Numerical Integration of Partial Differential Equations (PDEs)

- Time dependent Problems.

Time dependent Problems

- Time dependent PDEs in conservative form.
  - -Explicit schemes, Euler method.
  - -What is numerical stability? CFL-condition.
  - -Lax, Lax-Wendroff, Leap-Frog, upwind
- Diffusive processes.
  - -Diffusion equation in conservative form?
  - -Explicit and implicit methods.

Time dependent problems

Time dependent initial value problems in Flux-conservative form:

$$\frac{\partial \mathbf{u}}{\partial t} = -\frac{\partial \mathbf{F}(\mathbf{u})}{\partial x}$$

Where **F** is the conserved flux.

For simplicity we study only problems in one spatial dimension **u**=**u**(x,t)

Many relevant time dependent problems can be written in this form

For example the wave equation: $$\frac{\partial^2 u}{\partial t^2} = v^2 \frac{\partial^2 u}{\partial x^2}$$

Can be written as:

$$\frac{\partial r}{\partial t} = v \frac{\partial s}{\partial x} \qquad r \equiv v \frac{\partial u}{\partial x}$$

$$\frac{\partial s}{\partial t} = v \frac{\partial r}{\partial x} \qquad s \equiv \frac{\partial u}{\partial t}$$

$$\frac{\partial \mathbf{u}}{\partial t} = -\frac{\partial \mathbf{F}(\mathbf{u})}{\partial x} \qquad \mathbf{F}(\mathbf{u}) = \begin{pmatrix} 0 & -v \\ -v & 0 \end{pmatrix} \cdot \mathbf{u}$$

Remember derivation of wave equations from Maxwell equations. Here: 1D case

MHD in flux conservative form

$$\partial_t \rho + \nabla \cdot (\rho \boldsymbol{v}) = 0$$

$$\partial_t (\rho \boldsymbol{v}) + \nabla \cdot \left[\rho \boldsymbol{v} \otimes \boldsymbol{v} + \left(P + \frac{1}{8\pi} \boldsymbol{B}^2 \right) \mathbf{I} - \frac{1}{4\pi} \boldsymbol{B} \otimes \boldsymbol{B} \right] = 0$$

$$\partial_t \boldsymbol{B} + \nabla \cdot \left[\boldsymbol{v} \otimes \boldsymbol{B} - \boldsymbol{B} \otimes \boldsymbol{v} \right] = 0$$

$$\partial_t (\rho E) + \nabla \cdot \left[\left(\rho E + P + \frac{\boldsymbol{B}^2}{8\pi} \right) \boldsymbol{v} - \frac{1}{4\pi} (\boldsymbol{v} \cdot \boldsymbol{B}) \boldsymbol{B} \right] = 0$$

MHD in flux conservative form

$$\boldsymbol{u} = \begin{pmatrix} \rho \\ \rho\,\boldsymbol{v} \\ \boldsymbol{B} \\ e \end{pmatrix}; \boldsymbol{F} = \boldsymbol{F}(\boldsymbol{u})$$

$$\Rightarrow \partial_t \boldsymbol{u} + \nabla \cdot \boldsymbol{F} = 0$$

Advection Equation

$$\frac{\partial u}{\partial t} = -v\frac{\partial u}{\partial x}$$

The method we study used to solve this equation can be generalized:

- vectors **u**(x,y,z,t)
- 2D and 3D spatial dimensions
- Some nonlinear forms for **F**(**u**)

Explicit and Implicit Methods

- Explicit scheme:

$$Y(t + \Delta t) = F(Y(t))$$

- Implicit scheme:

$$G(Y(t), Y(t + \Delta t)) = 0$$

Aim: Find $Y(t + \Delta t)$

More afford necessary for implicit scheme.

$$\frac{\partial u}{\partial t} = -v\frac{\partial u}{\partial x}$$

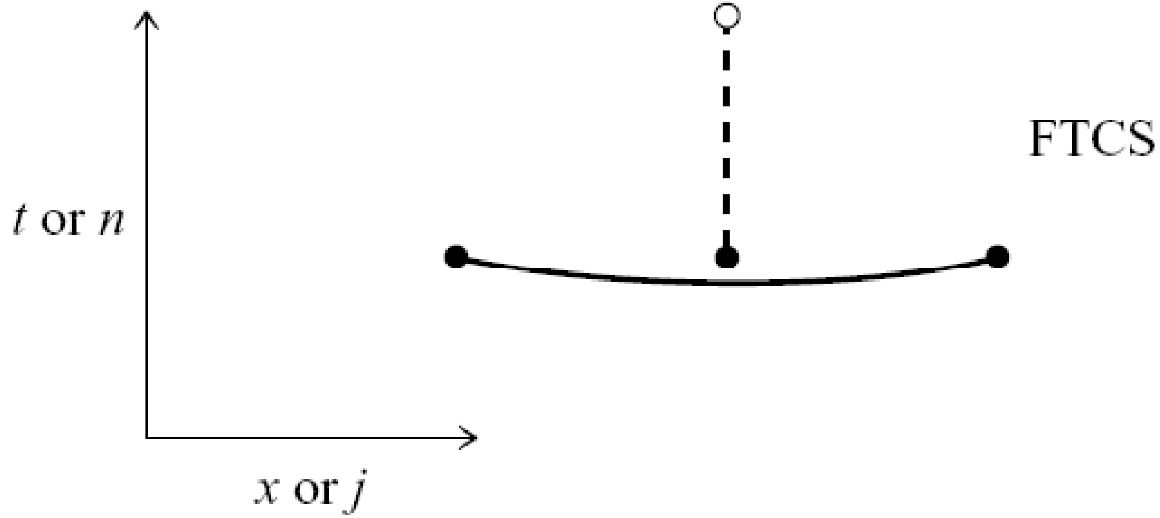
We try to solve this equation with discretisation in space and time:

$$\left.\frac{\partial u}{\partial t}\right|_{j,n} = \frac{u_j^{n+1} - u_j^n}{\Delta t} + O(\Delta t)$$ Forward in time

$$\left.\frac{\partial u}{\partial x}\right|_{j,n} = \frac{u_{j+1}^n - u_{j-1}^n}{2\Delta x} + O(\Delta x^2)$$ Centered in space

Euler method, FTCS

Forward in Time Centered in Space



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

Euler method, FTCS

Forward in Time Centered in Space

Show: demo_advection.pro

This is an IDL-program to
solve the advection equation
with different numerical schemes.

