Advanced Computational Techniques

Part 2
Time dependent problems

Krzysztof Banaś

Department of Applied Computer Science and Modelling
AGH University of Science and Technology, Kraków, Poland

Time dependent problems in 1D – parabolic equations

- Heat conduction – the rate of change in time of the temperature at a point is equal to the imbalance between the heat flux spatial derivative and heat source at the point

$$\frac{\partial T}{\partial t} - \frac{\partial}{dx}\left(k\frac{\partial T}{dx}\right) = s(x,t)$$

- Unknown fields as functions of time and space: $T(x,t)$, $u(x,t)$
- Partial differential equations with partial derivatives
- Initial condition(s) in addition to boundary conditions
  - for all points inside the computational domain
  - $T(x,0) = T_0(x)$ – for non-stationary heat conduction
  - $u(x,0) = u_0(x)$, $\frac{\partial u}{\partial t}(x,0) = v_0(x)$ – for elastodynamics
  - initial conditions must agree with boundary conditions
- Initial-boundary value problems
  - **well posed $\rightarrow$ existence and uniqueness of solutions**
- Stationary problems can be considered as limits of non-stationary processes, after reaching steady-state

Convection equation

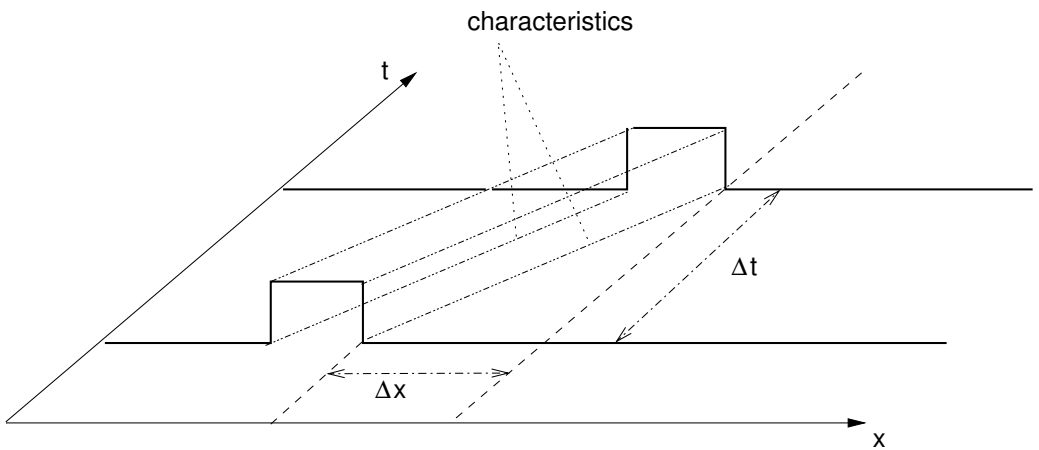
- The simplest time dependent problems are first order convection PDEs:

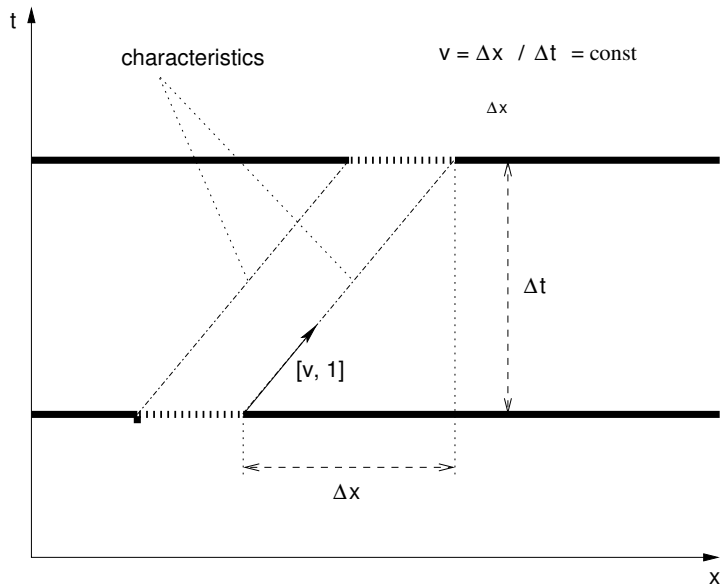
$$\frac{\partial u}{\partial t} + v_i \frac{\partial u}{\partial x_i} = f$$

  where $\boldsymbol{v}$ is the convection velocity
  - for $\boldsymbol{v}$ being the function of x only, the equation is linear
  - when $\boldsymbol{v}$ is the function of u (or its derivatives), the equation is non-linear
- Boundary conditions:
  - the part of the boundary where the velocity vector points inside – is the inflow boundary
    - on the inflow boundary the solution must be specified – to indicate what is convected into the computational domain
  - the part of the boundary where the velocity vector points outside – is the outflow boundary
    - on the outflow boundary the solution must not be specified – to allow for free departure from the computational domain

Convection equation

- The convection PDE requires an initial condition:
  - $u(\boldsymbol{x}, 0) = u_0(\boldsymbol{x})$
- In case of no source term f
  - the initial "shape" $u_0(\boldsymbol{x})$ is convected along the direction of velocity $\boldsymbol{v}$
- For the 1D case with no source term and constant v the solution is
  - $u(x,t) = u_0(x - vt)$
    - at time instant t_n: $u(x, t_n) = u_0(x - vt_n)$
  - the solution is just constant along characteristics $x = x_0 + vt$





The finite difference method for ODEs and PDEs (recall)

The finite difference method for ODEs and PDEs – approximation of derivatives that appear in equations with formulae that use the values at discrete points:

- based on Taylor's theorem

$$f(x_0+h) = f(x_0) + \frac{f'(x_0)}{1!}h + \frac{f^{(2)}(x_0)}{2!}h^2 + \cdots + \frac{f^{(n)}(x_0)}{n!}h^n + R_n(x_0+h)$$

where R_n is the remainder:

$$R_n(x_0+h) = \frac{f^{(n+1)}(\xi)}{(n+1)!}(h)^{n+1} = O(h^{n+1}) \qquad \text{for} \quad x_0 < \xi < x_0 + h$$

- for first order derivatives it gives:

$$f'(x_0) = \frac{f(x_0+h) - f(x_0)}{h} + O(h)$$

  - the formula is first order accurate, i.e. the discretization error is proportional to the first power of the distance (grid size) h

The finite difference method for PDEs (recall)

Approximation of partial derivatives appearing in equations using formulae with the values at discrete points:

- for example, given points $u^{t_n}_{x_{i-1}}$, $u^{t_n}_{x_i}$, $u^{t_n}_{x_{i+1}}$, $u^{t_{n+1}}_{x_i}$ $\left(u^{t_n}_{x_i} = u(t_n, x_i)\right)$
- and $\Delta t = t_{n+1} - t_n$, $\Delta x = x_{i+1} - x_i = x_i - x_{i-1}$:

$$\frac{\partial u}{\partial t}\Big|_{(t_n,x_i)} \approx \frac{u^{t_{n+1}}_{x_i} - u^{t_n}_{x_i}}{\Delta t}$$

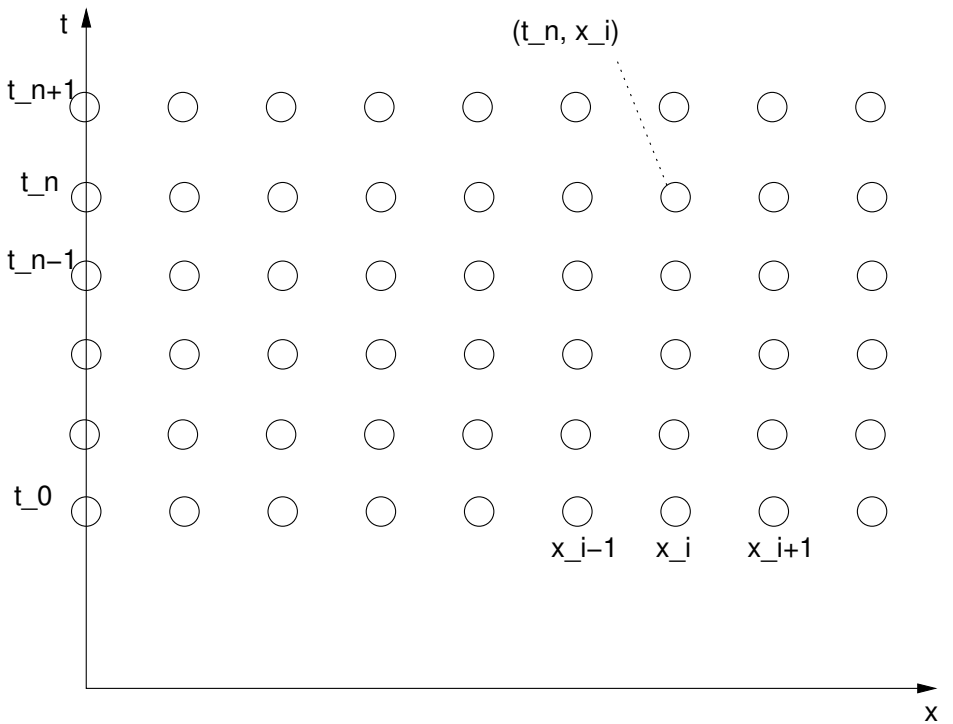
$$\frac{\partial u}{\partial x}\Big|_{(t_n,x_i)} \approx \frac{u^{t_n}_{x_{i+1}} - u^{t_n}_{x_i}}{\Delta x}$$

