Hydroinformatics
Module 4:Numerical Methods I

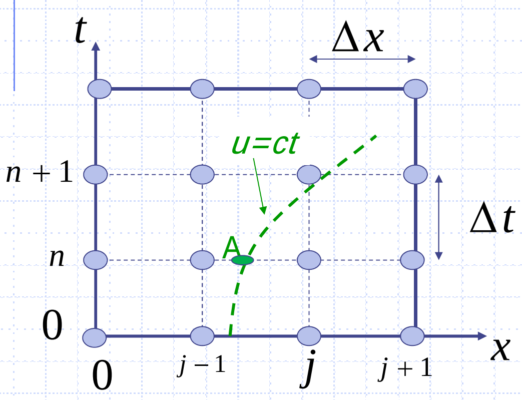
Lecture 5 and 6: PDE, Hyperbolic PDE, Stability, Accuracy, Parabolic PDE, Eliptic PDE

I.Popescu

5.4. Hyperbolic PDE-MoC schemes

$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = 0$$

MoC theory $\Rightarrow$ $u(x,t) = const$ along $\dfrac{dx}{dt} = a(x,t)$

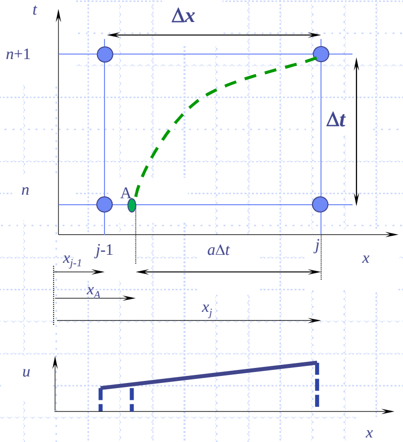


$$u_j^{n+1} = u(A) = u_A^n$$

u_A^n interpolated between u_{j-1}^n *and* u_j^n

5.4. Hyperbolic PDE-MoC schemes

$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = 0 \quad u_A^n \quad \text{liniar interpolaton between } u_{j-1}^n \quad and \quad u_j^n$$



$$Cr = \frac{a\Delta t}{\Delta x}$$

$$x_A = x_j - a\Delta t$$

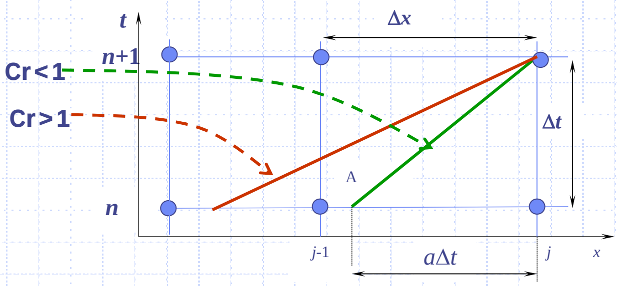
$$U_A^n = \frac{x_A - x_{j-1}}{x_j - x_{j-1}} U_j^n + \frac{x_j - x_A}{x_j - x_{j-1}} U_{j-1}^n$$

$$U_j^{n+1} = U_A^n = (1 - Cr)U_j^n + CrU_{j-1}^n$$

5.4.Hyperbolic PDE-MoC schemes

❑Special remarks

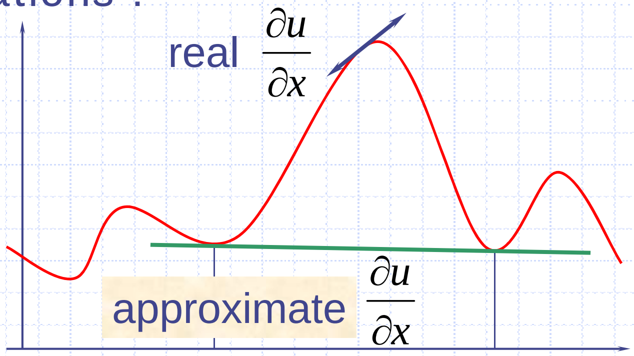
- The MoC numerical method is stable for Cr<1
- For Cr>1 extrapolation takes place



- Important: The formula $U_j^{n+1} = U_A^n = (1 - Cr)U_j^n + CrU_{j-1}^n$ is valid for positive values of a.

Accuracy / consistency

- The discretised equations are *not* the real ones
- The scheme does *not* solve the real equations !



Important Properties of Numerical Schemes

- **Convergence**

  Convergence → *Consistency*, *Stability*

  – numerical scheme solution is *convergent* if it comes closer and closer to the analytical solution of the real ODE/PDE when the time step decreases;
- **LaxTheorem**: 2 conditions needed for convergence
  - *Consistency*
    - A scheme is *consistent* if it gives a correct approximation of the ODE/PDE as the time/space step is decreased
    - verified using Taylor Series expansion
  - *Stability*
    - A scheme is *stable* if any initially finite perturbation remains bounded as time grows
    - Verification: Matrix method, Fourier method, Domain of dependence

Accuracy / consistency

Convergence → Consistency, Stability

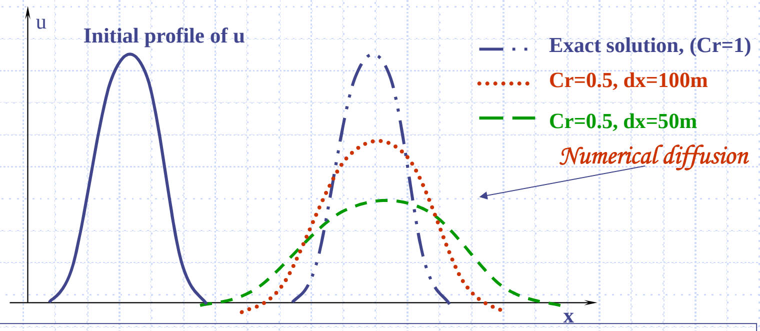
- To reduce the truncation error :
  - Decrease **both** Δt and Δx
  - (i.e. the Courant number must lie in a reasonable range) $Cr = \frac{a\Delta t}{\Delta x}$
- If the truncation error is 'small':
  - the discretised equation is **consistent** with the real one.

Accuracy and suitability

Convergence → Consistency, Stability

- Consistency of schemes for PDEs

$$\frac{\partial U}{\partial t} + a\frac{\partial U}{\partial x} = 0 \qquad U_j^{n+1} = Cr U_{j-1}^n + (1 - Cr)U_j^n$$

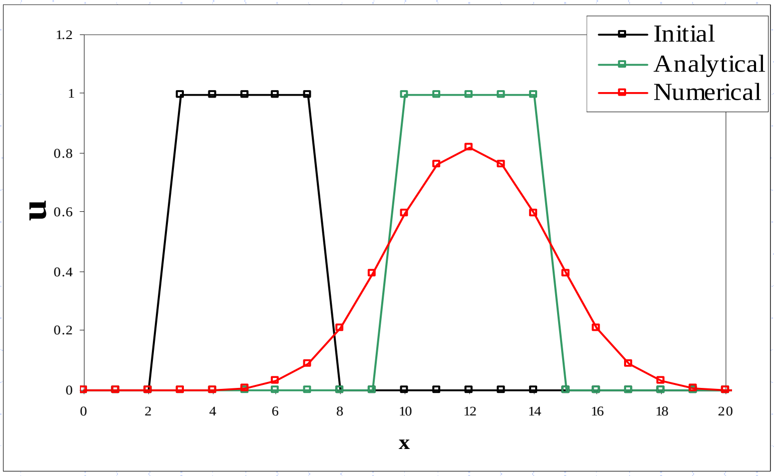


- **Numerical diffusion**
  - causes **amplitude** error

Accuracy / consistency

Convergence — Consistency / Stability

Numerical diffusion : profile smearing



Accuracy and suitability

Convergence — Consistency / Stability

- ❑ Consistency of explicit schemes

$$\frac{\partial U}{\partial t} + a\frac{\partial U}{\partial x} = 0 \qquad U_j^{n+1} = CrU_{j-1}^n + (1 - Cr)U_j^n$$

Taylor series expansions gives

$$U_{j-1}^n = U_j^n - \Delta x\frac{\partial U}{\partial x} + \frac{\Delta x^2}{2}\frac{\partial^2 U}{\partial x^2} + O(\Delta x^3)$$

$$U_j^{n+1} = U_j^n + \Delta t\frac{\partial U}{\partial t} + \frac{\Delta t^2}{2}\frac{\partial^2 U}{\partial t^2} + O(\Delta t^3)$$