Euler method, FTCS

- Explicit scheme and easy to derive.
- Needs little storage and executes fast.
- Big disadvantage:
  FTCS-Method is basically useless!
- Why?
- Algorithm is numerical unstable.

Leonard Euler
1707-1783

What is numerical stability?

Say we have to add 100 numbers of array a[i] using a computer with only 2 significant digits.

```
sum = 0
for i = 1 to 100 do sum = sum + a[i]
```

- Looks reasonable, doesn't it?
- But imagine a[0]=1.0 and all other a[i]=0.01
- Our two-digit computer gets: sum=1.0
- Better algorithm: Sort first a[i] by absolute values
- Two-digit comp gets: sum=2.0, which is a much better approximation of the true solution 1.99

Can we check if a numerical scheme is stable without computation? YES: Von Neumann stability analysis

John von Neumann
1903-1957

- Analyze if (or for which conditions) a numerical scheme is stable or unstable.
- Makes a local analysis, coefficients of PDE are assumed to vary slowly (our example: constant).
- How will unavoidable errors (say rounding errors) evolve in time?

Von Neumann stability analysis

Ansatz: $u_j^n = \xi^n e^{ikj\Delta x}$

Wave number k and amplification factor:

$$\xi = \xi(k)$$

A numerical scheme is unstable if:

$$|\xi(k)| > 1$$

Von Neumann stability analysis

We investigate the Euler scheme:

$$u_j^{n+1} = u_j^n + \frac{v\Delta t}{2\Delta x}(u_{j+1}^n - u_{j-1}^n)$$

and make the ansatz: $u_j^n = \xi^n \exp(ikj\Delta x)$

$$\xi^{n+1}\exp(ikj\Delta x) = \xi^n \exp(ikj\Delta x) + \frac{v\Delta t}{2\Delta x}(\xi^n \exp(ik(j+1)\Delta x) - \xi^n \exp(ik(j-1)\Delta x))$$

Von Neumann stability analysis

We divide by $\xi^n \exp(ikj\Delta x)$ and get:

$$\xi = 1 + \frac{v\Delta t}{2\Delta x}(\exp(ik\Delta x) - \exp(-ik\Delta x))$$

which can be written as:

$$\xi = 1 - i\frac{v\Delta t}{\Delta x}\sin(k\Delta x)$$

Von Neumann stability analysis

We get the amplitude $|\xi| = \sqrt{\xi\xi^*}$,
where ξ^* is the conjugate complex of ξ:

$$|\xi| = \sqrt{1 + \left(\frac{v\Delta t}{\Delta x}\right)^2 \sin^2(k\Delta x)} > 1$$

$\Rightarrow$ Euler scheme is unconditional unstable.

Lax method

A simple way to stabilize the FTCS method has been proposed by Peter Lax:

Peter Lax, born 1926

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

This leads to

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

Von Neumann stability analysis

We investigate the Lax scheme:

$$u_j^{n+1} = \frac{1}{2}(u_{j+1}^n + u_{j-1}^n) + \frac{v\Delta t}{2\Delta x}(u_{j+1}^n - u_{j-1}^n)$$

and make again the ansatz: $u_j^n = \xi^n \exp(ikj\Delta x)$

$$\xi^{n+1} \exp(ikj\Delta x) =$$
$$\frac{1}{2}(\xi^n \exp(ik(j+1)\Delta x) + \xi^n \exp(ik(j-1)\Delta x)) +$$
$$\frac{v\Delta t}{2\Delta x}(\xi^n \exp(ik(j+1)\Delta x) - \xi^n \exp(ik(j-1)\Delta x))$$

Von Neumann stability analysis

We divide by $\xi^n \exp(ikj\Delta x)$ and get:

$$\xi = \frac{1}{2}(\exp(ik\Delta x) + \exp(-ik\Delta x)) + \\ \frac{v\Delta t}{2\Delta x}(\exp(ik\Delta x) - \exp(-ik\Delta x))$$

which can be written as:

$$\xi = \cos(k\Delta x) - i\frac{v\Delta t}{\Delta x}\sin(k\Delta x)$$

Von Neumann stability analysis

We get the amplitude $|\xi| = \sqrt{\xi\xi^*}$:

$$|\xi| = \sqrt{\cos^2(k\Delta x) + \left(\frac{v\Delta t}{\Delta x}\right)^2 \sin^2(k\Delta x)} > 1$$

$\Rightarrow$ Lax scheme is conditional stable for:

$$\frac{|v|\Delta t}{\Delta x} \leq 1$$

Lax equivalence principle
or Lax Richtmyer theorem

A finite difference approximation **converges** (towards the solution of PDE) if and only if:

- The scheme is **consistent** (for dt->0 and dx->0 the difference-scheme agrees with original Differential equation.)
- And the difference scheme is **stable**.

Strictly proven only for linear initial value problem, but assumed to remain valid also for more general cases.

Courant Friedrichs Levy condition (1928)

