=1}^{N}\left|u_i - u^{exact}(x_i)\right|$$

L_∞ norm (also useful for hyperbolic equn's):

$$\Delta E_\infty = \max_{i=1,\dots,N} \left|u_i - u^{exact}(x_i)\right|$$

For hyperbolic equations that are better than second order accurate, our definition of the L_1 norm may need to be upgraded. We will see that in the chapters on WENO schemes.

The above points give practical ideas for *gauging the error* of a numerical scheme *via direct numerical experimentation*. It is always a good idea to gauge this error for each new scheme via direct numerical analysis. This can show deficiencies in scheme design and/or implementation.

The best way to gauge accuracy is to do the same simulation on meshes of increasing resolution.

One often finds that a more accurate scheme reaches its design accuracy faster, I.e. on smaller meshes. This is a good argument in favor of more accurate schemes.

Consistency of a numerical scheme.

In the previous sections we have seen how we can use the concept of “discretization error” to produce “finite difference approximation” (FDA) to a “partial differential equation” (PDE) that is “good enough”.

But what really determines “good enough”? Certainly, we want the computed solution from a finite difference approximation to approximate the solution of the PDE up to some specified (and specifiable) discretization error.

Formally speaking, we say that the finite difference approximation provides a ***consistent*** approximation to the PDE if the finite difference approximation tends to the PDE in the limit where Δt → 0 and Δx → 0.

Question: Is the finite difference approximation

$$\frac{u_i^{n+1} - 2u_i^n}{\Delta t} = \sigma\left(\frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{\Delta x^2}\right)$$

a consistent approximation of the PDE : $u_t = \sigma u_{xx}$?

We realize, therefore, that an accurate enough finite difference approximation will produce a consistent approximation of the PDE. But will the physics always be correctly represented if all we have is a consistent approximation? In other words, is a consistent finite difference approximation ***sufficient*** for correctly representing the physics? ***Answer : NO! It is possible to have consistent approximations to a PDE that will not represent the physics correctly!***

We need one more thing at a very minimum. That thing is ***stability***!

In other words, we can usually have *multiple*, consistent approximations to a PDE. Stability provides a further criterion by which we can winnow out several useless finite difference approximations. We only wish to retain the few consistent finite difference approximations to a PDE that pass the further test of stability.

2.5) The Stability of Finite Difference Approximations

Bridges, boats, cars and planes can fail if the natural oscillations that they are liable to experience from the wind, road or water cause them to jostle too much. Avoiding such situations plays a great role in the design of bridges, boats, cars and planes .

Even the slightest spurious effect can excite such oscillations – the butterfly effect!

The fact that computers have finite precision means that discretization errors can, in and of themselves, excite such spurious oscillations in a numerical method. Other sources of unwanted oscillations can be imperfectly specified initial conditions, non-linearly large fluctuations in the solution itself, source terms, discretization error and imperfect specification of boundary conditions, to name but a few. In other words, “Whatever can go wrong will go wrong”. ← Murphy’s law.

Our goal is to protect our solution process against ***all*** such errors.

A car should have a low enough center of gravity so that it does not turn turtle, a stable aircraft should want to fly right side up! A numerical code should, likewise, by virtue of its very design, want to produce the right physical solution.

It turns out that one can use the same “linear stability analysis” that one uses for cars and aircraft to also analyze the stability of a numerical scheme. Such a stability is known as ***Von Neumann Stability Analysis***.

The following examples give us our first exposure to stability analysis within the context of ordinary differential equations.

Consider the ordinary differential equation: $\frac{d\,y}{d\,t} = -\sigma y$

It has the solution: $y\,(\,t) = y_0\ e^{-\sigma\ t}$

Thus we can write: $y\,(\,t^{n+1}) = e^{-\sigma\ \Delta t}\ \ y\,(\,t^{n}\,)$

Thus, even for a numerical scheme for solving the ODE, $\frac{d\,y}{d\,t} = -\sigma y$, we can posit: $y^{n+1} = \lambda\ y^{n}$. Here, y^{n+1} is the numerical solution at t^{n+1}=$(n+1)\Delta t$ and y^{n} is the numerical solution at $t^{n} = n\ \Delta t$.

Whether the numerical scheme is stable or not depends on what λ's it produces!

λ is known as the amplification factor. It determines stability, or lack thereof.

Comparing λ to $e^{-\sigma\,\Delta t}$ also allows us to gauge the "goodness of our finite difference approximation".

Consider the following time-explicit scheme: $\dfrac{y^{n+1} - y^n}{\Delta t} = -\sigma\, y^n$

Which can be written as : $y^{n+1} = y^n - \sigma\,\Delta t\, y^n$

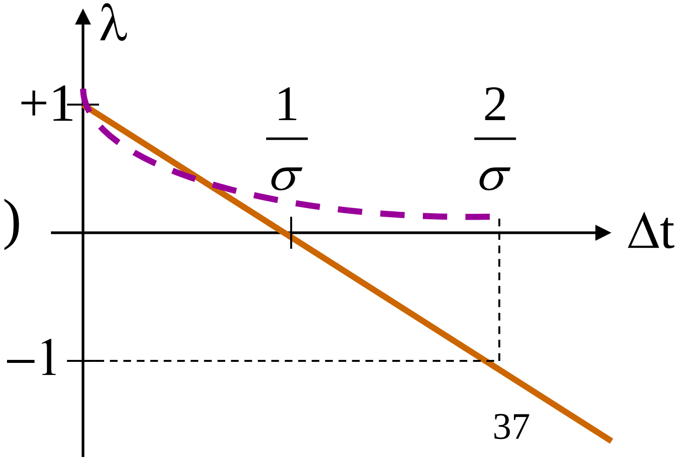
Use our ansatz that $y^{n+1} = \lambda y^n$ to get : $\lambda = 1 - \sigma\,\Delta t$

Notice that $\lambda \geq 0$ only when $\Delta t \leq \dfrac{1}{\sigma}$

Thus for a physically meaningful solution, the range of permissible Δt's is rather limited!

Say we take $|\lambda| \leq 1$ as our criterion for stability.(Why?)

Even then, we get $\Delta t \leq \dfrac{2}{\sigma}$. This is the domain of stability of the above scheme.



Question: What will happen to the solution computed from

$y^{n+1} = y^n - \sigma\, \Delta t\, y^n$ when $\Delta t \geq \dfrac{2}{\sigma}$? Find out by computing it on a computer.

Find out what happens when $\dfrac{1}{\sigma} < \Delta t < \dfrac{2}{\sigma}$.

Consider the following time-implicit scheme: $\dfrac{y^{n+1} - y^n}{\Delta t} = -\sigma\, y^{n+1}$

Which can be written as : $y^{n+1} + \sigma\, \Delta t\, y^{n+1} = y^n$

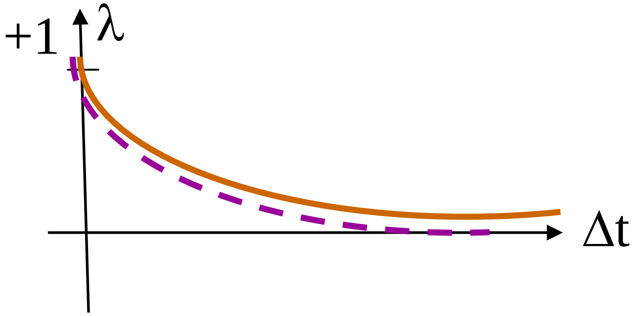
Use our ansatz that $y^{n+1} = \lambda y^n$ to get : $\lambda = 1/(1+\sigma\, \Delta t)$

Notice that $\lambda \geq 0$ for all Δt !

Also notice that $|\lambda| \leq 1$ for all Δt !

The scheme is unconditionally stable (or A-stable) !

λ also approximates $e^{-\sigma\, \Delta t}$ pretty well!



Von Neumann Stability is, in general, only *necessary* for stability. But by itself, it is *not sufficient* for stability in all cases.

A *convergent scheme* is one whose numerical solution tends to the solution of the PDE as Δt → 0 and Δx → 0.

Lax-Richtmeyer Theorem (see pgs 45-48 and 179 of Richtmeyer and Morton for proof) : Given a *properly posed linear* initial-boundary value problem and a finite difference approximation to it that satisfies the *consistency* condition, *stability* is a *necessary and sufficient condition* for *convergence*.
(***Mnemonic: consistency + stability → convergence***)

Note the word "*linear*" in the theorem above. Questions:Are the Euler equations linear? Identify some physically useful *linear* PDE's?

2.6) von Neumann Stability Analysis for Linear Parabolic Equations

The Lax-Richtmeyer theorem strictly applies to any linear PDE. But any PDE can be linearized about some local state; so it is also necessary (though not sufficient) in guaranteeing physical solutions for non-linear PDEs.

2.6.1) Stability Analysis for Time-explicit Linear Parabolic Equations

For now we focus on the linear parabolic PDE :

$\mathrm{u}_t = \sigma\, \mathrm{u}_{xx}$ with constant σ discretized on an infinite mesh (i.e. to avoid boundary conditions) with zone size Δx and timestep Δt. Mesh points: $x_j = j\Delta x$. $t^{n+1} = t^n + \Delta t$

Time-explicit scheme:

$$\boxed{\frac{\mathrm{u}_j^{n+1} - \mathrm{u}_j^{n}}{\Delta \mathrm{t}} = \sigma\left(\frac{\mathrm{u}_{j+1}^{n} - 2\mathrm{u}_j^{n} + \mathrm{u}_{j-1}^{n}}{\Delta \mathrm{x}^2}\right)} \Rightarrow$$

$$\mathrm{u}_j^{n+1} = \mathrm{u}_j^{n} + \mu\left(\mathrm{u}_{j+1}^{n} - 2\mathrm{u}_j^{n} + \mathrm{u}_{j-1}^{n}\right) \quad \text{where } \mu \equiv \frac{\sigma\,\Delta t}{\Delta x^2}$$

$$u_j^n = \mathrm{U}_\mathrm{k}^n \; e^{i\,\mathrm{k}\,x_j}\,;\; u_j^{n+1} = \mathrm{U}_\mathrm{k}^{n+1} \; e^{i\,\mathrm{k}\,x_j} \;\Rightarrow\; u_{j+1}^n = \mathrm{U}_\mathrm{k}^n \; e^{i\,\mathrm{k}\,x_j + i\,\mathrm{k}\,\Delta x} \text{ and } u_{j-1}^n = \mathrm{U}_\mathrm{k}^n \; e^{i\,\mathrm{k}\,x_j - i\,\mathrm{k}\,\Delta x}$$

It is worth demonstrating how this goes once:

$$\mathrm{U}_\mathrm{k}^{n+1} = \mathrm{U}_\mathrm{k}^n \left[1 + \mu\left(e^{i\,\mathrm{k}\,\Delta x} - 2 + e^{-i\,\mathrm{k}\,\Delta x}\right)\right] = \mathrm{U}_\mathrm{k}^n \left[1 + \mu\, 2\left(\cos\,(\mathrm{k}\,\Delta x) - 1\right)\right]$$

$$= \mathrm{U}_\mathrm{k}^n \left[1 - 4\,\mu\,\sin^2\,(\,\mathrm{k}\,\Delta x\,/\,2)\right] \quad \leftarrow \quad \text{Recall amplification factors?}$$

For the FDA we have : $\lambda_{\mathrm{FDA}}(\mathrm{k}) \equiv \dfrac{\mathrm{U}_\mathrm{k}^{n+1}}{\mathrm{U}_\mathrm{k}^n} = 1 - 4\,\mu\,\sin^2\,(\,\mathrm{k}\,\Delta x\,/\,2)$

For the PDE we have : $\lambda_{\mathrm{PDE}}(\mathrm{k}) = e^{-\sigma\,\mathrm{k}^2\,\Delta t} = e^{-(\mathrm{k}\Delta x)^2\left[\frac{\sigma\,\Delta t}{\Delta x^2}\right]} = e^{-\mu\,(\mathrm{k}\Delta x)^2}$

Notice that $\left|\lambda_{\mathrm{PDE}}(\mathrm{k})\right| \le 1$ for all $\Delta t \;\Rightarrow$ *PDE is unconditionally stable*

However, $\left|\lambda_{\mathrm{FDA}}(\mathrm{k})\right| \le 1$ for $\mu \le 1/2$ or $\Delta t \;\le\; \Delta x^2 \big/ \left(2\sigma\right)$