Into the equation this gives:

$$\cancel{U_j^n} + \Delta t\frac{\partial U}{\partial t} + \frac{\Delta t^2}{2}\frac{\partial^2 U}{\partial t^2} + O(\Delta t^3) = \left[U_j^n - \Delta x\frac{\partial U}{\partial x} + \frac{\Delta x^2}{2}\frac{\partial^2 U}{\partial x^2} + O(\Delta x^3)\right]a\frac{\Delta t}{\Delta x} + \left(\cancel{1} - a\frac{\Delta t}{\Delta x}\right)U_j^n$$

Accuracy and suitability

Convergence → Consistency, Stability

❑ Consistency of explicit schemes

After dividing with dt we obtain:

$$\frac{\partial U}{\partial t}+\frac{\Delta t}{2}\frac{\partial^2 U}{\partial t^2}+O(\Delta t^2)=\left[-\frac{\partial U}{\partial x}+\frac{\Delta x}{2}\frac{\partial^2 U}{\partial x^2}+O(\Delta x^2)\right]a$$

or:

$$\frac{\partial U}{\partial t}+a\frac{\partial U}{\partial x}=\left[-\frac{\Delta t}{2}\frac{\partial^2 U}{\partial t^2}+\frac{\Delta x}{2}\frac{\partial^2 U}{\partial x^2}\right]a+O(\Delta x^2)-O(\Delta t^2)$$

Since $\frac{\partial^2 U}{\partial t^2}=a^2\frac{\partial^2 U}{\partial x^2}$

$$\Rightarrow \frac{\partial U}{\partial t}+a\frac{\partial U}{\partial x}=\underbrace{\underbrace{(1-Cr)\frac{a\Delta x}{2}\frac{\partial^2 U}{\partial x^2}}_{TE_1}+\underbrace{O(\Delta x^2)}_{TE_2}-\underbrace{O(\Delta t^2)}_{TE_3}}_{TE}$$

Accuracy and suitability

Convergence → Consistency, Stability

❑ Consistency of explicit schemes

$$\frac{\partial U}{\partial t}+a\frac{\partial U}{\partial x}=\underbrace{\underbrace{(1-Cr)\frac{a\Delta x}{2}\frac{\partial^2 U}{\partial x^2}}_{TE_1}+\underbrace{O(\Delta x^2)}_{TE_2}-\underbrace{O(\Delta t^2)}_{TE_3}}_{TE}$$

- TE_1 cancels for Cr=1 ;
- first order accurate in x;
- consistent up to the second order;

Accuracy / consistency

Convergence → Consistency, Stability

What you want to solve :

$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = 0$$

- What the scheme 'sees' :

$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = \underbrace{k_1 \Delta x \frac{\partial^2 u}{\partial x^2} + k_2 \Delta x^2 \frac{\partial^3 u}{\partial x^3} + \dots}_{\text{Truncation error}}$$

$k_1 \Delta x \frac{\partial^2 u}{\partial x^2}$ → Numerical diffusion

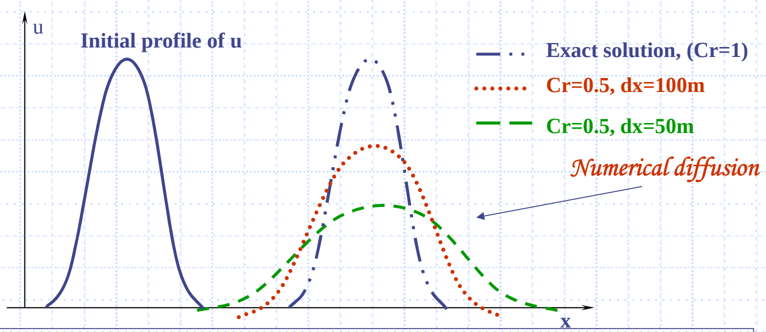
$k_2 \Delta x^2 \frac{\partial^3 u}{\partial x^3}$ → Numerical dispersion

Accuracy and suitability

Convergence → Consistency, Stability

- Stability of schemes for PDEs

$$\frac{\partial U}{\partial t} + a\frac{\partial U}{\partial x} = 0 \qquad U_j^{n+1} = CrU_{j-1}^n + (1 - Cr)U_j^n$$



- **Numerical diffusion**
  - causes **amplitude** error

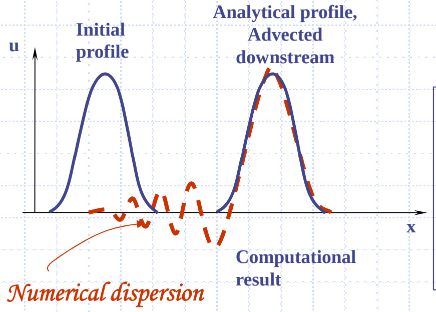
Accuracy and suitability

Convergence — Consistency / Stability

- Stability of schemes for PDEs
  - Because of numerical diffusion we try to use a better scheme,
    - like for instance Preismann scheme with psi=0.5;
    - a dispersion equation

$$\frac{\partial U}{\partial t} + a\frac{\partial U}{\partial x} = 0 \Longleftrightarrow \frac{\partial U}{\partial t} + a\frac{\partial U}{\partial x} = k\frac{\partial^3 U}{\partial x^3}$$

Dispersion equation

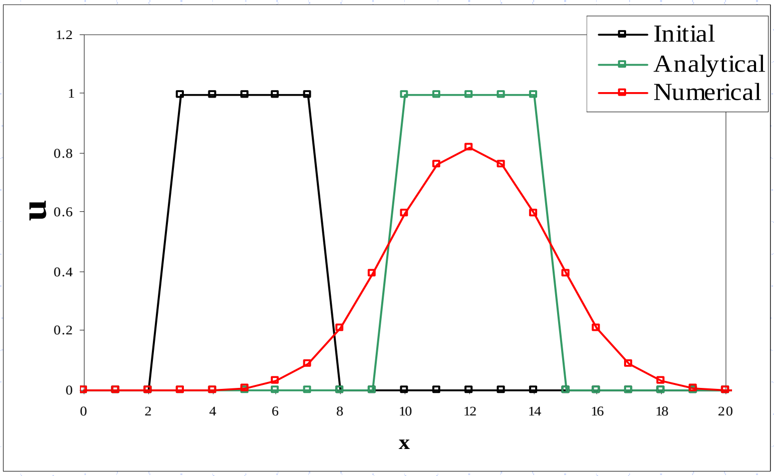


- Numerical dispersion
  - Due to derivatives estimation
  - Sharper profiles and oscillations
  - Create phase errors

Accuracy / consistency

Convergence — Consistency / Stability

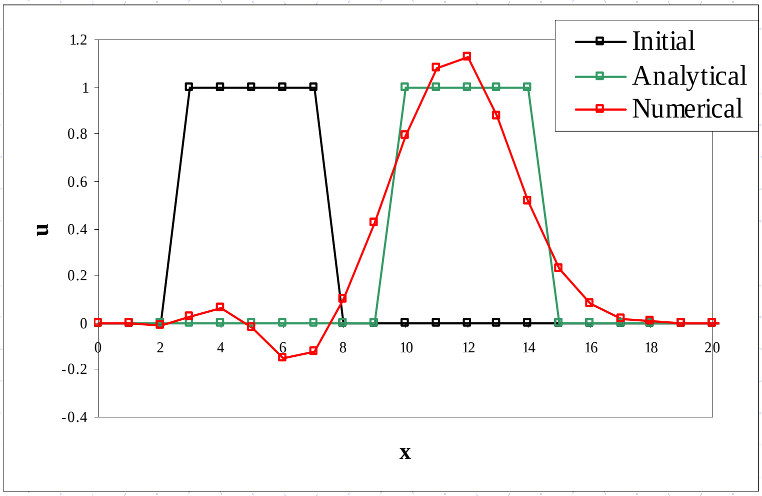
Numerical diffusion : profile smearing



Accuracy / consistency

Convergence — *Consistency* — ***Stability***

Numerical dispersion : oscillations



Accuracy and suitability

Convergence — *Consistency* — ***Stability***

- ❑ Stability of explicit schemes
  - Von Neumann stability method (Using Fourier analysis)
  - Same principle: amplitude factor is less than 1

Stability

- ❑ Stability of explicit schemes
  - Von Neumann stability method (Using Fourier analysis)
  - Same principle: amplitude factor is less than 1