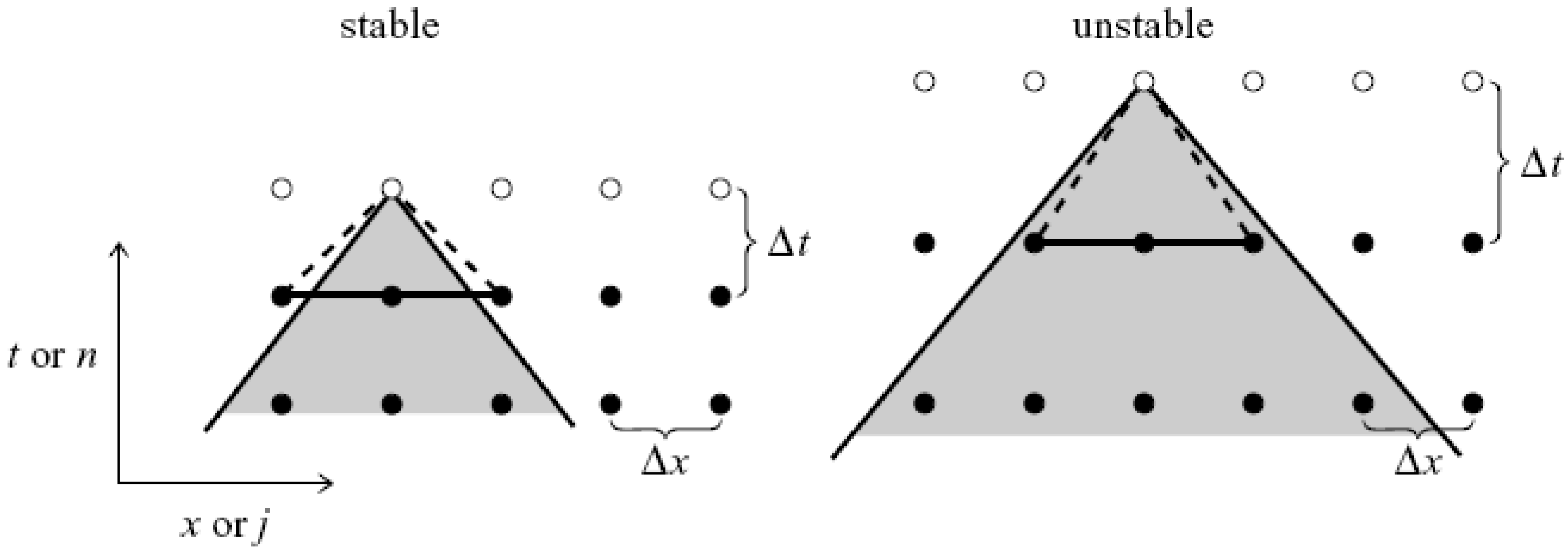
$$\frac{|v|\Delta t}{\Delta x} \leq 1$$

CFL-condition
Courant number

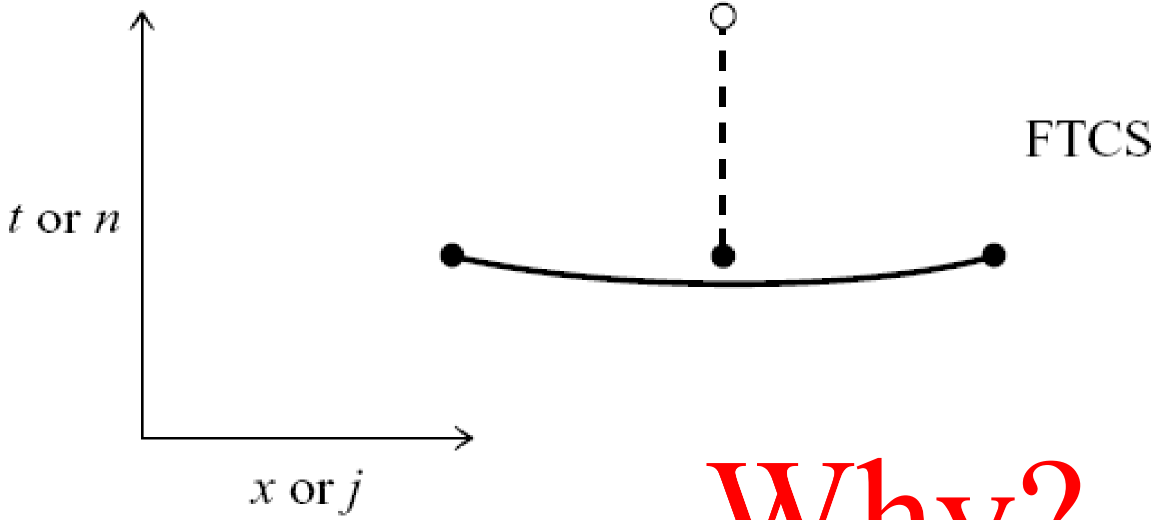
Famous stability condition in numerical mathematics
Valid for many physical applications, also in inhomogenous nonlinear cases like:

- Hydrodynamics (with **v** as sound speed)
- MHD (with **v** as Alfven velocity)

CFL-condition

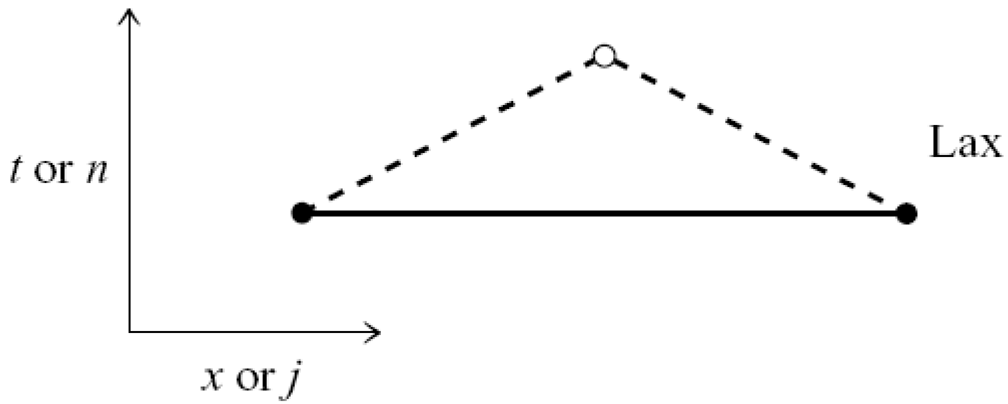


Value at a certain point depends on information within some area (shaded) as defined by the PDE. (say advection speed v, wave velocity or speed of light.) These physical points of dependency must be inside the computational used grid points for a stable method.



Unstable

Why?



Stable

Lax method

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

We write the terms a bit different:

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

and translate the difference equation back into a PDE in using the FTCS-scheme:

$$\underbrace{\frac{\partial u}{\partial t} = -v\frac{\partial u}{\partial x}}_{\text{Original PDE}} + \frac{(\Delta x)^2}{2\Delta t}\nabla^2 u$$

Original PDE

Diffusion term

Lax Method

- Stable numerical scheme (if CFL fulfilled)
- But it solves the wrong PDE!
- How bad is that?
- Answer: Not that bad.
  The dissipative term mainly damps small spatial structures on grid resolution, which we are not interested in. => **Numerical dissipation**
- The unstable FTCS-method blows this small scale structures up and spoils the solution.