$\Rightarrow$ *FDA is conditionally stable*

We say that the FDA approximates the PDE well when $\lambda_{\mathrm{FDA}}(\mathrm{k}) \;\to\; \lambda_{\mathrm{PDE}}(\mathrm{k})$!

$$\frac{\mathrm{u}_{\mathrm{i}}^{\mathrm{n+1}} - \mathrm{u}_{\mathrm{i}}^{\mathrm{n}}}{\Delta \mathrm{t}} = \sigma\left(\frac{\mathrm{u}_{\mathrm{i+1}}^{\mathrm{n}} - 2\mathrm{u}_{\mathrm{i}}^{\mathrm{n}} + \mathrm{u}_{\mathrm{i-1}}^{\mathrm{n}}}{\Delta \mathrm{x}^{2}}\right) \Rightarrow \mathrm{u}_{j}^{n+1} = \mathrm{u}_{j}^{n} + \mu\left(\mathrm{u}_{j+1}^{n} - 2\mathrm{u}_{j}^{n} + \mathrm{u}_{j-1}^{n}\right) \text{ with } \mu \equiv \frac{\sigma\,\Delta t}{\Delta x^{2}}$$

$$\mathrm{u}_j^n = \mathrm{U}_\mathrm{k}^n \; \mathrm{e}^{i\,\mathrm{k}\,x_j} \quad ; \quad \mathrm{u}_j^{n+1} = \mathrm{U}_\mathrm{k}^{n+1} \; \mathrm{e}^{i\,\mathrm{k}\,x_j} \;\; \Rightarrow$$

$$\mathrm{u}_{j+1}^n = \mathrm{U}_\mathrm{k}^n \; \mathrm{e}^{i\,\mathrm{k}\,x_j + i\,\mathrm{k}\,\Delta x} = \mathrm{U}_\mathrm{k}^n \; \mathrm{e}^{i\,\mathrm{k}\,x_j} \mathrm{e}^{i\,\mathrm{k}\,\Delta x} \text{ and } \mathrm{u}_{j-1}^n = \mathrm{U}_\mathrm{k}^n \; \mathrm{e}^{i\,\mathrm{k}\,x_j \, - i\,\mathrm{k}\,\Delta x} = \mathrm{U}_\mathrm{k}^n \; \mathrm{e}^{i\,\mathrm{k}\,x_j} \mathrm{e}^{-i\,\mathrm{k}\,\Delta x}$$

Question: Why do we want $-\pi \le k\Delta x \le \pi$?

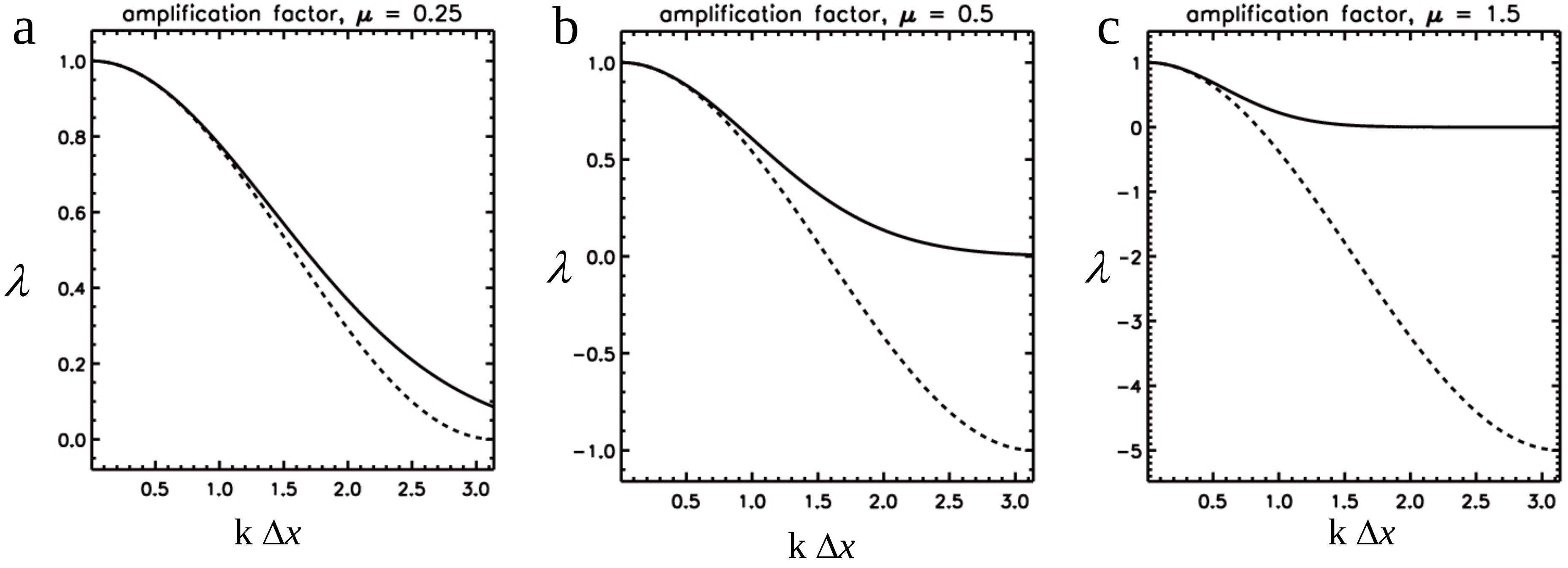
$$\mathrm{u}_j^{n+1} = \mathrm{u}_j^n + \mu\left(\mathrm{u}_{j+1}^n - 2\mathrm{u}_j^n + \mathrm{u}_{j-1}^n\right) \quad \Rightarrow$$

$$\lambda_{\mathrm{FDA}}(\mathrm{k}) \equiv \frac{\mathrm{U}_\mathrm{k}^{n+1}}{\mathrm{U}_\mathrm{k}^n} = 1 - 4\,\mu\,\sin^2\,(\,\mathrm{k}\,\Delta x\,/\,2)$$

$$\lambda_{\mathrm{PDE}}(\mathrm{k}) = \mathrm{e}^{-\,\sigma\,\mathrm{k}^2\,\Delta t} = \mathrm{e}^{-\,(\mathrm{k}\Delta x)^2 \left[\frac{\sigma\,\Delta t}{\Delta x^2}\right]} = \mathrm{e}^{-\,\mu\,(\mathrm{k}\Delta x)^2}$$

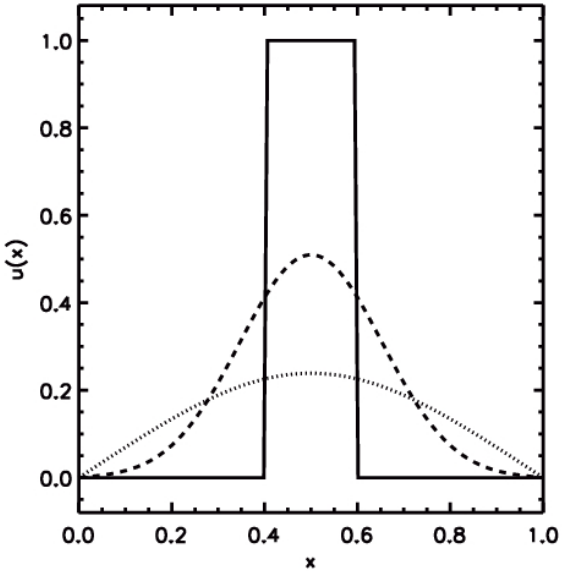
The amplification factors are shown for $\mu = 0.25, 0.5, 1.5$.

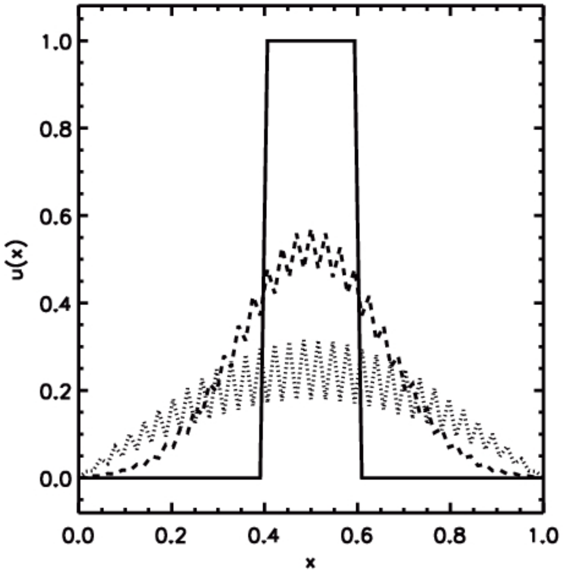
Dashed curve : FDA Solid curve : PDE <u>Question</u>: What do you see?



Evolution of Square Pulse:

$\mu = 0.4$





$\mu = 0.5008$

2.6.2) Stability Analysis for Time-Implicit and Semi-Implicit Linear Parabolic PDEs

Time-Implicit Scheme:

$$\boxed{\frac{u_j^{n+1}-u_j^{n}}{\Delta t}=\sigma\left(\frac{u_{j+1}^{n+1}-2u_j^{n+1}+u_{j-1}^{n+1}}{\Delta x^2}\right)} \Rightarrow$$

$$u_j^{n+1}-\mu\left(u_{j+1}^{n+1}-2u_j^{n+1}+u_{j-1}^{n+1}\right)=u_j^{n}$$

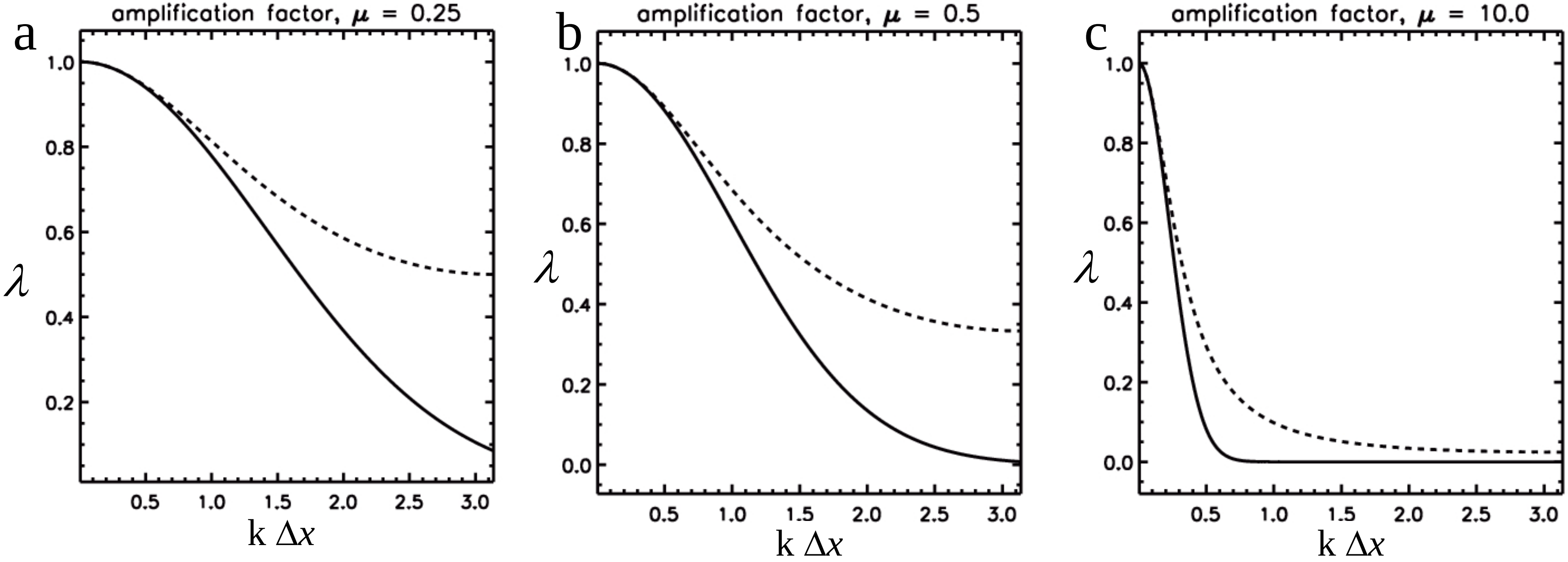
We get

$$\lambda_{FDA}(k)=\frac{U_k^{n+1}}{U_k^{n}}=\frac{1}{\left[1+4\,\mu\,\sin^2\left(k\,\Delta x/2\right)\right]}$$

← Question: What is this amplification factor telling you about stability?

