Finite Elements for Computational Fluid Dynamics

Dr. Albert Ruprecht

Former: Institute of Fluid Mechanics
and Hydraulic Machinery
University of Stuttgart

Basic equations

Conservation or transport equations

- Conservation equations for:
  - Mass
  - Momentum
  - Energy
- Transport equations for:
  - Concentrations
  - Turbulence
  - etc.

System of non-linear, coupled, partial differential equations of convection-diffusion type

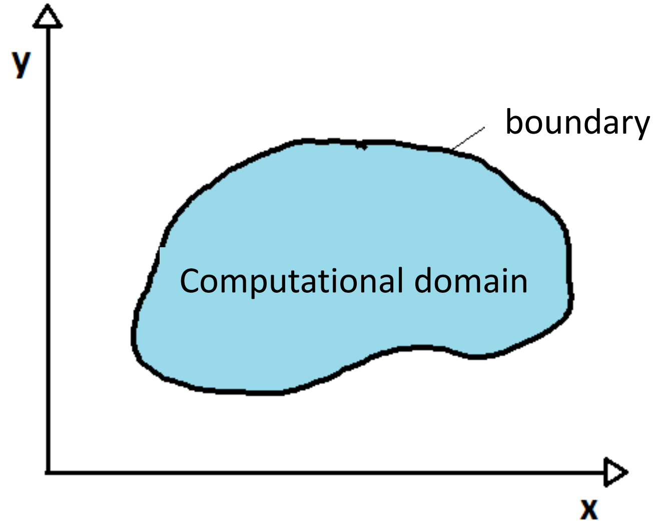
$$\frac{\partial \varphi}{\partial t} + U_j^* \frac{\partial \varphi}{\partial x_j} - \Gamma \frac{\partial^2 \varphi}{\partial x_j^2} - f = 0$$

1 2 3 4

1. Time derivative
2. Convection term
3. Diffusion term
4. Source term

Basic equations

Computational domain and boundary conditions



Boundary conditions:
- Dirichlet type (fixed values)
- Neumann type (gradient type b.c.)

For unsteady problems also **Initial Conditions** are required

Solution methods

Usually used solution methods:

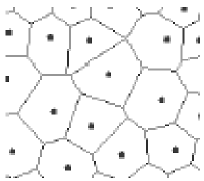
- **Finite Difference Methods (FDM)**
- **Finite Volume Methods (FVM)**
- **Finite Element Methods (FEM)**

Other Methods

- Spectral methods
- Lattice Boltzmann Methods
- Boundary Element Methods
- Etc.

Solution methods

FVM



$$\frac{\partial U}{\partial x}$$

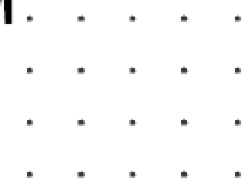
Integral form

$$\int_{CV} \frac{\partial U}{\partial x} dV$$

Divergence theorem

$$\int_{A} n \cdot U \, dA$$

FDM



$$\frac{\partial U}{\partial x}$$

Difference Equations

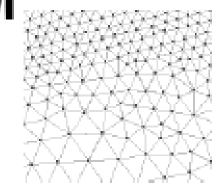
$$U_{i+1} = U_i + \Delta x \cdot \frac{\partial U}{\partial x}$$

$$U_{i-1} = U_i - \Delta x \cdot \frac{\partial U}{\partial x}$$

Taylor series

$$\frac{U_{i+1} - U_{i-1}}{2 \Delta x}$$

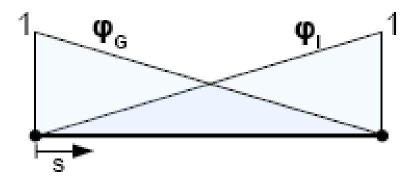
FEM



$$\frac{\partial U}{\partial x}$$

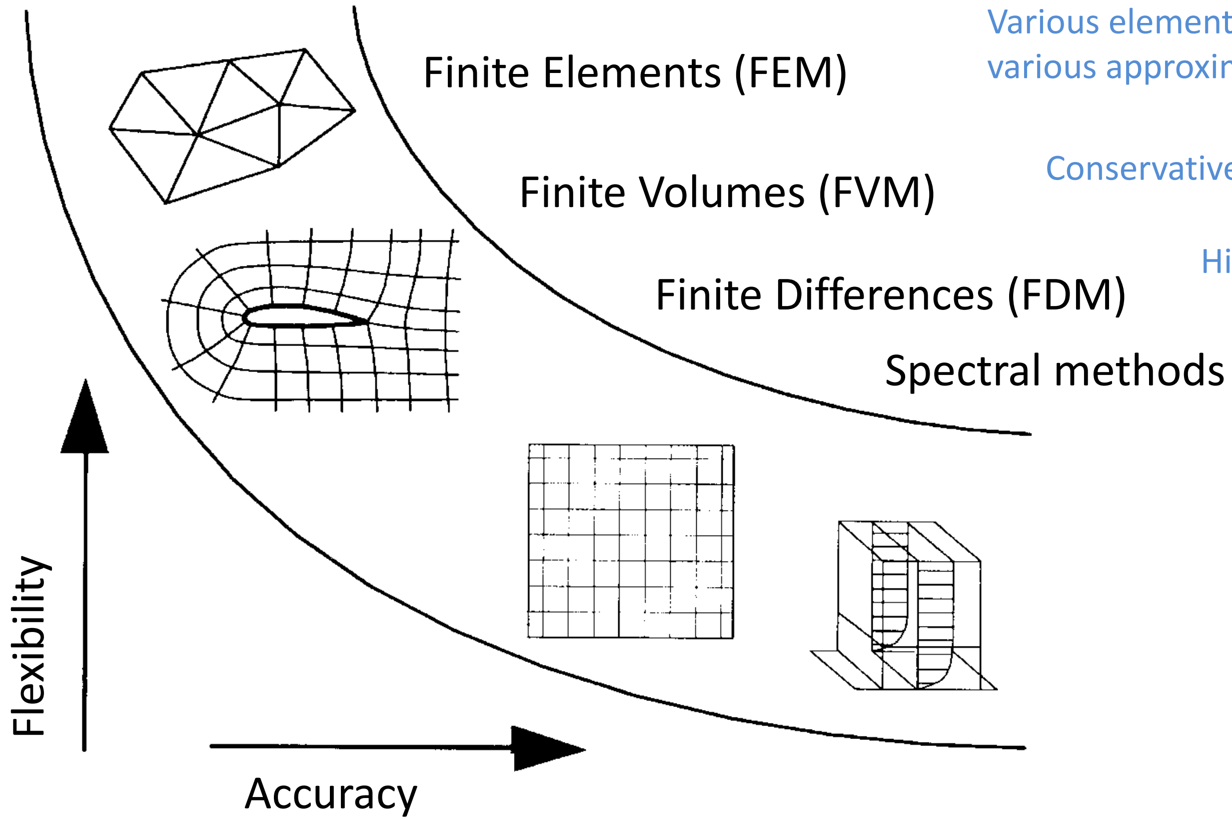
Using basis function

$$U = U_G \cdot \phi_G + U_I \cdot \phi_I$$



$$\frac{\partial U}{\partial x} = U_G \cdot \frac{\partial \phi_G}{\partial x} + U_I \cdot \frac{\partial \phi_I}{\partial x}$$

Solution methods



Various element types, various approximations

Conservative formulation

High order approximations

General procedure

1. Definition of describing equations
2. Definition of Computational domain, Boundary conditions Initial conditions
3. Dividing of the domain into Finite Elements
4. Definition of the local approximation of the solution quantities
5. Discrete form of the describing equations
6. Assembling of the relevant matrices
7. Solution of the system of linear equations
8. Solution (Linear equations)

Update of coefficients

Convergence check

Solution (non-linear equations)

1 Definition of describing equations

Example: Navier-Stokes equations, **steady-state**, incompressible

Mass conservation:

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

Momentum conservation:
(x-direction)

$$u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = -\frac{1}{\rho}\frac{\partial p}{\partial x} + \nu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right)$$

Momentum conservation:
(y-direction)

$$u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} = -\frac{1}{\rho}\frac{\partial p}{\partial y} + \nu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right)$$

- Coupled
- Non-linear
- 2. order for velocities
- 1. order for pressure

1 Definition of describing equations

Navier-Stokes equations, 3D, unsteady, incompressible

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0$$

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + w\frac{\partial u}{\partial z} = -\frac{1}{\rho}\frac{\partial p}{\partial x} + \nu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}\right)$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + w\frac{\partial v}{\partial z} = -\frac{1}{\rho}\frac{\partial p}{\partial y} + \nu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 v}{\partial z^2}\right)$$

$$\frac{\partial w}{\partial t} + u\frac{\partial w}{\partial x} + v\frac{\partial w}{\partial y} + w\frac{\partial w}{\partial z} = -\frac{1}{\rho}\frac{\partial p}{\partial z} + \nu\left(\frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} + \frac{\partial^2 w}{\partial z^2}\right)$$

In Index form

Mass conservation $$\frac{\partial u_i}{\partial x_i} = 0$$

Momentum conservation $$\frac{\partial u_i}{\partial t} + u_j\frac{\partial u_i}{\partial x_j} = -\frac{1}{\rho}\frac{\partial p}{\partial x_i} + \nu\frac{\partial^2 u_i}{\partial x_j^2}$$

Summation convention

1 Definition of describing equations

Example: Turbulent flow

Mass conservation $$\frac{\partial u_i}{\partial x_i} = 0$$

Momentum conservation $$\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} = -\frac{1}{\rho}\frac{\partial p}{\partial x_i} + (\nu + \nu_t)\frac{\partial^2 u_i}{\partial {x_j}^2}$$

Turbulent viscosity $$\nu_t = c_\mu \frac{k^2}{\varepsilon}$$

Turbulent kinetic energy $$\frac{\partial k}{\partial t} + u_i \frac{\partial k}{\partial x_i} = \frac{\partial}{\partial x_i}\left(\frac{\nu_t}{\sigma_\varepsilon}\frac{\partial k}{\partial x_i}\right) + \nu_t\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)\frac{\partial u_i}{\partial x_j} - \varepsilon$$

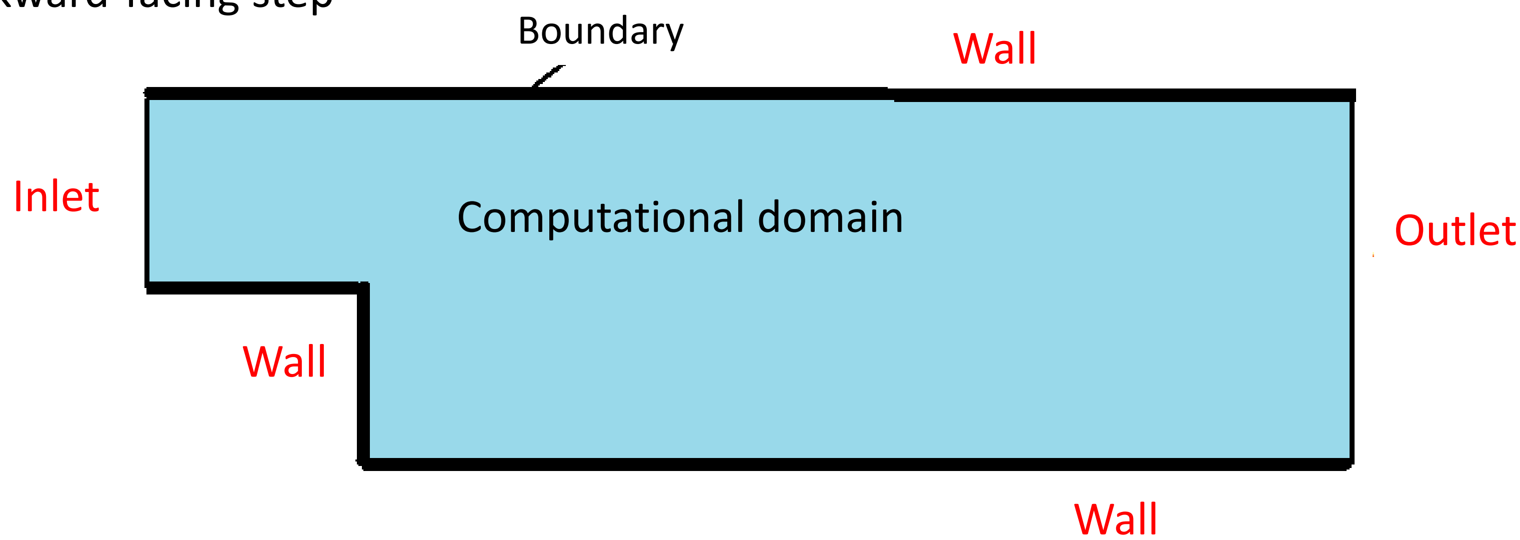
Dissipation rate $$\frac{\partial \varepsilon}{\partial t} + u_i \frac{\partial \varepsilon}{\partial x_i} = \frac{\partial}{\partial x_i}\left(\frac{\nu_t}{\sigma_\varepsilon}\frac{\partial \varepsilon}{\partial x_i}\right) + c_{1\varepsilon}\frac{\varepsilon}{k}\nu_t\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)\frac{\partial u_i}{\partial x_j} - c_{2\varepsilon}\frac{\varepsilon^2}{k}$$

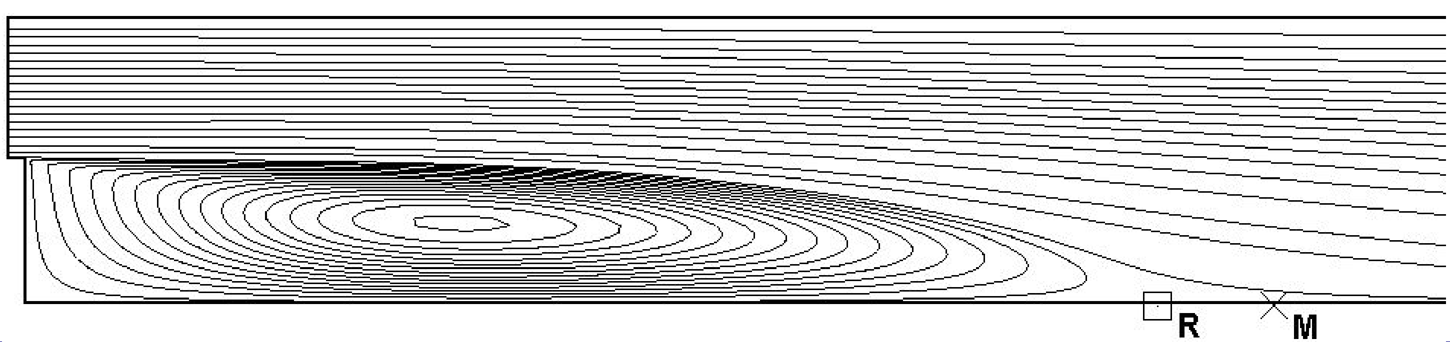
Model constants $c_\mu = 0.09; \quad \sigma_k = 1.0; \quad \sigma_\varepsilon = 1.3; \quad c_{1\varepsilon} = 1.44; \quad c_{2\varepsilon} = 1.92$

Advection – diffusion equation $$\frac{\partial \varphi}{\partial t} + u_j \frac{\partial \varphi}{\partial x_j} - \Gamma \frac{\partial^2 \varphi}{\partial x_j^2} - f = 0$$

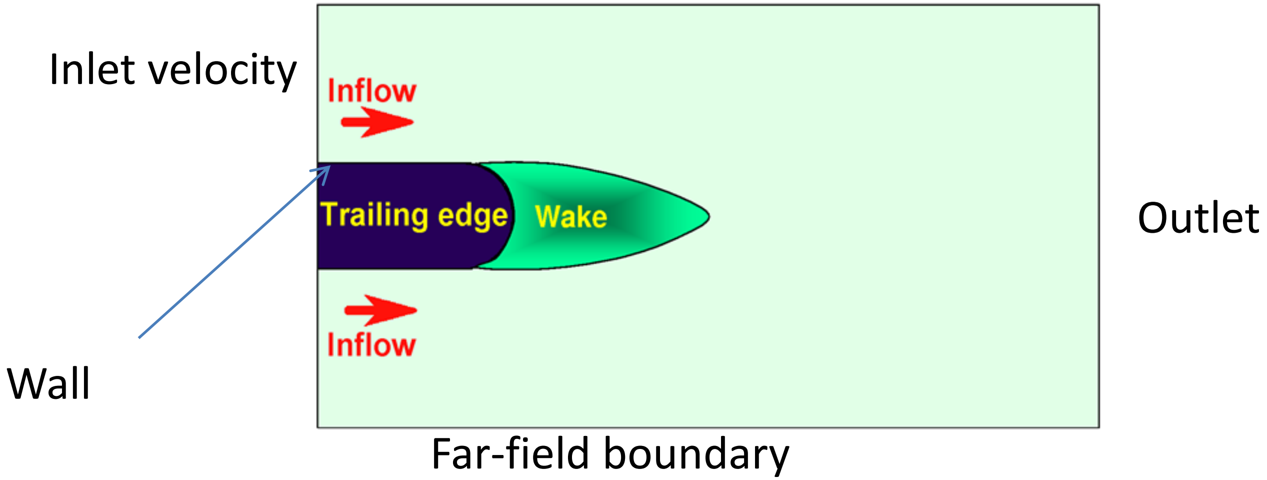
2 Definition of computational domain and boundary conditions