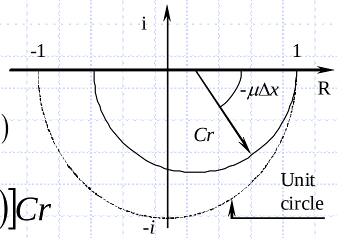
$$U_j^n = U_0^0 e^{(n\lambda\Delta t + ij\mu\Delta x)}$$

$$U_j^n = U_0^0 \exp(n\lambda_r \Delta t)[\cos(n\lambda_i \Delta t) + i\sin(n\lambda_i \Delta t)][\cos(j\mu\Delta x) + i\sin(j\mu\Delta x)]$$

$$A_N = \frac{U_j^{n+1}}{U_j^n} = (1 - Cr) + Cr\frac{U_{j-1}^n}{U_j^n}$$

$$\frac{U_{j-1}^n}{U_j^n} = \frac{U_0^0 \exp(n\lambda\Delta t + i(j-1)\mu\Delta x)}{U_0^0 \exp(n\lambda\Delta t + ij\mu\Delta x)} = \exp(-i\mu\Delta x)$$

$$A_N = 1 - Cr + [\cos(\mu\Delta x) - i\sin(\mu\Delta x)]Cr$$



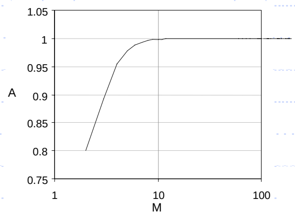
Amplitude and phase portraits

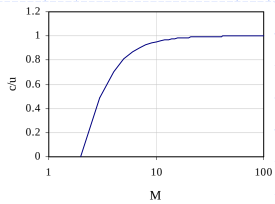
- ❑ Wave amplitude
  - Amplification factor =1
  - Propagation of Fourier waves – phase speed
    - – Any difference between the numerical phase speed and true phase speed is the phase error
- ❑ The graph that shows how the Fourier components are amplified is called an *amplitude portrait*.
- ❑ The graph that shows at what speed the Fourier components travel is called a *phase portrait*.

Amplitude and phase portraits

$$\mu\Delta x = \frac{2\pi}{M}$$

M- The wave number - represents the number of grid intervals needed to cover one period of the wave

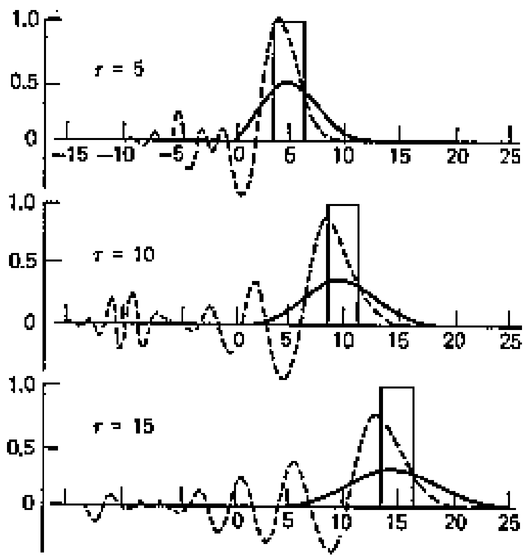




$$A_N = 1 - Cr + \left[\cos\left(\frac{2\pi}{M}\right) - i\sin\left(\frac{2\pi}{M}\right)\right]Cr$$

$$c_N = -\frac{Arg(A_N)}{\mu} = -\frac{a}{Cr}Arg(A_N)\frac{M}{2\pi}$$

Phase and amplitude errors

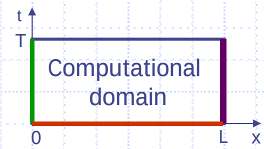


(B)Parabolic PDEs

Parabolic Equations – Initial Value Problems

A 1D time-dependent parabolic eqn (b=c=0)

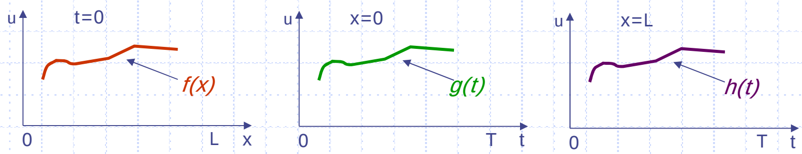
$$\frac{\partial u}{\partial t} = a\frac{\partial^2 u}{\partial x^2} \qquad \begin{array}{l} 0 \le x \le L \\ 0 \le t \le T \end{array}$$



With I.C $\quad u(x,0) = f(x)$

With B.C $\quad u(0,t) = g(t)$

$\quad u(L,t) = h(t)$



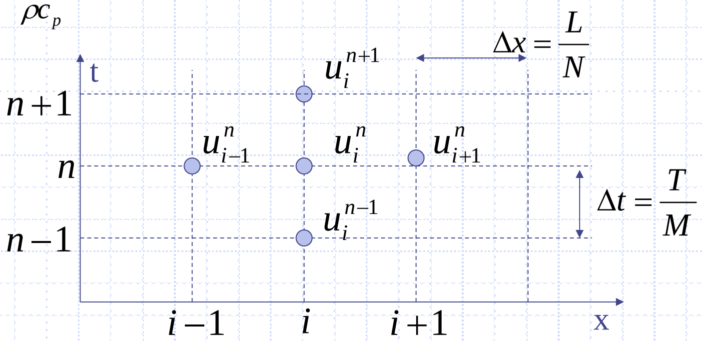
Example : Heat Conduction

Equation governing transfer of heat is:

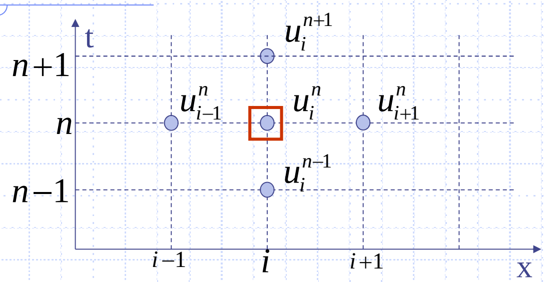
$$\rho c_p \frac{\partial T}{\partial t} = k \frac{\partial^2 T}{\partial x^2}$$

- T(x,t) – temperature
- ρ – density
- c_p – specific heat capacity
- k – thermal conductivity

$a = \frac{k}{\rho c_p}$ is also known as the thermal diffusivity



Parabolic PDE: Solution Methods – Explicit Methods



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

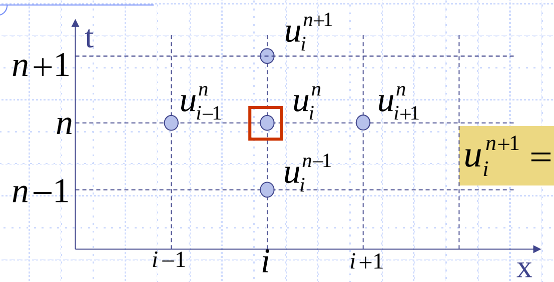
CS: $\frac{\partial^2 u}{\partial x^2} = \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{\Delta x^2}$

FT: $\frac{\partial u}{\partial t} = \frac{u_i^{n+1} - u_i^n}{\Delta t}$

$$\Rightarrow \frac{u_i^{n+1} - u_i^n}{\Delta t} = a \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{\Delta x^2}$$

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

Parabolic PDE:Solution Methods – Explicit Methods²



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

$$u_i^{n+1} = u_i^n + r\left(u_{i+1}^n - 2u_i^n + u_{i-1}^n\right)$$

I.C.: $u_i^0 = f(i\Delta x)$

B.C.: $u_0^n = g(n\Delta t)$

$u_N^n = h(n\Delta t)$

We can calculate the unknown values of u_i^{n+1} from the known values of u_i^n starting from the initial condition u_i^0

Parabolic PDE: Stability of the Explicit Method

- The explicit method is **unstable** if the time step is too large.
  - **Stability condition** for **fixed** boundary conditions is

$$0 < r \leq \tfrac{1}{2} \quad \Rightarrow \quad \Delta t \leq \frac{\Delta x^2}{2a}$$

  - **Stability condition** for **derivative** boundary conditions is

$$0 < r \leq \frac{1}{2 + \alpha/_k\,\Delta x} \quad \text{for} \quad -k\frac{\partial u}{\partial n} = \alpha(u - u_f)$$