The amplification factors are shown for $\mu = 0.25, 0.5, 10.0$.

Dashed curve : FDA Solid curve : PDE Question: What do you see?



$$\mathrm{u}_j^n = \mathrm{U}_\mathrm{k}^n \ \mathrm{e}^{i\ \mathrm{k}\, x_j} \quad ; \quad \mathrm{u}_j^{n+1} = \mathrm{U}_\mathrm{k}^{n+1} \ \mathrm{e}^{i\ \mathrm{k}\, x_j} \ \Rightarrow$$

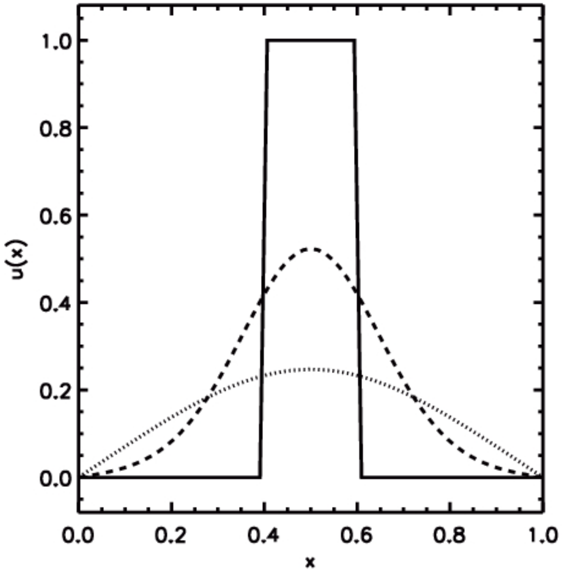
$$\mathrm{u}_{j+1}^n = \mathrm{U}_\mathrm{k}^n \ \mathrm{e}^{i\ \mathrm{k}\, x_j + i\ \mathrm{k}\, \Delta x} = \mathrm{U}_\mathrm{k}^n \ \mathrm{e}^{i\ \mathrm{k}\, x_j} \mathrm{e}^{i\ \mathrm{k}\, \Delta x} \text{ and } \mathrm{u}_{j-1}^n = \mathrm{U}_\mathrm{k}^n \ \mathrm{e}^{i\ \mathrm{k}\, x_j \ - i\ \mathrm{k}\, \Delta x} = \mathrm{U}_\mathrm{k}^n \ \mathrm{e}^{i\ \mathrm{k}\, x_j} \mathrm{e}^{- i\ \mathrm{k}\, \Delta x}$$

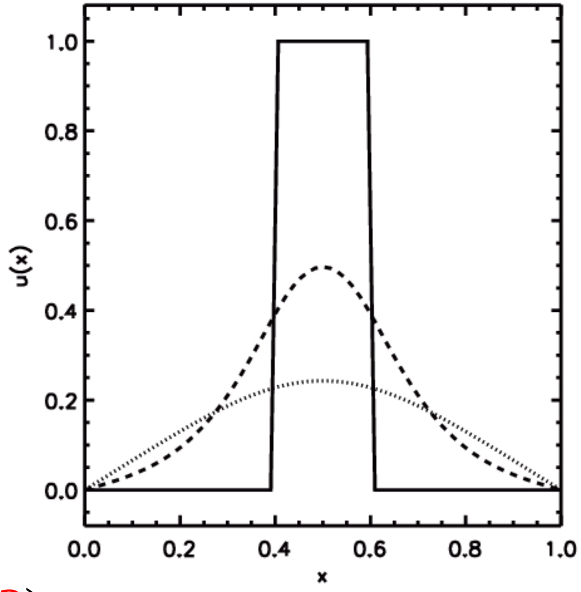
$$\mathrm{u}_j^{n+1} - \mu\left(\mathrm{u}_{j+1}^{n+1} - 2\mathrm{u}_j^{n+1} + \mathrm{u}_{j-1}^{n+1}\right) = \mathrm{u}_j^n \quad \Rightarrow$$

$$\lambda_{\mathrm{FDA}}(\mathrm{k}) = \frac{\mathrm{U}_\mathrm{k}^{n+1}}{\mathrm{U}_\mathrm{k}^n} = \frac{1}{\left[1+4\ \mu\ \sin^2\ (\ \mathrm{k}\ \Delta x\ /\ 2)\right]}$$

Evolution of Square Pulse:

$\mu = 6.55$





$\mu = 32.75$

Semi-Implicit Scheme (Crank-Nicholson,α=1/2):

$$\boxed{\frac{u_j^{n+1}-u_j^{n}}{\Delta t} = \alpha\ \sigma\left(\frac{u^{n}_{j+1}-2u^{n}_{j}+u^{n}_{j-1}}{\Delta x^2}\right) + (1-\alpha)\ \sigma\left(\frac{u^{n+1}_{j+1}-2u^{n+1}_{j}+u^{n+1}_{j-1}}{\Delta x^2}\right)\ (\,0\le\alpha\le 1\,)} \Rightarrow$$

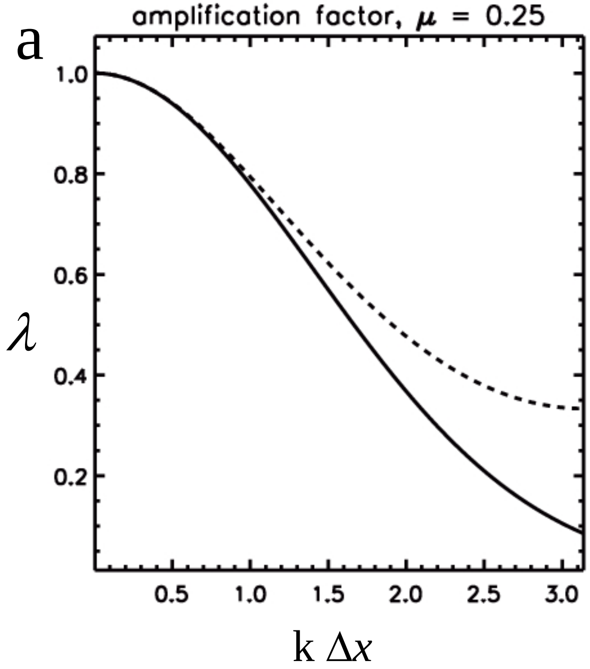
$$u_j^{n+1} - \mu(1-\alpha)\left(u^{n+1}_{j+1}-2u^{n+1}_{j}+u^{n+1}_{j-1}\right) = u^{n}_{j} + \mu\alpha\left(u^{n}_{j+1}-2u^{n}_{j}+u^{n}_{j-1}\right)$$

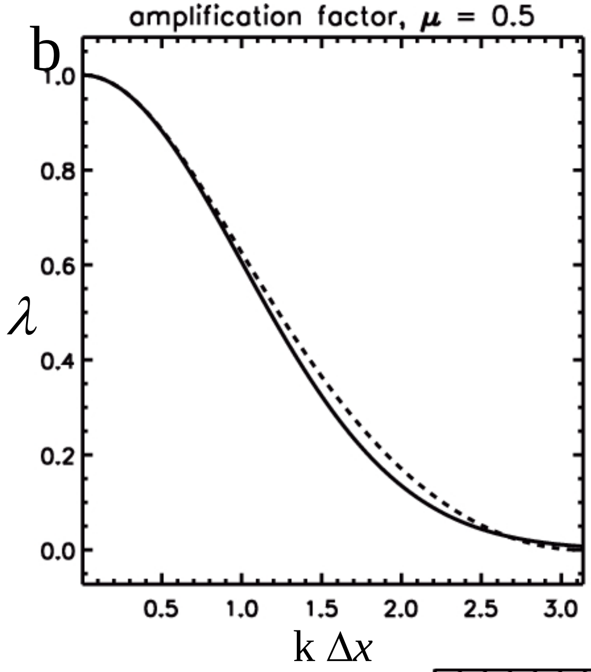
We get

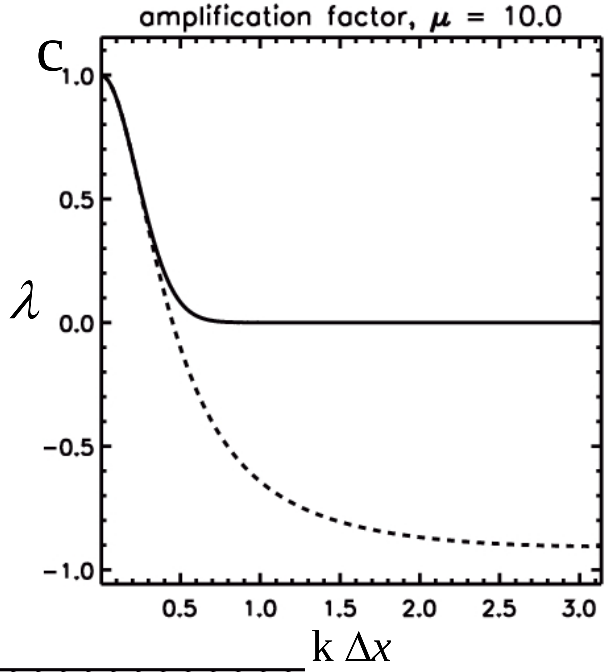
$$\lambda_{\mathrm{FDA}}(k) = \frac{U_k^{n+1}}{U_k^{n}} = \frac{\left[1-4\,\mu\,\alpha\,\sin^2\,(\,k\,\Delta x\,/\,2)\right]}{\left[1+4\,\mu\,(1-\alpha)\,\sin^2\,(\,k\,\Delta x\,/\,2)\right]}$$

The amplification factors are shown for μ = 0.25, 0.5, 10.0. and α=1/2

Dashed curve : FDA Solid curve : PDE Question: What do you see?

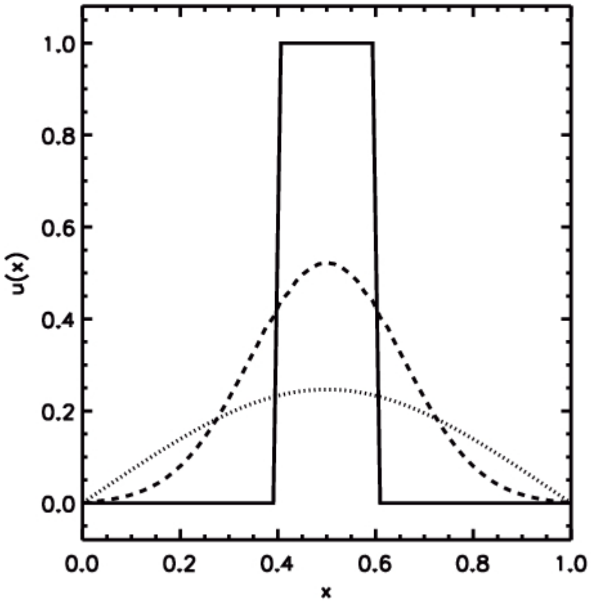


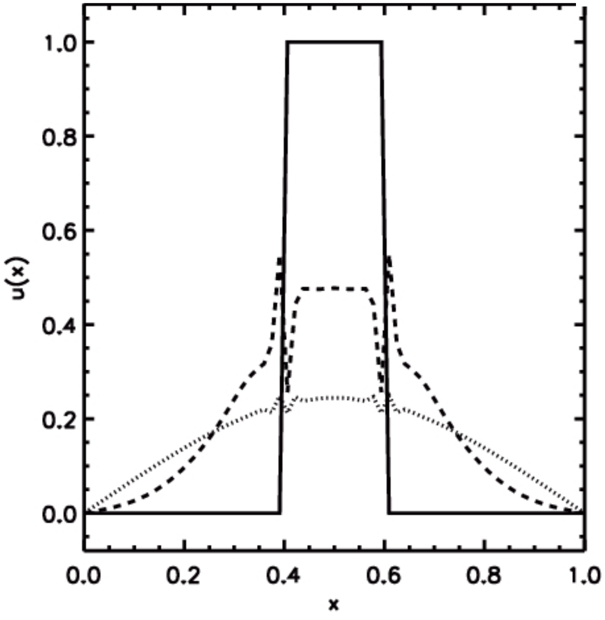




Evolution of Square Pulse:

μ = 3.5





μ = 10.0

2.6.3) Stability Analysis of the Time-Implicit TR-BDF2 Method

The Crank-Nicholson scheme, despite being second order accurate, has the deficiency that it produces spurious oscillations.

Can one obtain a second order accurate scheme for parabolic problems that is free of these oscillations? Ans: If one is willing to invert the matrix twice, then the answer is yes!