or

$$\frac{\partial u}{\partial x}\Big|_{(t_n,x_i)} \approx \frac{u^{t_n}_{x_i} - u^{t_n}_{x_{i-1}}}{\Delta x}$$

or

$$\frac{\partial u}{\partial x}\Big|_{(t_n,x_i)} \approx \frac{u^{t_n}_{x_{i+1}} - u^{t_n}_{x_{i-1}}}{2\Delta x}$$



The finite difference method and convection equations

- For the convection equation one of possible simple finite difference formulations is:
  - given the solution at time t_n (for t_0 taken from the initial condition) and each point x_i within the computational domain ...
  - calculate the solution at time t_{n+1} for each point x_i according to the formula:

$$\frac{u_{x_i}^{t_{n+1}} - u_{x_i}^{t_n}}{\Delta t} + v \cdot \frac{u_{x_{i+1}}^{t_n} - u_{x_i}^{t_n}}{\Delta x} = f(t_n, x_i)$$

- The formulated method is explicit, we do not have to solve a system of equations, but just calculate:

$$u_{x_i}^{t_{n+1}} = u_{x_i}^{t_n} - \Delta t \left(v \cdot \frac{u_{x_{i+1}}^{t_n} - u_{x_i}^{t_n}}{\Delta x} - f(t_n, x_i) \right)$$

- Usually explicit methods are stable (their solutions do not grow to infinity) only when suitable limits for time steps Δt are satisfied, e.g.:

$$\frac{\Delta t \cdot v}{\Delta x} < CFL_{\text{limit}} \quad (CFL \text{ is the, so called, Courant-Friedrichs-Lewy number})$$

The finite difference method and convection equations

- Hyperbolic problems pose important difficulties for numerical approximation
  - the methods that use characteristics (e.g. find characteristics and then solve the equations along characteristics) are difficult for systems of PDEs (PDEs for vector problems) and do not work well when, as is often found in practice, additional terms (e.g. with second order derivatives) appear in the equations
  - classical finite difference and finite element methods have problems with stability and accuracy (even for small time steps)
    - when exact solutions are not smooth the approximations exhibit spurious oscillations
    - for the finite difference methods, one of possible solutions is to use, so called, upwind differencing, where the choice of the difference formulae used for spatial derivatives depends upon the actual direction of velocity:

$$\frac{u_{x_i}^{t_n+1} - u_{x_i}^{t_n}}{\Delta t} + v \cdot \frac{u_{x_i}^{t_n} - u_{x_{i-1}}^{t_n}}{\Delta x} = f(t_n, x_i) \qquad \text{for } v > 0$$

$$\frac{u_{x_i}^{t_n+1} - u_{x_i}^{t_n}}{\Delta t} + v \cdot \frac{u_{x_{i+1}}^{t_n} - u_{x_i}^{t_n}}{\Delta x} = f(t_n, x_i) \qquad \text{for } v < 0$$

    - the above upwind scheme is stable for $CFL < 1$ and is first order accurate in time and in space

Accuracy, stability, consistency, convergence

Numerical approximation methods have several properties related to the behaviour of the discretization error $e_h = u - u_h$:

- The step size or grid size parameter h is assumed to be the most important parameter in studying the behaviour of e_h
  - it can be taken as the largest step size or grid cell (element) size (in the latter case some definition for 2D and 3D grids has to be adopted, e.g. the radius of the smallest ball (circle) that contains each grid cell or the longest element edge for the whole mesh)
- The most important property is the convergence of a numerical method
  - A numerical method converges if some suitable norm of discretization error tends to zero with the discretization parameter h going to zero

$$\|e_h\| \to 0 \quad \text{for} \quad h \to 0$$

- The order of accuracy of a discretization error specifies how fast the method converges to the exact solution with the step size or grid size tending to zero
  - it does not say how large is the error for a given value of h

Accuracy, stability, consistency, convergence

- Consistency is a measure to which extent the exact solution u satisfies the discrete problem
  - A numerical method is consistent if the exact solution satisfies the discrete problem in the limit $h \to 0$
- Stability determines whether the numerical (discrete) solution does not amplifies too much disturbances in problem parameters
  - the stability of numerical schemes correspond to the well-posedness of differential problems (continuous dependence on data)
  - in practical applications stability says whether the numerical solution can grow significantly (e.g. tend to infinity) in some circumstances
    - conditionally stable numerical schemes are stable for specific values of h
    - unconditionally stable schemes are stable for all values of h
- The fundamental theorem of numerical analysis states that the solutions of a scheme that is stable and consistent converge to the exact solution of the discretized problem

The finite difference method for ODEs (recall)

- There are several fundamental simple finite difference approximations for ODEs of the form

$$\frac{du}{dt} = f(t,u)$$

  - The first order accurate explicit (forward) Euler method:

$$\frac{u^{t_{n+1}} - u^{t_n}}{\Delta t} = f(t_n, u^{t_n}) \qquad \rightarrow \qquad u^{t_{n+1}} = u^{t_n} + \Delta t f(t_n, u^{t_n})$$

  - The first order accurate implicit (backward) Euler method:

$$\frac{u^{t_{n+1}} - u^{t_n}}{\Delta t} = f(t_{n+1}, u^{t_{n+1}}) \qquad \rightarrow \qquad u^{t_{n+1}} - \Delta t f(t_{n+1}, u^{t_{n+1}}) = u^{t_n}$$

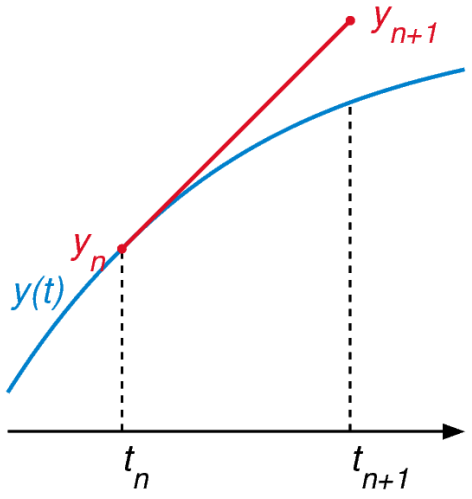
  - The second order accurate implicit Crank-Nicolson method:

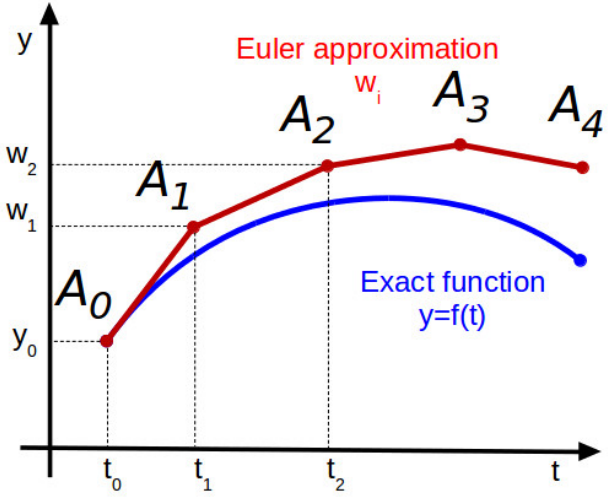
$$\frac{u^{t_{n+1}} - u^{t_n}}{\Delta t} = \frac{1}{2}\left(f(t_{n+1}, u^{t_{n+1}}) + f(t_n, u^{t_n})\right) \rightarrow u^{t_{n+1}} - \frac{\Delta t}{2} f(t_{n+1}, u^{t_{n+1}}) = u^{t_n} + \frac{\Delta t}{2} f(t_n, u^{t_n})$$

  - Implicit methods are more stable than explicit methods, but require the solution of an algebraic equation, that may be non-linear, at each step