Sorry to Leonard Euler

- We should not refer to Euler entirely negative for developing an unstable numerical scheme.
- He lived about 200 years before computers have been developed and the performance of schemes has been investigated.
- Last but not least:
  The Euler-scheme is indeed stable for some other applications, e.g. the Diffusion equation.

Phase Errors

- We rewrite the stability condition:

$$\xi = e^{-ik\Delta x} + i\left(1 - \frac{v\Delta t}{\Delta x}\right)\sin k\Delta x$$

- A wave packet is a superposition of many waves with different wave numbers k.
- Numerical scheme multiplies modes with different phase factors.
- => Numerical **dispersion**.
- The method is exact if CFL is fulfilled exactly:

$$\Delta t = \Delta x / v$$

(Helps here but not in inhomogenous media.)

Lax method

Show: demo_advection.pro

This is an IDL-program to
solve the advection equation
with different numerical schemes.

Nonlinear instabilities

- Occur only for nonlinear PDEs like:

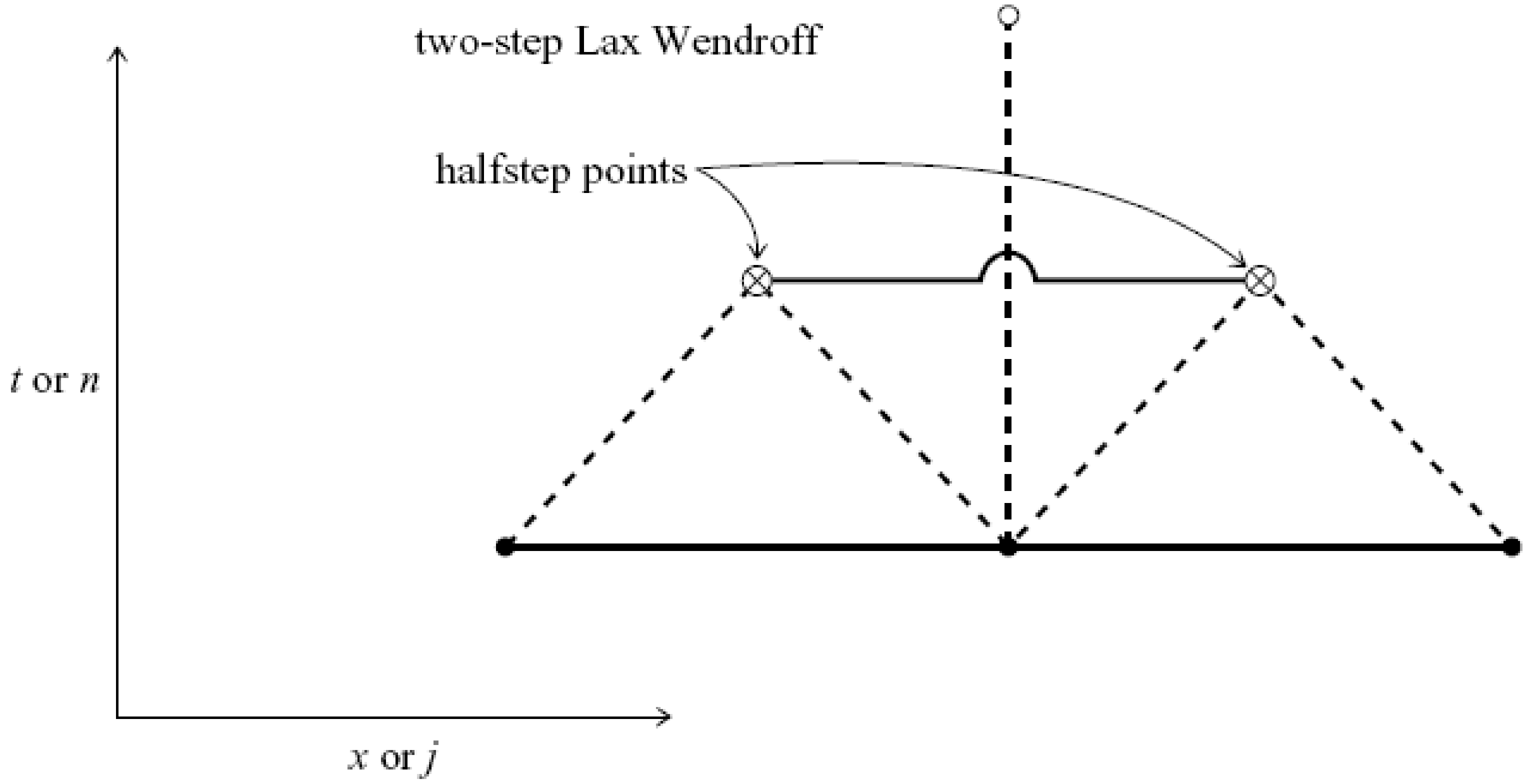
$$\frac{\partial v}{\partial t} = -v\frac{\partial v}{\partial x} + \ldots$$

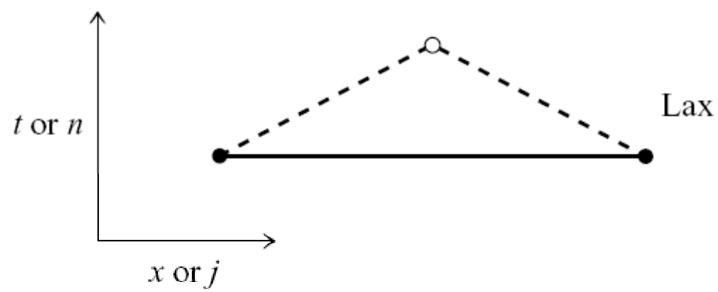
- Von Neumann stability analysis linearizes the nonlinear term and suggests stability.
- For steep profiles (shock formation) the nonlinear term can transfer energy from long to small wavelength.
- Can be controlled (stabilized) by numerical viscosity.
- Not appropriate if you actually want to study shocks.