One uses a **TR**apezoidal scheme for the first step which only takes us up to a time of $t^n + \Delta t/2$ from a time of t^n . This is written as:

$$\mathrm{u}_j^{n+1/2} - \frac{\mu}{4}\left(\mathrm{u}_{j+1}^{n+1/2} - 2\mathrm{u}_j^{n+1/2} + \mathrm{u}_{j-1}^{n+1/2}\right) = \mathrm{u}_j^n + \frac{\mu}{4}\left(\mathrm{u}_{j+1}^n - 2\mathrm{u}_j^n + \mathrm{u}_{j-1}^n\right)$$

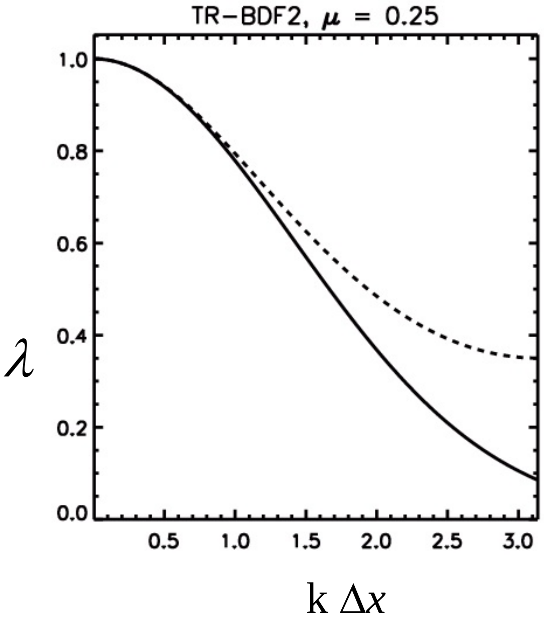
Using time levels t^n and $t^{n+1/2}$, we now use a **B**ackward **D**ifference **F**ormula of **2**nd order as:

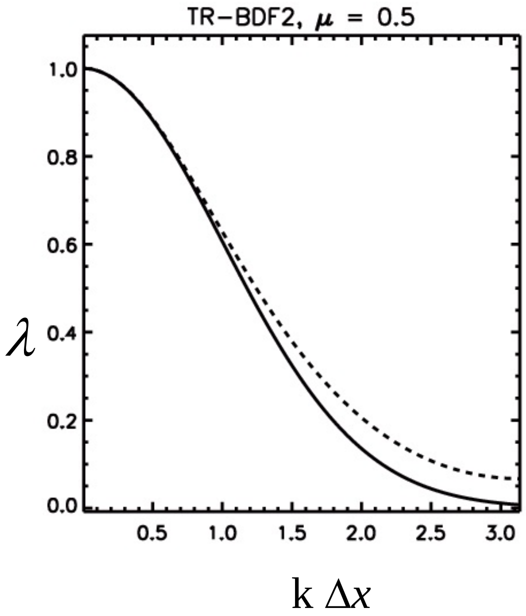
$$\mathrm{u}_j^{n+1} - \frac{\mu}{3}\left(\mathrm{u}_{j+1}^{n+1} - 2\mathrm{u}_j^{n+1} + \mathrm{u}_{j-1}^{n+1}\right) = -\frac{1}{3}\,\mathrm{u}_j^n + \frac{4}{3}\,\mathrm{u}_j^{n+1/2}$$

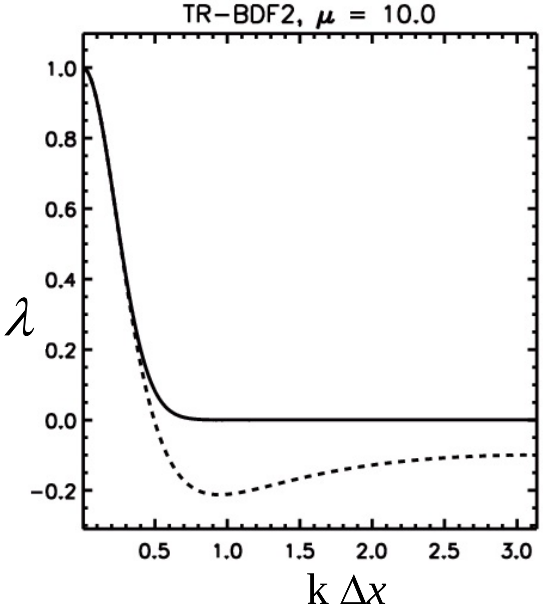
Hence the name TR-BDF2. This scheme is also useful when stiff source terms are present in addition to the diffusion terms.

The amplification factors are shown for μ = 0.25, 0.5, 10.0. and TR-BDF2

Dashed curve : FDA   Solid curve : PDE   Question:What do you see?Compare with C-N.

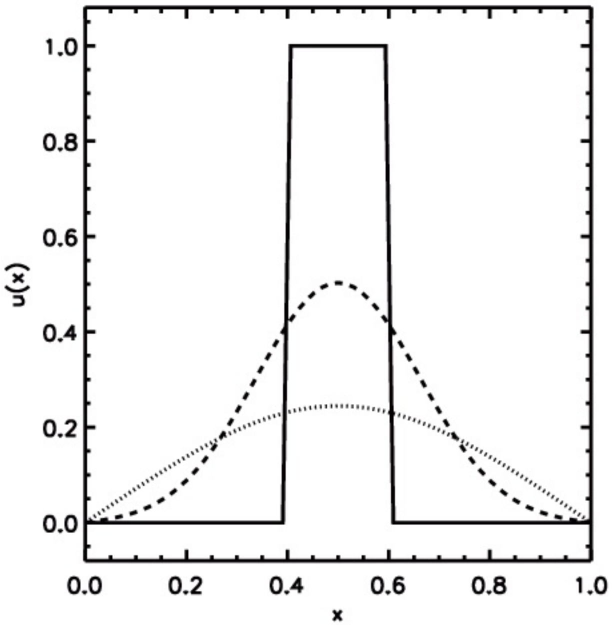




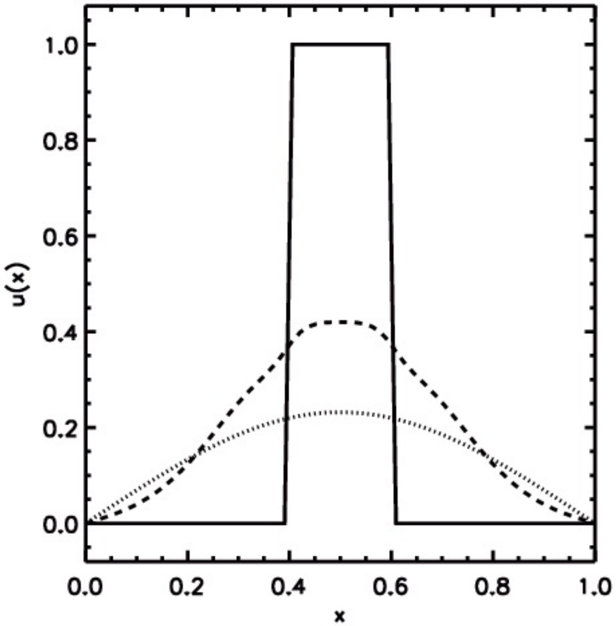


Evolution of Square Pulse:

μ = 6.55



μ = 32.75



2.6.4) Boundary Conditions for Parabolic Equations

Our *parabolic FDA* looks very much like the elliptic *Poisson equation*.

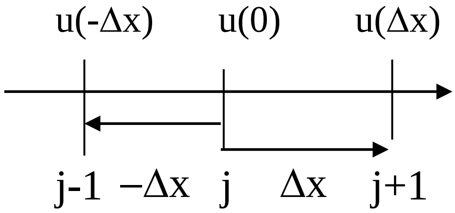
There is a theorem which states that for the Poisson problem we can either specify the value of the potential at the boundary or specify the gradient of the potential at the boundary. However, we can never specify the value of the potential and its gradient at a boundary.

For parabolic equations, the boundary conditions can change in time, but the same restrictions apply – we can't specify variable and its gradient at a boundary at any given time.

The boundary conditions we used in our previous example are called **Dirichlet** boundary conditions and consist of specifying the solution at the boundary of the domain.

Specifying the gradient gives us **Neumann** boundary conditions.

We may also specify a linear combination of the potential and its gradient, known as **mixed** boundary conditions.



$$a_l \; u_x + b_l \; u = c_l \quad ; \quad a_r \; u_x + b_r \; u = c_r$$

We may also require the boundary conditions to be **periodic**, which changes the dimension of the resulting matrix when implicit/semi-implicit formulations are used.

2.6.5) Introduction to Matrix Methods for Parabolic Equations

Consider the fully-implicit formulation on a 1d mesh. The mesh points are indexed from j=0 to j=J At the boundaries one can have the most general form of mixed boundary conditions by discretizing the boundary conditions as:

$$(b_l\Delta x - a_l)\, u_0^{n+1} + a_l \; u_1^{n+1} = c_l\Delta x \quad ; \quad -a_r u_{J-1}^{n+1} + (b_r\Delta x + a_r)\, u_J^{n+1} = c_r\Delta x$$

In the interior we have the FDA:

$$-\mu \; u_{j+1}^{n+1} + \left(1+2\mu\right) u_j^{n+1} - \mu \; u_{j-1}^{n+1} = u_j^n \quad \text{for} \quad j=1,..,J-1$$

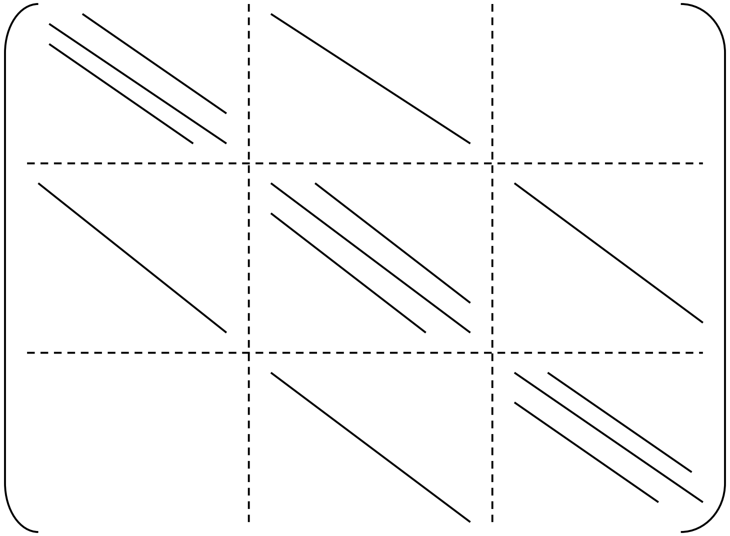
Solve : $\mathrm{u}_j^{n+1} - \mu\left(\mathrm{u}_{j+1}^{n+1} - 2\mathrm{u}_j^{n+1} + \mathrm{u}_{j-1}^{n+1}\right) = \mathrm{u}_j^n$ for $1 \le j \le J-1$

with boundary conditions : $\mathrm{a}_l\ \mathrm{u}_x + \mathrm{b}_l\ \mathrm{u} = \mathrm{c}_l$ at $j=0$; $\mathrm{a}_r\ \mathrm{u}_x + \mathrm{b}_r\ \mathrm{u} = \mathrm{c}_r$ at $j=J$

The result is a ***banded sparse matrix*** with dimension $(J+1)\times(J+1)$:

$$\begin{bmatrix} \mathrm{b}_l\Delta x-\mathrm{a}_l & \mathrm{a}_l & & & & \\ -\mu & (1+2\mu) & -\mu & & & \\ & -\mu & (1+2\mu) & -\mu & & \\ & & \cdots & \cdots & \cdots & \\ & & & -\mu & (1+2\mu) & -\mu \\ & & & & -\mathrm{a}_r & \mathrm{b}_r\Delta x+\mathrm{a}_r \end{bmatrix} \begin{bmatrix} \mathrm{u}_0^{n+1} \\ \mathrm{u}_1^{n+1} \\ \mathrm{u}_2^{n+1} \\ . \\ \mathrm{u}_{J-1}^{n+1} \\ \mathrm{u}_J^{n+1} \end{bmatrix} = \begin{bmatrix} \mathrm{c}_l\Delta x \\ \mathrm{u}_1^{n} \\ \mathrm{u}_2^{n} \\ . \\ \mathrm{u}_{J-1}^{n} \\ \mathrm{c}_r\Delta x \end{bmatrix}$$

Such matrices also arise when discretizing elliptic and parabolic equations in multiple dimensions. For 2d problems we have the form:



2.7) von Neumann Stability Analysis of Linear Hyperbolic Equations

Even the simplest 1d PDE : $u_t + a\, u_x = 0$ has much to teach us.