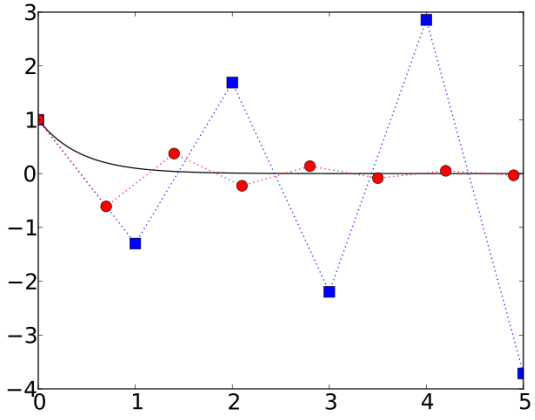
The finite difference method for ODEs (recall)

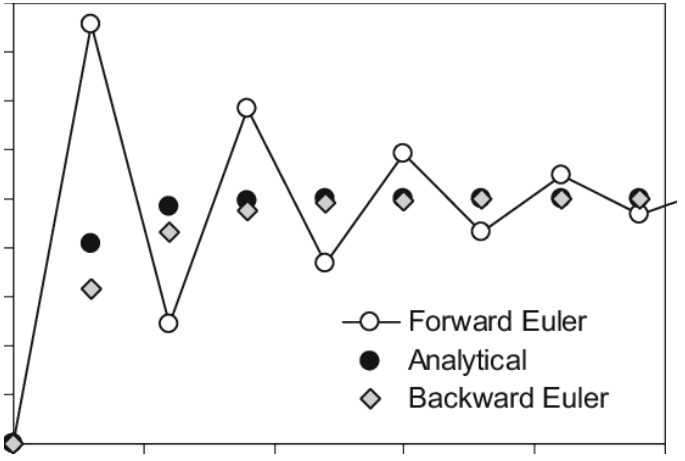
Explicit Euler time integration





Stability of time integration - dependence on the size of time step (blue – large, red – small) ... and the type of approximation





The finite difference method for the heat conduction equation

1D heat conduction equation (no convection - parabolic problem):

$$\frac{\partial T}{\partial t} - \frac{\partial}{\partial x}\left(k\frac{\partial T}{\partial x}\right) = s(x,t)$$

- Finite difference approximations:
  - second order derivative in space - the most popular approach: central difference

$$\frac{\partial}{dx}\left(k\frac{\partial T}{dx}\right)|_{(t,x_i)} \approx \frac{\partial}{dx}\left(k\frac{T^t_{x_{i+1}} - T^t_{x_i}}{\Delta x}\right)|_{(t,x_i)} \approx k\frac{\frac{T^t_{x_{i+1}} - T^t_{x_i}}{\Delta x} - \frac{T^t_{x_i} - T^t_{x_{i-1}}}{\Delta x}}{\Delta x}$$

$$\approx \frac{k}{\Delta x^2}\left(T^t_{x_{i+1}} - 2T^t_{x_i} + T^t_{x_{i-1}}\right)$$

  - the PDE becomes an ODE

$$\frac{dT}{dt}|_{(t,x_i)} = \frac{k}{\Delta x^2}\left(T^t_{x_{i+1}} - 2T^t_{x_i} + T^t_{x_{i-1}}\right) + s(x_i,t) = f(t)$$

The finite difference method for the heat conduction equation

1D heat conduction equation (no convection - parabolic problem):

$$\frac{\partial T}{\partial t} - \frac{\partial}{\partial x}\left(k\frac{\partial T}{\partial x}\right) = s(x,t)$$

- Finite difference approximations:
  - the obtained ODE is solved by one of basic methods, giving:
    - for the explicit Euler method

$$T_{x_i}^{t_{n+1}} = T_{x_i}^{t_n} + k\frac{\Delta t}{\Delta x^2}\left(T_{x_{i+1}}^{t_n} - 2T_{x_i}^{t_n} + T_{x_{i-1}}^{t_n}\right) + s(x_i, t_n)$$

    - for the implict Euler method

$$T_{x_i}^{t_{n+1}} - k\frac{\Delta t}{\Delta x^2}\left(T_{x_{i+1}}^{t_{n+1}} - 2T_{x_i}^{t_{n+1}} + T_{x_{i-1}}^{t_{n+1}}\right) - s(x_i, t_{n+1}) = T_{x_i}^{t_n}$$

    - for the Crank-Nicolson method

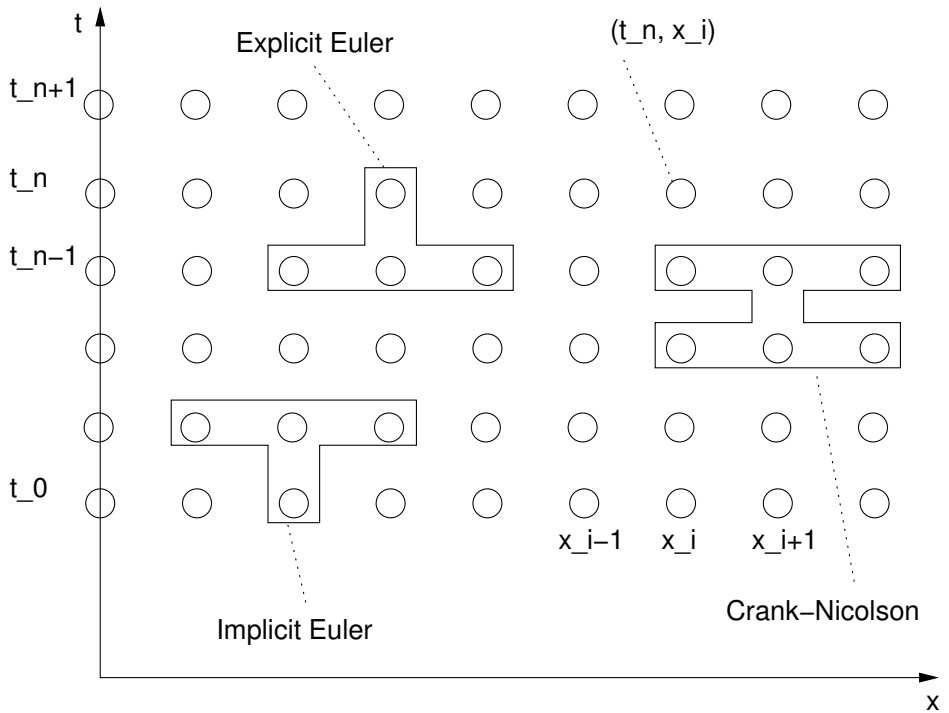
$$2T_{x_i}^{t_{n+1}} - k\frac{\Delta t}{\Delta x^2}\left(T_{x_{i+1}}^{t_{n+1}} - 2T_{x_i}^{t_{n+1}} + T_{x_{i-1}}^{t_{n+1}}\right) - s(x_i, t_{n+1}) = 2T_{x_i}^{t_n} + k\frac{\Delta t}{\Delta x^2}\left(T_{x_{i+1}}^{t_n} - 2T_{x_i}^{t_n} + T_{x_{i-1}}^{t_n}\right) + s(x_i, t_n)$$

The finite difference method for the heat conduction equation

1D heat conduction equation (no convection - parabolic problem):

$$\frac{\partial T}{\partial t} - \frac{\partial}{\partial x}\left(k\frac{\partial T}{\partial x}\right) = s(x,t)$$

- Finite difference approximations:
  - the finite difference formulae link several values at particular points and time instants, creating, so called, finite difference stencils:



The combined FEM+FDM approach for non-stationary problems

Non-stationary heat conduction in several space variables – notation

- $T(\boldsymbol{x}, t)$ - temperature as function of time and space
  - $T^n(\boldsymbol{x}) = T(\boldsymbol{x}, t^n)$
- assumption for finite element discretization:

$$T(\boldsymbol{x}, t) = \sum_{L=1}^{N} \mathrm{T}_L(t)\psi_L(\boldsymbol{x})$$

$\mathbf{T}(t) = \{\mathrm{T}_L(t)\}$ - the set of degrees of freedom for the approximation of T

- hence

$$\frac{\partial T(\boldsymbol{x}, t)}{\partial t} = \sum_{L=1}^{N} \frac{d\mathrm{T}_L}{dt}\psi_L(\boldsymbol{x}) = \sum_{L=1}^{N} \dot{\mathrm{T}}_L\psi_L(\boldsymbol{x}) \qquad T^n(\boldsymbol{x}) = \sum_{L=1}^{N} \mathrm{T}_L^n\psi_L(\boldsymbol{x})$$