Lax-Wendroff Method

- 2 step method based on Lax Method.
- Apply first one step “Lax step” but advance only half a time step.
- Compute fluxes at this points $\mathbf{t^{n+1/2}}$
- Now advance to step $\mathbf{t^{n+1}}$ by using points at $\mathbf{t^{n}}$ and $\mathbf{t^{n+1/2}}$
- Intermediate Results at $\mathbf{t^{n+1/2}}$ not needed anymore.
- Scheme is second order in space and time.

Lax-Wendroff Method





Lax-Wendroff Method

Lax step

$$u_{j+1/2}^{n+1/2} = \frac{1}{2}(u_{j+1}^{n} + u_{j}^{n}) - \frac{\Delta t}{2\Delta x}(F_{j+1}^{n} - F_{j}^{n})$$

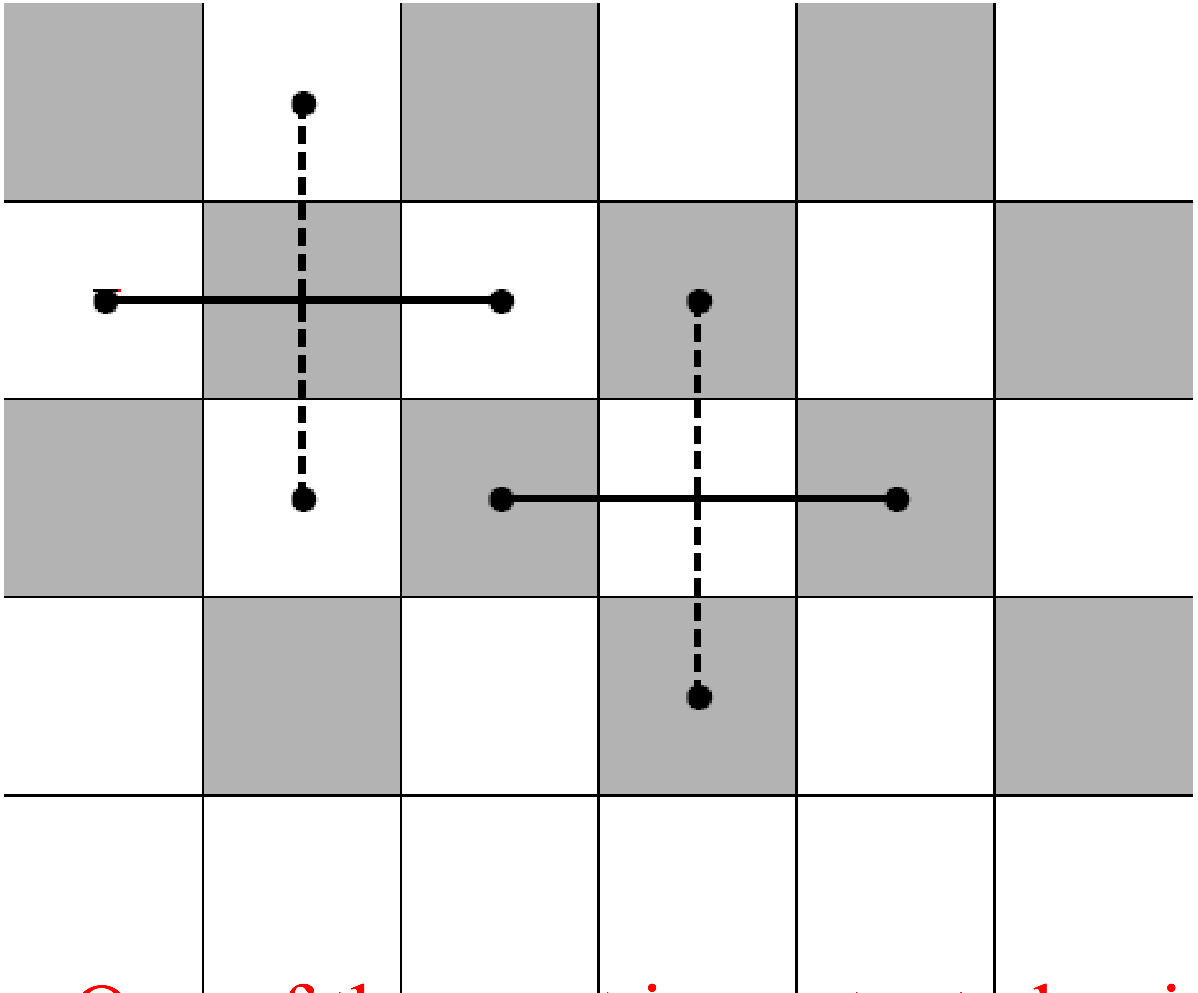
Compute Fluxes at n+1/2 and then:

$$u_{j}^{n+1} = u_{j}^{n} - \frac{\Delta t}{\Delta x}\left(F_{j+1/2}^{n+1/2} - F_{j-1/2}^{n+1/2}\right)$$

- Stable if CFL-condition fulfilled.
- Still diffusive, but here this is only 4th order in k, compared to 2th order for Lax method.

=> Much smaller effect.

Leap-Frog Method



Children playing leapfrog

Harlem, ca. 1930.

Scheme uses second order **central differences** in space and time.

One of the most important classical methods.

Commonly used to solve MHD-equations.