On infinite domains, the initial conditions $u_0(x)$ evolves as $u\,(x, t) = u_0(x - a\, t)$, as shown in the figure below.

Shape is preserved; characteristics are straight lines in space-time.

Question: On finite domains, we need more. What do we need?

How do we specify the boundaries of the domain?

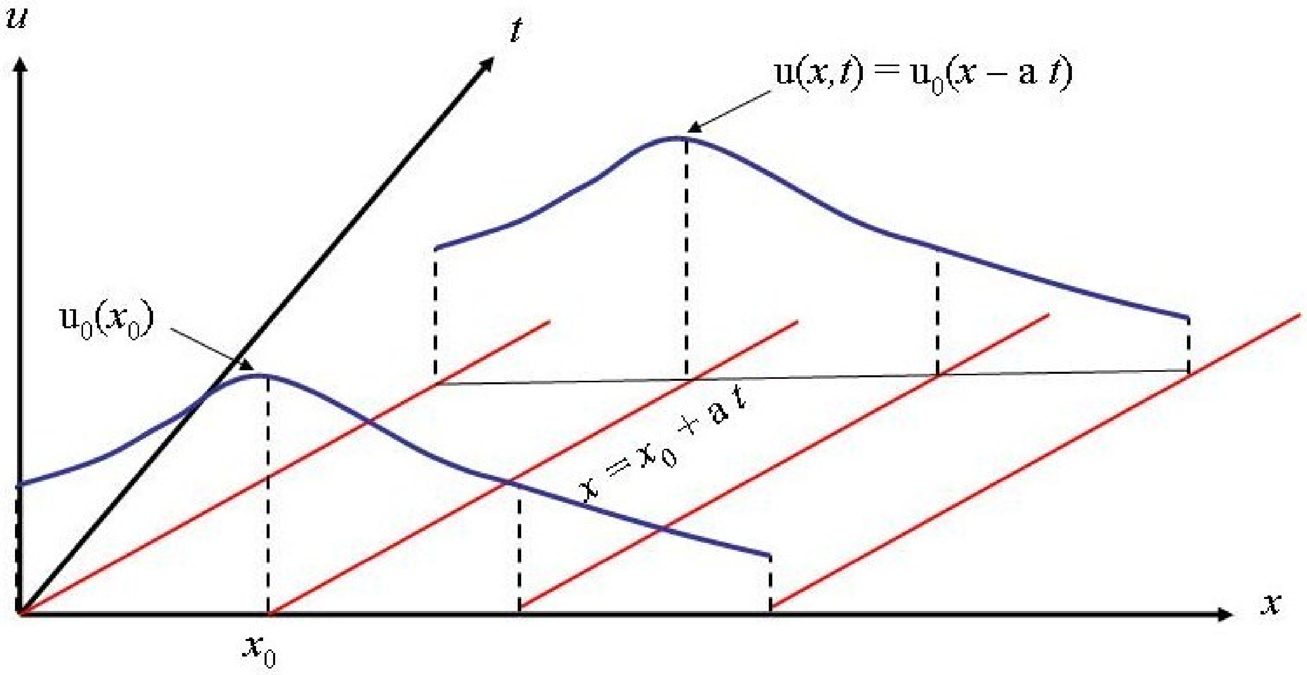


Fig. 2.13 showing the time-evolution of the advection equation $u_t + a\, u_x = 0$ *. The red lines show the characteristics in the x,t plane while the blue curves represent the solution "u" and its evolution in time.*

First analyze amplification factor for PDE, $u_t + a\,u_x = 0$,

using Fourier modes $u_j^n = U_k^n \, e^{i\,k\,x_j}$:-

$$\lambda_{\text{PDE}}(k) = e^{-i\,a\,k\,\Delta t} = e^{-i\,(k\,\Delta x)\left[\frac{a\,\Delta t}{\Delta x}\right]} = e^{-i\,\mu\,(k\,\Delta x)}$$

where we define $\mu \equiv \dfrac{a\,\Delta t}{\Delta x}$

$$\left|\lambda_{\text{PDE}}(k)\right| = 1 \;\forall\; k$$

$\Rightarrow$ PDE is not *dissipative*

$$\theta_{\text{PDE}}(k) \equiv \tan^{-1}\left\{\frac{\operatorname{Im}\left[\lambda_{\text{PDE}}(k)\right]}{\operatorname{Re}\left[\lambda_{\text{PDE}}(k)\right]}\right\} = -k\,a\,\Delta t$$

$\Rightarrow$ PDE is not *dispersive*

2.7.1) Forward Euler (Never Used)

The forward Euler scheme is an example of an *unconditionally unstable* scheme that should never be used. It is only meant to illustrate that it is easy to do something that seems "reasonable" and arrive at a bad scheme.

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = -a\left(\frac{u_{j+1}^n - u_{j-1}^n}{2\,\Delta x}\right) \Leftrightarrow u_j^{n+1} = u_j^n - \frac{\mu}{2}\left(u_{j+1}^n - u_{j-1}^n\right) \qquad \mu \equiv \frac{a\,\Delta t}{\Delta x}$$

$$\lambda_{\mathrm{FDA}}(\mathrm{k}) = \frac{U_k^{n+1}}{U_k^n} = 1 - i\,\mu\,\sin(\,\mathrm{k}\,\Delta x) \quad ; \quad \left|\lambda_{\mathrm{FDA}}(\mathrm{k})\right| > 1 \;\; \forall\; \mathrm{k} \Rightarrow \text{unconditionally unstable!}$$

Set $u_j^n = U_{\mathrm{k}}^n\ e^{i\,\mathrm{k}\,x_j}$ and $u_j^{n+1} = U_{\mathrm{k}}^{n+1}\ e^{i\,\mathrm{k}\,x_j}$ to derive amplification factor.

2.7.2) Lax-Friedrichs Scheme

Slightly modify the forward Euler scheme:

$$\boxed{\frac{u_j^{n+1} - \frac{1}{2}\left(u_{j+1}^n + u_{j-1}^n\right)}{\Delta t} = -a\left(\frac{u_{j+1}^n - u_{j-1}^n}{2\,\Delta x}\right) \Leftrightarrow u_j^{n+1} = \frac{1}{2}\left(u_{j+1}^n + u_{j-1}^n\right) - \frac{\mu}{2}\left(u_{j+1}^n - u_{j-1}^n\right)}$$

$$\mu \equiv \frac{a\,\Delta t}{\Delta x}$$

is CFL #

Why does it work? Rewrite the scheme as:

$$u_j^{n+1} = u_j^n - \frac{\Delta t}{\Delta x}\left(f_{j+1/2}^n - f_{j-1/2}^n\right) \text{ with } f_{j+1/2}^n = \frac{a}{2}\left(u_{j+1}^n + u_j^n\right) - \frac{\Delta x}{2\,\Delta t}\left(u_{j+1}^n - u_j^n\right) \leftarrow \text{dissipation term}$$

$$\lambda_{\text{FDA}}(k) = \frac{U_k^{n+1}}{U_k^n} = \cos(k\,\Delta x) - i\,\mu\,\sin(k\,\Delta x)$$

$$\mathrm{u}_j^n = \mathrm{U}_\mathrm{k}^n \ \mathrm{e}^{i\ \mathrm{k}\, x_j} \quad ; \quad \mathrm{u}_j^{n+1} = \mathrm{U}_\mathrm{k}^{n+1} \ \mathrm{e}^{i\ \mathrm{k}\, x_j} \ \Rightarrow$$

$$\mathrm{u}_{j+1}^n = \mathrm{U}_\mathrm{k}^n \ \mathrm{e}^{i\ \mathrm{k}\, x_j + i\ \mathrm{k}\, \Delta x} = \mathrm{U}_\mathrm{k}^n \ \mathrm{e}^{i\ \mathrm{k}\, x_j} \mathrm{e}^{i\ \mathrm{k}\, \Delta x} \text{ and } \mathrm{u}_{j-1}^n = \mathrm{U}_\mathrm{k}^n \ \mathrm{e}^{i\ \mathrm{k}\, x_j \ - i\ \mathrm{k}\, \Delta x} = \mathrm{U}_\mathrm{k}^n \ \mathrm{e}^{i\ \mathrm{k}\, x_j} \mathrm{e}^{-\, i\ \mathrm{k}\, \Delta x}$$

$$\mathrm{u}_j^{n+1} = \frac{1}{2}\left(\mathrm{u}_{j+1}^n + \mathrm{u}_{j-1}^n\right) \ - \ \frac{\mu}{2} \ \left(\mathrm{u}_{j+1}^n - \mathrm{u}_{j-1}^n\right)$$

$$\lambda_{\mathrm{FDA}}(\mathrm{k}) = \frac{\mathrm{U}_k^{n+1}}{\mathrm{U}_k^n} = \cos\,(\ \mathrm{k}\ \Delta x) - i\ \mu\ \sin\,(\ \mathrm{k}\ \Delta x)$$

$$\lambda_{\mathrm{FDA}}(\mathrm{k}) = \frac{\mathrm{U}_k^{n+1}}{\mathrm{U}_k^{n}} = \cos(\mathrm{k}\,\Delta x) - i\,\mu\,\sin(\mathrm{k}\,\Delta x)$$

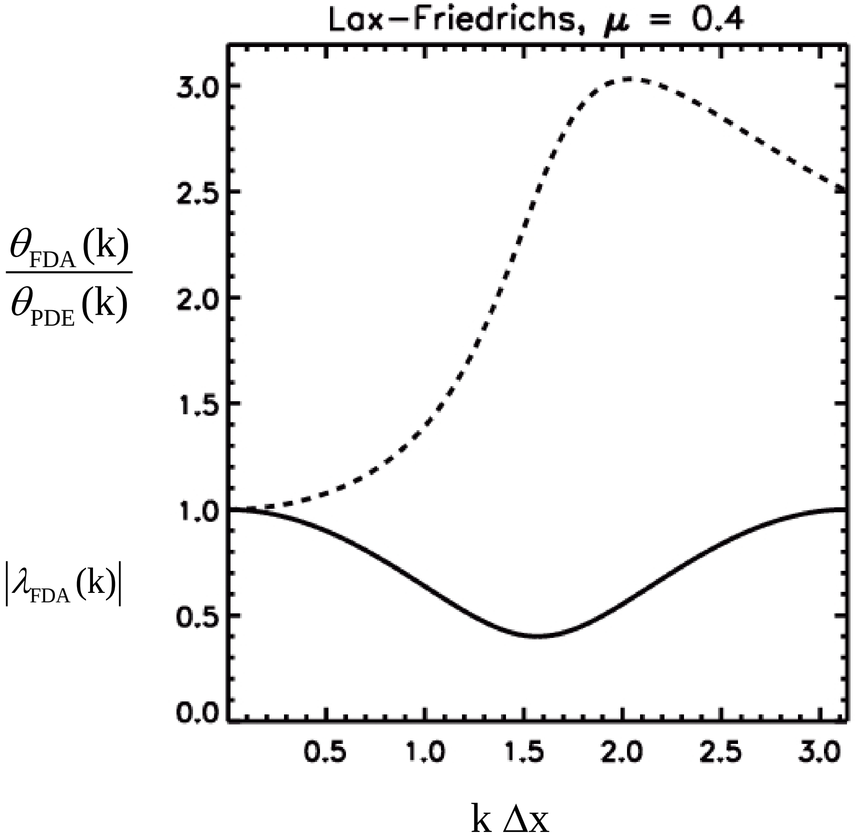
$$\left|\lambda_{\mathrm{FDA}}(\mathrm{k})\right| = \sqrt{1 + (\mu^2 - 1)\sin^2(\mathrm{k}\,\Delta x)}$$ ← Scheme is stable for CFL number $|\mu| \le 1$

$$\frac{\theta_{\mathrm{FDA}}(\mathrm{k})}{\theta_{\mathrm{PDE}}(\mathrm{k})} = \frac{\tan^{-1}\left[\mu\,\tan(\mathrm{k}\,\Delta x)\right]}{\mu\,(\mathrm{k}\,\Delta x)}$$

Notice that $\lim_{k\Delta x \to 0} \left|\lambda_{\mathrm{FDA}}(\mathrm{k})\right| = 1$

and $\lim_{k\Delta x \to 0} \frac{\theta_{\mathrm{FDA}}(\mathrm{k})}{\theta_{\mathrm{PDE}}(\mathrm{k})} = 1$

⇒ long wavelength modes are advected faithfully!