Parabolic Equation –An example calculation

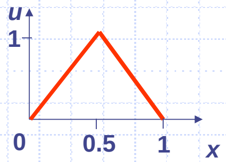
Solve

$$\frac{\partial u}{\partial t} = a\frac{\partial^2 u}{\partial x^2} \quad \text{for} \quad 0 \le x \le 1, \quad 0 \le t \le \tau$$

(Where u represents temperature)

with initial condition

$$u(x,0) = \begin{cases} 2x & \text{for} \quad 0 \le x \le 0.5 \\ 2(1-x) & \text{for} \quad 0.5 \le x \le 1.0 \end{cases}$$



and boundary conditions

$$u(0,t) = 0$$
$$u(1,t) = 0$$

Parabolic Equation –An example calculation2

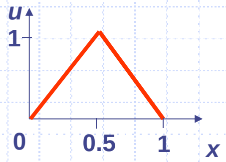
Solve

$$\frac{\partial u}{\partial t} = a\frac{\partial^2 u}{\partial x^2} \quad \text{for} \quad 0 \le x \le 1, \quad 0 \le t \le \tau$$

(Where u represents temperature)

with initial condition

$$u(x,0) = \begin{cases} 2x & \text{for} \quad 0 \le x \le 0.5 \\ 2(1-x) & \text{for} \quad 0.5 \le x \le 1.0 \end{cases}$$



and boundary conditions

$$u(0,t) = 0$$
$$u(1,t) = 0$$

Analytical (exact) solution

$$u(x,t) = \frac{8}{\pi^2}\sum_{n=1}^{\infty}\left[n^{-2}\exp\left(-n^2\pi^2 t\right) \times \sin\left(\tfrac{1}{2}n\pi\right)\sin\left(n\pi x\right)\right]$$

Parabolic Example : Case 1 ($r<0.5$)

Values for a, space and time discretisation are:

$$a=1 \qquad \Delta x = 0.1 \qquad \Delta t = 0.001 \qquad \Rightarrow \qquad r = \frac{a\Delta t}{\Delta x^2} = 0.1$$

$$u_i^{n+1} = u_i^n + r(u_{i+1}^n - 2u_i^n + u_{i-1}^n)$$

n=0
t=0

$$u_0^0 = 0, \quad u_1^0 = 0.2, \quad u_2^0 = 0.4, \ \ldots, \quad u_4^0 = 0.8, \quad u_5^0 = 1.0, \quad u_6^0 = 0.8, \ \ldots.$$

n=1
t=0.001

$$u_1^1 = 0.2 + 0.1(0.4 - 0.4 + 0.0) = 0.2$$
...
$$u_5^1 = 1.0 + 0.1(0.8 - 2.0 + 0.8) = 0.96$$

n=2
t=0.002

$$u_1^2 = 0.2 + 0.1(0.4 - 0.4 + 0.0) = 0.2$$
...
$$u_5^2 = 0.96 + 0.1(0.8 - 1.92 + 0.8) = 0.928$$

Parabolic Example : Case 1 ($r<0.5$)

- First 10 time steps

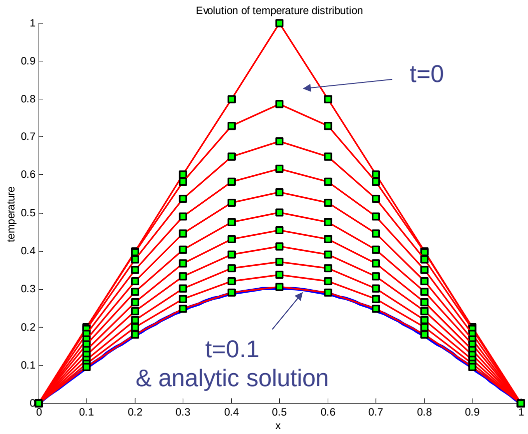
i =	0	1	2	3	4	5
x =	0	0.1000	0.2000	0.3000	0.4000	0.5000
n=0	0	0.2000	0.4000	0.6000	0.8000	1.0000
n=1	0	0.2000	0.4000	0.6000	0.8000	0.9600
n=2	0	0.2000	0.4000	0.6000	0.7960	0.9280
n=3	0	0.2000	0.4000	0.5996	0.7896	0.9016
n=4	0	0.2000	0.4000	0.5986	0.7818	0.8792
n=5	0	0.2000	0.3998	0.5971	0.7732	0.8597
n=6	0	0.2000	0.3996	0.5950	0.7643	0.8424
n=7	0	0.1999	0.3992	0.5924	0.7551	0.8268
n=8	0	0.1999	0.3986	0.5893	0.7460	0.8125
n=9	0	0.1998	0.3978	0.5859	0.7370	0.7992
n=10	0	0.1996	0.3968	0.5822	0.7281	0.7867

symmetrical about x=0.5

Parabolic Example : Case 1 ($r<0.5$)

- Plot of the temperature distribution every 10 discretisation points in space for 100 time steps

STABLE solution



Parabolic Example : Case 2 ($r=0.5$)

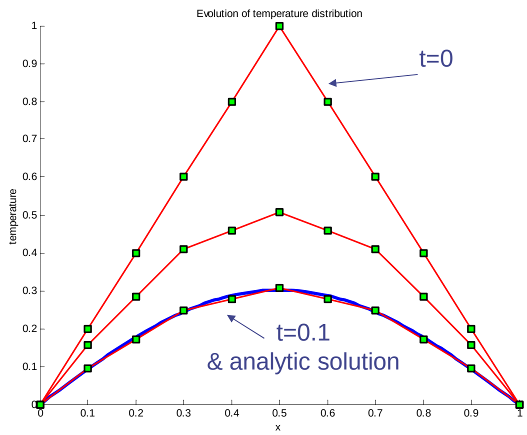
- Plot of the temperature distribution every 10 space step for 20 time steps

$$\Delta x = 0.1$$

$$\Delta t = 0.005$$

$$\Rightarrow r = \frac{a\Delta t}{\Delta x^2} = 0.5$$

STABLE solution but not very accurate



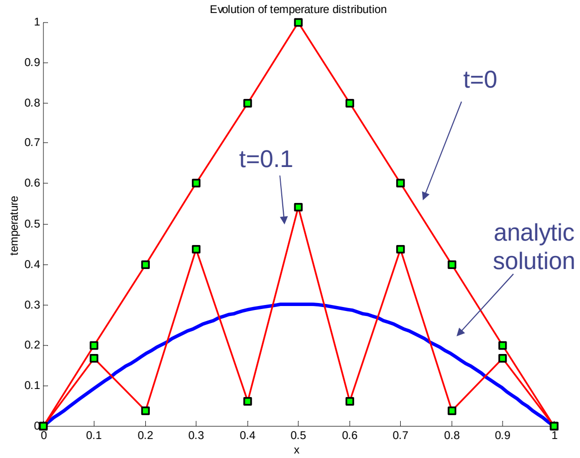
Parabolic Example : Case 3 (r>0.5)

- Plot of the temperature distribution at 0 and 18 time steps

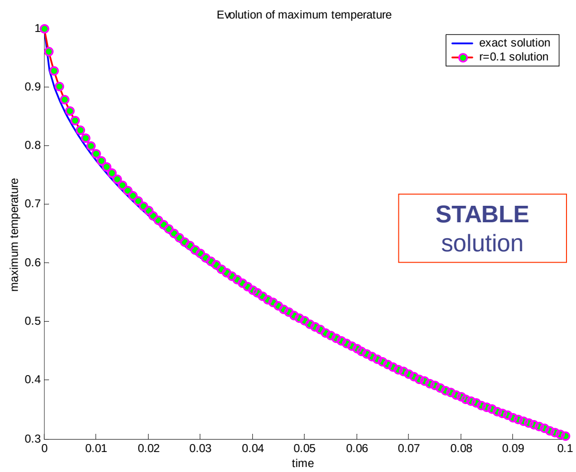
$$\Delta x = 0.1$$
$$\Delta t = 0.0055$$
$$\Rightarrow r = \frac{a\Delta t}{\Delta x^2} = 0.55$$