Leap-Frog method

$$u_j^{n+1} - u_j^{n-1} = -\frac{\Delta t}{\Delta x}(F_{j+1}^n - F_{j-1}^n)$$

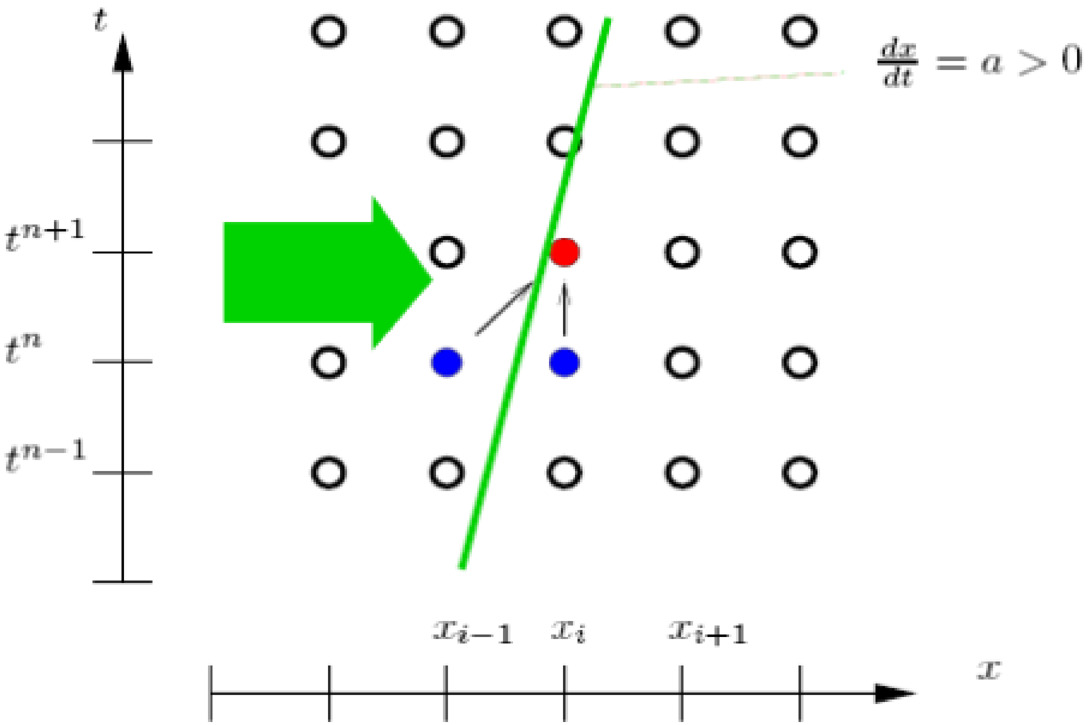
- Requires storage of previous time step.
- Von Neumann analysis shows stability under CFL-condition. $\xi = -i\frac{v\Delta t}{\Delta x}\sin k\Delta x \pm \sqrt{1 - \left(\frac{v\Delta t}{\Delta x}\sin k\Delta x\right)^2}$
- We get $|\xi|^2 = 1$ for any $v\Delta t \leq \Delta x$
- Big advantage of Leap-Frog method:
  No amplitude diffusion.

Leap-Frog method

- Popular in fluid dynamics and MHD.
- No diffusion in the Leap-Frog scheme.
- For nonlinear problems the method can become unstable if sharp gradients form.
- This is mainly because the two grids are uncoupled.
- Cure: Couple grids by adding artificial viscosity.

This is also how nature damps shocks/discontinuities: producing viscosity or resistivity by micro-instabilities.

Upwind method: A more physical approach to the transport problem.



$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = -v_j^n \begin{cases} \dfrac{u_j^n - u_{j-1}^n}{\Delta x}, & v_j^n > 0 \\ \dfrac{u_{j+1}^n - u_j^n}{\Delta x}, & v_j^n < 0 \end{cases}$$

Upwind method: A more physical approach to the transport problem.

- Upwind methods take into consideration the flow direction (different from central schemes).
- Here: only first order accuracy in space and time.
- CFL-stable for upwind direction;
  downwind direction unstable.
- Upwind methods can be generalized to higher order and combined with other methods:
  -use high order central schemes for smooth flows
  -upwind methods in regions with shocks.

Exercise: Leap-Frog, Lax-Wendroff, Upwind

lecture_advection_draft.pro

This is a draft IDL-program to
solve the advection equation.
Task: implement Leap-Frog, Lax-Wendroff, Upwind

Can be used also for other equations
in conservative form, e.g.
the nonlinear Burgers equation (see exercises)

Time dependent PDEs
Summary

- Very simple numerical schemes often do not work, because of **numerical instabilities**.
- Lax: Consistency + stability = convergence.
- **CFL-condition** (or Courant number) limits **maximum allowed time step**.
- Important are second order accurate schemes:
  -**Leap-Frog** method.
  -**Lax-Wendroff** scheme.

Diffusive processes.

- One derivation of diffusion equation.
- Diffusion equation in conservative form?
- Try to solve diffusion equation with our explicit solvers from last section.
- Application to a nonlinear equation: (Diffusive Burgers equation)
- Implicit methods: Crank-Nicolson scheme.

Maxwell Equations (displacement current neglected and without electric charges) and Ohm's law:

$$\nabla \times \mathbf{B} = \mu_0 \mathbf{j}, \;\; \nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t}, \;\; \nabla \cdot \mathbf{B} = 0$$
$$\nabla \cdot \mathbf{E} = 0, \qquad \mathbf{E} = \eta \mathbf{j}$$

We take the curl of Ohm's law: $\nabla \times \mathbf{E} = \eta \nabla \times \mathbf{j}$

For $\mathbf{E}$ and $\mathbf{j}$ we insert Maxwell equations: $-\frac{\partial \mathbf{B}}{\partial t} = \frac{\eta}{\mu_0} \nabla \times \nabla \times \mathbf{B}$
which leads to: $-\frac{\partial \mathbf{B}}{\partial t} = \frac{\eta}{\mu_0} \left(\nabla(\nabla \cdot \mathbf{B}) - \Delta \mathbf{B}\right)$

Finally we get a Diffusion equation for $\mathbf{B}$:

$$\boxed{\frac{\partial \mathbf{B}}{\partial t} = \frac{\eta}{\mu_0} \Delta \mathbf{B}}$$