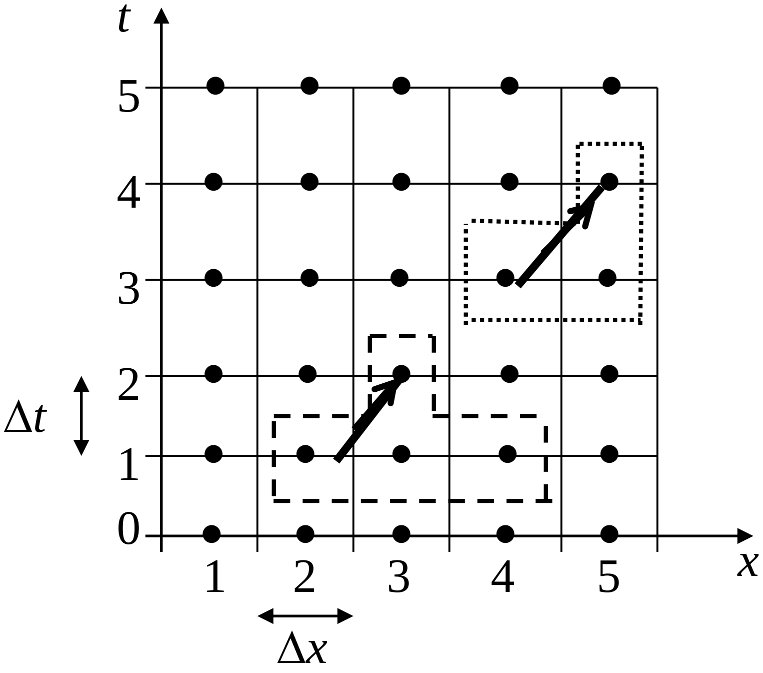


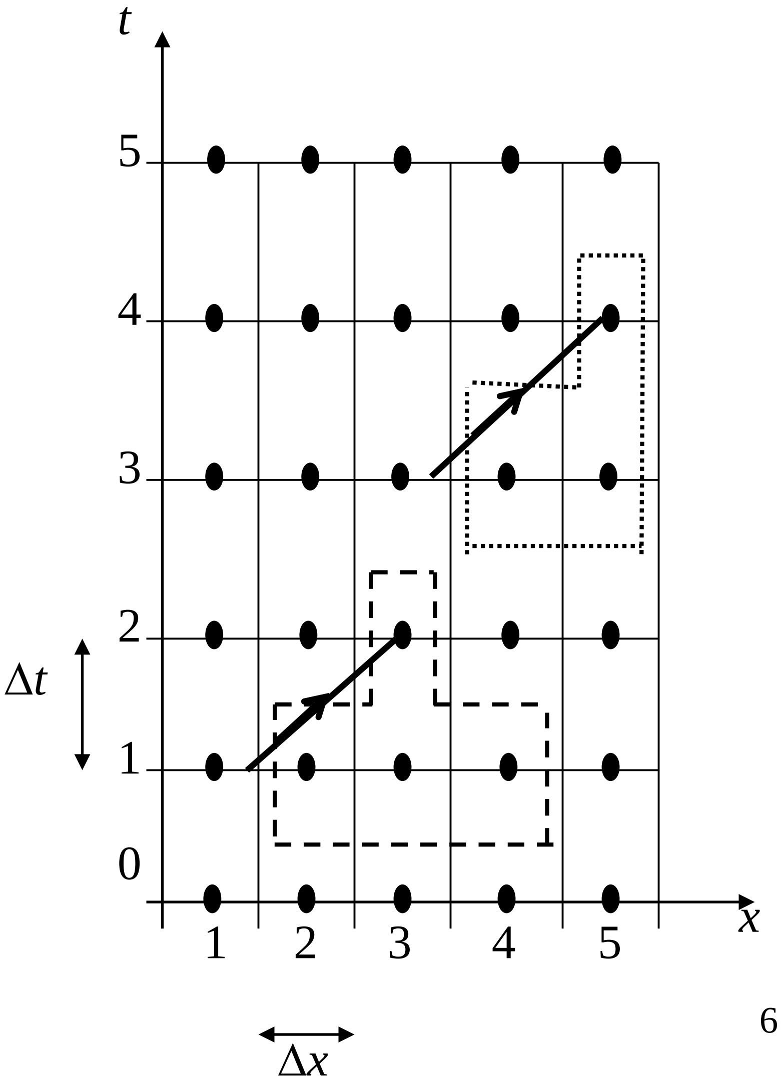
Question: What can you say about advection of short wavelength modes?

Observe the domain of dependence of the Lax-Friedrich scheme (dashed stencil)

The solid arrow also shows the characteristics.

Question: What do the characteristics tell us about the CFL time step restriction?





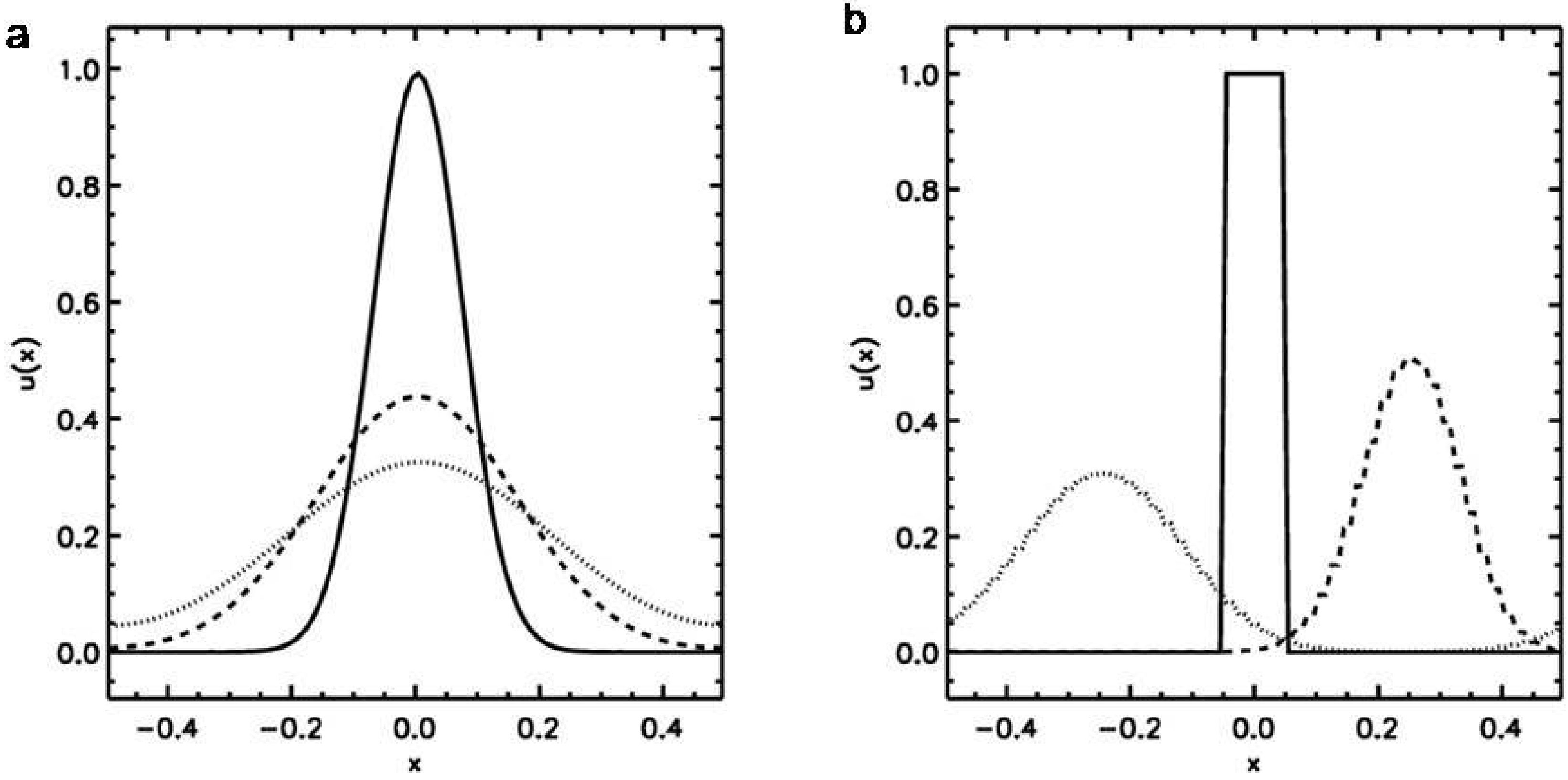


Fig 2.15a and 2.15b show the solution from the Lax-Friedrich scheme for the scalar advection equation with initial conditions given by a Gaussian profile and a top-hat profile respectively. The times in Fig. 2.15a are t=0 (solid line), t=1 (dashed line) and t=2 (dotted line). The times in Fig. 2.15b are t=0 (solid line), t=0.25 (dashed line) and t=0.75 (dotted line).

The above figures show advection of a Gaussian and a top-hat profile.

Question: Relate the deficiencies that you see in these simulations to the above dispersion analysis.

2.7.3) Lax-Wendroff Scheme

The Lax-Friedrichs scheme was first order and very dissipative. So we *try to build a second order scheme*. Resort to Taylor series expansion.

The *Lax-Wendroff procedure* shown here is, in fact, a common building block for numerical schemes, even though the Lax-Wendroff scheme designed here is seldom used.

$$\mathrm{u}\,(x_j, t^n + \Delta t) = \mathrm{u}\,(x_j, t^n) + \Delta t\ \mathrm{u}_t\ (x_j, t^n) + \frac{1}{2}\Delta t^2\ \mathrm{u}_{tt}\ (x_j, t^n) + \ldots$$

Now use the governing equation: $\mathrm{u}_t + \mathrm{a}\,\mathrm{u}_x = 0$ to get : $\mathrm{u}_t = -\,\mathrm{a}\,\mathrm{u}_x$ and $\mathrm{u}_{tt} = -\,\mathrm{a}\,\mathrm{u}_{xt} = -\,\mathrm{a}\,\mathrm{u}_{tx} = \mathrm{a}^2\ \mathrm{u}_{xx}$.

Substituting one gets: $\mathrm{u}\,(x_j, t^n + \Delta t) = \mathrm{u}\,(x_j, t^n)\ -\ \mathrm{a}\,\Delta t\ \mathrm{u}_x\ (x_j, t^n) + \dfrac{1}{2}\ \mathrm{a}^2\ \Delta t^2\ \mathrm{u}_{xx}\ (x_j, t^n)$

Which yields:

$$\boxed{\begin{aligned}
&\frac{\mathrm{u}_j^{n+1} - \mathrm{u}_j^n}{\Delta t} = -\,\mathrm{a}\left(\frac{\mathrm{u}_{j+1}^n - \mathrm{u}_{j-1}^n}{2\ \Delta x}\right) + \frac{1}{2}\ \Delta t\ \mathrm{a}^2\ \left(\frac{\mathrm{u}_{j+1}^n - 2\ \mathrm{u}_j^n + \mathrm{u}_{j-1}^n}{\Delta x^2}\right) \Leftrightarrow \\
&\mathrm{u}_j^{n+1} = \mathrm{u}_j^n - \frac{\mu}{2}\left(\mathrm{u}_{j+1}^n - \mathrm{u}_{j-1}^n\right) + \frac{\mu^2}{2}\left(\mathrm{u}_{j+1}^n - 2\ \mathrm{u}_j^n + \mathrm{u}_{j-1}^n\right)
\end{aligned}}$$

$$\mathrm{u}\,(x_j, t^n + \Delta t) = \mathrm{u}\,(x_j, t^n) + \Delta t\ \mathrm{u}_t\ (x_j, t^n) + \frac{1}{2}\Delta t^2\ \mathrm{u}_{tt}\ (x_j, t^n) + \ldots$$