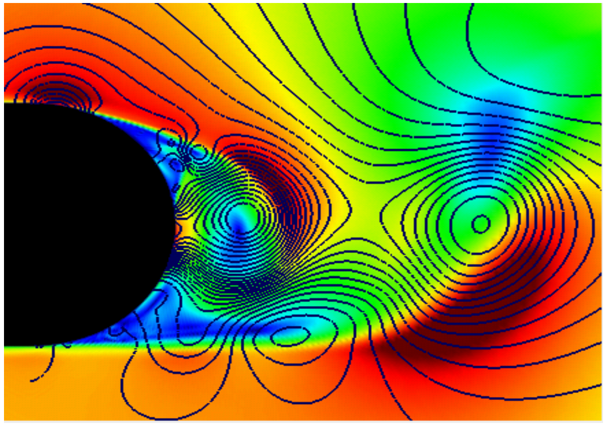
Example backward-facing step

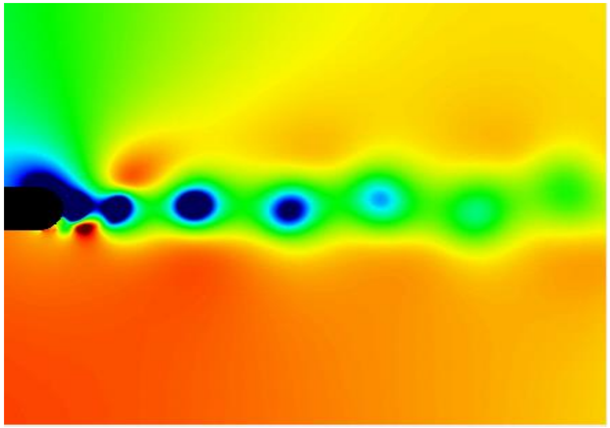




2 Definition of computational domain and boundary conditions







General procedure

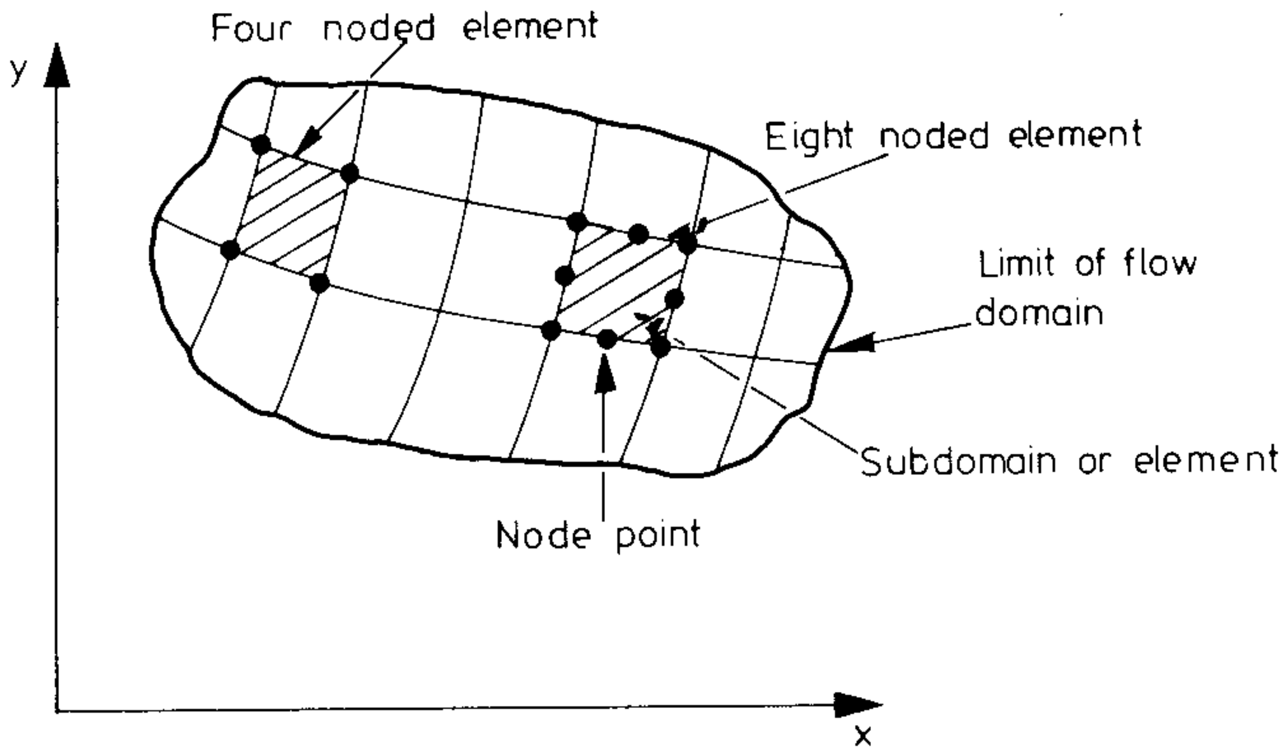
1. Definition of describing equations
2. Definition of Computational domain, Boundary conditions Initial conditions
3. Dividing of the domain into Finite Elements
4. Definition of the local approximation of the solution quantities
5. Discrete form of the describing equations
6. Assembling of the relevant matrices
7. Solution of the system of linear equations
8. Solution (Linear equations)

Update of coefficients → 6

7 → Convergence check

Solution (non-linear equations)

3 Dividing of the domain into Finite Elements

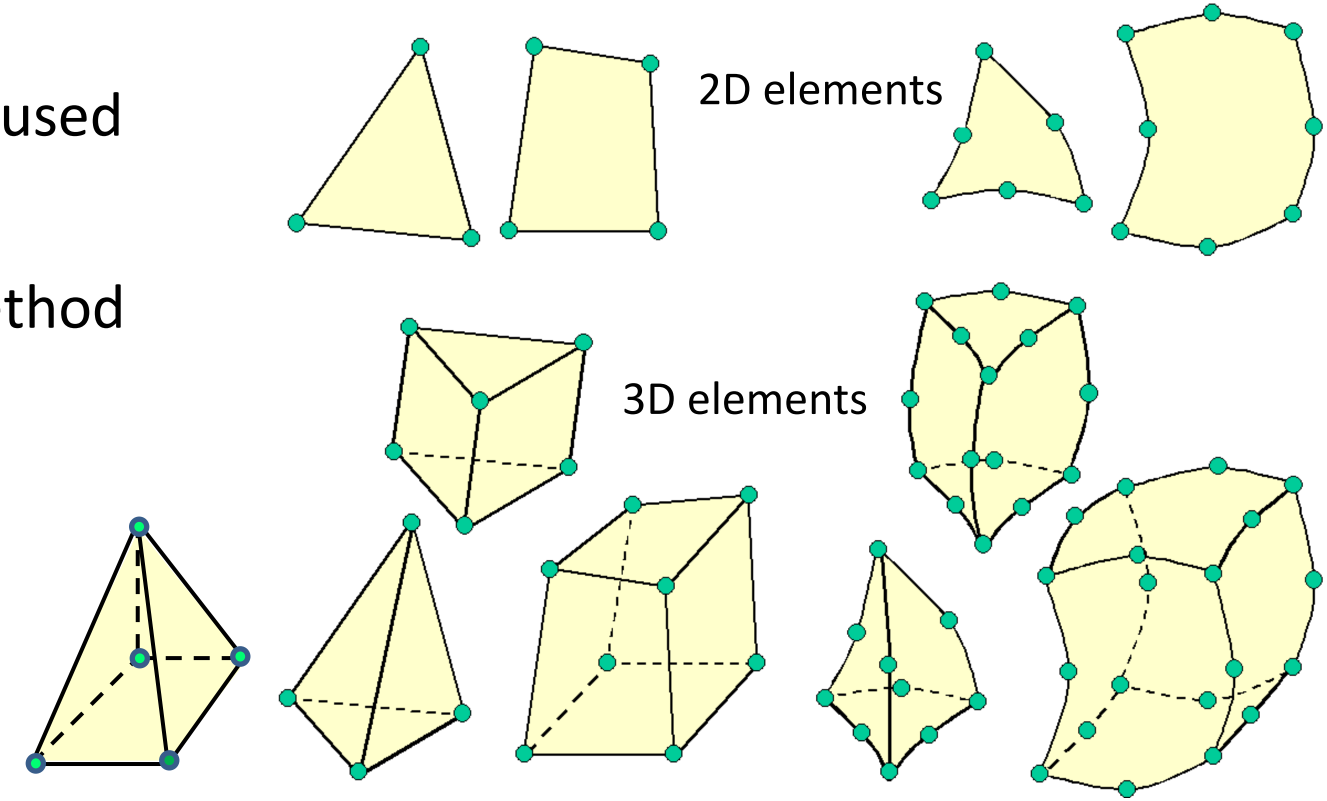


Various types of elements

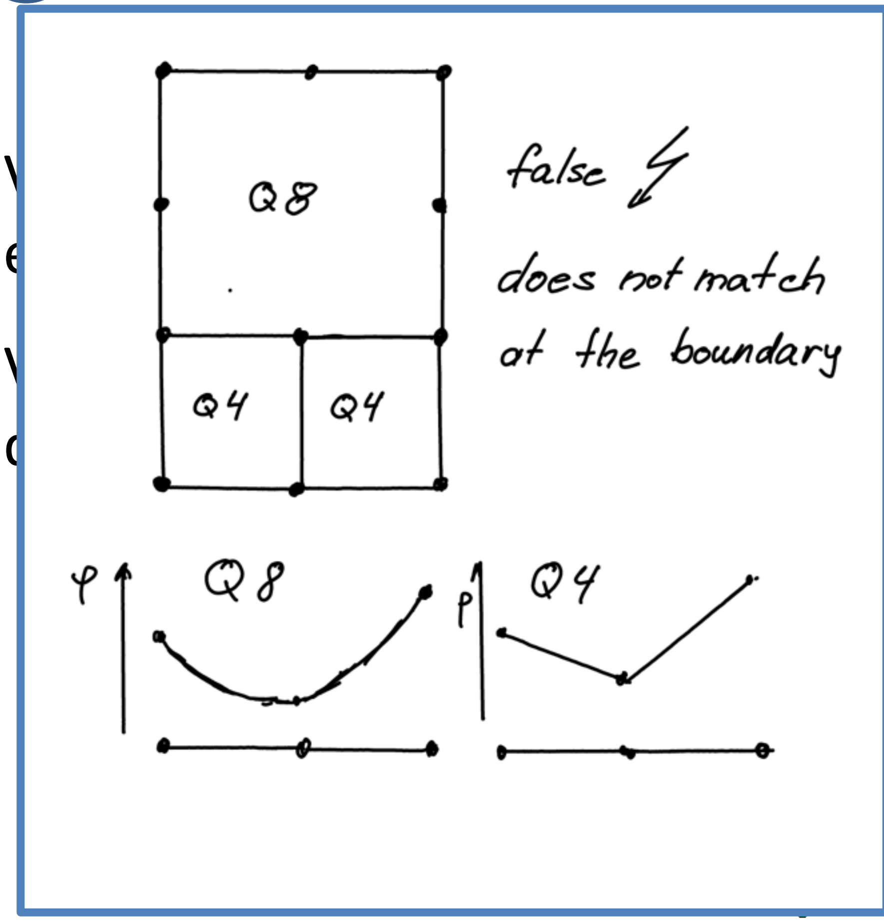
3 Dividing of the domain into Finite Elements

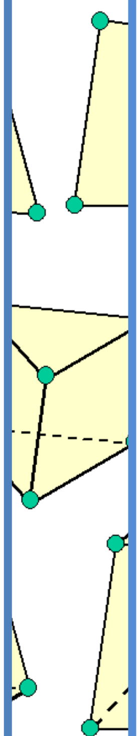
Various types of elements can be used

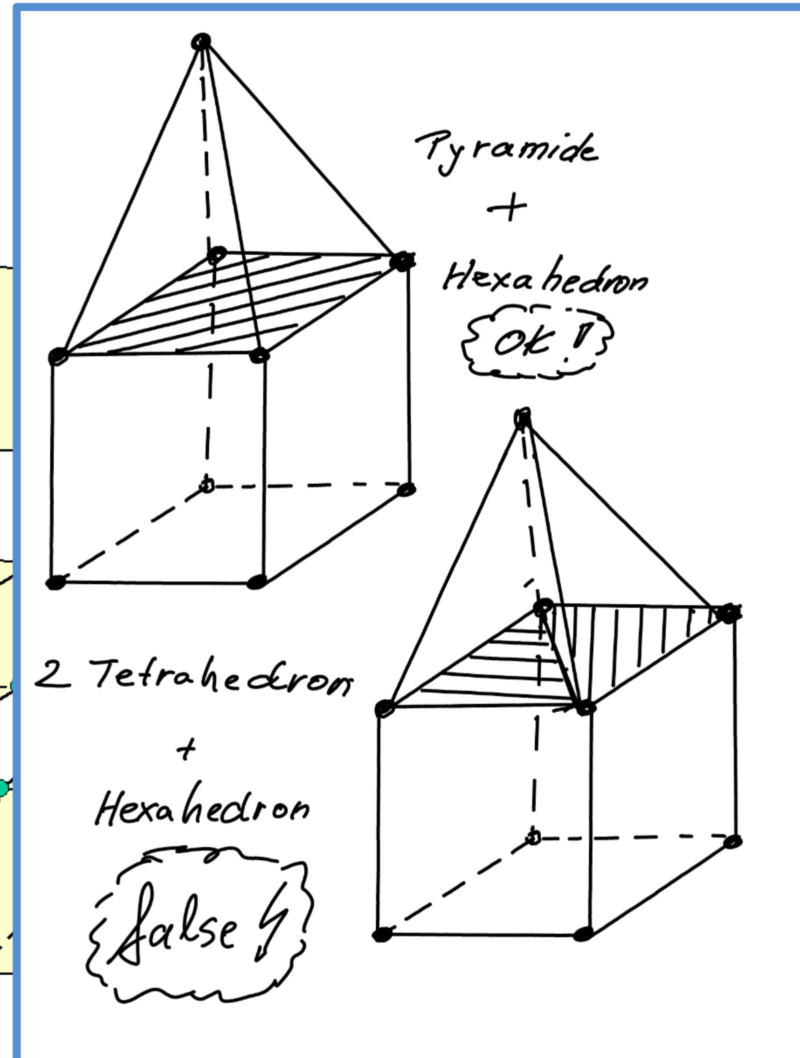
Very flexible discretization method



3 Dividing of the domain into Finite Elements

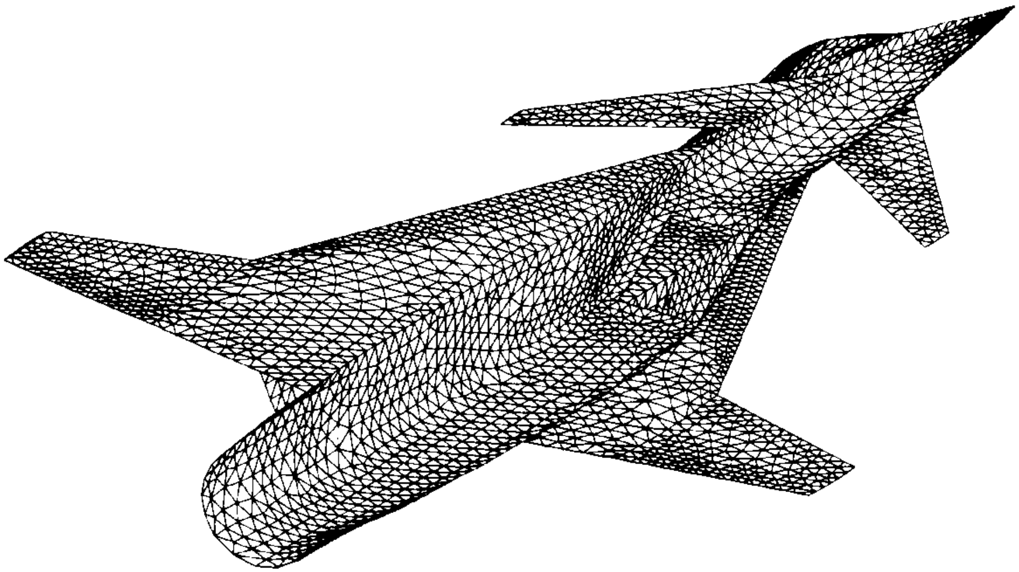




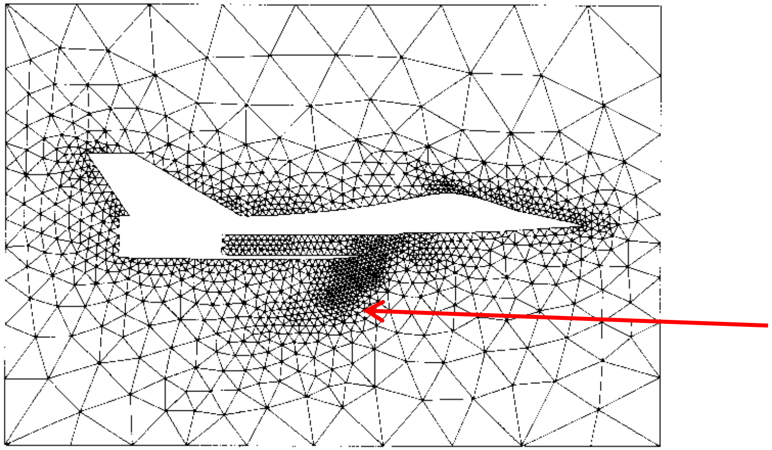


Elements

Example: Flow around a jet



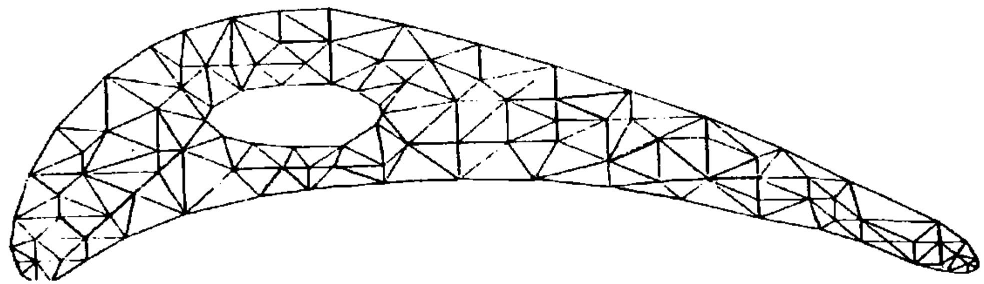
Zienkiewicz & Taylor (2000)