Parabolic PDEs: Diffusion equation

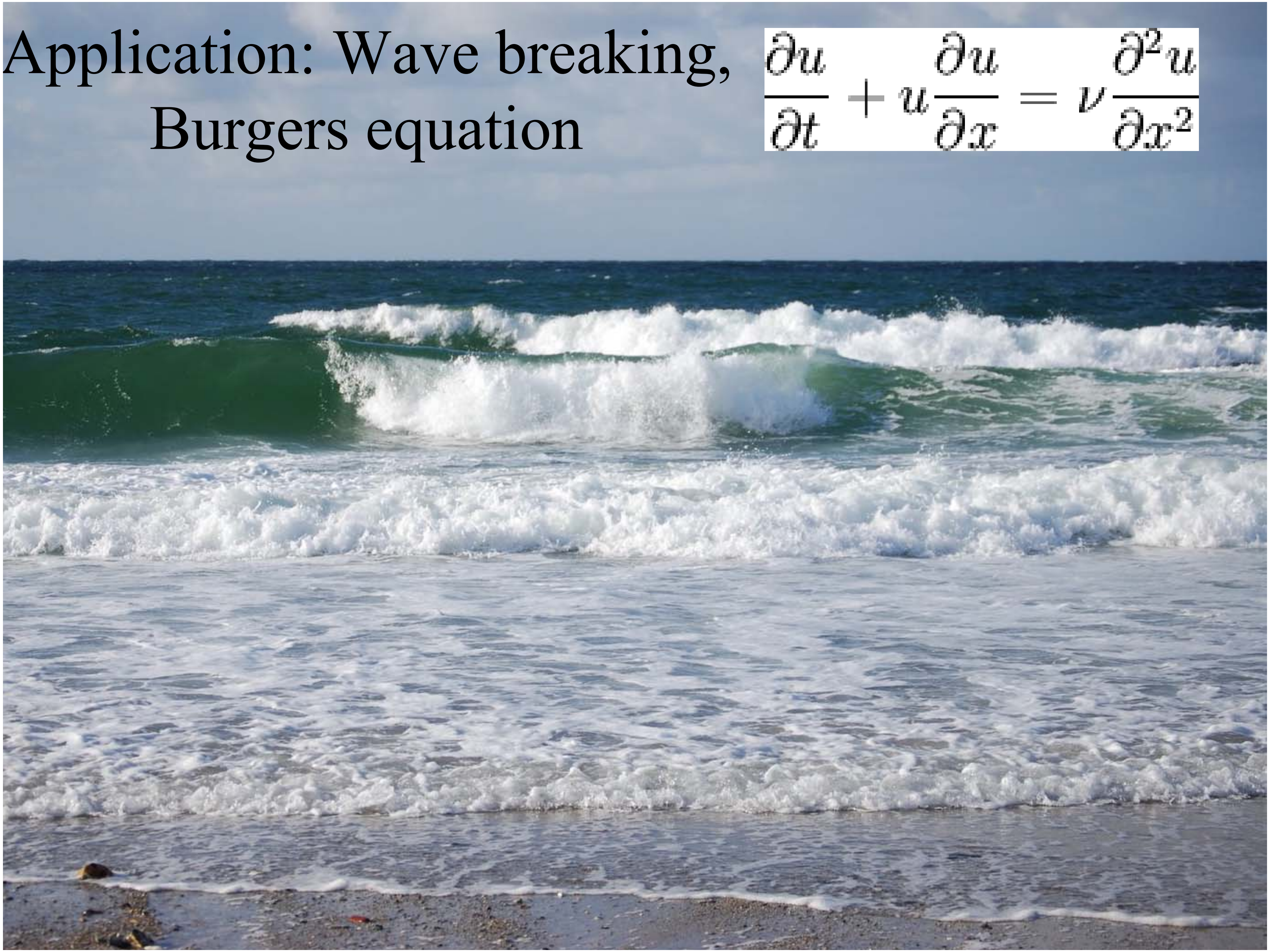
$$\frac{\partial u}{\partial t} = \frac{\partial}{\partial x}\left(D\frac{\partial u}{\partial x}\right)$$

In principle we know already how to solve this equation in the conservative form:

$$\frac{\partial \mathbf{u}}{\partial t} = -\frac{\partial \mathbf{F}(\mathbf{u})}{\partial x} \qquad F = -D\frac{\partial u}{\partial x}$$

Application: Wave breaking, Burgers equation

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = \nu\frac{\partial^2 u}{\partial x^2}$$



Diffusion equation and diffusive Burgers Equation

demo_advection.pro

- Apply our methods and check stability for: (Euler, Leap-Frog, upwind, Lax, Lax-Wendroff):

- Diffusion equation: $$\frac{\partial u}{\partial t} = \frac{\partial}{\partial x}\left(D\frac{\partial u}{\partial x}\right)$$

- Diffusive Burgers equation: $$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = \nu\frac{\partial^2 u}{\partial x^2}$$

Euler-method FTCS

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = D \left[\frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{(\Delta x)^2} \right]$$

- Euler method is conditional stable for

$$\frac{2D\Delta t}{(\Delta x)^2} \leq 1$$

- Time step way more demanding (has to be very small) compared to hyperbolic equations.
- Becomes even more restrictive if higher spatial derivatives are on the right hand side. $dt \sim (dx)^n$ for the n'th spatial derivative.

Time step restrictions

- We have to resolve the diffusion time across a spatial scale

$$\tau \sim \frac{\lambda^2}{D}$$

- And in our explicit scheme we have to resolve the smallest present spatial scale, which is the grid resolution.
- Often we are only interested in scales $\lambda \gg \Delta x$
- It takes about $\lambda^2/(\Delta x)^2$ steps until these scales are effected.

Implicit schemes

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = D \left[\frac{u_{j+1}^{n+1} - 2u_j^{n+1} + u_{j-1}^{n+1}}{(\Delta x)^2} \right]$$

- Looks very similar as FTCS-method, but contains new (t+dt) step on right side.
- This is called ‘fully implicit’ or ‘backward in time’ scheme.
- Disadvantage: We do not know the terms on the right side, but want to obtain them.
- Advantages of the method?
  Do a stability analysis!

Implicit scheme

- Von Neumann stability analysis:

$$\xi = \frac{1}{1 + 4\alpha \sin^2\left(\frac{k\Delta x}{2}\right)} \qquad \alpha \equiv \frac{D\Delta t}{(\Delta x)^2}$$

- Fully implicit method is **unconditional stable**. No restrictions on timestep.
- Stable does not mean accurate. The method is only first order accurate.

How to use an implicit scheme?

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = D\left[\frac{u_{j+1}^{n+1} - 2u_j^{n+1} + u_{j-1}^{n+1}}{(\Delta x)^2}\right] \qquad \alpha \equiv \frac{D\Delta t}{(\Delta x)^2}$$

can be rewritten to

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

and at every time step one has to solve a system of linear equations to find u_j^{n+1}. This is a large extra afford, but pays off by allowing an unrestricted time step.