UNSTABLE

solution is meaningless

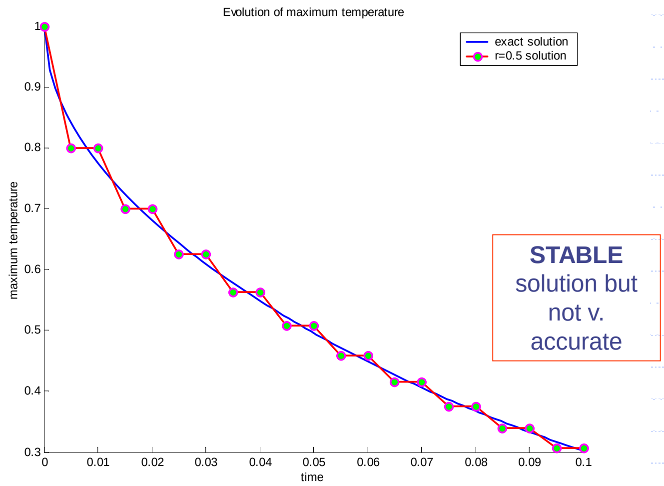


Parabolic Example : Evolution of maximum temperature (r=0.1)

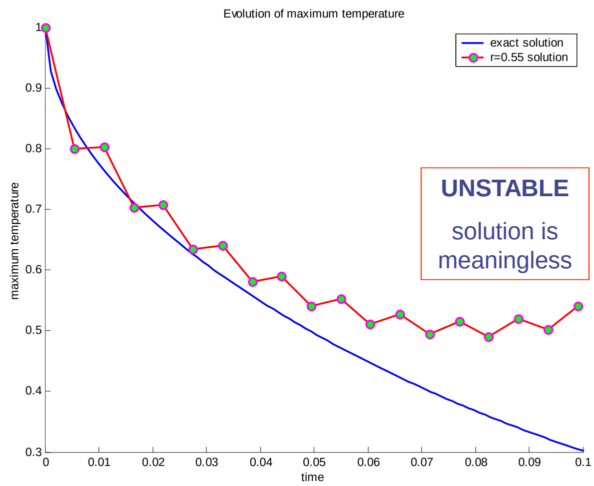


STABLE solution

Parabolic Example :
Evolution of maximum temperature (r=0.5)



Parabolic Example :
Evolution of maximum temperature (r=0.55)



Parabolic Example : Explicit Method Error

Comparison of temperature values at x=0.5 and t=0.1 (for N=11)

r	M	u	% error
analytic	-	0.3021	-
0.001	10000	0.3071	1.65
0.01	1000	0.3070	1.60
0.1	100	0.3056	1.16
0.5	20	0.3071	1.64
0.55	18	0.6336	109
0.6	16	7.2340	2294

unstable

need to increase N to reduce error further

(C)Eliptic PDEs

Elliptic Equations –No time variable

$$\frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial x^2} + \lambda u(x,y) = f(u,x,y) \qquad \begin{array}{l} 0 \le x \le L_x \\ 0 \le y \le L_y \end{array}$$

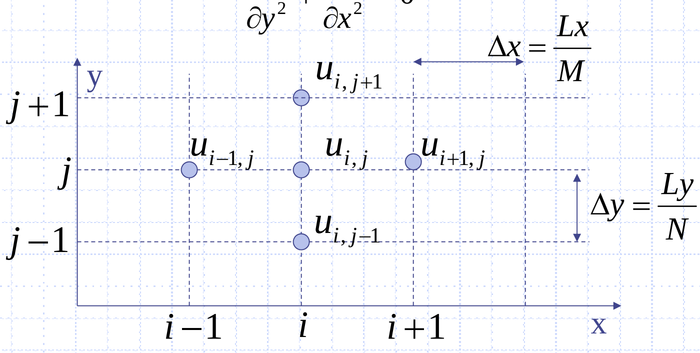
- With the particular cases:
  - Poisson equation $\lambda = 0 \rightarrow \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial x^2} = f(u,x,y)$
  - Laplace equation $\lambda = 0$ and $f = 0 \rightarrow \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial x^2} = 0$
- With different typs of B.C.:
  - Dirichlet : u is specified at the boundary
  - Neumann : derivative of u is specified at the boundary
  - Mixed(robin): both u and its derivative is specified at the boundary

t

Example : Laplace equation

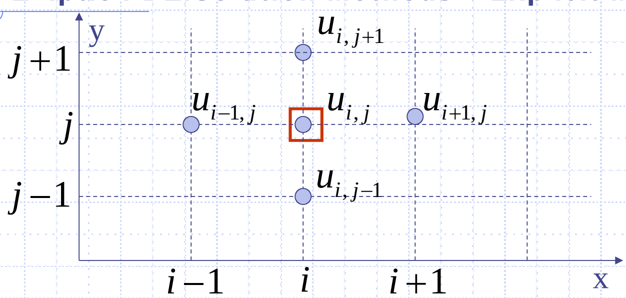
Equation is:

$$\frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial x^2} = 0$$



Approximate solution determined at all grid points simultaneously by solving single system of algebraic equations

Elliptic PDE:Solution Methods – Explicit Methods



$$\frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial x^2} = 0$$

$$\frac{\partial^2 u}{\partial x^2} = \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{\Delta x^2}$$

$$\frac{\partial^2 u}{\partial y^2} = \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{\Delta y^2}$$

$$\Rightarrow \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{\Delta x^2} + \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{\Delta y^2} = 0$$

$$u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} - 4u_{i,j} = 0$$

Holds for all interior points of the domain

Finite difference methods

What you should remember

What you should remember

- Numerical solutions to PDEs can be obtained by discretising both space and time.
- Explicit numerical schemes for PDEs are subject to stability constraints.
- Implicit numerical schemes for PDEs are always stable.
- Iterations are also needed for the implicit solution of non-linear PDEs.
- The notions of consistency, stability and convergence also hold for numerical schemes for PDEs.

What you should remember

- First-order accurate schemes produce numerical diffusion; numerical profiles obtained are smoothed and may lead to peak underestimation. Numerical diffusion leads to amplitude error.
- Second-order accurate schemes produce numerical dispersion; numerical profiles exhibit artificial oscillations. Undesirable behaviours (such as negative concentrations) may appear. Numerical dispersion causes phase error.

What you should remember

- Decreasing Δt or Δx alone is not sufficient to reduce the truncation error : Δt and Δx should be reduced together.
- The MOC is a specific kind of numerical method used for advection modelling. Its main drawback is that it is generally not conservative (some water, pollutant, or energy, may be lost artificially).
- At least three points in space are needed to solve a diffusion equation.
- The design of 2-D and 3-D computational grid should be carried out with care, long and narrow grids should be avoided.

6 –Finite volume methods

(FVM)

FVM – principle (1)

FVMs are applicable to conservative equations, i.e. equations of the form

$$\frac{\partial U}{\partial t}+\frac{\partial F}{\partial x}=0 \qquad \text{(1D problems)}$$

$$\frac{\partial U}{\partial t}+\frac{\partial F}{\partial x}+\frac{\partial G}{\partial y}=0 \qquad \text{(2D problems)} \qquad (5.1)$$

$$\frac{\partial U}{\partial t}+\frac{\partial F}{\partial x}+\frac{\partial G}{\partial y}+\frac{\partial H}{\partial z}=0 \qquad \text{(3D problems)}$$

F, *G*, *H* : Fluxes in *x*, *y* and *z*
U : Conserved variable

$$F=F\left(U,\frac{\partial U}{\partial x},\frac{\partial U}{\partial y},\frac{\partial U}{\partial z},\ldots,\frac{\partial^n U}{\partial x^n},\frac{\partial^n U}{\partial y^n},\frac{\partial^n U}{\partial z^n}\right)$$

$$G=G\left(U,\frac{\partial U}{\partial x},\frac{\partial U}{\partial y},\frac{\partial U}{\partial z},\ldots,\frac{\partial^n U}{\partial x^n},\frac{\partial^n U}{\partial y^n},\frac{\partial^n U}{\partial z^n}\right)$$

$$H=H\left(U,\frac{\partial U}{\partial x},\frac{\partial U}{\partial y},\frac{\partial U}{\partial z},\ldots,\frac{\partial^n U}{\partial x^n},\frac{\partial^n U}{\partial y^n},\frac{\partial^n U}{\partial z^n}\right)$$

FVM – principle (2)

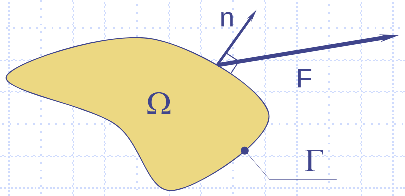
Eqs. (5.1) express the conservation of *U* over any bounded volume of space

$$\frac{\partial U}{\partial t}+\nabla\left(F\vec{x}+G\vec{y}+H\vec{z}\right)=0 \qquad (5.2)$$