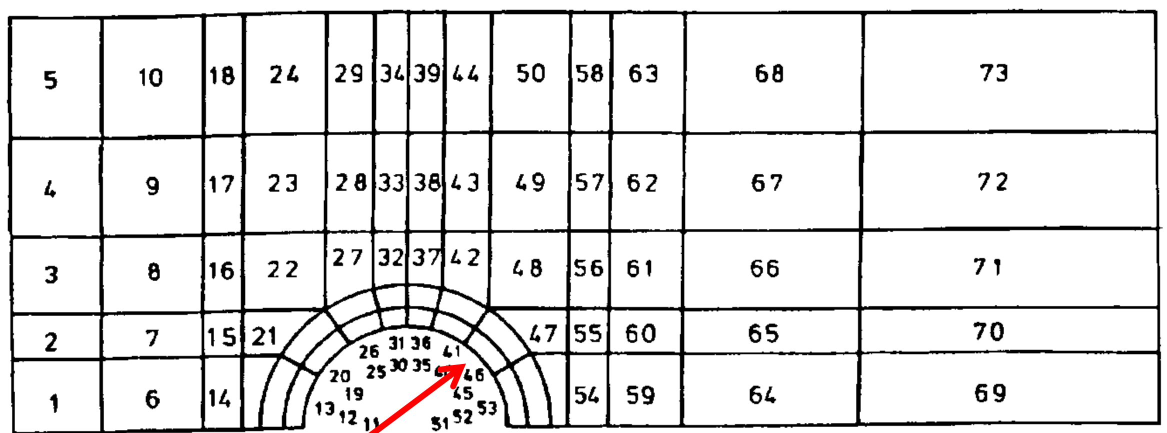


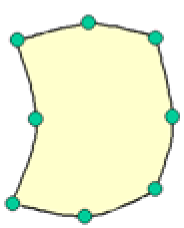
Grid concentration in regions with steep gradients

Elements

Simple grids







Curvilinear elements

Source: Taylor & Hughes

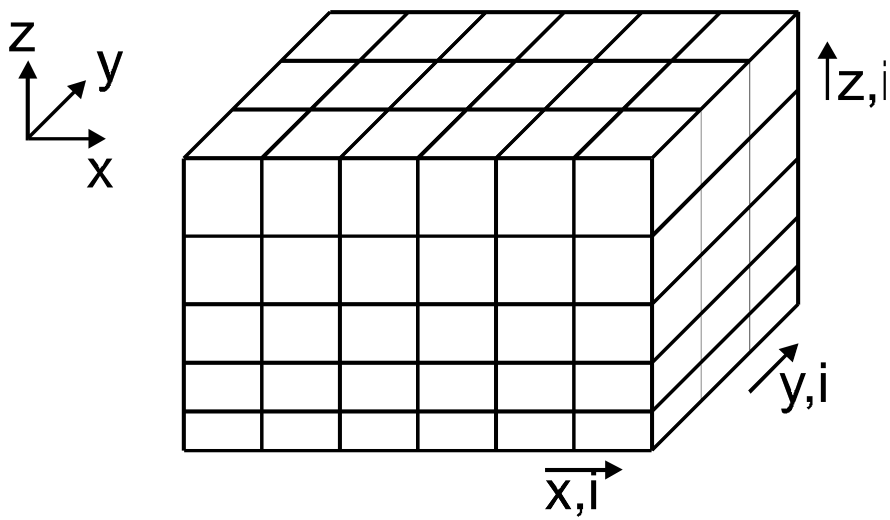
Insertion: Characteristic grid structures

Cartesian Grid

$P_{ijk}(x_i,y_j,z_k)$

Is defined by 1D arrays

real x(nx), y(ny), z(nz), u(nx,ny,nz)



All grid lines are straight lines parallel to the coordinate axis
All cells are rectangular
All angles are right angles
Partial differentials can be build in 1D

For FDM

<u>Example program</u>

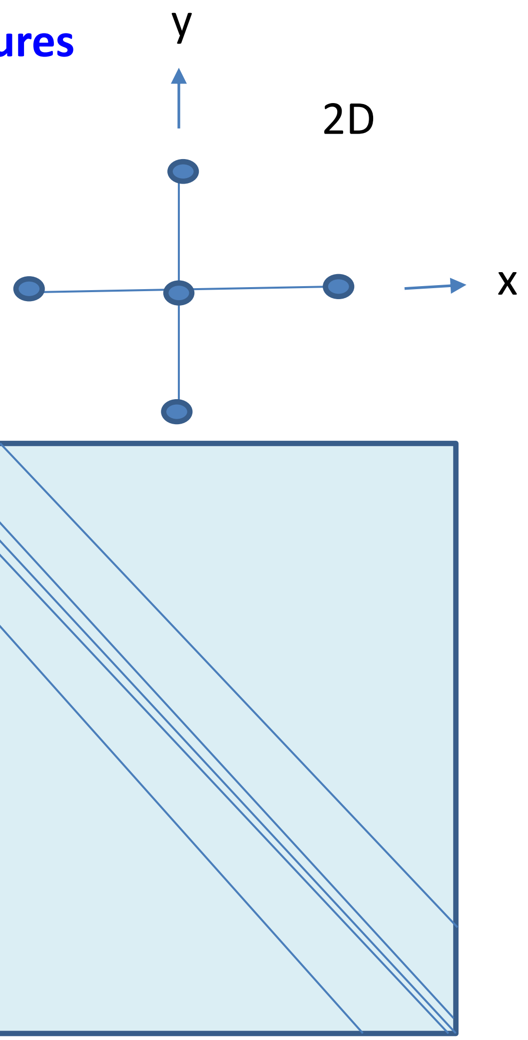
```
delx = Lx/(nx-1)
dely = Ly/(ny-1)
delz = Lz/(nz-1)
   do i = 1,nx
   do j = 1,nz
   do k = 1,nz
      x (i,j,k) = delx*(i-1)
      y (i,j,k) = delx*(j-1)
      z (i,j,k) = delz*(k-1)
      u (i,j,k) = 0.0
   end do
   end do
   end do
```

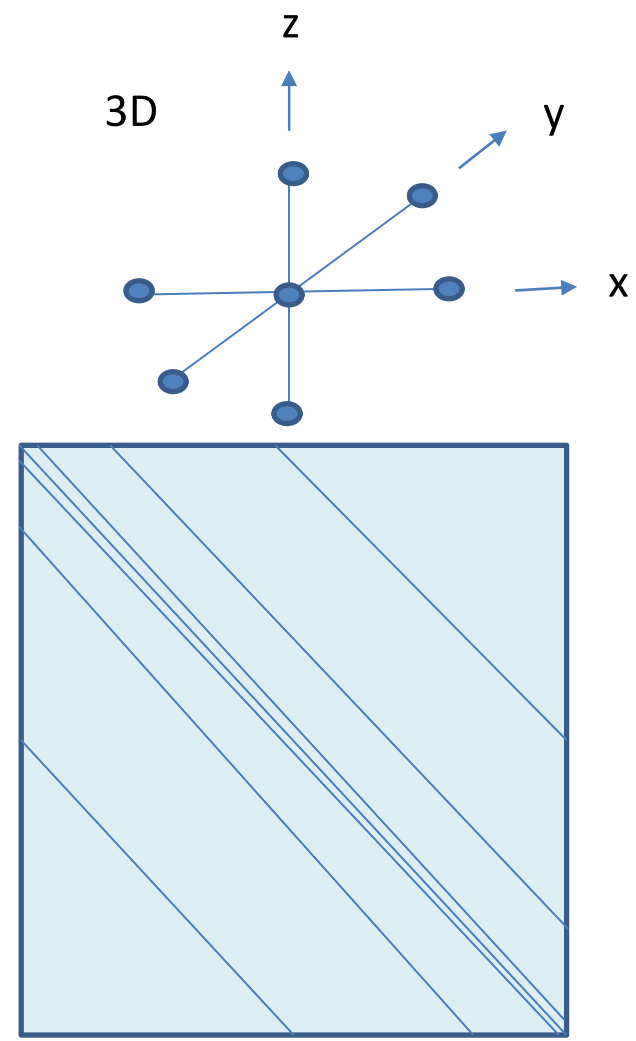
Characteristic grid structures

Discretization
results in band matrices

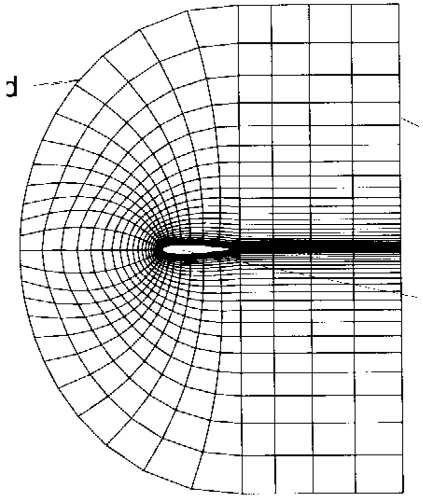
2D: 5 bands
3D: 7 bands

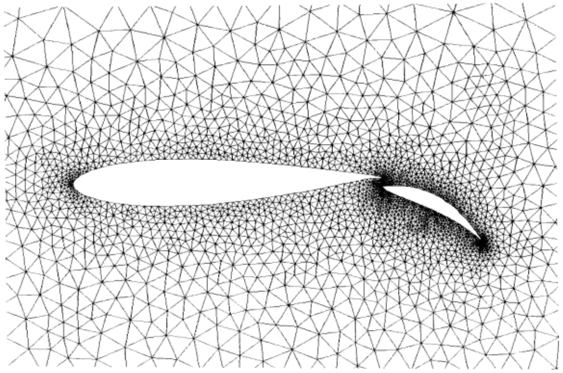
Number of columns depend on the number of neighbours





Characteristic grid structures





Structured grids

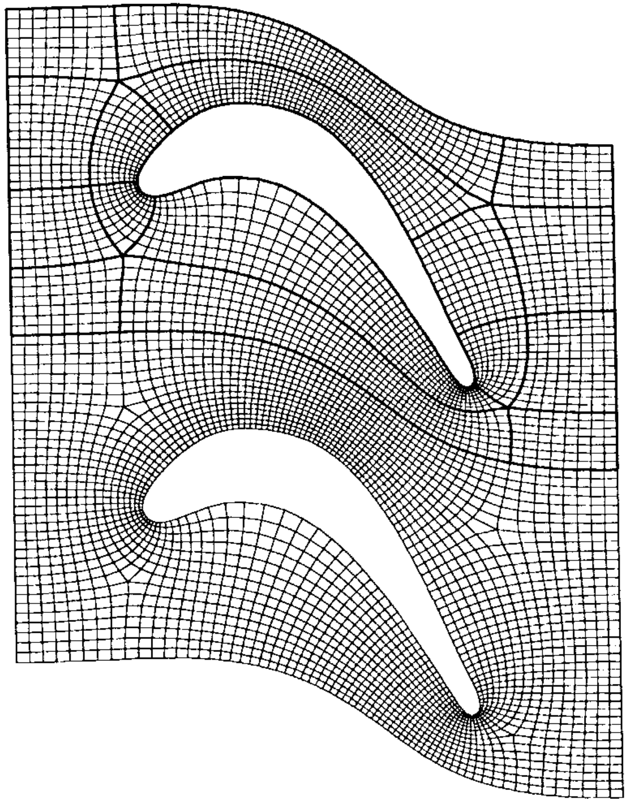
2D	**3D**
Two regimes of grid lines	**three**
Cells with four corners	**Hexahedron**
Each grid point has 4 neighbors	**six**

Body-fitted coordinates by using curvilinear grid lines

Unstructured grids

No regimes of grid lines	
Cells can have different shapes	
e. g. triangles …	**Prisms...**
No of neighbors are different	

Bock-structured grids



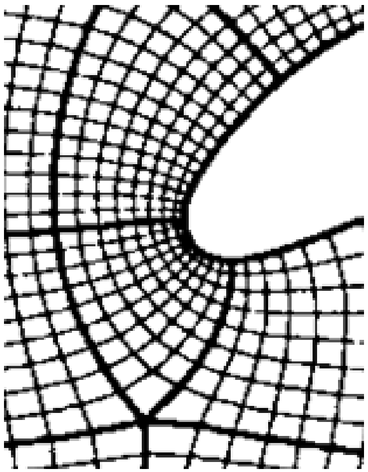
Turbine blades
(Geometry is periodic)

Each block consists of a structured grid

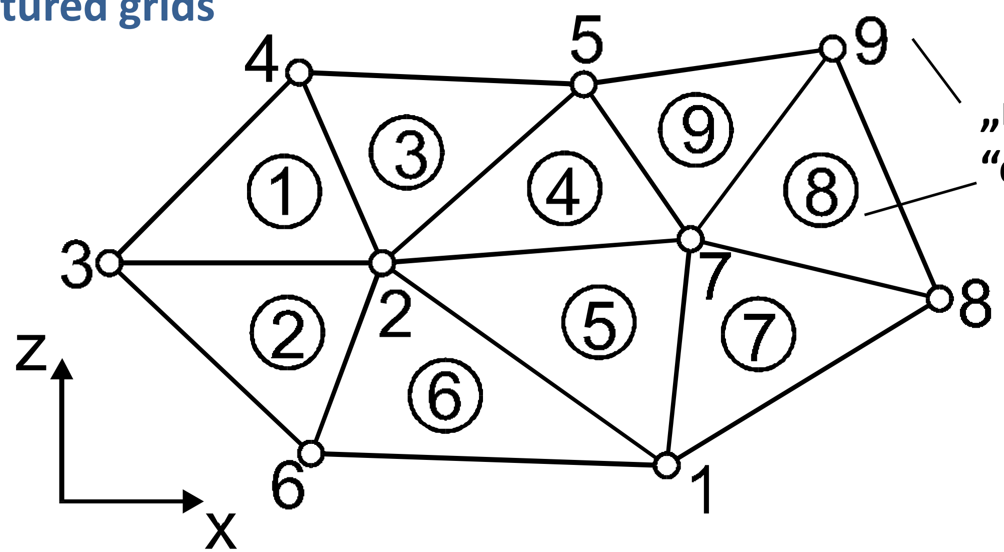
On the block boundaries the grid points are identical between neighbor blocks

Grid lines should be smooth on block boundaries

Advantages:
Suitable for complex geometries
Easy to parallelize



Unstructured grids

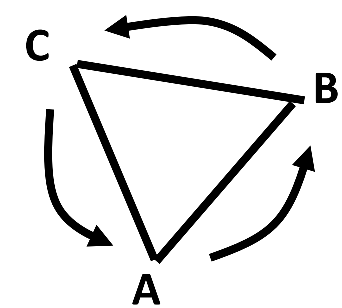


Connectivity table

„local" nodes

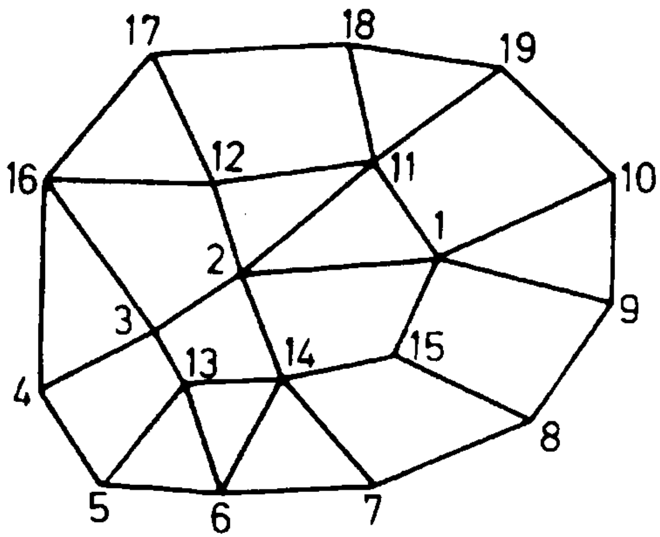
	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*
A	3	6	2	2	1	1	1	7	7
B	2	2	5	7	7	2	8	8	9
C	4	3	4	5	2	6	7	9	5

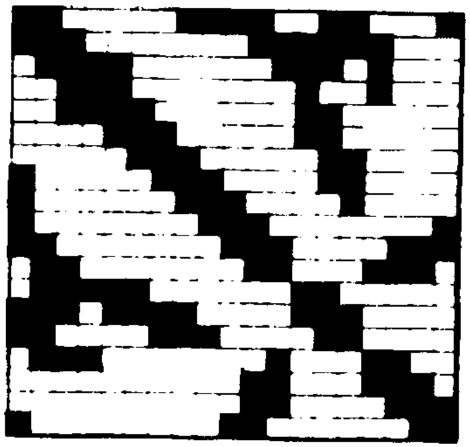
„global nodes



Unstructured grids

Unlike for structured meshes, where band matrices occur, for unstructured meshes, unstructured, sparse matrices arise instead.