Crank-Nicolson scheme

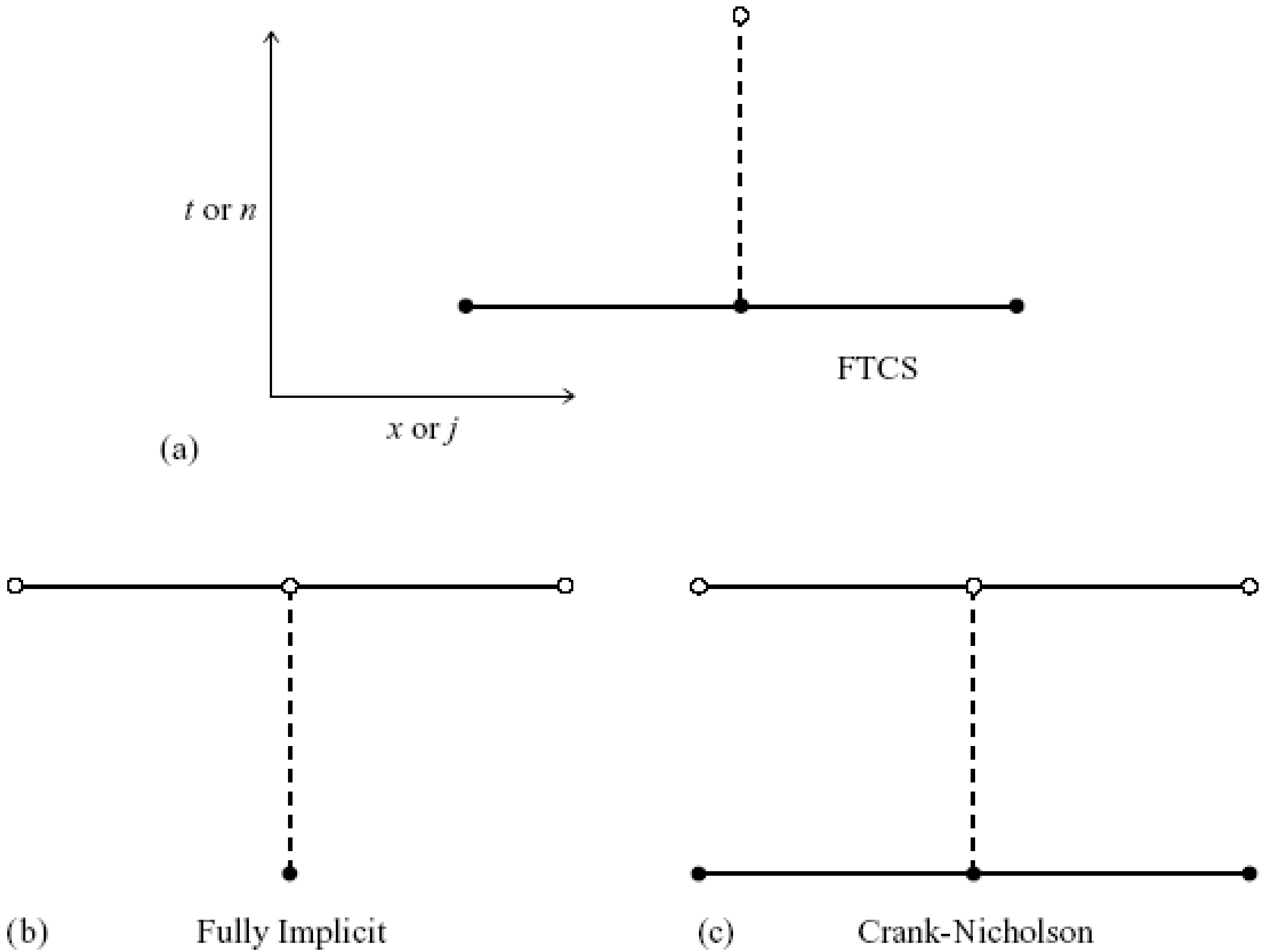
John Crank
1916-2006

Phyllis Nicolson
1917-1968

Now lets average between the FTCS and the fully implicit scheme:

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = \frac{D}{2}\left[\frac{(u_{j+1}^{n+1} - 2u_j^{n+1} + u_{j-1}^{n+1}) + (u_{j+1}^n - 2u_j^n + u_{j-1}^n)}{(\Delta x)^2}\right]$$

The Crank-Nicolson method is unconditional stable and second order accurate. (Because it is a centered scheme in space and time.)


t or n
x or j
(a)
FTCS
(b)
Fully Implicit
(c)
Crank-Nicholson

Diffusive Equations, Generalization

$$\frac{\partial u}{\partial t} = F\left(u, x, t, \frac{\partial u}{\partial x}, \frac{\partial^2 u}{\partial x^2}\right)$$

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = F_i^n\left(u, x, t, \frac{\partial u}{\partial x}, \frac{\partial^2 u}{\partial x^2}\right) \qquad \text{(forward Euler)}$$

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = F_i^{n+1}\left(u, x, t, \frac{\partial u}{\partial x}, \frac{\partial^2 u}{\partial x^2}\right) \qquad \text{(backward Euler)}$$

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \frac{1}{2}\left(F_i^{n+1}\left(u, x, t, \frac{\partial u}{\partial x}, \frac{\partial^2 u}{\partial x^2}\right) + F_i^n\left(u, x, t, \frac{\partial u}{\partial x}, \frac{\partial^2 u}{\partial x^2}\right)\right)$$

(Crank-Nicolson)

Crank-Nicolson scheme

- Scheme is unconditional stable.
- This allows using long time steps.
- Method has second order accuracy.
- Implicit scheme: One has to solve system of equation to advance in time.
- This is straight forward for linear PDEs.
- Method works also for nonlinear PDEs.
- But this requires to solve a system of nonlinear coupled algebraic equations, which can be tricky.

Parabolic (diffusive) PDEs
Summary

- Explicit Euler-scheme is stable, but with severe **restrictions on time step**.
- Doubling the spatial grid resolution requires reduction of time step by a factor 4 for explicit schemes.
- The **implicit Crank-Nicolson** scheme is **unconditional stable**.
- Implicit codes are more difficult to implement.