∇ : Divergence operator

By definition of the divergence, Eq. (5.2) can be rewritten as

$$\frac{\partial}{\partial t}\int_{\Omega} U\,\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z+\oint_{\Gamma}\left(F\vec{x}+G\vec{y}+H\vec{z}\right)\vec{n}\,\mathrm{d}\gamma=0$$

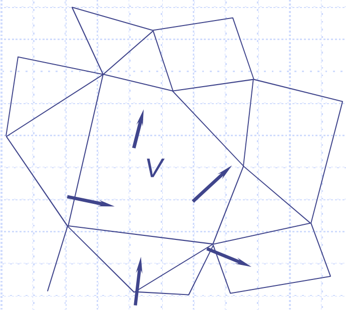


FVM – principle (3)

Application:

1) Discretise space into volumes
2) Compute the fluxes at the edges between the volumes
3) Determine the changes in U via a balance equation

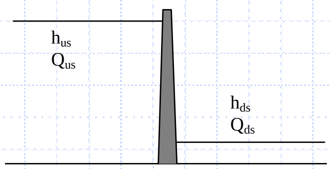
$$U^{n+1} = U^{n} + \left(\text{Flux}_{\text{in}} - \text{Flux}_{\text{out}}\right)\frac{\Delta t}{V} \qquad (5.3)$$

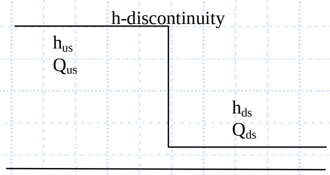


FVM – application (1)

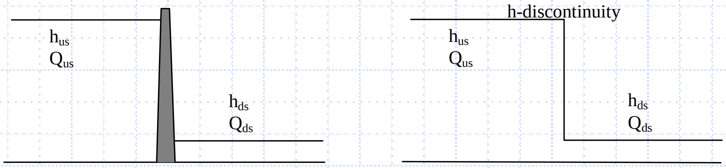
The conservation law together with piecewise constant data having a single discontinuity is known as ***the Riemann problem***.

$$q_x = \begin{cases} q_l & \text{if } x \le 0 \\ q_r & \text{if } x \ge 0 \end{cases}$$





FVM – application (2)



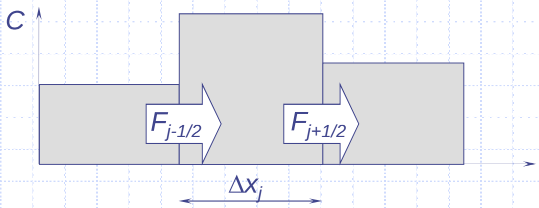
Three possible patterns:

- zones of constant state (depth and velocity are homogeneous over such zones),
- a shock wave (information coming from upstream catches and information downstream),
- a rarefaction wave (information downstream travels faster than the information upstream).

FVM – applications (3)

Transport equations

$$\frac{\partial C}{\partial t} + \frac{\partial}{\partial x}\underbrace{\left(uC - D\frac{\partial C}{\partial x}\right)}_{F} = 0 \quad \Longrightarrow \quad C_j^{n+1} = C_j^n + \frac{\Delta t}{\Delta x_j}\left(F_{j-1/2} - F_{j+1/2}\right)$$



$$F_{j-1/2} = \begin{cases} uC_{j-1}^n - 2D\dfrac{C_j^n - C_{j-1}^n}{\Delta x_{j-1} + \Delta x_j} & \text{if } u \geq 0 \\ uC_j^n - 2D\dfrac{C_j^n - C_{j-1}^n}{\Delta x_{j-1} + \Delta x_j} & \text{if } u \leq 0 \end{cases}$$

FVM – applications (4)

Applications in 1D

Discontinuous flows

$$\frac{\partial U}{\partial t}+\frac{\partial U}{\partial x}=0 \quad \text{(scalar PDE)}$$

$$\frac{\partial \mathbf{U}}{\partial t}+\frac{\partial \mathbf{F}}{\partial x}=0 \quad \text{(system of PDEs)}$$

When F [**F**] is a non-linear function of U [**U**], the solution may become discontinuous

Ex. Burgers equation

$$\frac{\partial u}{\partial t}+\frac{\partial}{\partial x}\left(\frac{u^2}{2}\right)=0 \quad \text{(conservation form)}$$

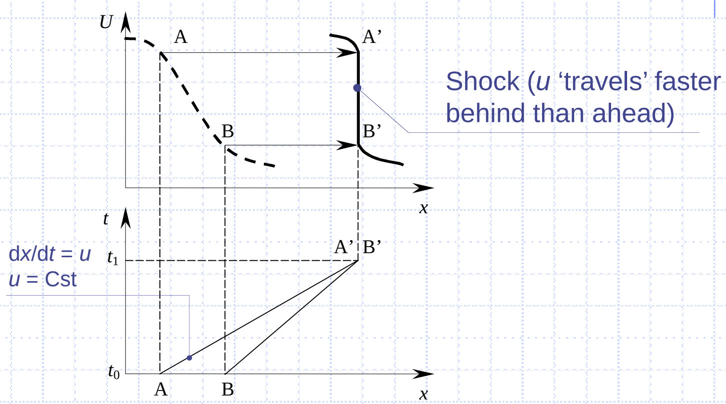
$$\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}=0 \quad \text{(characteristic form)}$$

$$\frac{\mathrm{D}u}{\mathrm{D}t}=0 \ \text{along} \ \frac{\mathrm{d}x}{\mathrm{d}t}=u \quad (u \text{ is a Riemann invariant})$$

FVM – applications (5)

Applications in 1D

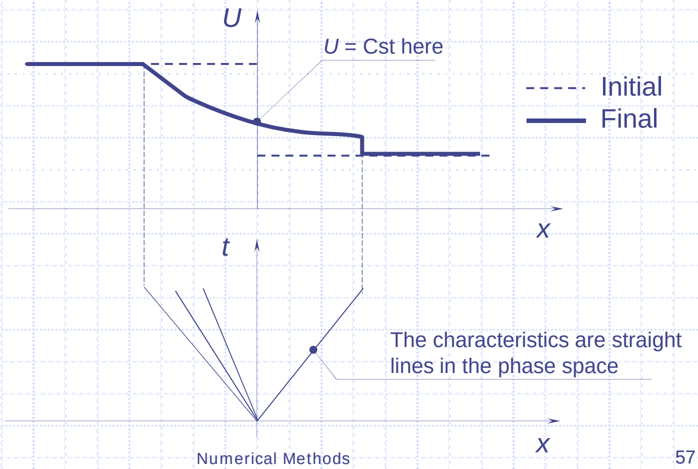
Burgers Eq. (continued): formation of shocks from initially smooth profiles



FVM – applications (6)

Applications in 1D (4)

In finite volume methods: the flux is calculated from the solution of a Riemann problem



FVM – applications (7)

Applications in 1D

The solution of the Riemann problem exists even though the initial profile is discontinuous

Algorithm

1) At each interface *j*-1/2, define the Riemann problem $\left(U_{j-1}^{n}, U_{j}^{n}\right)$

2) Solve it => solution $U_{j-1/2}$

3) Compute the flux $F_{j-1/2} = F\left(U_{j-1/2}\right)$

4) For each cell, carry out balance for *U*

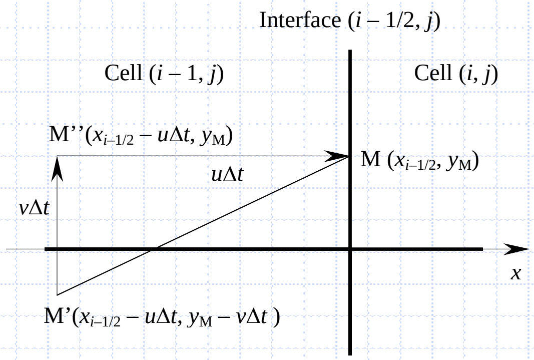
$$U_{j}^{n+1} = U_{j}^{n} + \frac{\Delta t}{\Delta x_{j}}\left(F_{j-1/2} - F_{j+1/2}\right)$$

FVM – applications (8)

Multidimensional problems

Wave splitting (scalar equations)

$$\frac{\partial U}{\partial t} + u\frac{\partial U}{\partial x} + v\frac{\partial U}{\partial y} = 0$$



FVM – applications (9)

Multidimensional problems (2)

Wave splitting (systems of equations)

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} + \frac{\partial \mathbf{G}}{\partial y} = 0$$