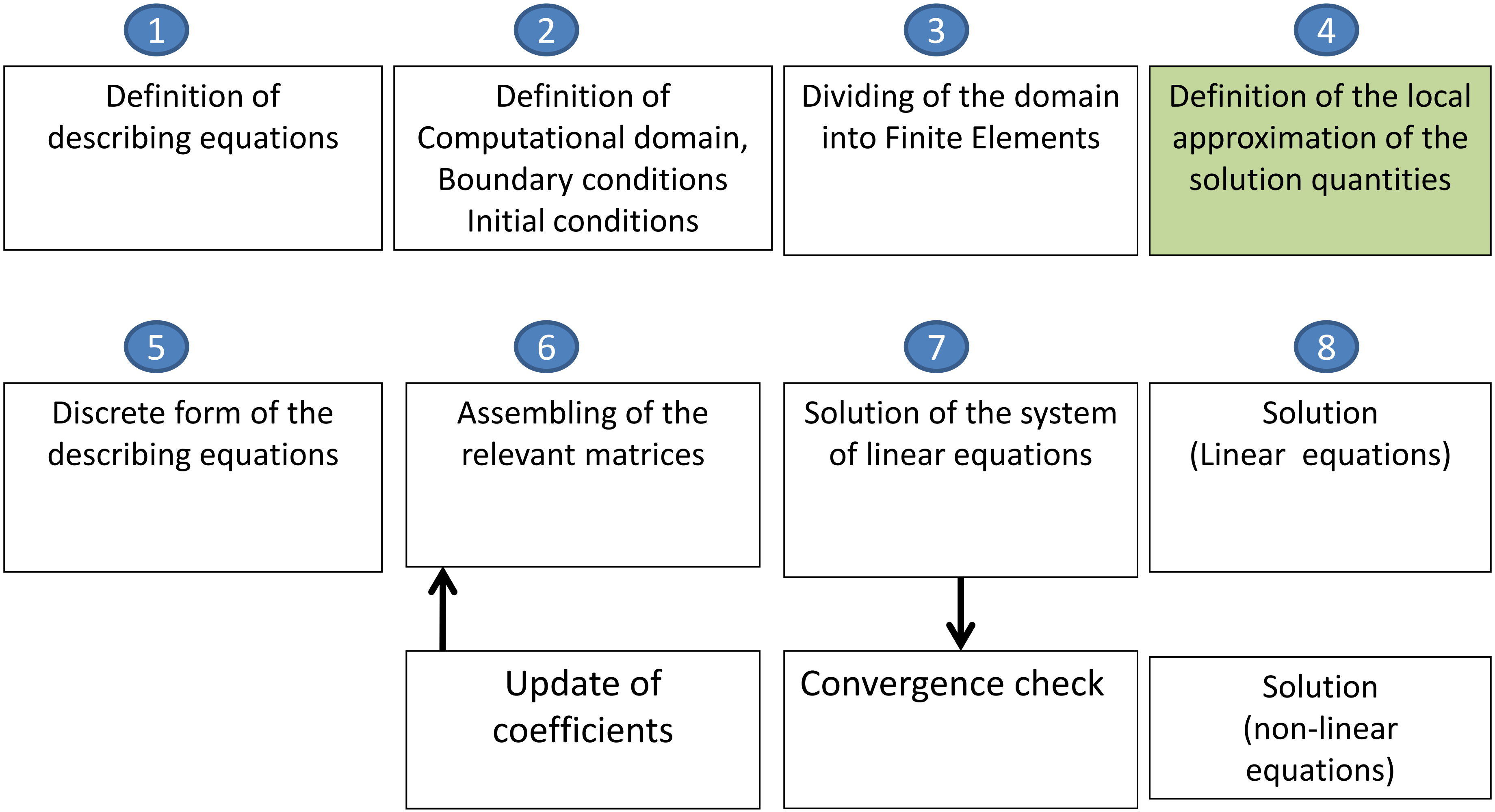




Structure of the matrices is symmetric

Matrices are not stored completely, instead only the relevant parts are stored, for this special storage technologies exists.

General procedure



Definition of approximation

In each element an approximation for the required quantity ϕ is defined. This means it is defined, how the quantity behaves within the element.

Examples:

2D linear:

$$\hat{\phi}(x,y) = \alpha_1 + \alpha_2 x + \alpha_3 y$$

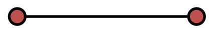
2D quadratic:

$$\hat{\phi}(x,y) = \alpha_1 + \alpha_2 x + \alpha_3 y + \alpha_4 x^2 + \alpha_5 xy + \alpha_6 y^2$$

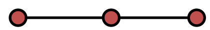
The coefficients are the solution

Definition of approximation

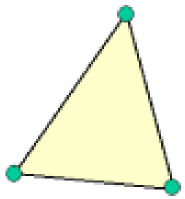
Examples:



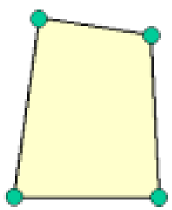
$$\hat{\Phi} = \alpha_0 + \alpha_1 x$$



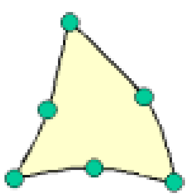
$$\hat{\Phi} = \alpha_0 + \alpha_1 x + \alpha_2 x^2$$



$$\hat{\Phi} = \alpha_0 + \alpha_1 x + \alpha_2 y$$



$$\hat{\Phi} = \alpha_0 + \alpha_1 x + \alpha_2 y + \alpha_3 xy$$



$$\hat{\Phi} = \alpha_0 + \alpha_1 x + \alpha_2 y + \alpha_3 x^2 + \alpha_4 y^2 + \alpha_5 xy$$

Definition of approximation

The approximations must fulfill certain continuity conditions:

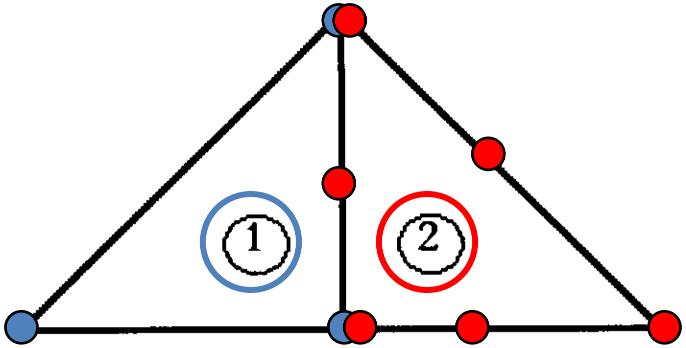
1.) Conforming elements

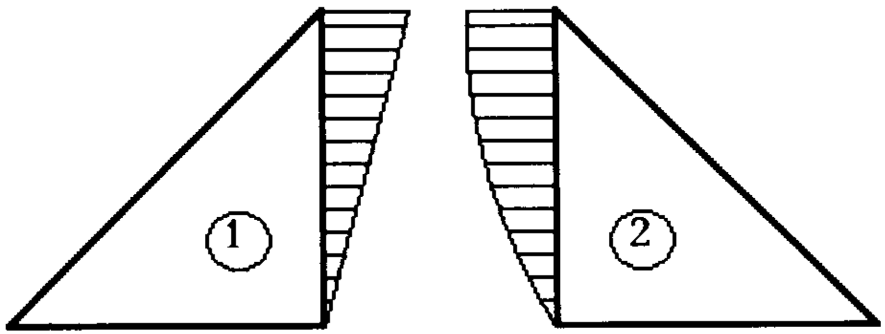
The approximation on neighbor element boundaries must be identical

These are called ***conforming elements.***

Only conforming elements are considered here.

EXAMPLE: Non-conforming velocity approximation





Because the approximations are different in each element, this acts as a source or sink between the elements

Definition of approximation

2.) C_{n-1} continuity requirement

If the integrant contains derivatives of n-th order, the approximations have to be continuous up to derivatives of (n-1)-th order.

For the Navier-Stokes equations this means:

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = -\frac{1}{\rho}\frac{\partial p}{\partial x} + \nu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right)$$

Velocities are 2nd order, pressure 1st order

$\Rightarrow$ Velocity approximation must be continuous and differentiable

$\Rightarrow$ Pressure approximation must be continuous

(The continuity requirements can be reduced by applying Galerkin method by means of Green-Gauss theorem, see later)

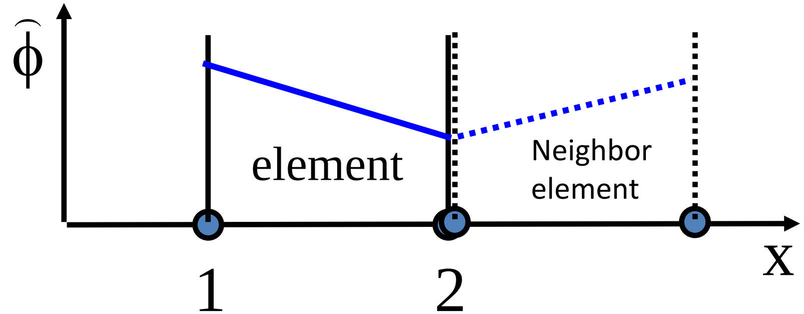
Definition of approximation

To fulfill the C_{n-1} continuity requirements the polynomial approximations are not suitable. Between the coefficients of neighboring elements constraints has to be satisfied.

It is much easier to use node values instead of coefficients to prescribe the approximation

1-D Example:

Linear Approximation: $\hat{\Phi} = \alpha_0 + \alpha_1 x$



The continuity can be satisfied easier when expressing the approximation by node values

$$\hat{\phi} = \hat{\phi}^1 + \left(\hat{\phi}^2 - \hat{\phi}^1\right)\frac{(x - x_1)}{x_2 - x_1}$$

Definition of approximation

Generalized: $\widehat{\phi}(x_i) = \sum \widehat{\phi}^k N^k(x_i) \quad < k = 1, \ldots \text{ No nodes } >$

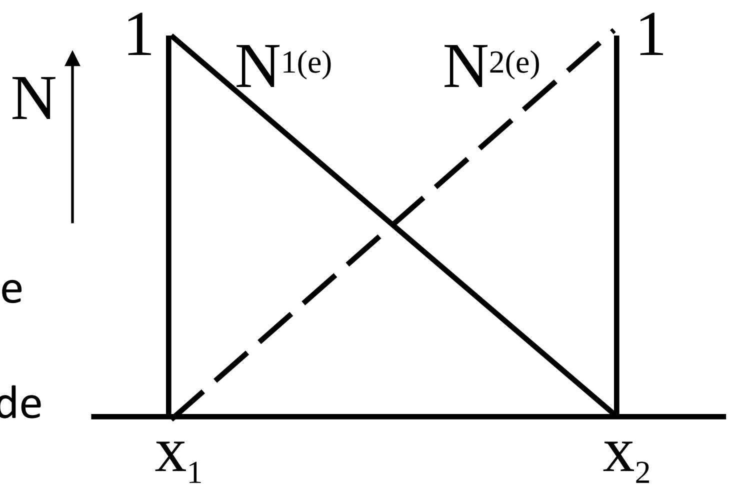
Linear Approximation: $\widehat{\phi}^{(e)} = \widehat{\phi}^{1(e)} N^{1(e)} + \phi^{2(e)} N^{2(e)}$

N are the shape or trial functions (der index (e) means that it is related to the local element numbering).

Introducing the dimensionless coordinate $\xi = \dfrac{x - x_1}{x_2 - x_1}$

the shape functions are

$N^{1(e)} = 1 - \xi$ and $N^{2(e)} = \xi$



The shape function characterize the influence of a node value to the solution.
The shape function must be 1 in the corresponding node and must be 0 in all other nodes

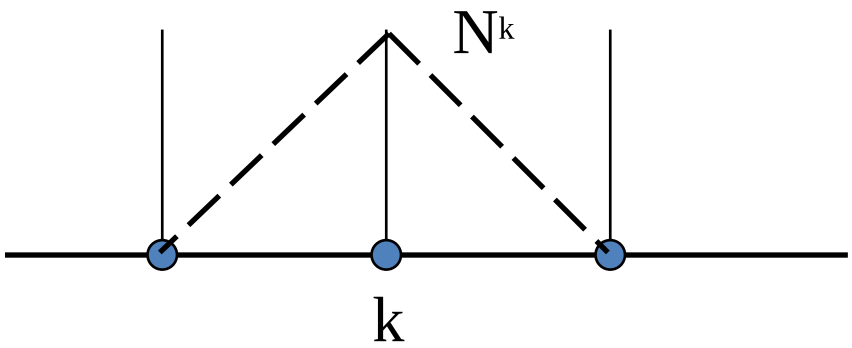
Definition of approximation

By a global numbering of all nodes the local approximations can be put together to a global approximation.

$$\hat{\phi}(x_i) = \sum \hat{\phi}^k N^k(x_i) \qquad < k = 1,\ldots,\ \text{Number of nodes}\ | >$$

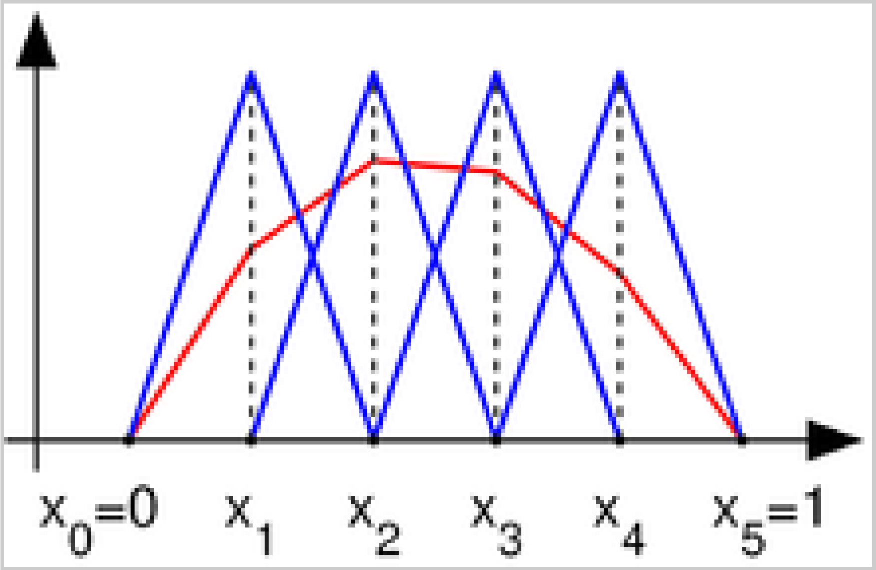
The global shape functions are composition of the local shape functions

Example: 1D linear



Definition of approximation

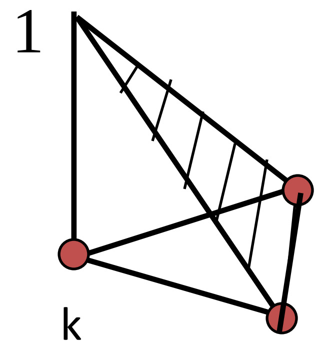
$$\hat{\phi}(x_i) = \sum \hat{\phi}^k N^k(x_i) \quad < k = 1, \dots \text{ no nodes } >$$



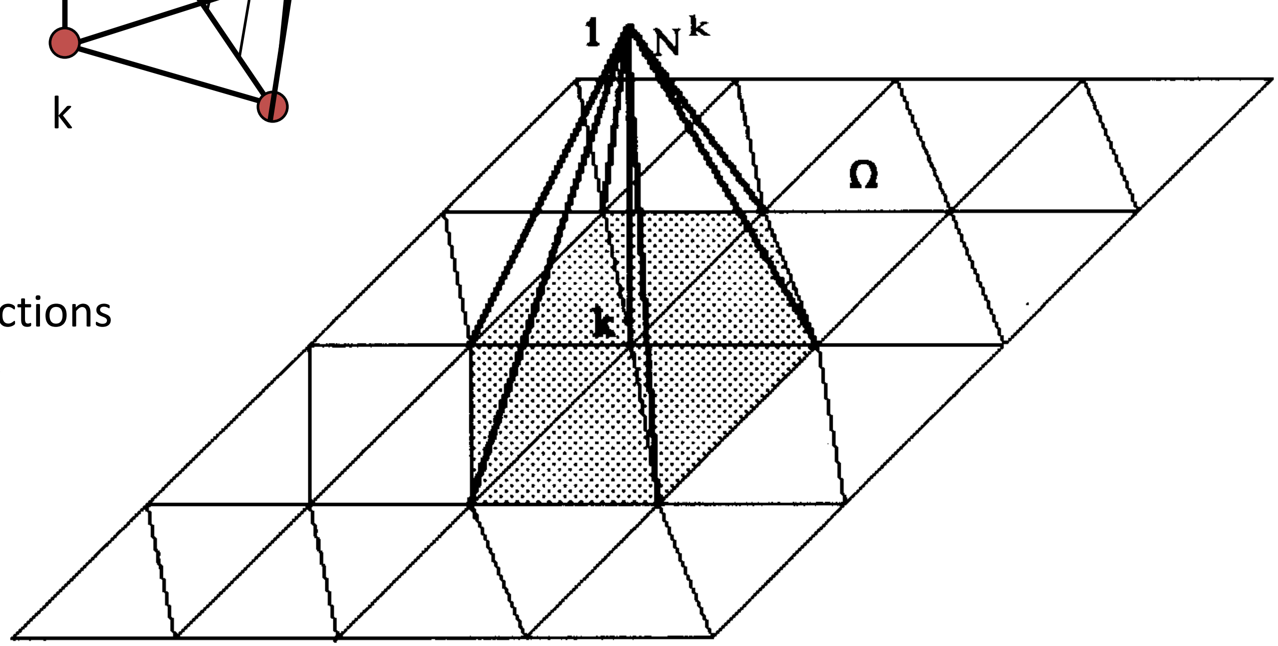
Basis functions v_k (blue) and a linear combination of them, which is piecewise linear (red).

Definition of approximation

For 2D linear triangles the following shape function for node k is obtained



and the global shape functions for node k are given here

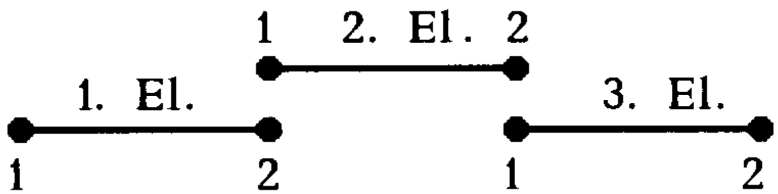


Definition of approximation

The introduction of global shape functions simplifies the formal description,

For programming only the element shape functions are usually used.