Use Lax-Wendroff procedure (switch time derivatives with spatial derivatives)

using $\mathrm{u}_t + \mathrm{a}\,\mathrm{u}_x = 0$

$$\mathrm{u}\,(x_j, t^n + \Delta t) = \mathrm{u}\,(x_j, t^n) - \mathrm{a}\,\Delta t\ \mathrm{u}_x\ (x_j, t^n) + \frac{1}{2}\ \mathrm{a}^2\ \Delta t^2\ \mathrm{u}_{xx}\ (x_j, t^n)$$

$$\frac{\mathrm{u}_j^{n+1} - \mathrm{u}_j^n}{\Delta t} = -\mathrm{a}\left(\frac{\mathrm{u}_{j+1}^n - \mathrm{u}_{j-1}^n}{2\ \Delta x}\right) + \frac{1}{2}\ \Delta t\ \mathrm{a}^2\ \left(\frac{\mathrm{u}_{j+1}^n - 2\ \mathrm{u}_j^n + \mathrm{u}_{j-1}^n}{\Delta x^2}\right) \quad \leftarrow \quad \text{Forward Euler + What?}$$

$$\mathrm{u}_j^n = \mathrm{U}_\mathrm{k}^n \ \mathrm{e}^{i\,\mathrm{k}\,x_j} \quad ; \quad \mathrm{u}_j^{n+1} = \mathrm{U}_\mathrm{k}^{n+1} \ \mathrm{e}^{i\,\mathrm{k}\,x_j} \ \Rightarrow$$

$$\mathrm{u}_{j+1}^n = \mathrm{U}_\mathrm{k}^n \ \mathrm{e}^{i\,\mathrm{k}\,x_j + i\,\mathrm{k}\,\Delta x} = \mathrm{U}_\mathrm{k}^n \ \mathrm{e}^{i\,\mathrm{k}\,x_j} \mathrm{e}^{i\,\mathrm{k}\,\Delta x} \text{ and } \mathrm{u}_{j-1}^n = \mathrm{U}_\mathrm{k}^n \ \mathrm{e}^{i\,\mathrm{k}\,x_j \ - i\,\mathrm{k}\,\Delta x} = \mathrm{U}_\mathrm{k}^n \ \mathrm{e}^{i\,\mathrm{k}\,x_j} \mathrm{e}^{-i\,\mathrm{k}\,\Delta x}$$

$$\mathrm{u}_j^{n+1} = \mathrm{u}_j^n - \frac{\mu}{2}\left(\mathrm{u}_{j+1}^n - \mathrm{u}_{j-1}^n\right) + \frac{\mu^2}{2}\left(\mathrm{u}_{j+1}^n - 2\ \mathrm{u}_j^n + \mathrm{u}_{j-1}^n\right)$$

$$\lambda_{\mathrm{FDA}}(\mathrm{k}) = \frac{\mathrm{U}_k^{n+1}}{\mathrm{U}_k^n} = 1 - i\ \mu\ \sin\,(\ \mathrm{k}\ \Delta x)\ -\ 2\ \mu^2\ \sin^2\ (\ \mathrm{k}\ \Delta x\ /\ 2)$$

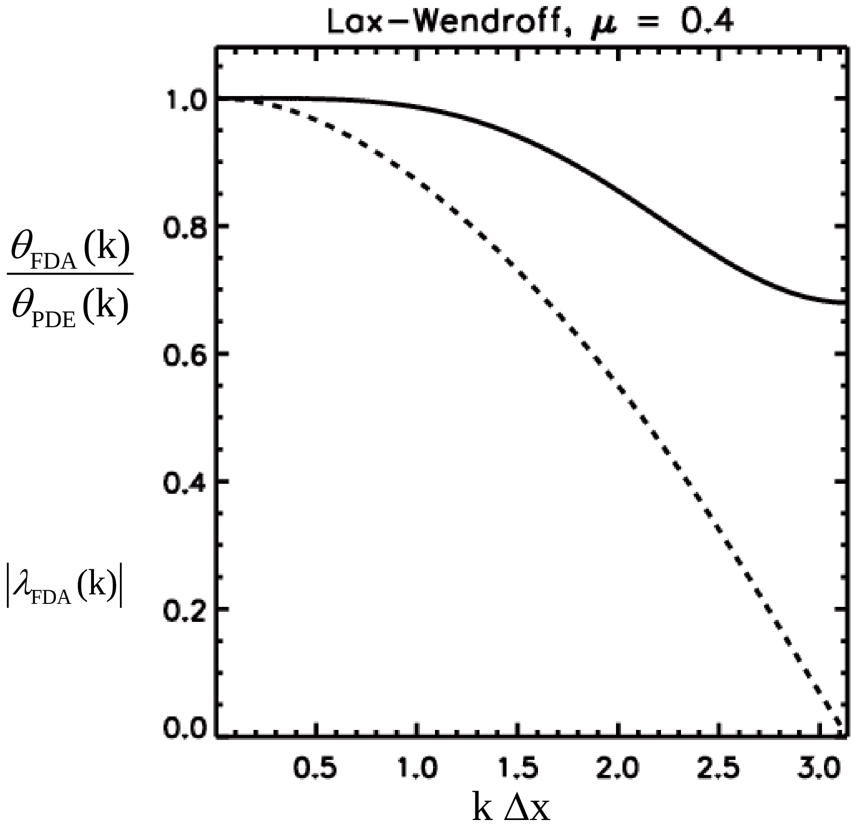
$$\lambda_{\mathrm{FDA}}(\mathrm{k}) = \frac{\mathrm{U}_k^{n+1}}{\mathrm{U}_k^{n}} = 1 - i\,\mu\,\sin(\,\mathrm{k}\,\Delta x) - 2\,\mu^2\,\sin^2(\,\mathrm{k}\,\Delta x\,/\,2)$$

$$\left|\lambda_{\mathrm{FDA}}(\mathrm{k})\right| = \sqrt{1 - 4\,\mu^2\,(1-\mu^2)\,\sin^4(\,\mathrm{k}\,\Delta x\,/\,2)}$$ ← Scheme is stable for CFL number $|\mu| \le 1$

$$\frac{\theta_{\mathrm{FDA}}(\mathrm{k})}{\theta_{\mathrm{PDE}}(\mathrm{k})} = \frac{1}{\mu\,(\mathrm{k}\,\Delta x)}\,\tan^{-1}\left\{\frac{\mu\,\sin(\,\mathrm{k}\,\Delta x)}{1 - 2\,\left[\mu\,\sin(\,\mathrm{k}\,\Delta x\,/\,2)\right]^2}\right\}$$

Questions: What can you say about advection of long wavelength modes?

What can you say about advection of short wavelength modes?



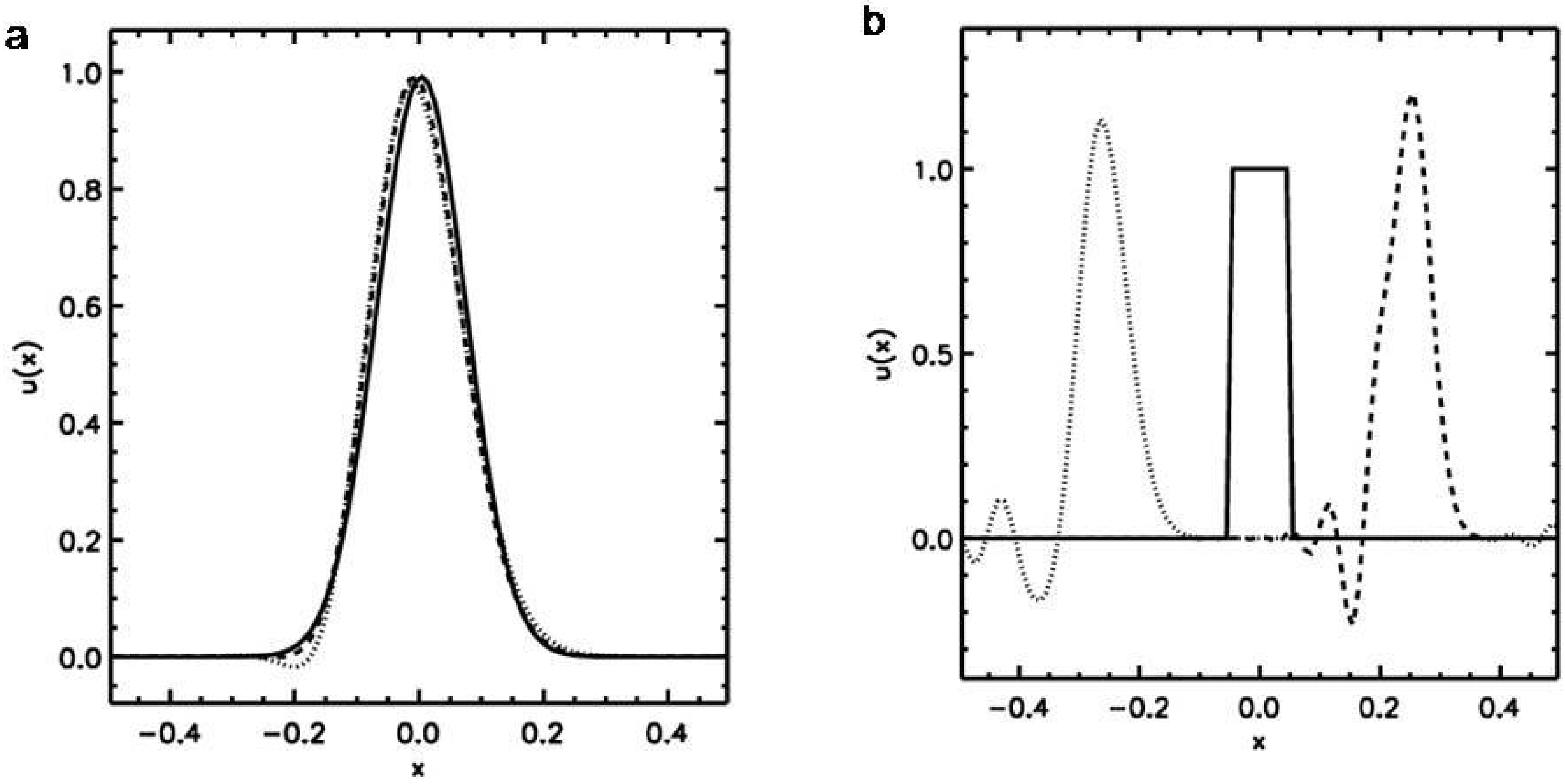


Fig 2.17a and 2.17b show the solution from the Lax-Wendroff scheme for the scalar advection equation with initial conditions given by a Gaussian profile and a top-hat profile respectively. The times in Fig. 2.17a are t=0 (solid line), t=1 (dashed line) and t=2 (dotted line). The times in Fig. 2.17b are t=0 (solid line), t=0.25 (dashed line) and t=0.75 (dotted line).

The above figures show advection of a Gaussian and a top-hat profile. The ***Gaussian is almost perfect*** The top **hat profile is very oscillatory**, ***non-positive***.

Question: Relate the deficiencies that you see in these simulations to the above dispersion analysis. What will a non-positive method do at shocks?

If the square pulse represented a pulse of fluid density, the Lax-Wendroff scheme would produce negative densities, a very undesirable situation.

The ability of an advection scheme to evolve a solution so that positive initial conditions remain so for all time is called the *positivity property*.

To see it, rewrite the scheme and observe the negative coefficients below:

$$\mathrm{u}_j^{n+1} = \left(1-\mu^2\right)\mathrm{u}_j^n - \frac{\mu}{2}\left(1-\mu\right)\mathrm{u}_{j+1}^n + \frac{\mu}{2}\left(1+\mu\right)\mathrm{u}_{j-1}^n$$

A rather pessimistic theorem by Godunov says that all *linear* *positivity-preserving* schemes are *condemned to be first order accurate*!

2.7.4) Two-Stage Runge-Kutta Scheme

We *try to build a second order scheme* by resorting to second order Runge-Kutta in time and centered fluxes in space.

Nice thing here is that we can split the spatial operator from the temporal operator, making the resulting scheme easy to implement

The Runge-Kutta time stepping shown here is, in fact, a popular building block for numerical schemes, even though the Runge-Kutta scheme designed here is seldom used.