Decomposed into

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} = 0 \quad \Leftrightarrow \quad \frac{\partial \mathbf{U}}{\partial t} + \mathbf{A}\frac{\partial \mathbf{U}}{\partial x} = 0$$

followed by

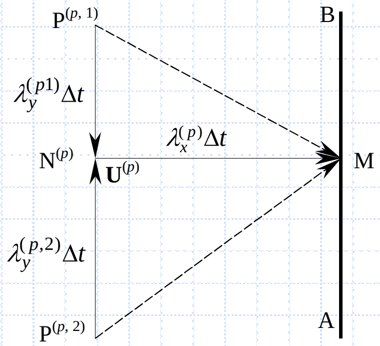
$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{G}}{\partial y} = 0 \quad \Leftrightarrow \quad \frac{\partial \mathbf{U}}{\partial t} + \mathbf{B}\frac{\partial \mathbf{U}}{\partial y} = 0$$

=> Decompose the Riemann problem into 2 R.Ps: 1 along *x* and 1 along *y*

FVM – applications (10)

Multidimensional problems (3)

Wave splitting (2)



What you should remember

- Finite Volume Methods (FVMs) are well-suited for the solution of conservative PDEs. The *weak* solution of the PDE is sought.
- FVMs ensure mass conservation automaticly and can handle shocks and discontinuities.
- The solution of the advection PDE by the Godunov-type FVMs involves the definition and the solution of a Riemann problem.

7. Finite Element Method

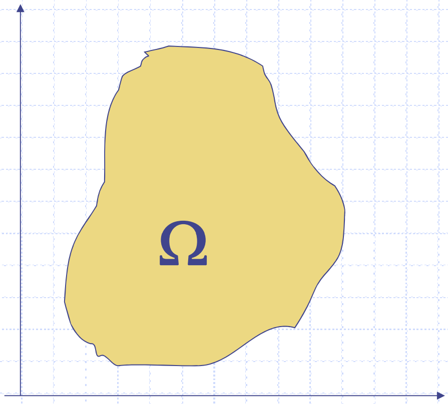
(FEM) - useful for problems with complicated geometries and discontinuities, where analytical solutions can not be obtained

What is it FEM?

- The finite element method is a numerical method for solving problems of engineering and mathematical physics, useful for problems with complicated geometries and discontinuities, where analytical solutions can not be obtained.

Principle of the method -(1)

- Methodology
  - Approximating the field of a dependent variable by a finite series expansion in terms of linearly independent analytical functions.
  - The form of the expansion functions in the finite-element method is in a such way that they are only locally non zero



$$f(U) = 0$$

$$\int_{\Omega} f(U) w d\omega = 0$$

Principle of the method -(2)

Methodology: Two basic steps in the finite-element:

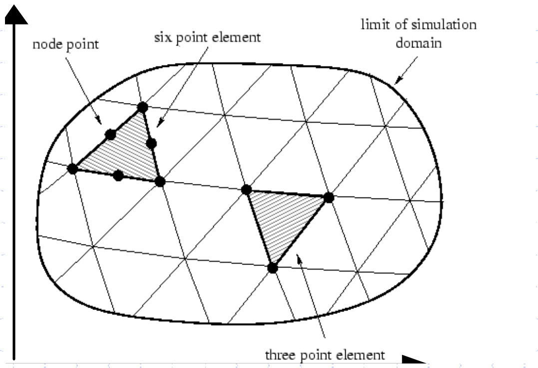
- expand the dependent variables in terms of a set of low-order polynomials (the basis functions) which are only locally non-zero;
- insert these expansions into the governing equations and orthogonalize the error with respect to some test functions.

$$f(U) = 0 \qquad U(x, y) = \sum_{j=1}^{N_e} N_j U_j$$

$$\int_{\Omega} f(U) w d\omega = 0$$

Principle of the method -(3)

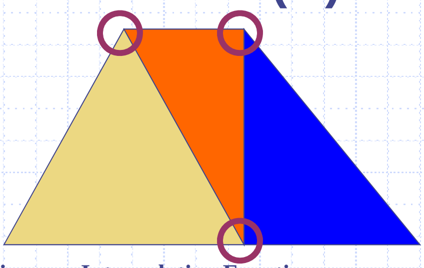
- Discretization
  - Model the domain by dividing it into an equivalent system of smaller domains or units (*finite elements*) interconnected at points common to two or more elements (nodes or nodal points) and/or boundary lines and/or surfaces.



Principle of the method -(4)

Discretization

$$U(x,y) = \sum_{j=1}^{N_e} N_j U_j$$



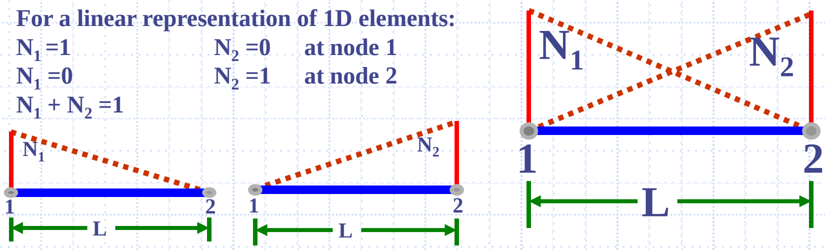
N_1 and N_2 are called Shape Functions or Interpolation Functions. They express the shape of the assumed U.

For a linear representation of 1D elements:

$N_1 = 1$ $N_2 = 0$ at node 1

$N_1 = 0$ $N_2 = 1$ at node 2

$N_1 + N_2 = 1$



Principle of the method -(5)

- Methodology
  - Obtain a set of algebraic equations to solve for unknown nodal quantity (displacement).
  - Secondary quantities (stresses and strains) are expressed in terms of nodal values of primary quantity
- History
  - Hrennikoff [1941] - Lattice of 1D bars
  - McHenry [1943] - Model 3D solids
  - **Courant [1943] - Variational form**
  - Levy [1947, 1953] - Flexibility & Stiffness
  - Argryis and Kelsey [1954] - Energy Prin. for Matrix Methods
  - Turner, Clough, Martin and Topp [1956] - 2D elements
  - Clough [1960] - Term “Finite Elements”

Applications

- Fluid Flow
- Heat Transfer
- Structural/Stress Analysis
- Electro-Magnetic Fields
- Soil Mechanics
- Acoustics

Advantages

- Irregular Boundaries
- Boundary Conditions
- Variable Element Size
- Easy Modification
- Dynamics
- Nonlinear Problems (Geometric or Material)
- General Loads
- Different Materials

Algorithm

- Discretize and Select Element Type
- Select a Function Representation for U
- Define Relationships between U and other physical elements (where applicable)
- Derive Element Stiffness Matrix & Eqs.
- Assemble Equations and Introduce B.C.'s
- Solve for the Unknown the system (obtain U)
- Solve for the other elemnts (depending on U-where applicable)
- Interpret the Results

Application -1D advection equation

$$\frac{\partial U}{\partial t} + a\frac{\partial U}{\partial x} = 0 \quad \Rightarrow \quad \int_0^L \left(\frac{\partial y}{\partial t} + c\frac{\partial y}{\partial x}\right) w.dx = 0 \quad \forall w(x,y)$$

$$\int_0^L \left(\sum_i \frac{U_i^{n+1} - U_i^n}{\Delta t} + c\sum_i U_i^{n+1}\frac{\partial N_i}{\partial x}\right) N_j^*.dx = 0$$

$$\int_0^L \sum_i \frac{U_i^{n+1} - U_i^n}{\Delta t} N_i N_j^*.dx + c\int_0^L \sum_i U_i^{n+1}\frac{\partial N_i}{\partial x} N_j^*.dx = 0$$

$$\int_0^L \sum_i U_i^{n+1}\frac{\partial N_i}{\partial x} N_j dx = \frac{1}{2}\left(U_{j+1}^{n+1} - U_{j-1}^{n+1}\right)$$