This is explained at a 1D example.



elements with local numbering

Elements with global numbering

Applied is a linear approximation

Definition of approximation

Approximation for element 1: $\hat{\phi}^{(e)} = \alpha_1 + \alpha_2 x$

Approximation for element 2: $\phi^{(e)} = \beta_1 + \beta_2 x$

As already mentioned with these approximations it is difficult to satisfy the continuity requirements. There are constrains between the coefficients. Therefore the approximations are expressed in node values

$$\hat{\phi}^{(e)} = \hat{\phi}^{1(e)} + \left(\phi^{2(e)} - \phi^{1(e)}\right)\frac{x - x^{1(e)}}{x^{2(e)} - x^{1(e)}}$$

This results in

for element 1: $$\hat{\phi}^{(e)} = \hat{\phi}^{1} + \left(\hat{\phi}^{2} - \hat{\phi}^{1}\right)\frac{x - x^{1}}{x^{2} - x^{1}}$$

for element 2: $$\hat{\phi}^{(e)} = \hat{\phi}^{2} + \left(\hat{\phi}^{3} - \hat{\phi}^{2}\right)\frac{x - x^{2}}{x^{3} - x^{2}}$$

Definition of approximation

So the continuity is satisfied without additional measures.

By introducing a dimensionless coordinate for each element:

$$\xi = \frac{x - x^{1(e)}}{x^{2(e)} - x^{1(e)}}$$

the approximation can be written in the form

$$\hat{\phi}^{(e)} = \hat{\phi}^{1(e)}(1-\xi) + \phi^{2(e)}\xi$$

or with the shape functions

$$\hat{\phi}^{(e)} = \hat{\phi}^{1(e)} N^{1(e)} + \phi^{2(e)} N^{2(e)}$$

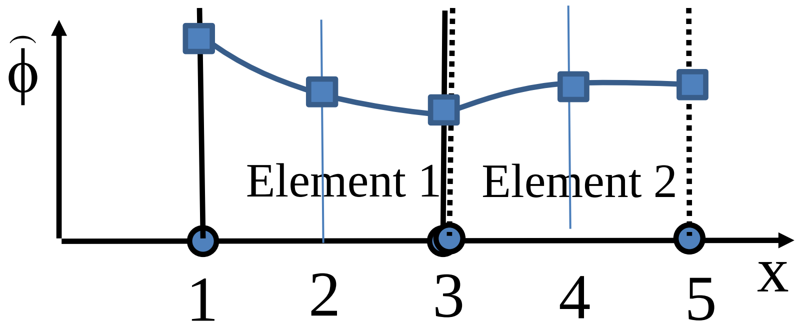
Where the shape functions are defined as

$$N^{1(e)} = 1 - \xi \quad \text{and} \quad N^{2(e)} = \xi$$

Definition of approximation

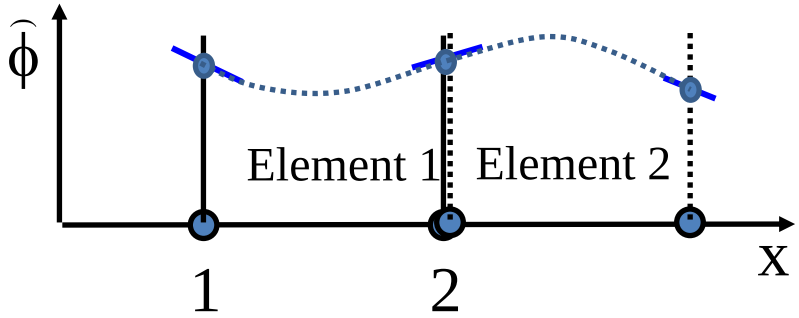
The continuity is automatically satisfied by using node values

Example:
1D quadratic



But differentiability is not yet obtained

This could be obtained by introducing also the derivatives on the nodes as generalized node values



This works, but it is very complicated in 2D and 3D

If not Galerkin method is applied (reduction of continuity requirements!) for higher order differential equations mostly the higher order differential equation is transformed into a system of differential equation of first order.

Example:

Basic equation $$\frac{\partial^2 u}{\partial x^2} + f = 0$$

Introduction of function g $$g = \frac{\partial u}{\partial x}$$

Resulting system of 1st order equation

$$\frac{\partial g}{\partial x} + f = 0$$

$$\frac{\partial u}{\partial x} - g = 0$$

By introducing the vorticity vector the Navier-Stokes equations can be transferred into a system of 6 equations of first order

General procedure

1. Definition of describing equations
2. Definition of Computational domain, Boundary conditions Initial conditions
3. Dividing of the domain into Finite Elements
4. Definition of the local approximation of the solution quantities
5. Discrete form of the describing equations
6. Assembling of the relevant matrices
7. Solution of the system of linear equations
8. Solution (Linear equations)

Update of coefficients

Convergence check

Solution (non-linear equations)

Discrete form of the describing equations

In opposition to the Finite Difference Method, where a direct discretization of the differential equations is undertaken, the FEM is based on a variation principle:

$$\Pi = \iiint_{\Omega} F\left(\phi, \frac{\partial \phi}{\partial x}, \frac{\partial \phi}{\partial y}, \frac{\partial \phi}{\partial z}, \frac{\partial^2 \phi}{\partial x^2}, \ldots., x, y, z \right) dxdydz \ = \min$$

ϕ is the solution function looked for. It can be a single function or a set of unknown functions (e. g. for the Navier-Stokes equations ϕ stands for the velocity components and the pressure).

In structural mechanics an equivalent variation principle to the differential equation formulation exists (e. g. principle of virtual work).

For the Navier-Stokes equations an equivalent variation principle is not known! As a consequence an approximation must be used.

Discrete form of the describing equations

Method of weighted residuals (MWR)

Partial Differential equation (PDE) $D(\Phi) = f \quad im \quad Gebiet \quad \Omega$

Approximation of the solution $\hat{\Phi} = \sum_{k=1}^{n} \alpha^k \cdot \phi^k$

$\phi^k \ldots linear\ independent\ functions$
$\alpha^k \ldots coefficients\ (looked\ for)$

Introduced into the in PDE results in a residuum $\varepsilon = D(\hat{\Phi}) - f$

Postulated: $\int_\Omega w^k \cdot \varepsilon d\Omega = \int_\Omega w^k \cdot (D(\hat{\Phi}) - f) d\Omega = 0 < k = 1, \ldots, n >$

linear independent weighting functions

There are several methods to choose the weighting functions

Discrete form of the describing equations

Least Square Method

For the LSM it is postulated

$$\iiint_{\Omega} \varepsilon^2 \, d\Omega = \min$$

This is equivalent to

$$\iiint_{\Omega} \frac{\partial \varepsilon}{\partial \alpha^k} \varepsilon \; d\Omega = 0$$

Compared to the general MWR approach this results in the following weighting functions

$$w^k = \frac{\partial \varepsilon}{\partial \alpha^k}$$

<u>Note</u>: Using LSM no reduction of the continuity requirements can be obtained by partial integration (Green-Gauss theorem)

Discrete form of the describing equations

Galerkin Formulation

This is the mostly used formulation for the Navier-Stokes equation.

The weighting function are chosen equal to the shape functions

$$w^k = N^k$$

This results in a variation principle

$$\iiint_{\Omega} N^k \, \varepsilon \, d\Omega = 0$$

Discrete form of the describing equations

Collocation method

Using the collocation method Dirac-functions are used as weighting functions

$$w^k = \delta\left(x_i - x_i^k\right) \qquad \delta\left(x_i - x_i^k\right) = \begin{cases} 0 & \text{for } \left(x_i \neq x_i^k\right) \\ \infty & \text{for } \left(x_i = x_i^k\right) \end{cases} \qquad \int_{-\infty}^{\infty} \delta \, dx = 1$$

This is equivalent to the postulation, that the residuals at given points (collocation points) is zero.

$$\iiint_{\Omega} \varepsilon \cdot \delta\left(x_i - x_i^k\right) d\Omega = \varepsilon\big|_{x_i^k} = 0$$

Discrete form of the describing equations

Subdomain Method

The weighting function is chosen to be 1 in the subdomain and 0 outside the subdomain

This means the residuum zero in an „integral“ way in the subdomain.

The weighting function is given as

$$w^k = \begin{cases} 1 & \text{in} \quad D_m \\ 0 & \text{out of} \quad D_m \end{cases}$$

The subdomain method is equivalent to the Finite Volume Method (FVM)

General procedure

1. Definition of describing equations
2. Definition of Computational domain, Boundary conditions Initial conditions
3. Dividing of the domain into Finite Elements
4. Definition of the local approximation of the solution quantities
5. Discrete form of the describing equations
6. Assembling of the relevant matrices
7. Solution of the system of linear equations
8. Solution (Linear equations)

Convergence check (after step 7)

Update of coefficients (leads back to step 6)

Solution (non-linear equations)

Assembling of the relevant matrices

Calculation of the element matrices

Based on the method of weighted residuals

$$\iiint_{\Omega} w^k \varepsilon d\Omega = 0 \quad < k = 1, \ldots, \text{no of nodes} >$$

the integral can be expressed by the sum of the integrals over all elements

$$\iiint_{\Omega} w^k \varepsilon d\Omega = \sum \iiint_{\Omega^{(e)}} w^{k(e)} \varepsilon d\Omega^{(e)}$$

This results in a linear system of equations (LSE)

$$\underline{\underline{A}}\,\underline{\phi} = \underline{b}$$

The matrix A can be assembled from the element matrices, also the vector of the right-hand side is calculated from all element contributions

Vector ϕ are the node values, searched for.

Assembling of the relevant matrices

$$\underline{\underline{A}} = \sum \underline{\underline{A}}^{(e)} \quad \text{und} \quad \underline{b} = \sum \underline{b}^{(e)}$$

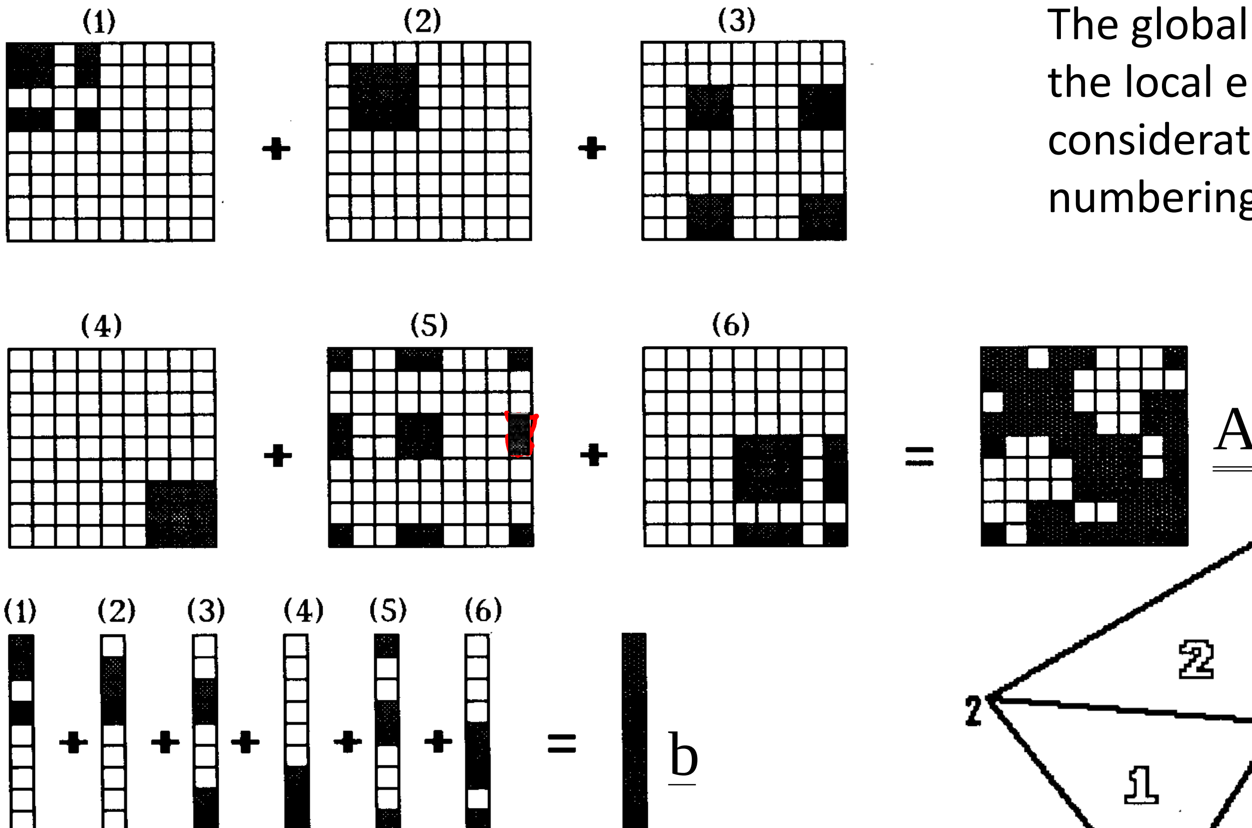
The element matrices depend on the equations, the types of elements, approximations etc.

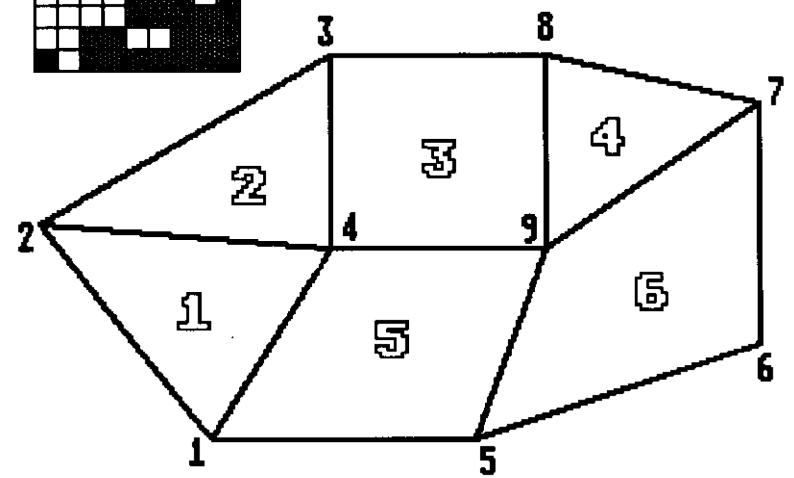
For linear PDEs and rather simple types of elements the element matrices can be calculated directly. For non-linear PDEs and/or more complex elements the element matrices must be calculated numerically.

The calculation of the element matrices will be shown later at an example.

Assembling of the relevant matrices

The global matrix is assembled from the local element matrices under consideration of the global node numbering





Introduction of boundary conditions

Through the assembly of the element matrices one obtains the global linear system of equations

Dirichlet boundary conditions must be introduces into this LSE

Neuman boundary conditions result in an additional contribution to the global matrix

Example: Diriclet b.c.

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \cdot \begin{bmatrix} \widehat{\phi}_1 \\ \widehat{\phi}_2 \\ \widehat{\phi}_3 \\ \widehat{\phi}_4 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix}$$

Node value 1 is given $\widehat{\phi}_1 = \phi_{fix}$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \cdot \begin{bmatrix} \widehat{\phi}_1 \\ \widehat{\phi}_2 \\ \widehat{\phi}_3 \\ \widehat{\phi}_4 \end{bmatrix} = \begin{bmatrix} \phi_{fix} \\ b_2 \\ b_3 \\ b_4 \end{bmatrix}$$

General procedure

1. Definition of describing equations
2. Definition of Computational domain, Boundary conditions Initial conditions
3. Dividing of the domain into Finite Elements
4. Definition of the local approximation of the solution quantities
5. Discrete form of the describing equations
6. Assembling of the relevant matrices
7. Solution of the system of linear equations
8. Solution (Linear equations)

Linearization

- Update of coefficients
- Convergence check
- Solution (non-linear equations)

Linearization

Basic equations

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = -\frac{1}{\rho}\frac{\partial p}{\partial x} + \nu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right)$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} = -\frac{1}{\rho}\frac{\partial p}{\partial y} + \nu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right)$$

Non-linear terms

Stokes-Linearization: Convection term is completely taken from the previous iteration step

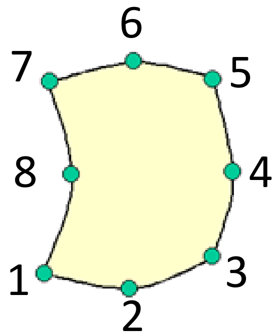
Picard-Iteration: $u\frac{\partial u}{\partial x} \approx u^{old}\frac{\partial u^{new}}{\partial x}$ Robust, slower convergence, large convergence radius

Newton-Iteration: fast convergence, small convergence radius

Steady-state convection-diffusion equation

Basic equation

$$U_j^* \frac{\partial \phi}{\partial x_j} - \Gamma \frac{\partial^2 \phi}{\partial x_j^2} - f = 0$$



Approximation within the element

$$\hat{\phi} = \sum_{l=1}^{8} N^l \hat{\phi}^l$$

Residuum

$$\varepsilon = U_j^* \frac{\partial \hat{\phi}}{\partial x_j} - \Gamma \frac{\partial^2 \hat{\phi}}{\partial x_j^2} - f$$

Method of weighted residuals

$$\iiint_{\Omega} w^k \varepsilon \, d\Omega = \sum_{\text{Elemente}} w^{k(e)} \varepsilon \, d\Omega$$

Steady-state convection-diffusion equation

Galerkin formulation $$w^k = N^k$$

$$\iiint_{\Omega} N^k \varepsilon \, d\Omega = \sum_{\text{elements}} \iiint_{\text{El } i} N^{k(e)} \varepsilon \, d\Omega = 0$$

with
$$\varepsilon = \left(U_j^* \frac{\partial N^l}{\partial x_j} - \Gamma \frac{\partial^2 N^l}{\partial x_j^2} \right) \hat{\phi}^l - f$$

$$\sum_{\text{Elemente}} \iiint_{\text{El } i} N^k \left(\left(U_j^* \frac{\partial N^l}{\partial x_j} - \Gamma \frac{\partial^2 N^l}{\partial x_j^2} \right) \hat{\phi}^l - f \right) d\Omega = 0$$