- and $\dot{\mathbf{T}} = \{\dot{\mathrm{T}}_L\}$ $\mathbf{T}^n = \{\mathrm{T}_L^n\}$

- test functions in the standard way:

$$w = \sum_{K=1}^{N} \mathrm{W}_K\psi_K(\boldsymbol{x})$$

The combined FEM+FDM approach for non-stationary problems

Non-stationary heat equations in several space variables

$$\frac{\partial T}{\partial t} - k\frac{\partial^2 T}{\partial x_i^2} = s(x,t) \qquad T_{,t} - kT_{,ii} = s(x,t)$$

- weak formulation (Dirichlet boundary conditions for simplicity)

$$(T_{,t}, w) + k(T_{,i}, w_{,i}) = (s, w) \qquad \forall w \in V_0$$

- finite element space discretization leads to

$$\sum_{L=1}^{N} (\psi_K, \psi_L)\,\dot{\mathrm{T}}_L + \sum_{L=1}^{N} k\left(\frac{d\psi_K}{dx}, \frac{d\psi_L}{dx}\right)\mathrm{T}_L = (s, \psi_K) \quad \text{for} \quad K = 1, 2, \ldots, N$$

- that can be written as

$$\mathbf{M}\dot{\mathbf{T}} + \mathbf{K}\mathbf{T} = \mathbf{b}$$

with

$$\mathrm{M}_{K,L} = (\psi_K, \psi_L) \quad \mathrm{K}_{K,L} = k\left(\frac{d\psi_K}{dx_i}, \frac{d\psi_L}{dx_i}\right) \quad \mathrm{b}_K^n = (\psi_K, \psi_L)\,\mathrm{T}_L^n + \Delta t(s(t_n, x), \psi_K)$$

The combined FEM+FDM approach for non-stationary problems

- FDM for linear ODEs of the form

$$\mathbf{M}\dot{\mathbf{T}} + \mathbf{K}\mathbf{T} = \mathbf{b}$$

  - explicit Euler time integration:

$$\mathbf{M}\mathbf{T}^{n+1} = \mathbf{M}\mathbf{T}^{n} - \Delta t\mathbf{K}\mathbf{T}^{n} + \Delta t\mathbf{b}^{n}$$

  - implicit Euler time integration:

$$\mathbf{M}\mathbf{T}^{n+1} + \Delta t\mathbf{K}\mathbf{T}^{n+1} = \mathbf{M}\mathbf{T}^{n} + \Delta t\mathbf{b}^{n+1}$$

  - Crank-Nicolson time integration:

$$\mathbf{M}\mathbf{T}^{n+1} + \frac{1}{2}\Delta t\mathbf{K}\mathbf{T}^{n+1} = \mathbf{M}\mathbf{T}^{n} - \frac{1}{2}\Delta t\mathbf{K}\mathbf{T}^{n} + \frac{1}{2}\Delta t\left(\mathbf{b}^{n} + \mathbf{b}^{n+1}\right)$$

  - the above time integration schemes can be generalized into the so called α-method

$$\mathbf{M}\mathbf{T}^{n+1} + \alpha\Delta t\mathbf{K}\mathbf{T}^{n+1} = \mathbf{M}\mathbf{T}^{n} - (1-\alpha)\Delta t\mathbf{K}\mathbf{T}^{n} + \Delta t\left((1-\alpha)\mathbf{b}^{n} + \alpha\mathbf{b}^{n+1}\right)$$

    with: $\alpha = 0$ – for explicit Euler, $\alpha = 1$ – for implicit Euler, $\alpha = 0.5$ – for Crank-Nicolson,

The combined FEM+FDM approach for non-stationary problems

- The method of lines
  - the same discretization steps as for the non-stationary heat equation can be done for other time-dependent problems
  - for first order equations the resulting systems of ordinary differential equations may be written as:

$$\mathbf{M}\dot{\mathbf{T}} + \mathbf{K}\mathbf{T} = \mathbf{b}$$

  - for second order hyperbolic problems the ODEs have the form:

$$\mathbf{M}\ddot{\mathbf{U}} + \mathbf{C}\dot{\mathbf{U}} + \mathbf{K}\mathbf{U} = \mathbf{b}$$

  - due to interpretations in mechanics the matrix M is usually caled ”the mass matrix”, while the matrix K is ”the stiffness matrix”
  - any method, explicit or implicit, can be used to solve the above ODEs
    - typical choices include: the introduced variations of the α-method, Runge-Kutta methods, a family of Newmark methods for second order equations, discontinuous Galerkin time discretization, etc.
    - in order not to solve a system of linear equations at each time step, so called ”mass lumping” (diagonalization of **M**) is performed for explicit methods

Linear convection-diffusion-reaction equations

- General form for time dependent convection-diffusion-reaction equations for a vector valued function $\boldsymbol{u} = [u_1, u_2, ..., u_{N_u}]$, posed in the computational domain $\Omega \subset \boldsymbol{R}^s$, $s = 2$ or 3, with boundary Γ

$$\boldsymbol{M}\frac{\partial \boldsymbol{u}}{\partial t} - \left(\boldsymbol{A}^{ij}\boldsymbol{u}_{,j}\right)_{,i} + \left(\boldsymbol{B}^{i}\boldsymbol{u}\right)_{,i} + \boldsymbol{C}\boldsymbol{u} = \boldsymbol{s} - \boldsymbol{q}^{i}_{,i}$$

  - equivalent form

$$\sum_{l=1}^{N_u} m_{kl}\frac{\partial u_l}{\partial t} - \boldsymbol{\nabla}\cdot\left(\sum_{l=1}^{N_u}\boldsymbol{A}_{kl}\boldsymbol{\nabla}u_l\right) + \boldsymbol{\nabla}\cdot\left(\sum_{l=1}^{N_u}\boldsymbol{b}_{kl}u_l\right) + \sum_{l=1}^{N_u} c_{kl}u_l = s_k - \boldsymbol{\nabla}\boldsymbol{q}_k$$

    with $k = 1, 2, ...N_u$.
  - a particular case of time dependent heat equation for the temperature $T(\boldsymbol{x}, t)$

$$\rho c\left(\frac{\partial T}{\partial t} + \boldsymbol{v}\cdot\boldsymbol{\nabla}T\right) - \boldsymbol{\nabla}\cdot(k\boldsymbol{\nabla}T) = s$$

    - with density ρ, specific heat c, velocity field $\boldsymbol{v}$, heat conductivity k and source s

FEM+FDM for linear convection-diffusion-reaction equations

- Boundary conditions on Γ
  - Dirichlet (essential) on Γ_{D}:
    $$\boldsymbol{u} = \boldsymbol{f}^{\mathrm{D}}(\boldsymbol{x}, t)$$
  - Neumann (natural) on Γ_{N}:
    $$\boldsymbol{A}^{ij}\boldsymbol{u}_{,j}n_i = \boldsymbol{f}^{\mathrm{N}}(\boldsymbol{x}, t)$$
  - Robin (third type) on Γ_{R}:
    $$\boldsymbol{A}^{ij}\boldsymbol{u}_{,j}n_i = \left(\boldsymbol{u} - \boldsymbol{f}^{\mathrm{R}}(\boldsymbol{x}, t)\right)\boldsymbol{K}^{\mathrm{R}}(\boldsymbol{x}, t)$$

    with $\boldsymbol{K}^{\mathrm{R}}, \boldsymbol{f}^{\mathrm{D}}, \boldsymbol{f}^{\mathrm{N}}$ and $\boldsymbol{f}^{\mathrm{R}}$ given matrix and vector valued functions
  - for the special case of heat equation
    - Dirichlet on Γ_T
      $$T = T_D$$
    - Neumann on Γ_q (with $\boldsymbol{n}$ the unit outward vector normal to the boundary)
      $$-kT_{,i}n_i = q_N$$
    - convection-radiation on Γ_R
      $$-kT_{,i}n_i = -h_c(T - T_A) - \Sigma\epsilon(T^4 - T_A^4)$$

      with: h_c – heat transfer coefficient, T_A – ambient temperature, ϵ – emissivity and Σ – Stefan-Boltzmann constant