Application - 1D advection equation

$$\frac{\partial U}{\partial t} + a\frac{\partial U}{\partial x} = 0$$

$$\int_0^L \sum_i U_i^{n+1}\frac{\partial N_i}{\partial x} N_j dx = \frac{1}{2}\left(U_{j+1}^{n+1} - U_{j-1}^{n+1}\right)$$

$$\Downarrow$$

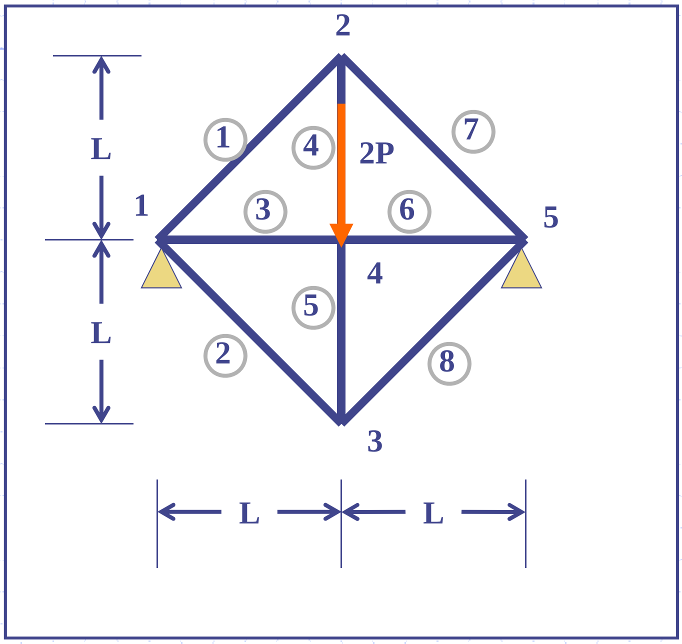
$$a_j y_{j-1}^{n+1} + b_j y_j^{n+1} + c_j y_{j+1}^{n+1} = d_j$$

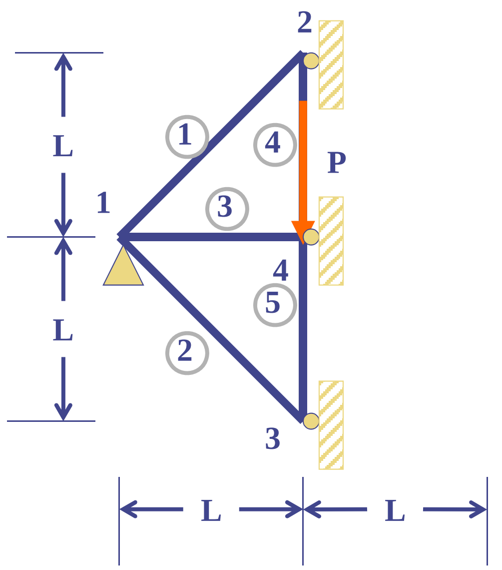
$$a_j = \frac{\Delta x_{j-1/2}}{3c\Delta t} + 1 \qquad b_j = \frac{2}{3}\frac{\Delta x_{j-1/2} + \Delta x_{j+1/2}}{c\Delta t} \qquad c_j = \frac{\Delta x_{j+1/2}}{3c\Delta t} - 1$$

$$d_j = \frac{2}{c\Delta t}\left(\frac{\Delta x_{j-1/2}}{6}U_{j-1}^n + \frac{\Delta x_{j-1/2} + \Delta x_{j+1/2}}{3}U_j^n + \frac{\Delta x_{j+1/2}}{6}U_{j+1}^n\right)$$

Modeling Considerations

- Solve the tridiagonal system [A]Y=B
  - *Symmetry* - means correspondence in size, shape and position of U and boundary conditions that are on opposite sides of a dividing line or plane;
    - Use of symmetry allows us to consider a reduced problem instead of the actual problem.
    - The order of the total (global) stiffness matrix and the total number of equations can be reduced.
    - Solution time is reduced!
  - *Bandwidth* - An envelope that begins with the first nonzero component in each column of the [A] matrix





X	X	0	X	0	0	0	0	0	0	0	0
X	X	X	X	X	0	0	0	0	0	0	0
0	X	X	0	X	X	0	0	0	0	0	0
X	X	0	X	X	0	X	0	0	0	0	0
0	X	X	X	X	X	0	X	0	0	0	0
0	0	X	0	X	X	0	X	X	0	0	0
0	0	0	X	X	0	X	X	0	X	0	0
0	0	0	0	X	X	X	X	X	X	X	0
0	0	0	0	0	X	0	X	X	0	X	X
0	0	0	0	0	0	X	X	0	X	0	0
0	0	0	0	0	0	0	X	X	0	X	0
0	0	0	0	0	0	0	0	X	0	0	X

X is a nonzero 2 x 2 block n_b

Bandwidth

$n_b = n_{dof}\,(m + 1)$

Where:

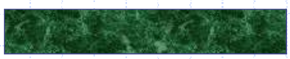
n_b is the semibandwidth

n_{dof} is the number of degrees of freedom per node.

m is the maximum difference in node numbers for any element.

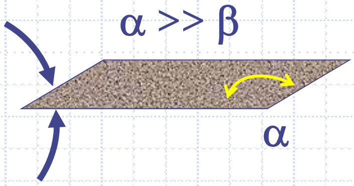
Poor Shapes

b >> h



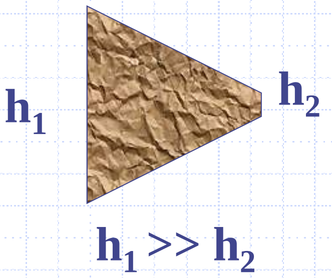
Large aspect ratio

α >> β



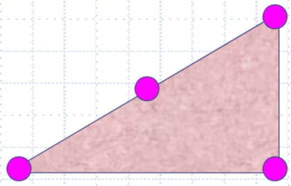
Very large and very small corner angles

Poor Shapes



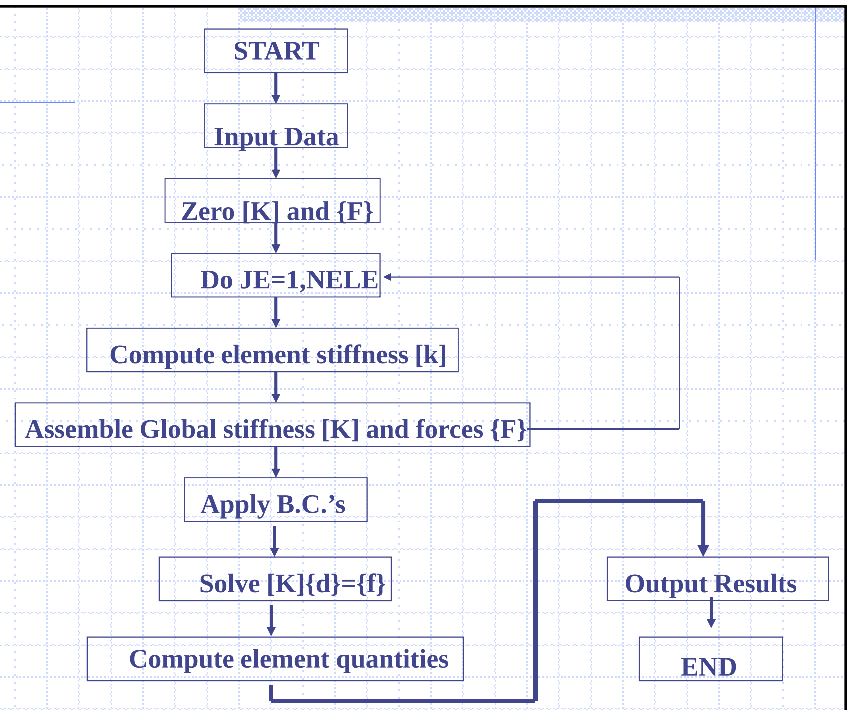
Quadrilateral approaching triangular shape

Quadrilateral degenerating into triangular shape



Flowchart for a

Finite Element Program.



INPUT

- ❑ Control parameters
  - Number of Elements
  - Number of Nodes
  - Number of B.C.'s
- ❑ Geometry
  - x,y,z location of each node
  - Element connectivity (which nodes are associated with which elements)

INPUT

- ❑ Element Properties
  - Area
  - Moment of Inertia
  - Thickness
  - Location of Neutral Axis
- ❑ Physical parameters Information

Programs

- ❑ALGOR
- ❑ANSYS
- ❑COSMOS/M
- ❑STARDYNE
- ❑IMAGES-3D
- ❑MSC/NASTRAN
- ❑SAP90
- ❑ADINA
- ❑NISA

What you should remember

- Finite Element Methods (FEMs) seek a *weak* solution to the PDEs.
- The solution is sought as the sum of a set of basis – or shape – functions. Those can be piecewise linear, or parabolic, etc.
- The Galerkin technique uses a weighting of the solution by functions that are the same as the shape functions