General procedure

1. Definition of describing equations
2. Definition of Computational domain, Boundary conditions Initial conditions
3. Dividing of the domain into Finite Elements
4. Definition of the local approximation of the solution quantities
5. Discrete form of the describing equations
6. Assembling of the relevant matrices
7. Solution of the system of linear equations
8. Solution (Linear equations)

Update of coefficients

Convergence check

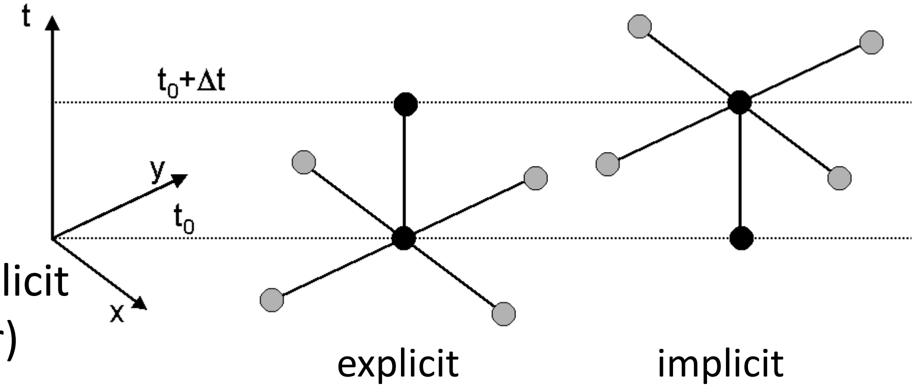
Solution (non-linear equations)

Time integration

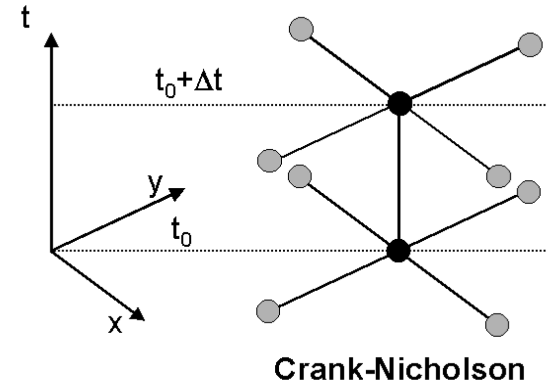
Time discretization

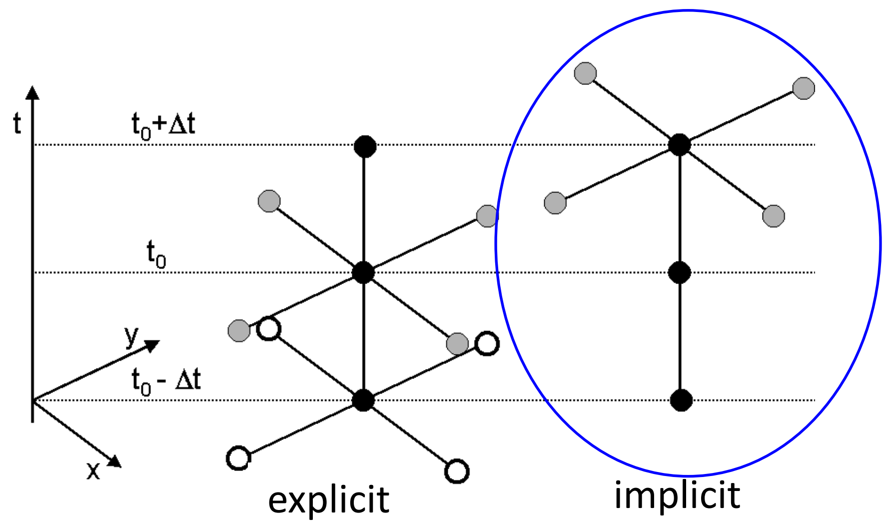
The time discretization is done by means of finite differences in time, FEM is used only in space

Explicit or implicit Euler (1. order)



Crank Nicholson (2. order)





3 Level Schemes explicit order implicit (2. order)

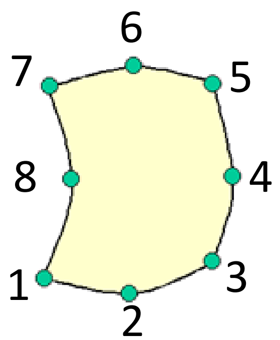
Unsteady convection-diffusion equation

Basic equation

$$\frac{\partial \phi}{\partial t} + U_j^* \frac{\partial \phi}{\partial x_j} - \Gamma \frac{\partial^2 \phi}{\partial x_j^2} - f = 0$$

Approximation within the elements

$$\hat{\phi} = \sum_{l=1}^{8} N^l \hat{\phi}^l$$



Residuum

$$\varepsilon = \frac{\partial \hat{\phi}}{\partial t} + U_j^* \frac{\partial \hat{\phi}}{\partial x_j} - \Gamma \frac{\partial^2 \hat{\phi}}{\partial x_j^2} - f$$

MWR

$$\iiint_{\Omega} w^k \varepsilon \, d\Omega = \sum_{\text{elements}} w^{k(e)} \varepsilon \, d\Omega$$

Galerkin formulation

$$w^k = N^k$$

Unsteady convection-diffusion equation

Galerkin formulation

$$\iiint_{\Omega} N^k \varepsilon \, d\Omega = \sum_{\text{elements}} \iiint_{\text{El i}} N^{k(e)} \varepsilon \, d\Omega = 0$$

With the residuum

$$\varepsilon = N^l \hat{\dot{\phi}} + \left(U_j^* \frac{\partial N^l}{\partial x_j} - \Gamma \frac{\partial^2 N^l}{\partial x_j^2} \right) \hat{\phi}^l - f$$

$$\sum_{\text{Elemente}} \left(\iiint_{\text{Eli}} N^k N^l d\Omega \right) \hat{\dot{\phi}}^l + \iiint_{\text{El i}} N^k \left(\left(U_j^* \frac{\partial N^l}{\partial x_j} - \Gamma \frac{\partial^2 N^l}{\partial x_j^2} \right) \hat{\phi}^l - f \right) d\Omega = 0$$

Discretized equations

$$M^{kl} \cdot \hat{\dot{\phi}}^l + D^{kl} \cdot \hat{\phi}^l - f^l = 0$$

Mass matrix (M^{kl}) — Stiffness matrix (D^{kl})

Time discretisation

$$M^{kl} \cdot \frac{\hat{\phi}^l - \hat{\phi}^l}{\Delta t} + D^{kl} \hat{\phi}^l - f^l = 0$$

FD discretisation

Implicit (Euler backward)

$$M^{kl} \cdot \frac{\hat{\phi}^l_{new} - \hat{\phi}^l_{old}}{\Delta t} + D^{kl} \hat{\phi}^l_{new} - f^l_{new} = 0$$

Explicit (Euler forward)

$$M^{kl} \cdot \frac{\hat{\phi}^l_{new} - \hat{\phi}^l_{old}}{\Delta t} + D^{kl} \hat{\phi}^l_{old} - f^l_{old} = 0$$

Still a linear system of equations must be solved

=> In FEM usually an implicit time descretization is applied

Example: 1D Helmholtz equation

$$\frac{\partial^2\phi}{\partial x^2} - \lambda^2\phi = 0 \qquad \lambda = 0.5$$

Boundary conditions

$$\phi(0)=1; \quad \phi(2)=1.54308$$

Exact solution: $\phi = c_1 e^{-\lambda x} + c_2 e^{\lambda x}$ with $c_1 = c_2 = 0.5$

MWR

$$\int_0^2 w^k \left(\frac{\partial^2 \hat{\phi}}{\partial x^2} - \lambda^2 \hat{\phi} \right) dx = 0$$

Approximation $\hat{\phi} = \hat{\phi}^k N^k$

In the direct form the approximation must be continuous and differentiable.

The requirement can be reduced by applying Galerkin method and the Green Gauss theorem

Example: 1D Helmholtz equation

Galerkin formulation

$$w^k = N^k$$

Weighting functions = Shape functions

C_{n-1}- continuity

Integral contains derivations 2. order => approximation continuous and differentiable

$$\left[\int_0^2 N^k \left(\frac{\partial^2 N^l}{\partial x^2} - \lambda^2 N^l \right) dx \right] \hat{\phi}^l = 0$$

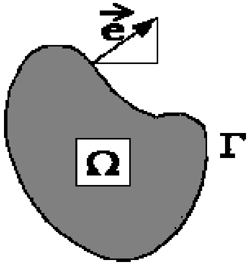
k and l count from 1 to the no of nodes n

System matrix (n x n)

Node values

By applying Green-Gauss theorem the continuity requirements can be reduced by one order.

Green-Gauss theorem



$$\iiint_{\Omega} \phi \frac{\partial \theta}{\partial x_i} d\Omega = -\iiint_{\Omega} \frac{\partial \phi}{\partial x_i} \theta d\Omega + \int_{\Gamma} \phi \theta e_i d\Gamma$$

Example: 1D Helmholtz equation

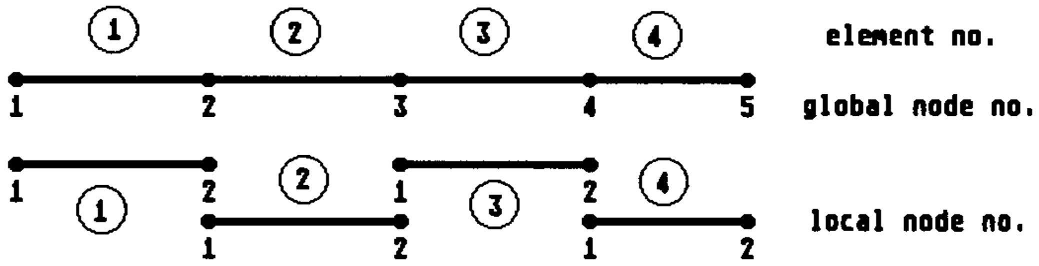
Weak formulation $$\left[\int_0^2 \left(\frac{\partial N^k}{\partial x} \frac{\partial N^l}{\partial x} + \lambda^2 N^k N^l \right) dx \right] \hat{\phi}^l = \underline{\underline{A}}\,\underline{\hat{\phi}} = 0$$

(Change of sign!!)

The resulting surface integral can be neglected since Diriclet boundary conditions are given. For Neuman boundary conditions the surface integral has to be calculated and results in an addition to the system matrix.

In the weak formulation the integral contains only derivatives of first order. Therefor only C_0-continuity is required. (This is the big advantage of the Galerkin formulation)

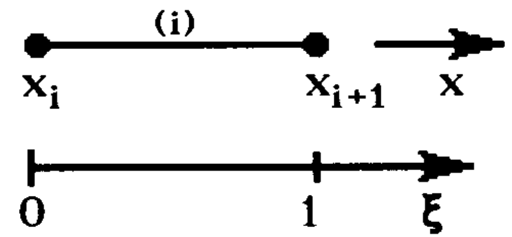
Solution with 4 linear elements



Example: 1D Helmholtz equation

Calculation of the global matrix from the local element matrices

$$\left[\sum \int_{x_i}^{x_{i+1}} \left(\frac{\partial N^k}{\partial x} \frac{\partial N^l}{\partial x} + \lambda^2 N^k N^l \right) dx \right] \hat{\phi}^l = \sum \underline{\underline{A}}^{(e)} \underline{\hat{\phi}} = 0$$



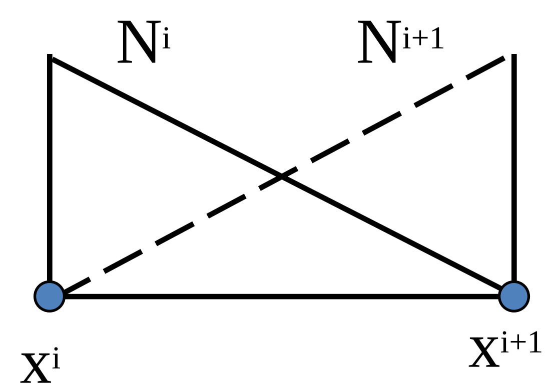
Element matrix

Dimensionless coordinate

$$\xi = \frac{x - x_i}{L} \qquad \text{with} \qquad L = x_{i+1} - x_i$$

Linear shape functions

$$N^i = 1 - \xi; \qquad N^{i+1} = \xi$$



Example: 1D Helmholtz equation

with $$dx = L\, d\xi$$

and $$\frac{\partial N^k}{\partial x} = \frac{\partial N}{\partial \xi}\frac{\partial \xi}{\partial x} = \frac{1}{L}\frac{\partial N}{\partial \xi}$$

Tis results in $$\underline{\underline{A}}^{(e)} = \int_0^1 \left(\frac{1}{L^2}\frac{\partial N^k}{\partial \xi}\frac{\partial N^l}{\partial \xi} + \lambda^2 N^k N^l \right) \cdot L\, d\xi$$

In components

$$\underline{\underline{A}}^{(e)} = \int_0^1 \begin{bmatrix} \frac{1}{L}\left(\frac{\partial N^1}{\partial \xi}\right)^2 + \lambda^2 L \left(N^1\right)^2 & \frac{1}{L}\left(\frac{\partial N^1}{\partial \xi}\frac{\partial N^2}{\partial \xi}\right) + \lambda^2 L \left(N^1 N^2\right) \\ \frac{1}{L}\left(\frac{\partial N^1}{\partial \xi}\frac{\partial N^2}{\partial \xi}\right) + \lambda^2 L \left(N^1 N^2\right) & \frac{1}{L}\left(\frac{\partial N^2}{\partial \xi}\right)^2 + \lambda^2 L \left(N^2\right)^2 \end{bmatrix} d\xi$$

Example: 1D Helmholtz equation

$$\textit{with} \quad \frac{\partial N^1}{\partial \xi} = -1 \qquad \textit{and} \quad \frac{\partial N^2}{\partial \xi} = 1$$

$$\textit{and} \quad \int_0^1 \xi^2 d\xi = \frac{1}{3}; \qquad \int_0^1 (1-\xi)\xi d\xi = \frac{1}{6} \qquad \textit{and} \qquad \int_0^1 (1-\xi)^2 d\xi = \frac{1}{3}$$

With L = 0.5 and λ = 0.5 this results in the element matrix

$$\underline{\underline{A}}^{(e)} = \begin{bmatrix} \frac{1}{L} + \lambda^2 \frac{L}{3} & -\frac{1}{L} + \lambda^2 \frac{L}{6} \\ -\frac{1}{L} + \lambda^2 \frac{L}{6} & \frac{1}{L} + \lambda^2 \frac{L}{3} \end{bmatrix} = \begin{bmatrix} 2.042 & -1.979 \\ -1.979 & 2.042 \end{bmatrix}$$

Example: 1D Helmholtz equation

Assembling of the global matrix

$$\underline{\underline{A}} = \begin{bmatrix} 2.042 & -1.979 & 0 & 0 & 0 \\ -1.979 & 4.083 & -1.979 & 0 & 0 \\ 0 & -1.979 & 4.083 & -1.979 & 0 \\ 0 & 0 & -1.979 & 4.083 & -1.979 \\ 0 & 0 & 0 & -1.979 & 2.042 \end{bmatrix}$$

Introduction of the boundary conditions $\widehat{\phi}^1 = 1. \qquad \widehat{\phi}^5 = 1.54308$

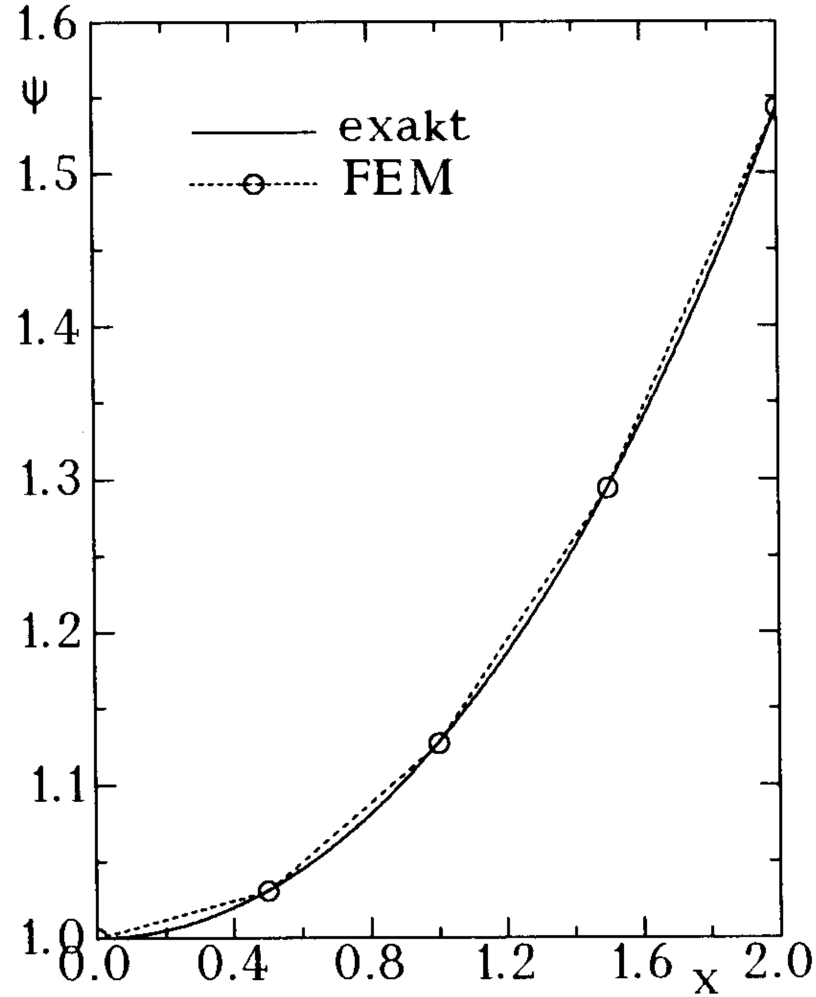
Final equation

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ -1.979 & 4.083 & -1.979 & 0 & 0 \\ 0 & -1.979 & 4.083 & -1.979 & 0 \\ 0 & 0 & -1.979 & 4.083 & -1.979 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \widehat{\phi}^1 \\ \widehat{\phi}^2 \\ \widehat{\phi}^3 \\ \widehat{\phi}^4 \\ \widehat{\phi}^5 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 1.54308 \end{bmatrix}$$