FEM+FDM for linear convection-diffusion-reaction equations

- Standard finite element procedures (multiplication by test functions, generalized integration by parts, i.e. Green-Gauss-Ostrogradski theorem) lead to the weak statement (valid for any test function $\boldsymbol{w}$)

$$\int_{\Omega}\left(\boldsymbol{M}\boldsymbol{w}\frac{\partial \boldsymbol{u}}{\partial t}+\boldsymbol{A}^{ij}\boldsymbol{w}_{,i}\boldsymbol{u}_{,j}-\boldsymbol{B}^{i}\boldsymbol{w}_{,i}\boldsymbol{u}+\boldsymbol{C}\boldsymbol{w}\boldsymbol{u}\right)d\Omega+$$

$$+\int_{\Gamma^{+}\cup\Gamma_{\mathrm{R}}^{-}}\boldsymbol{B}^{i}n^{i}\boldsymbol{w}\boldsymbol{u}d\Gamma-\int_{\Gamma_{\mathrm{R}}}\boldsymbol{K}^{\mathrm{R}}\boldsymbol{w}\boldsymbol{u}d\Gamma=$$

$$\int_{\Omega}\boldsymbol{s}\boldsymbol{w}d\Omega+\int_{\Omega}\boldsymbol{q}^{i}\boldsymbol{w}_{,i}d\Omega-\int_{\Gamma}\boldsymbol{q}^{i}n^{i}\boldsymbol{w}d\Gamma+\int_{\Gamma_{\mathrm{N}}}\boldsymbol{w}\boldsymbol{f}^{\mathrm{N}}d\Gamma-\int_{\Gamma_{\mathrm{R}}}\boldsymbol{K}^{\mathrm{R}}\boldsymbol{w}\boldsymbol{f}^{\mathrm{R}}d\Gamma$$

  - for the special case of the heat equation (no Robin conditions)

$$\int_{\Omega}\rho c\frac{\partial T}{\partial t}wd\Omega+\int_{\Omega}\rho cv_{i}T_{,i}wd\Omega+\int_{\Omega}kT_{,i}w_{,i}d\Omega=\int_{\Omega}swd\Omega-\int_{\Gamma_{q}}q_{N}wd\Gamma$$

FEM+FDM for linear convection-diffusion-reaction equations

- Scalar linear convection diffusion equation

$$u_t + v_i u_{,i} - k u_{,ii} = s$$

  - the equation is formally parabolic (due to the second order terms)
  - for $\|\boldsymbol{v}\| >> k$ the equation has dominating convection
  - in the limit $k \to 0$ the equation becomes hyperbolic

- The equation has the standard weak formulation of the form

$$\int_\Omega \frac{\partial u}{\partial t} w d\Omega + \int_\Omega v_i u_{,i} w d\Omega + \int_\Omega k u_{,i} w_{,i} d\Omega = \int_\Omega s w d\Omega$$

  (for homogeneous Dirichlet boundary conditions for simplicity)

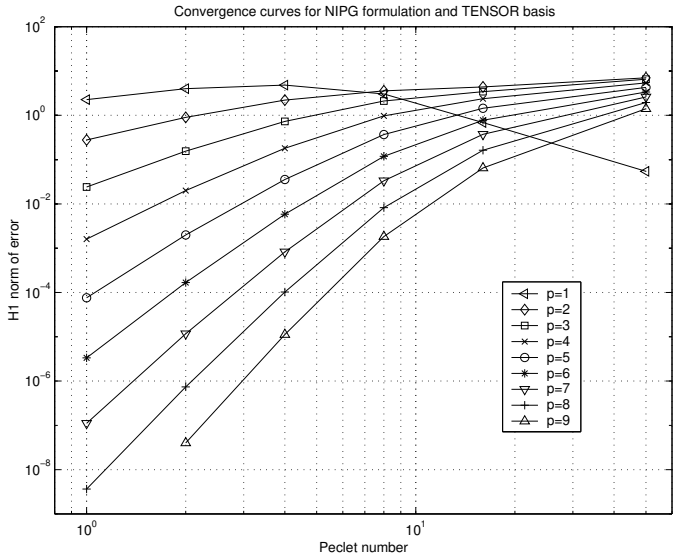
- For both types. hyperbolic and with dominating convection, standard finite element procedures lead to oscillations of the solution
  - there are many methods for obtaining stable solutions to the equation
    - some of them are similar to the upwinding used for FDM
  - one of the most popular technique is to stabilize the formulation by adding suitable second order terms (in the form of a special additional diffusivity)

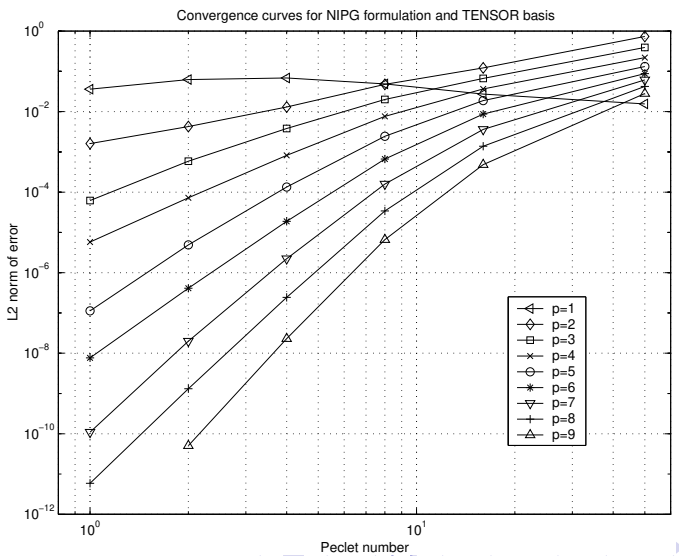
FEM+FDM for linear convection-diffusion-reaction equations

- The important parameter for the stability and convergence of discretization of convection dominated equations is the element Péclet number

$$\mathrm{Pe} = \frac{h\|\boldsymbol{v}\|}{2k}$$

  - for linear convection diffusion equations the solutions of standard finite element formulations and linear elements are stable when $\mathrm{Pe} < 1$
    - for every proportion of velocity magnitude (inertia forces) to diffusion coefficient there is a mesh size that guarantees the stability of solution
    - for many practical applications such mesh sizes are impractical, leading to billions of degrees of freedom in the mesh





FEM+FDM for linear convection-diffusion-reaction equations

- Stabilized formulation for the scalar convection-diffusion equation:
  - one of many formulations for the stabilized finite element method uses residuals of the original equation:

$$R(u) = \frac{\partial u}{\partial t} + v_i u_{,i} - k u_{,ii} - s$$

with a similar expression for test functions:

$$\bar{R}(w) = v_i w_{,i} - k w_{,ii}$$

  - the stabilized formulation uses the following integral statement:

$$\int_\Omega \frac{\partial u^h}{\partial t} w^h d\Omega + \int_\Omega v_i u^h_{,i} w^h d\Omega + \int_\Omega k u^h_{,i} w^h_{,i} d\Omega + \sum_e \int_{\Omega_e} R(u^h) \omega \bar{R}(w^h) d\Omega = \int_\Omega s w^h d\Omega$$

    - ω is the coefficient of stabilization, integrals are calculated over element interiors
    - thanks to the use of residuals the formulation is consistent
    - it can be proven that the solutions to the stabilized problems converge to the exact solution of the original equations

FEM+FDM for linear convection-diffusion-reaction equations

- For the special case of the time dependent heat equation (no Robin boundary conditions) the stabilized integral statement is the following:

$$\int_\Omega \rho c \frac{\partial T^h}{\partial t} w^h d\Omega + \int_\Omega \rho c v_i T^h_{,i} w^h d\Omega + \int_\Omega k T^h_{,i} w^h_{,i} d\Omega$$
$$+ \sum_e \int_{\Omega_e} R^F(T^h) \omega \bar{R}^F(w^h) d\Omega = \int_\Omega s w^h d\Omega - \int_{\Gamma_q} q_N w^h d\Gamma$$

  - for stationary problems and linear elements (with zero second order derivatives of shape functions inside elements), the stabilization term becomes in essence

$$\int_{\Omega_e} v_i T^h_{,i} \omega v_i w^h_{,i} d\Omega$$

    and the stabilized method can be interpreted as the standard weak form for the equation with suitably modified test functions $w^h + \omega v_i w^h_{,i}$
  - the procedure of modifying test functions can be interpreted as corresponding to the upwinding in the FDM and the family of related stabilization techniques is often named streamline upwind Petrov-Galerkin (SUPG) method