The scheme shares many strengths and weakness with Lax-Wendroff.

$$\boxed{\begin{array}{l} \mathrm{u}_j^{n+1/2} = \mathrm{u}_j^n - \dfrac{\Delta t}{2\,\Delta x}\left(\mathrm{f}_{j+1/2}^{\,n} - \mathrm{f}_{j-1/2}^{\,n}\right) \ \text{ with } \ \mathrm{f}_{j+1/2}^{\,n} = \dfrac{1}{2}\,\mathrm{a}\left(\mathrm{u}_{j+1}^n + \mathrm{u}_j^n\right) \ \leftarrow \ \text{Predictor Stage} \\[2ex] \mathrm{u}_j^{n+1} = \mathrm{u}_j^n - \dfrac{\Delta t}{\Delta x}\left(\mathrm{f}_{j+1/2}^{\,n+1/2} - \mathrm{f}_{j-1/2}^{\,n+1/2}\right) \ \text{ with } \ \mathrm{f}_{j+1/2}^{\,n+1/2} = \dfrac{1}{2}\,\mathrm{a}\left(\mathrm{u}_{j+1}^{n+1/2} + \mathrm{u}_j^{n+1/2}\right) \ \leftarrow \ \text{Corrector Stage} \end{array}}$$

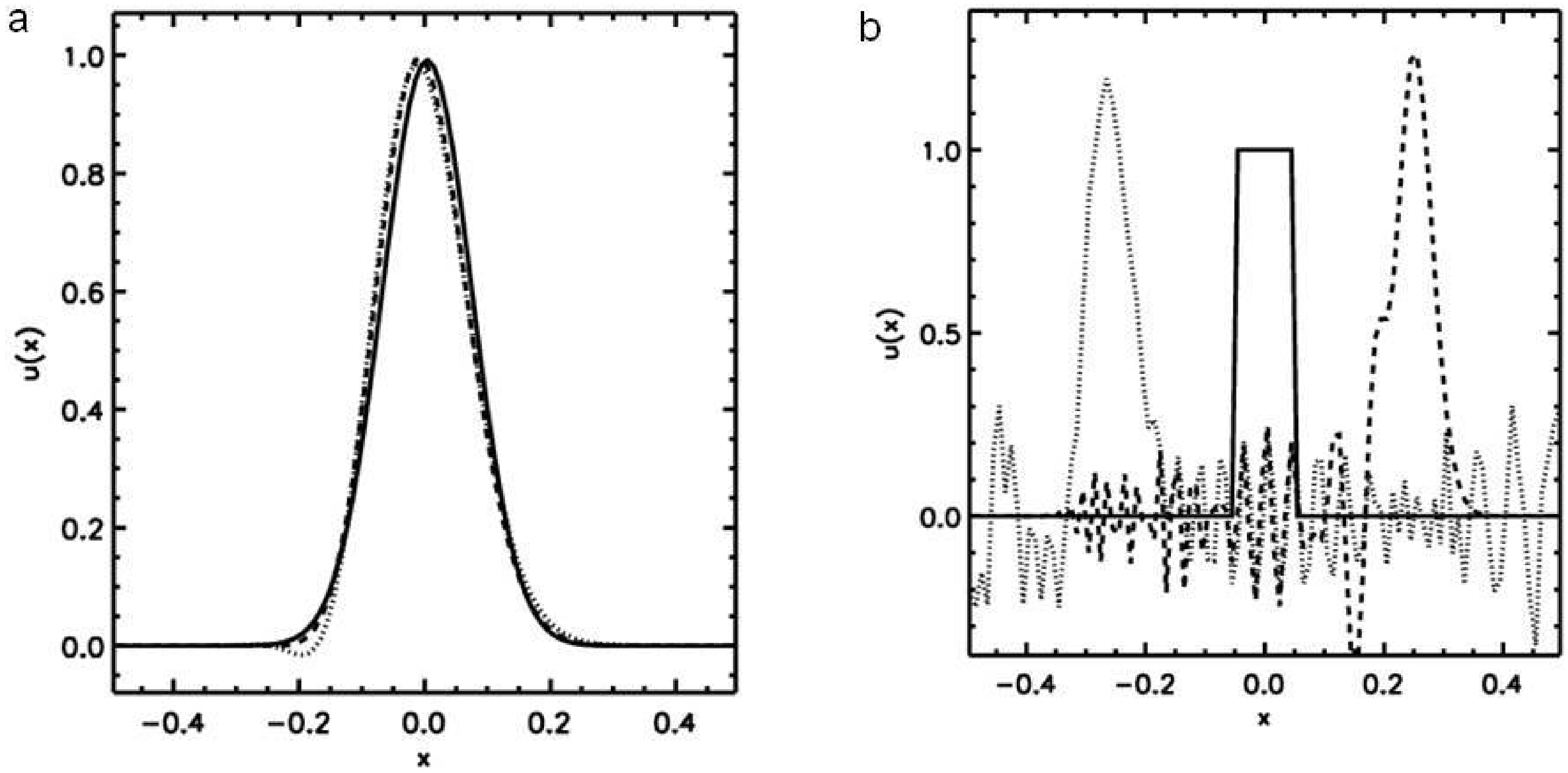


Fig 2.18a and 2.18b show the solution from the two-stage Runge-Kutta scheme for the scalar advection equation with initial conditions given by a Gaussian profile and a top-hat profile respectively. The times in Fig. 2.18a are t=0 (solid line), t=1 (dashed line) and t=2 (dotted line). The times in Fig. 2.18b are t=0 (solid line), t=0.25 (dashed line) and t=0.75 (dotted line).

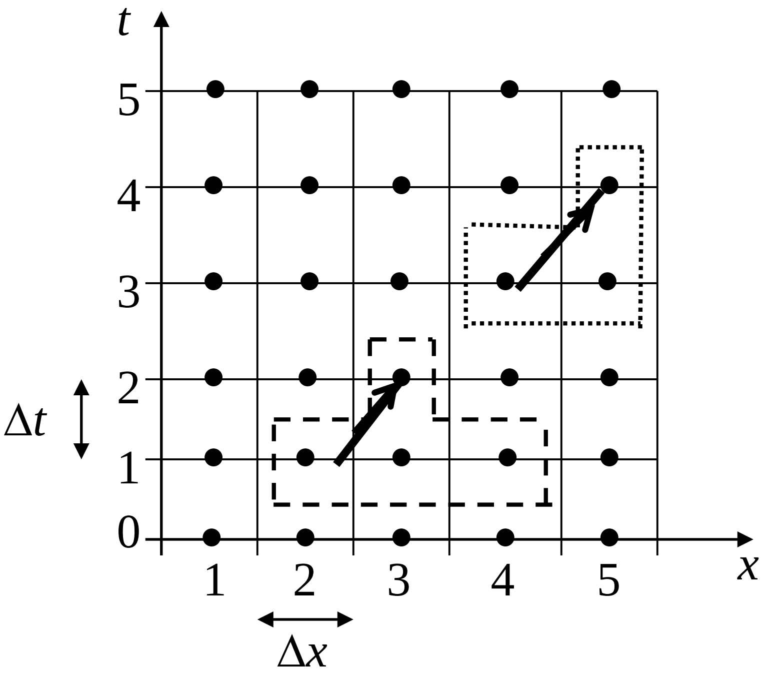
The ability of an advection scheme to evolve a solution so that positive initial conditions remain so for all time is called the ***positivity property***. The Lax-Wendroff and Runge-Kutta schemes clearly lack such a property.

2.7.5) First Order Upwind Scheme

Realize that information always flows from the *upwind direction* in the advection equation. We try to build that intuition into our scheme in the simplest way.

For a > 0 we have: $$\boxed{\frac{u_j^{n+1} - u_j^n}{\Delta t} = -a\left(\frac{u_j^n - u_{j-1}^n}{\Delta x}\right)}$$

The scheme is also called the *donor cell scheme*. It is *positivity preserving*.



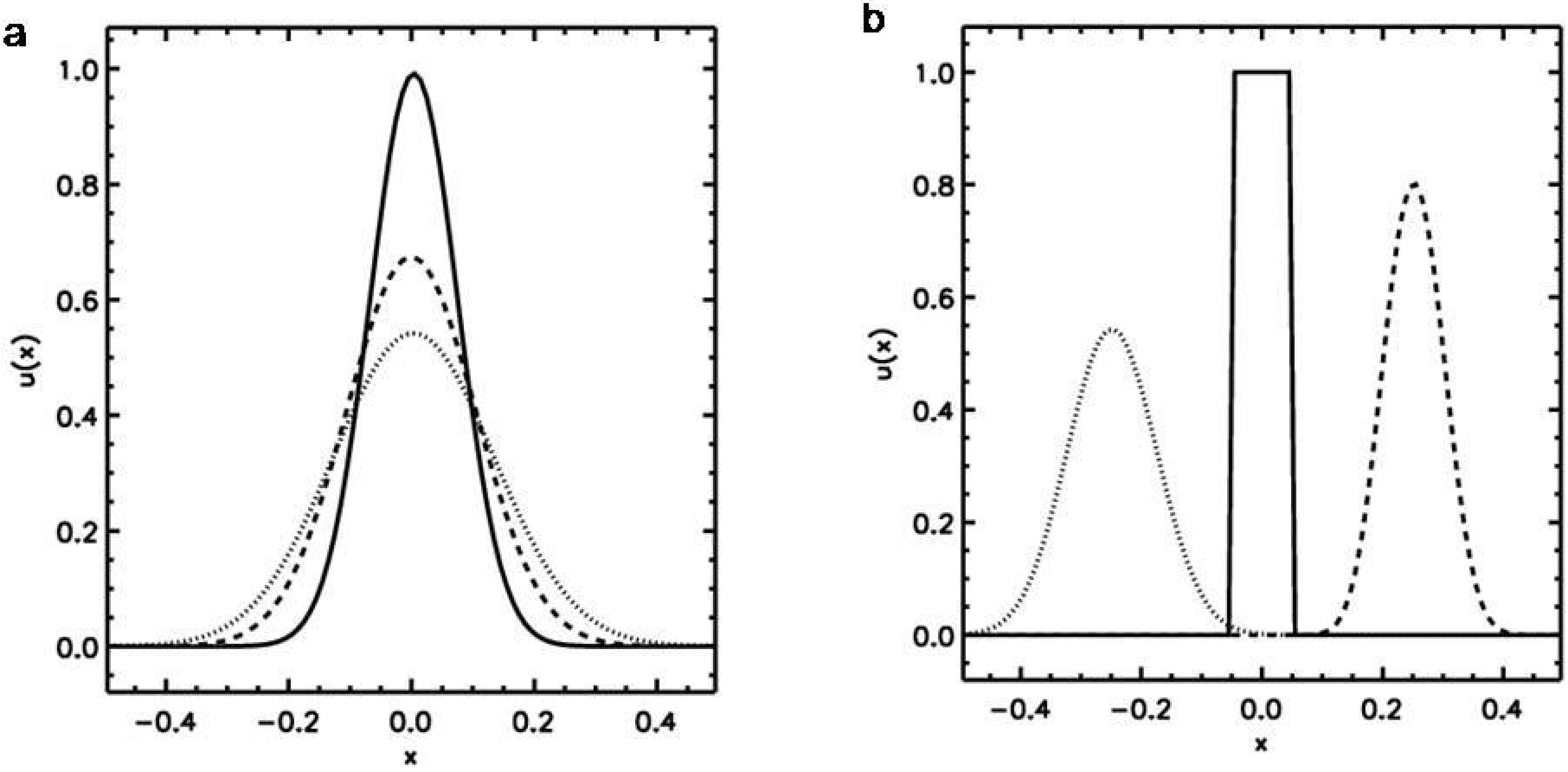


Fig 2.19a and 2.19b show the solution from the first order upwind scheme for the scalar advection equation with initial conditions given by a Gaussian profile and a top-hat profile respectively. The times in Fig. 2.19a are t=0 (solid line), t=1 (dashed line) and t=2 (dotted line). The times in Fig. 2.19b are t=0 (solid line), t=0.25 (dashed line) and t=0.75 (dotted line).

Gaussian shows dissipation because of first order accuracy. However, the ***top-hat profile is oscillation-free***! We wish to retain this desirable trait.

2.7.6) Section Summary for Hyperbolic Systems

Second order schemes did very well for smooth profiles, like the Gaussian.

They are deficient for discontinuous profiles, like the top-hat profile.

First order upwind scheme did the best of all for discontinuous profiles.

Desirable to combine the best traits of both: In smooth regions, retain full second order accuracy; at discontinuities, use elements of the first order solution. ***Positivity*** at discontinuities is an important issue.

Within the confines of ***linear*** *schemes (even for linear PDEs)* the *Lax-Richtmeyer theorem* tells us that this is not possible.

The only way out is to resort to ***non-linear hybridization*** *(even for linear PDEs).* We will find a way to pick the second order solution in regions of smooth flow while backing away from it locally at discontinuities.