Example: 1D Helmholtz equation

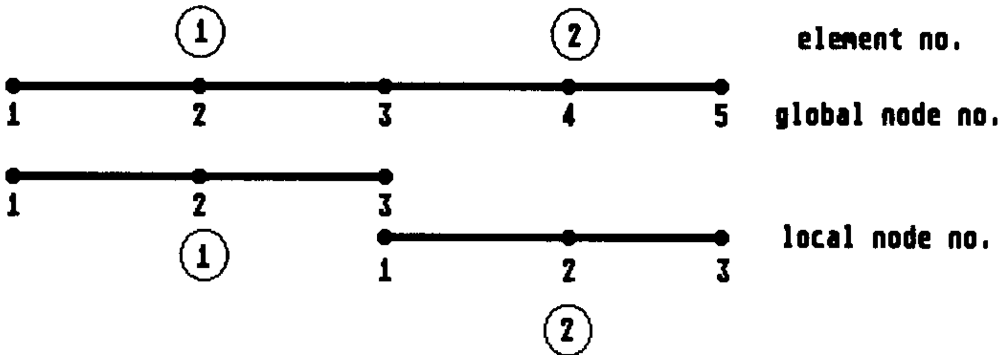
Solution

node	numerical	exact
1	1	1
2	1.03092	1.03141
3	1.12694	1.12763
4	1.29414	1.29468
5	1.54308	1.54308



Example: 1D Helmholtz equation

Solution using two quadratic elements



Shape functions

$$N^{1(e)} = (1-\xi)(1-2\xi)$$

$$N^{2(e)} = 4\xi(1-\xi)$$

$$N^{3(e)} = -\xi(1-2\xi)$$

In analogy to the linear elements the element matrices can be calculated

Example: 1D Helmholtz equation

Element matrix

$$\underline{\underline{A}}^{(e)} = \begin{bmatrix} \frac{7}{3L}+\frac{4L}{30} & \frac{-8}{3L}+\frac{2L}{30} & \frac{1}{3L}-\frac{L}{30} \\ \frac{-8}{3L}+\frac{2L}{30} & \frac{16}{3L}+\frac{16L}{30} & \frac{-8}{3L}+\frac{2L}{30} \\ \frac{1}{3L}-\frac{L}{30} & \frac{-8}{3L}+\frac{2L}{30} & \frac{7}{3L}+\frac{4L}{30} \end{bmatrix}$$

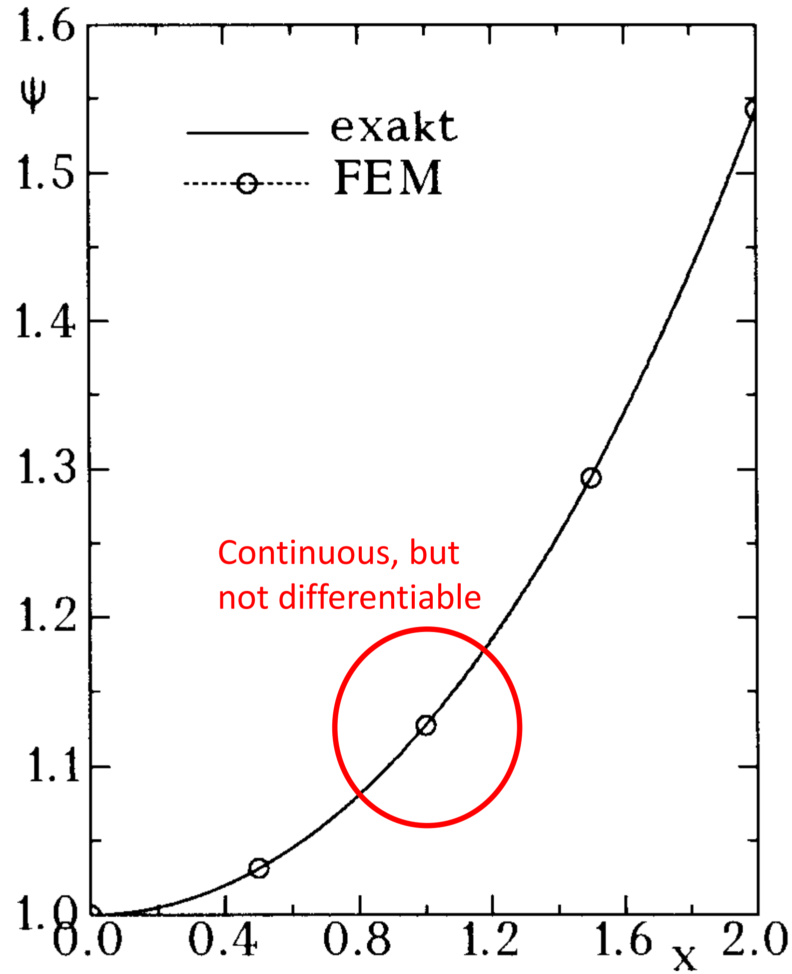
Both element matrices will be summed up and the boundary conditions must be introduced.

Example: 1D Helmholtz equation

Global matrix

$\frac{7}{3L_1}$	$-\frac{8}{3L_1}$	$\frac{1}{3L_1}$		
$-\frac{8}{3L_1}$	$\frac{16}{3L_1}$	$-\frac{8}{3L_1}$		
$\frac{1}{3L_1}$	$-\frac{8}{3L_1}$	$\frac{7}{3L_1}+\frac{7}{3L_2}$	$-\frac{8}{3L_2}$	$\frac{1}{3L_2}$
		$-\frac{8}{3L_2}$	$\frac{16}{3L_2}$	$-\frac{8}{3L_2}$
		$\frac{1}{3L_2}$	$-\frac{8}{3L_2}$	$\frac{7}{3L_2}$

Solution



Gauss Integration

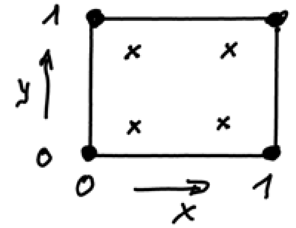
For linear PDEs and rather simple types of elements the element matrices can be calculated directly. For non-linear PDEs and/or more complex elements the element matrices must be calculated numerically.

Gauss Integration

2D: $\int_E f(x,y)\,dx\,dy$

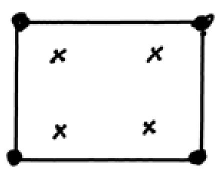
$$\approx A \cdot \sum_{i=1}^{n_G} f(x_i, y_i) \cdot \omega_i$$

A ↑ Area, (x_i, y_i) ↑ Gauss points, ω_i ↑ weight



	X	Y	ω
1	0.2113	0.2113	0.25
2	0.7887	0.2113	0.25
3	0.2113	0.7887	0.25
4	0.7887	0.7887	0.25

Example: 2D rectangle



Accuracy 2nd order is sufficient.
↳ 4 Gauss points

$$\int_E f(x,y)\,dx\,dy = A \cdot \sum_{i=1}^{4} f(x_i, y_i) \cdot \omega_i$$

Transport equation

$$\frac{\partial \phi}{\partial t} + U_j^* \frac{\partial \phi}{\partial x_j} - \Gamma \frac{\partial^2 \phi}{\partial x_j^2} - f = 0$$

Picard-Iteration of the convection term

Approximation: $\hat{\phi} = N^k \Phi^k$

N^k … *Shape function*
Φ^k … *Node values*

Method of weighted residuals (MWR)

Galerkin method: $\int_\Omega N^k \varepsilon d\Omega = 0$ with $\varepsilon = \frac{\partial \hat{\phi}}{\partial t} + U_j^* \frac{\partial \hat{\phi}}{\partial x_j} - \Gamma \frac{\partial^2 \hat{\phi}}{\partial x_j^2} - f$

„weak“ Formulation

$$\left\{\int_\Omega N^k N^l d\Omega\right\}\dot{\Phi}^l + \left\{\int_\Omega \left(U_j^* N^k \frac{\partial N^l}{\partial x_j} + \Gamma \frac{\partial N^k}{\partial x_j}\frac{\partial N^l}{\partial x_j}\right) d\Omega - \int_\Gamma \Gamma N^k \frac{\partial N^l}{\partial x_j} e_j d\Gamma\right\}\Phi^l$$

$$= \int_\Omega N^k f d\Omega$$

Navier-Stokes equation

Mass conservation

$$\frac{\partial u_i}{\partial x_i} = 0$$

Momentum conservation

$$\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} = -\frac{1}{\rho}\frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j}\left[\nu_{eff}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)\right]$$

Approximations

$$\hat{u}_i = N^k \hat{u}_i^k$$

N^k *and* M^k *Shape functions*

$\hat{u}_i^k$ *Node values*

$$\hat{p} = M^k \hat{p}^k$$

Navier-Stokes equation

Momentum equations

$$\int_{\Omega}\left[N^k N^l \dot{U}_i^l + \tilde{U}_j N^k \frac{\partial N^l}{\partial x_j} U_i^l + \upsilon_{eff} \frac{\partial N^k}{\partial x_j}\left(\frac{\partial N^l}{\partial x_j} U_i^l + \frac{\partial N^l}{\partial x_i} U_j^l\right)\right] d\Omega$$

$$-\int_{\Gamma} \upsilon_{eff} N^k \left(\frac{\partial N^l}{\partial x_j} U_i^l + \frac{\partial N^l}{\partial x_i} U_j^l\right) n_j \, d\Gamma - \int_{\Omega} \frac{1}{\rho} \frac{\partial N^k}{\partial x_i} M^q P^q + \int_{\Gamma} N^k M^q P^q n_i \, d\Gamma = 0$$

$$< k \_ 1,\ldots,n >, \; < l = 1,\ldots,n >, \; < q = 1,\ldots,m >$$

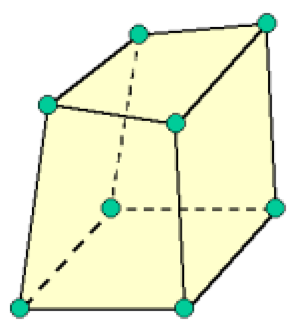
Continuity equation $\int_{\Omega} M^q \frac{\partial N^l}{\partial x_i} U_i^l \; d\Omega = 0$ $< q = 1,\ldots,m >, \; < l = 1,\ldots,n >$

Navier-Stokes equation

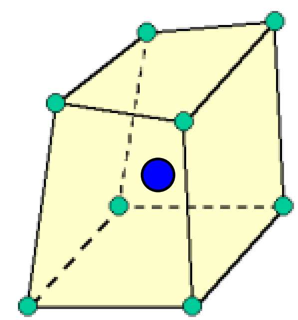
Minimum continuity requirements:

Continuous approximation for velocity components

Discontinuous approximatio for pressure



Tri-linear Hexahedron for velocity components



Constant pressure

LBB Condition

Ladyzhenskaya–Babuška–Brezzi condition

$$\sup_{v \in V, v \neq 0} \frac{b(v,q)}{||v||_V} \geq \beta ||q||_Q$$

The approximations for velocity and pressure cannot be defined independently.

For the Navier-Stokes equation the velocity components can be interpreted as main variables (degrees of freedom),

The pressure can be seen as constrains

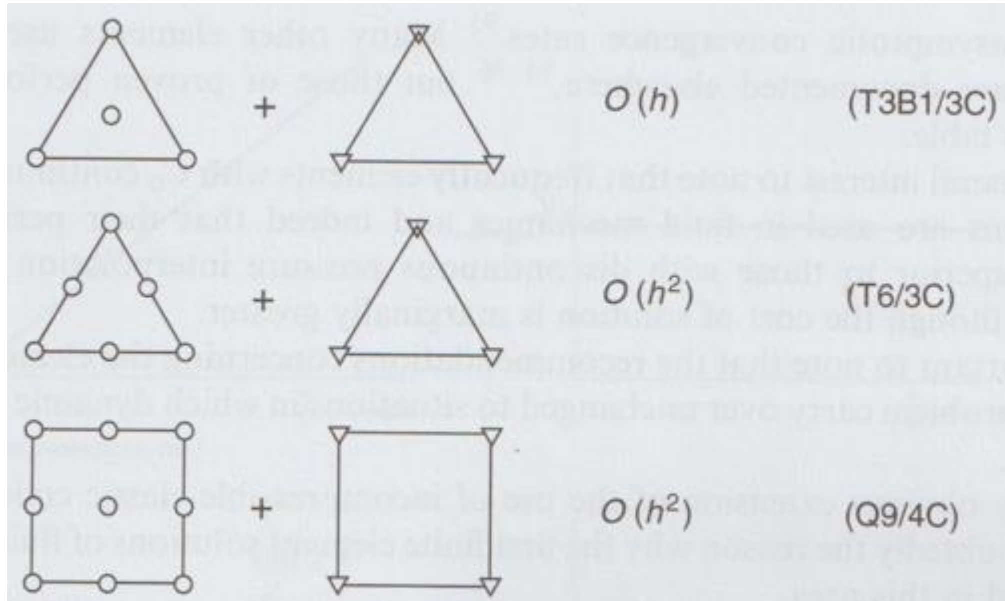
It is necessary to have more degrees of freedom as constrains

To guaranty a unique solution, the approximations must fulfill the LBB condition, this condition is very complicated and cannot be discussed here.

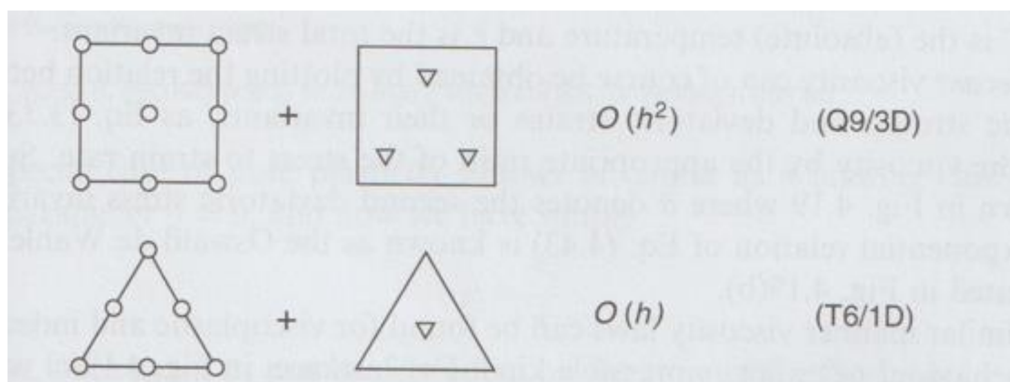
As a rule of thumb, it can be said, that the approximation of the velocities should be one order higher compared to the approximation of the pressure