FEM+FDM for linear convection-diffusion-reaction equations

- After applying the α-method for time integration the final integral equation becomes (with Robin boundary conditions included):

$$\int_\Omega \rho c \frac{T^{n+1}}{\Delta t} w d\Omega + \alpha \int_\Omega \rho c v_i T^{n+1}_{,i} w d\Omega + \alpha \int_\Omega k T^{n+1}_{,i} w_{,i} d\Omega$$
$$+ \sum_e \int_{\Omega_e} R^F(T^{n+1}) \omega \bar{R}^F(w) d\Omega - \int_{\Gamma_R} h_c T^{n+1} w^h d\Gamma$$
$$= \int_\Omega s w d\Omega - \int_{\Gamma_q} q_N w^h d\Gamma - \int_{\Gamma_R} h_c T_A w^h d\Gamma$$
$$+ \int_\Omega \rho c \frac{T^n}{\Delta t} w d\Omega + (\alpha - 1) \int_\Omega \rho c v_i T^n_{,i} w d\Omega + (\alpha - 1) \int_\Omega k T^n_{,i} w_{,i} d\Omega$$

- The whole discretization procedure leads to the system of equations (linear or non-linear) for each time step
  - the integrals for entries in the system matrix and the right hand side vector directly correspond to the finite element weak statement above

The fundamental equations in mechanics and thermodynamics

The continuity equation (mass balance)

$$\frac{\partial \rho}{\partial t} + (\rho v_i)_{,i} = 0$$

Momentum equation

$$\frac{\partial (\rho v_j)}{\partial t} + (\rho v_j v_i)_{,i} + p_{,i} - \tau_{ji,i} = b_j$$

Energy balance equation

$$\frac{\partial (\rho e)}{\partial t} + ((\rho e + p) v_i - \tau_{ji} v_j + q_i)_{,i} = 0$$

Compressible fluid flow equations in conservative form

The equations for the three conservation principles in vector form

$$\begin{pmatrix} \rho \\ \rho v_j \\ \rho e \end{pmatrix}_{,t} + \begin{pmatrix} \rho v_i \\ \rho v_i v_j + p\delta_{ij} - \tau_{ij} \\ (\rho e + p)v_i - \tau_{ij} v_j + q_i \end{pmatrix}_{,i} = 0$$

Defining:

- $\boldsymbol{U} = (\rho, \rho v_j, \rho e)^T$ - vector of conservation variables
- $\boldsymbol{f}_i^E = (\rho v_i, \rho v_i v_j + p\delta_{ij}, (\rho e + p)v_i)^T$ – vector of Eulerian (inviscid) fluxes
- $\boldsymbol{f}_i^\mu = (0, \tau_{ij}, \tau_{ij} v_j - q_i)^T$ - vector of viscous and heat fluxes

leads to:

The Navier-Stokes equations of compressible fluid flow

$$\boldsymbol{U}(\boldsymbol{x}, t)_{,t} + \boldsymbol{f}_i^E(\boldsymbol{U})_{,i} = \boldsymbol{f}_i^\mu(\boldsymbol{U}, \nabla \boldsymbol{U})_{,i}$$

Finite volume method

Finite volume method (FVM) is a method for finding approximate solutions to partial differential equations, especially PDEs formulated as conservation laws

- The idea of the finite volume method for compressible Euler equations:

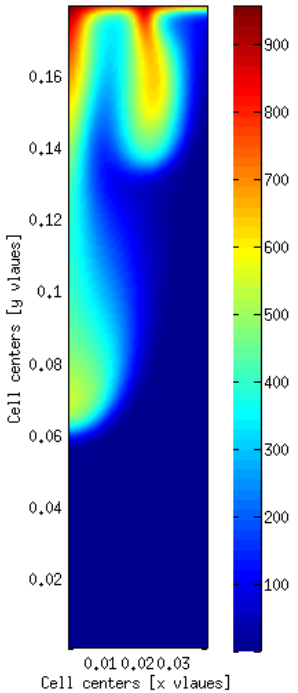
$$\boldsymbol{U}(\boldsymbol{x},t)_{,t} + \boldsymbol{f}_i^E(\boldsymbol{U})_{,i} = 0$$

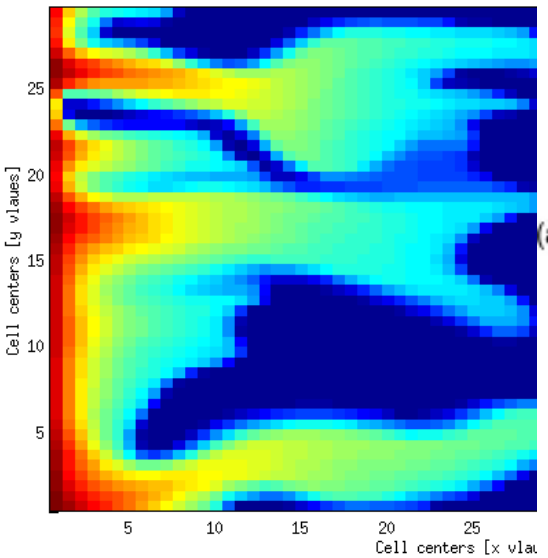
  - discretization of the computational domain into cells (control volumes) – the cells can be either simple (as finite elements) or can be constructed as Voronoi diagrams based on standard meshes
  - integration of the PDEs over the computational domain, splitting the global integral into the sum of integrals over each cell, the application of the Green-Gauss-Ostrogradski theorem for fluxes – the exchange of volume integrals to surface integrals

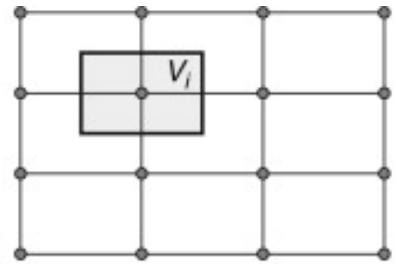
$$\int_{V_c} \boldsymbol{U}(\boldsymbol{x},t)_{,t} dV = -\int_{\partial V_c} \boldsymbol{f}_i^E(\boldsymbol{U}) n_i dS$$

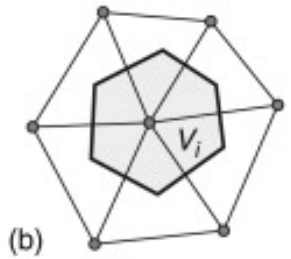
  - the application of special techniques for calculating approximations to integrals of fluxes based on discrete values and for updating these values based on calculated fluxes

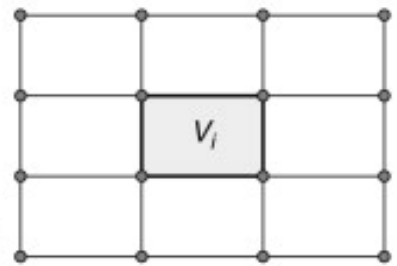
Finite volume method

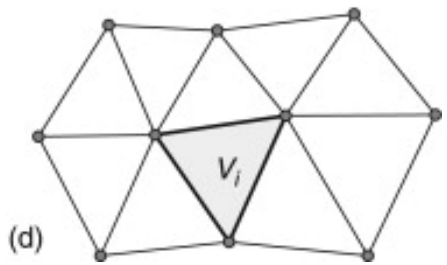


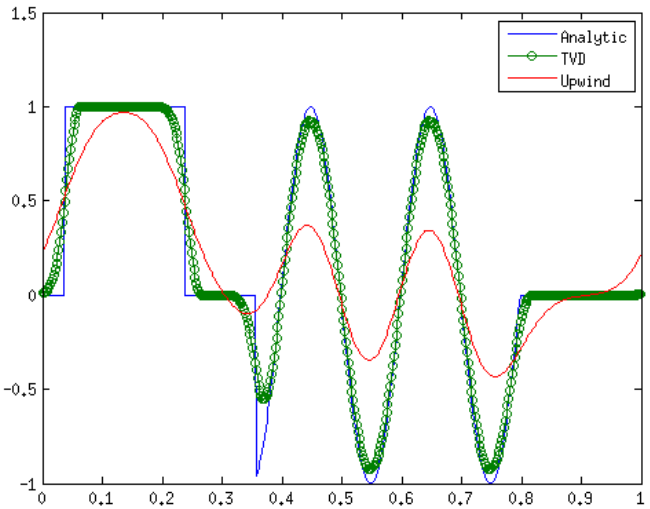


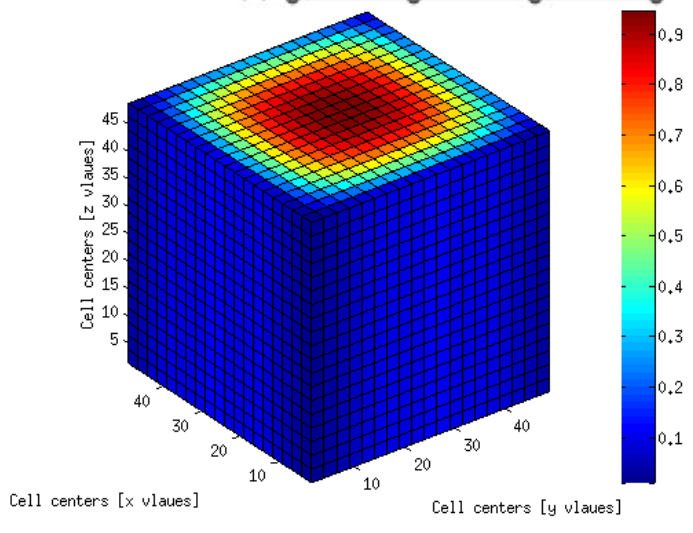












Finite volume method

- Based on suitable extensions of the basic scheme, the FVM can be applied to inviscid and viscous, compressible and incompressible flows, as well as different than fluid dynamics application areas
- The main advantage of the finite volume method is its conservative character, related to the direct application of conservative principles for each cell
- Moreover, due to special methods for constructing flux approximations, the FVM, for many problems, can eliminate oscillations of solutions in the regions of rapid gradient changes (such as e.g. shocks)
- The FVM does not introduce higher order cells – the values are associated either with cells or mesh vertices
- *h*-adaptivity and remeshing (usually called Adaptive Mesh Refinement (AMR) techiques) are often applied to locally increase accuracy of solutions
- The FVM method for time dependent problems can use explicit or implicit time integration schemes
- In practical applications FVM is today the most popular discretization technique in computational fluid dynamics (CFD)

The fundamental equations in mechanics and thermodynamics

The continuity equation (mass balance)

$$\frac{\partial \rho}{\partial t} + (\rho v_i)_{,i} = 0$$

Momentum equation

$$\frac{\partial (\rho v_j)}{\partial t} + (\rho v_j v_i)_{,i} + p_{,i} - \tau_{ji,i} = b_j$$

Energy balance equation

$$\frac{\partial (\rho e)}{\partial t} + ((\rho e + p) v_i - \tau_{ji} v_j + q_i)_{,i} = 0$$

Incompressible fluid flow – the Navier-Stokes equations

Derivation of the Navier-Stokes equations for incompressible fluid flow:

- due to incompressibility density is assumed to be constant
  - $\rho = \rho_0 = const$, the volume for a group of particles remains the same
  - the terms in the fluid flow equations with the gradient of density vanish
  - the analysis of the thermodynamic relations indicates that the speed of sound is infinite for incompressible media
  - therefore the changes in the pressure field propagate immediately throughout the whole computational domain
  - although the model of incompressible flow is purely theoretical (there are no fully incompressible materials) the approximations to the real flows based on the model can be relatively easily computed and have great practical importance

Mass balance (the continuity equation) for incompressible fluid flow

$$\frac{\partial \rho}{\partial t} + \nabla(\rho \boldsymbol{v}) = 0 \quad \Leftrightarrow \quad \nabla \cdot \boldsymbol{v} = 0$$

Incompressible fluid flow – the Navier-Stokes equations

Derivation of the Navier-Stokes equations for incompressible fluid flow:

- since the density is constant, the pressure becomes the function of velocity field only, not the thermodynamic quantity
- the continuity and momentum equations decouple from the energy equation (the pressure and velocity fields do not depend on the temperature field)

Momentum balance for incompressible fluid flow

$$\rho_0\left(\frac{\partial \boldsymbol{v}}{\partial t} + (\boldsymbol{v}\cdot\boldsymbol{\nabla})\boldsymbol{v}\right) - \mu\boldsymbol{\nabla}^2\boldsymbol{v} + \boldsymbol{\nabla}p = \boldsymbol{f}$$

Historically, the Navier-Stokes equations denoted the equations for fluid velocities, derived form the momentum balance. The name is also used for the whole system that must solved to find the velocity field in fluid flow and, hence, denotes the coupled equations for mass, momentum and energy balance for compressible flows and coupled equations for mass and momentum balance for incompressible flows

Incompressible fluid flow – the Navier-Stokes equations

Derivation of the Navier-Stokes equations for incompressible fluid flow:

- the Navier-Stokes equations assume the viscous stress tensor for the Newtonian incompressible fluid in the form:
  - $\tau_{ij} = \mu(v_{i,j} + v_{j,i})$
- usually the system is closed with the typical boundary conditions

Boundary conditions for incompressible fluid flow

$$\begin{aligned} \boldsymbol{v} &= \hat{\boldsymbol{v}}_0 \quad \text{on } \Gamma_D \\ (\mu \boldsymbol{\nabla} \boldsymbol{v})\boldsymbol{n} - p\boldsymbol{n} &= \boldsymbol{g} \quad \text{on } \Gamma_N \end{aligned}$$

- in the so called Boussinesq approximation for the buoyancy driven flows, the density in the momentum balance may be treated as linear function of the temperature
  - e.g. $\rho = \rho_0 + \Delta\rho(T) = \rho_0 - \alpha\rho_0(T - T_0)$
  - this approximation is used for modifying the body force due to gravity: $\boldsymbol{f} = \rho_0 \mathbf{g}\,(1 - \alpha(T - T_0))$

Energy equation for incompressible fluid flow

- the energy balance equation formulated for the total energy $e = e_I + e_K$

$$\left(\rho(e_I + \frac{1}{2}v_j v_j)\right)_{,t} + \left((\rho(e_I + \frac{1}{2}v_j v_j) + p)v_i - \tau_{ij}v_j + q_i\right)_{,i} = 0$$

can be transformed to the equation for the internal energy alone (by using the momentum balance to eliminate the mechanical energy $e_K = \frac{1}{2}v_j v_j$ from the equation):

$$(\rho e_I)_{,t} + (\rho e_I v_i)_{,i} + p v_{i,i} - \tau_{ij}v_{i,j} + q_{i,i} = 0$$

- the assumption of incompressibility leads to further simplifications of the model:

$$\rho c_p \left(\frac{\partial T}{\partial t} + T_{,i}v_i\right) - (\kappa T_{,i})_{,i} = 0$$

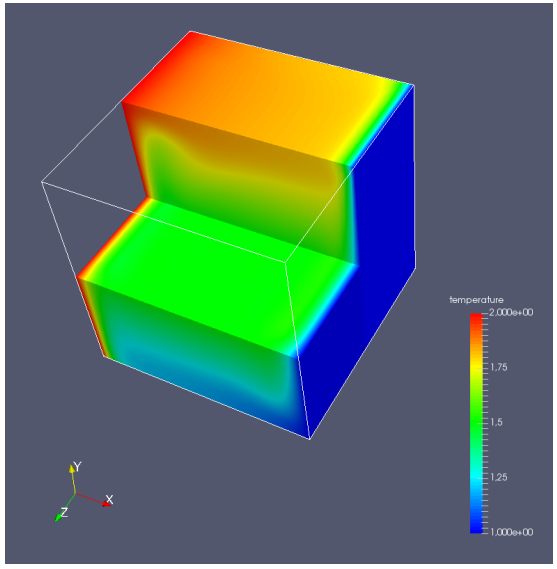
due to the fact that: $v_{i,i} = 0$, $c_p = c_V = const$, $e_I = c_V T$, $q_i = -\kappa T_{,i}$ and $\tau_{ij}v_{i,j}$ (heat produced by fluid viscosity) is small and can be neglected

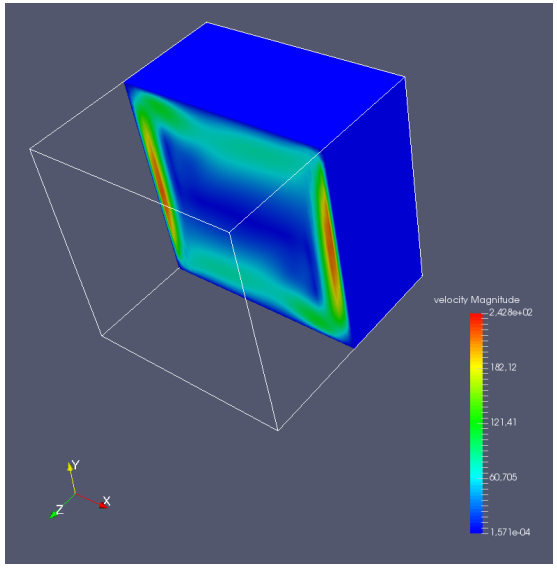
Energy equation for incompressible fluid flow

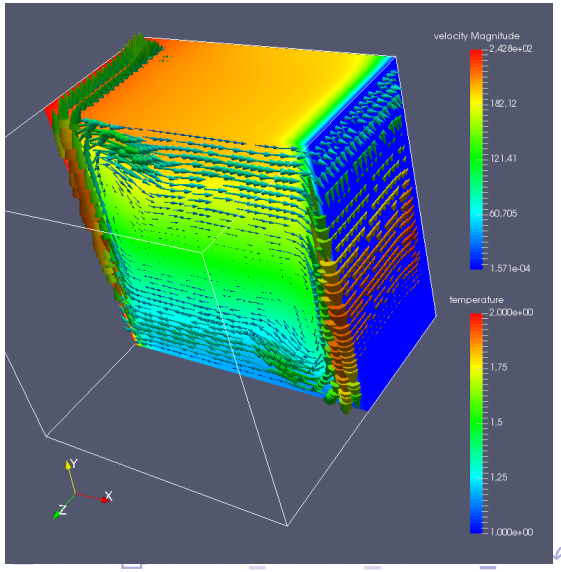
Convective heat transfer equation

$$\rho c \left(\frac{\partial T}{\partial t} + \boldsymbol{v} \cdot \boldsymbol{\nabla} T \right) - \boldsymbol{\nabla} \cdot (\kappa \boldsymbol{\nabla} T) = s$$

- temperature distribution depends on the velocity field
- velocity and pressure fields do not depend on the temperature field
  - the exception is the Bussinesq approximation with gravity causing hotter fluid to move up and the cooler fluid to move down







Incompressible fluid flow – the Navier-Stokes equations

The Navier-Stokes equations for incompressible fluid flow is an example of non-linear convection-diffusion equations

Mass balance

$$\frac{\partial \rho}{\partial t} + \nabla(\rho \boldsymbol{v}) = 0 \Leftrightarrow \nabla \cdot \boldsymbol{v} = 0$$

Momentum balance

$$\rho_0\left(\frac{\partial \boldsymbol{v}}{\partial t} + (\boldsymbol{v} \cdot \nabla)\boldsymbol{v}\right) - \mu \nabla^2 \boldsymbol{v} + \nabla p = \boldsymbol{f}$$

Boundary conditions

$$\boldsymbol{v} = \hat{\boldsymbol{v}}_0 \quad \text{on } \Gamma_D$$

$$(\mu \nabla \boldsymbol{v})\boldsymbol{n} - p\boldsymbol{n} = \boldsymbol{g} \quad \text{on } \Gamma_N$$

Comments

Different character of the two equations

- elliptic mass balance (divergence free condition) – equivalent to the so called pressure Poisson equation (derived from the momentum balance, taking into account the mass balance):
  - $\rho_0 \boldsymbol{v}_{i,j}\boldsymbol{v}_{j,i} = p_{,jj} + \boldsymbol{f}_{j,j}$
- usually convection dominated momentum balance
  - historically, the Navier-Stokes equations denoted the equations for fluid velocities, derived form the momentum balance
  - the name is also used for the whole system that must solved to find the velocity field in fluid flow and, hence, denotes the coupled equations for mass and momentum balance
- the convective heat transfer equation is often combined with the Navier-Stokes equations to describe the whole phenomena related to mass, momentum and energy balance for incompressible fluids

Approximating the Navier-Stokes equations by the FEM

- The Navier-Stokes equations and the convective heat transfer equation can be approximated using the finite element method
- For the momentum balance equations and the heat transfer equation, in the case of dominating convection, standard finite element procedures lead to the solutions with strong oscillations
  - the family of SUPG stabilization methods (introduced already for the time dependent heat equation) can be used to effectively remove oscillations, maintaining the high accuracy of the approximate solution
- Moreover, the fact that divergence free condition is equivalent to the elliptic equation for the pressure leads to the instability of the whole formulation (so called ”chequerboard patterns” appearing for the pressure)
  - in order to deal with the stability problems several approaches have been developed including
    - mixed formulations with different approximation spaces for velocities and pressure
    - stabilized formulations, that introduce additional second order terms into the weak formulations of the problem

Finite element formulation for the Navier-Stokes equations

Space discretization - with SUPG and pressure stabilization terms

$$\int_\Omega \rho_0 \frac{\partial v_j}{\partial t} w_j d\Omega + \int_\Omega \left(\rho_0 v_{j,k} v_k w_j + \mu v_{j,k} w_{j,k} - p w_{j,j}\right) d\Omega$$

$$+ \sum_e \int_{\Omega_e} \left\{R_j(\boldsymbol{v}, p)\omega \bar{R}_j(\boldsymbol{w}, r) + v_{j,j}\delta w_{k,k}\right\} d\Omega$$

$$+ \int_\Omega v_{j,j} r d\Omega = \int_\Omega f_j w_j d\Omega - \int_{\partial\Omega} g_j w_j dS$$

$$R_j(\boldsymbol{v}, p) = \rho_0 \frac{\partial v_j}{\partial t} + \rho_0 v_{j,k} v_k - \mu v_{j,kk} + p_{,j} - f_j$$

$$\bar{R}_j(\boldsymbol{w}, r) = \rho_0 w_{j,k} v_k - \mu w_{j,kk} + r_{,j}$$

- $\boldsymbol{w}$ and r – test functions
- ω and δ – stabilization parameters
- equal order linear approximation for velocities and pressure

Finite element formulation for the Navier-Stokes equations

System of ODEs for the vector of degrees of freedom $\boldsymbol{u}$

Rearrangement of DOFs: $\boldsymbol{u}_v$ - velocity DOFs, $\boldsymbol{u}_p$ - pressure DOFs

$$\begin{pmatrix} \boldsymbol{M}_{vw}(v)\cdot\dot{\boldsymbol{u}}_v \\ 0 \end{pmatrix} + \begin{pmatrix} \boldsymbol{A}_{vw}(v) & \boldsymbol{A}_{pw}(v) \\ \boldsymbol{A}_{vr}(v) & \boldsymbol{A}_{pr} \end{pmatrix} \cdot \begin{pmatrix} \boldsymbol{u}_v \\ \boldsymbol{u}_p \end{pmatrix} = \begin{pmatrix} \boldsymbol{b}_w \\ \boldsymbol{b}_r \end{pmatrix}$$

α scheme for time integration

$$\begin{aligned} &\frac{\boldsymbol{M}_{vw}(v^{n+1})\cdot\boldsymbol{u}_v^{n+1} - \boldsymbol{M}_{vw}(v^n)\cdot\boldsymbol{u}_v^n}{\Delta t} + \\ &\quad +\alpha\cdot\left(\boldsymbol{A}_{vw}(v^{n+1})\boldsymbol{u}_v^{n+1} + \boldsymbol{A}_{pw}(v^{n+1})\boldsymbol{u}_p^{n+1}\right) \\ = &\quad (\alpha-1)\cdot\left(\boldsymbol{A}_{vw}(v^n)\boldsymbol{u}_v^n + \boldsymbol{A}_{pw}(v^n)\boldsymbol{u}_p^n\right) + \alpha\boldsymbol{b}_w^{n+1} + (1-\alpha)\boldsymbol{b}_w^n \end{aligned}$$

$$\boldsymbol{A}_{vr}(v^{n+1})\boldsymbol{u}_v^{n+1} + \boldsymbol{A}_{pr}\boldsymbol{u}_p^{n+1} = \boldsymbol{b}_r^{n+1}$$

Finite element formulation for the Navier-Stokes equations

- Algorithms for time integration and non-linear problem solution can be applied to the weak finite element statement
- An example finite element formulation for a single fixed-point (Picard's) nonlinear iteration within the implicit Euler time integration for the stabilized finite element method approximating the Navier-Stokes equations reads:

Final formulation

$$\int_\Omega \rho \frac{(v_{k+1}^{n+1})_j}{\Delta t} w_j d\Omega + \int_\Omega \rho (v_{k+1}^{n+1})_{j,l} (v_k^{n+1})_l w_j d\Omega + \int_\Omega \mu (v_{k+1}^{n+1})_{j,l} w_{j,l} d\Omega$$
$$- \int_\Omega p_{k+1}^{n+1} w_{j,j} d\Omega - \int_\Omega (v_{k+1}^{n+1})_{j,j} r d\Omega + \sum_e \int_{\Omega_e} R_j(\boldsymbol{u}, p) \omega \bar{R}_j(\boldsymbol{w}, r) d\Omega$$
$$+ \sum_e \int_{\Omega_e} (u_{k+1}^{n+1})_{j,l} \delta w_{j,l} d\Omega = \int_\Omega \rho \frac{(v^n)_j}{\Delta t} w_j d\Omega + \int_\Omega f_j w_j d\Omega - \int_{\Gamma_N} g_j w_j d\Gamma$$

Finite element formulation for the Navier-Stokes equations

An example: the simulation of the von Karman vortex street by the adaptive finite element method