LBB Condition

Example of elements, satisfying the LBB condition



Continuous pressure approximation

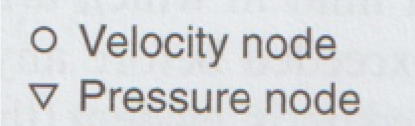


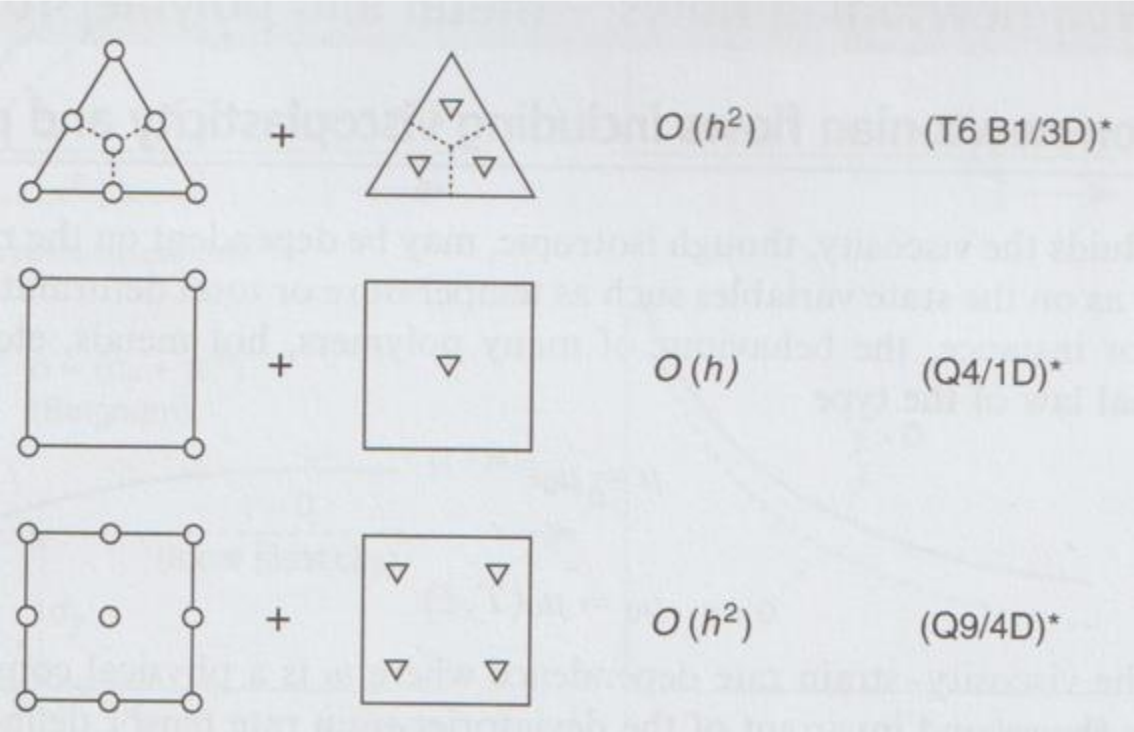
Discontinuous pressure approximation

○ Velocity node
▽ Pressure node

LBB Condition

Elements failing the LBB condition, but still performing reasonable



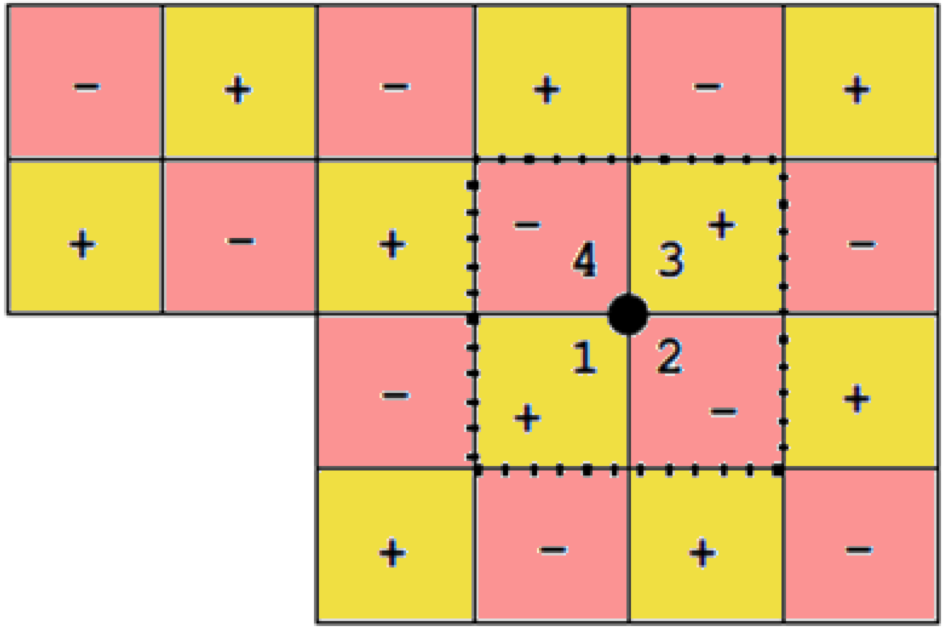


Problem with decoupling of different pressure modes

LBB Condition

Problem with decoupling of different pressure modes

Checkerboard oscillation

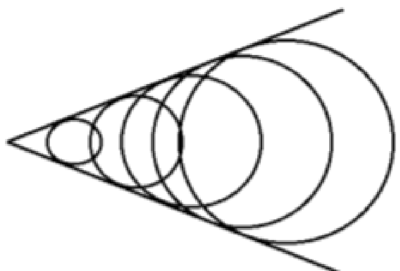


Cure: Smoothing of the pressure

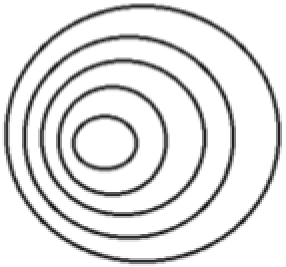
Convection-diffusion problem

$$\frac{\partial \phi}{\partial t} + U_j^* \frac{\partial \phi}{\partial x_j} - \Gamma \frac{\partial^2 \phi}{\partial x_j^2} - f = 0$$

Convection Diffusion

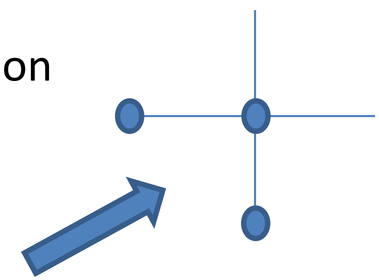


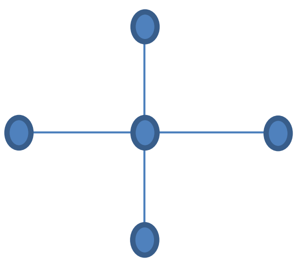
hyperbolic



elliptic

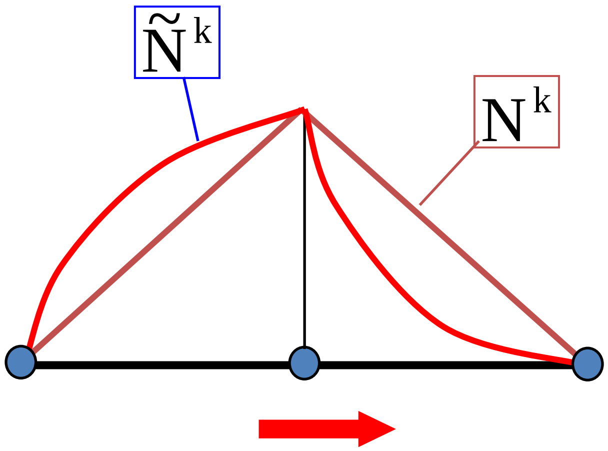
FD discretization





Upwind discretization

"Streamline-upwind" Petrov-Galerkin Method



Approximation: $\hat{\phi} = N^k \Phi^k$

Galerkin: $\int_\Omega N^k \varepsilon d\Omega = 0$

Petrov-Galerkin: $\int_\Omega \tilde{N}^k \varepsilon d\Omega = 0$

Residuum: $\varepsilon = \frac{\partial \hat{\phi}}{\partial t} + U_j^* \frac{\partial \hat{\phi}}{\partial x_j} - \Gamma \frac{\partial^2 \hat{\phi}}{\partial x_j^2} - f$

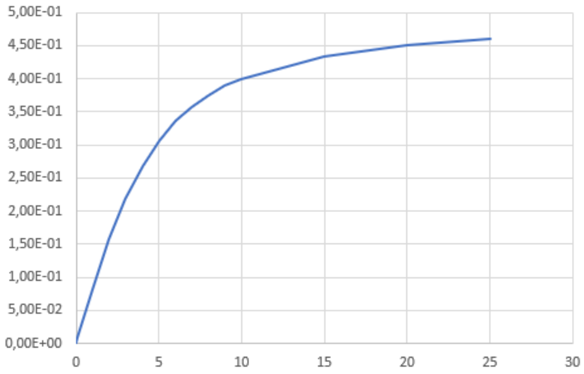
Upwind discretization

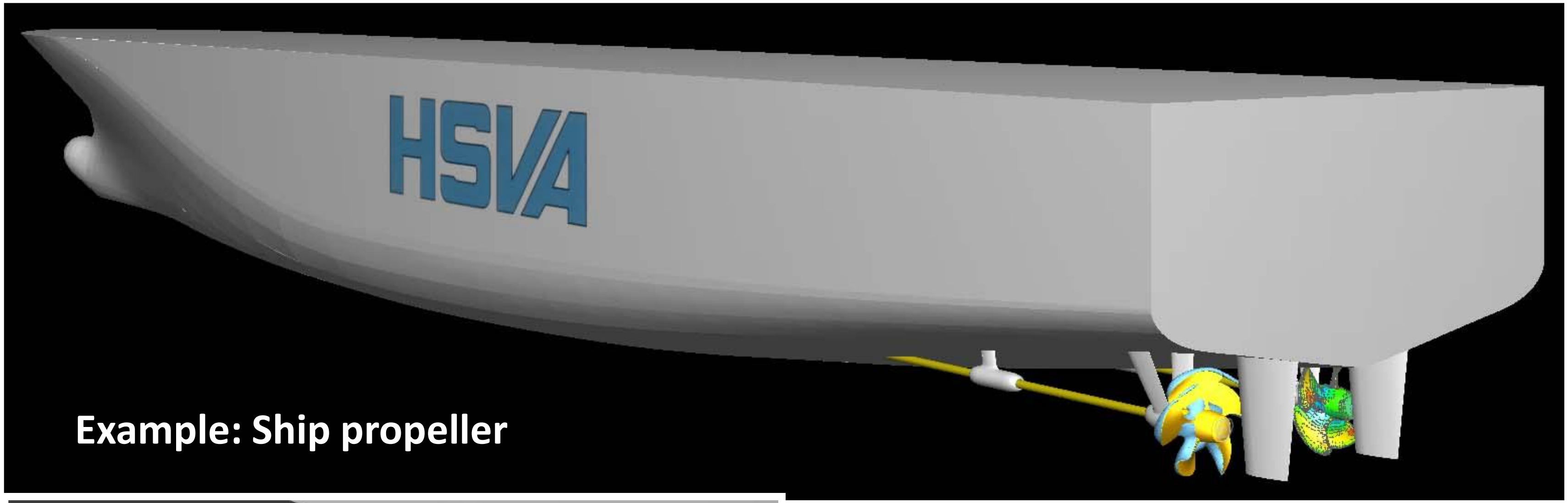
SUPG weighting function

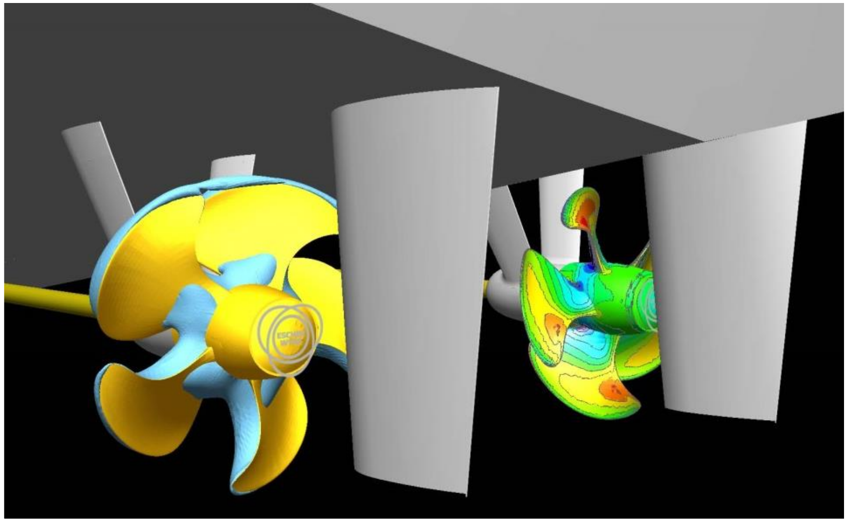
$$\tilde{N}^k = \begin{cases} N^k + \dfrac{\partial N^k}{\partial x_j} \dfrac{U_j}{\sqrt{U_i U_i}} \cdot k'' & \text{für } Re^{(e)} > 2 \\ N^k & \text{für } Re^{(e)} \leq 2 \end{cases}$$

$$Re^{(e)} = \frac{U^{(e)} h}{\Gamma}$$

$$k'' = \left(\coth\left(Re^{(e)}/2\right) - 2/Re^{(e)}\right) \cdot h/2$$

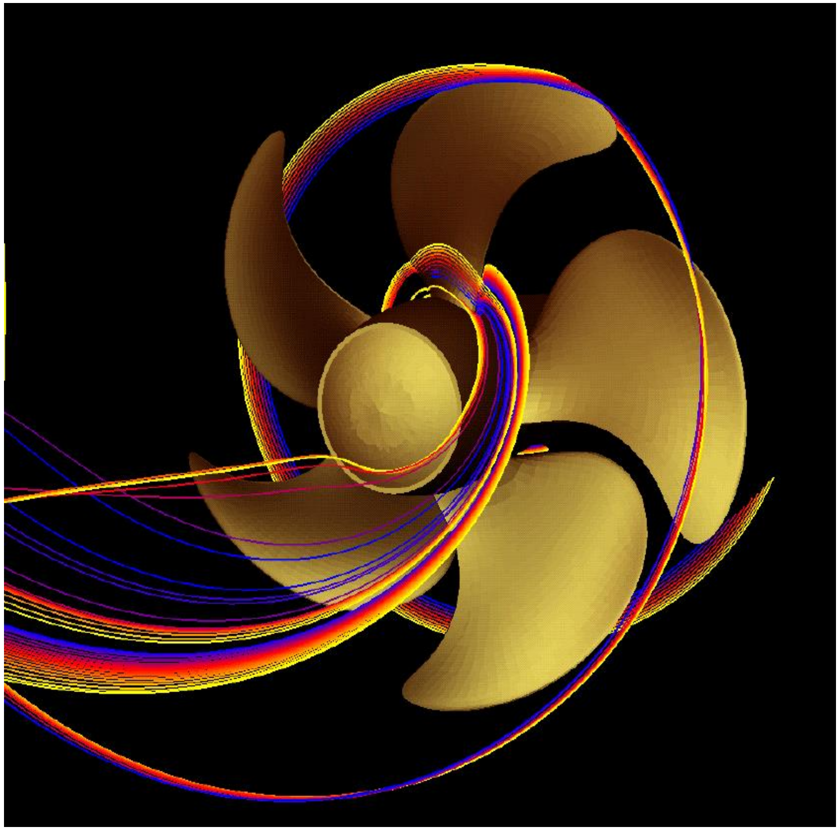


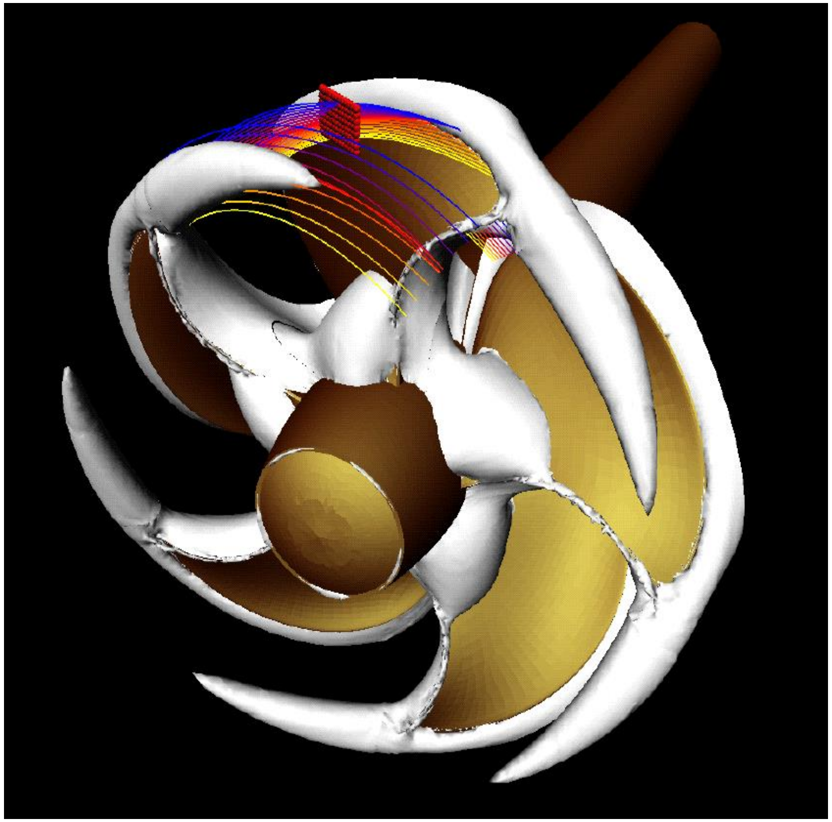

HSVA
Example: Ship propeller



Example ship propeller

Tip vortex

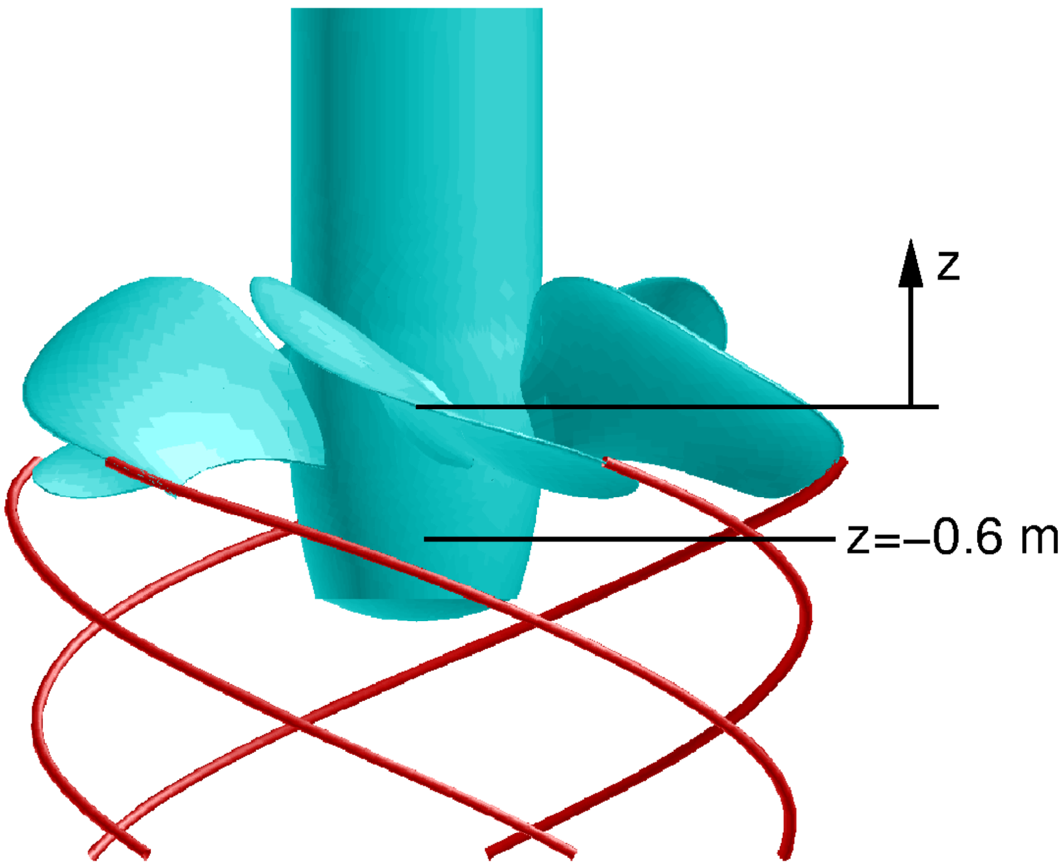




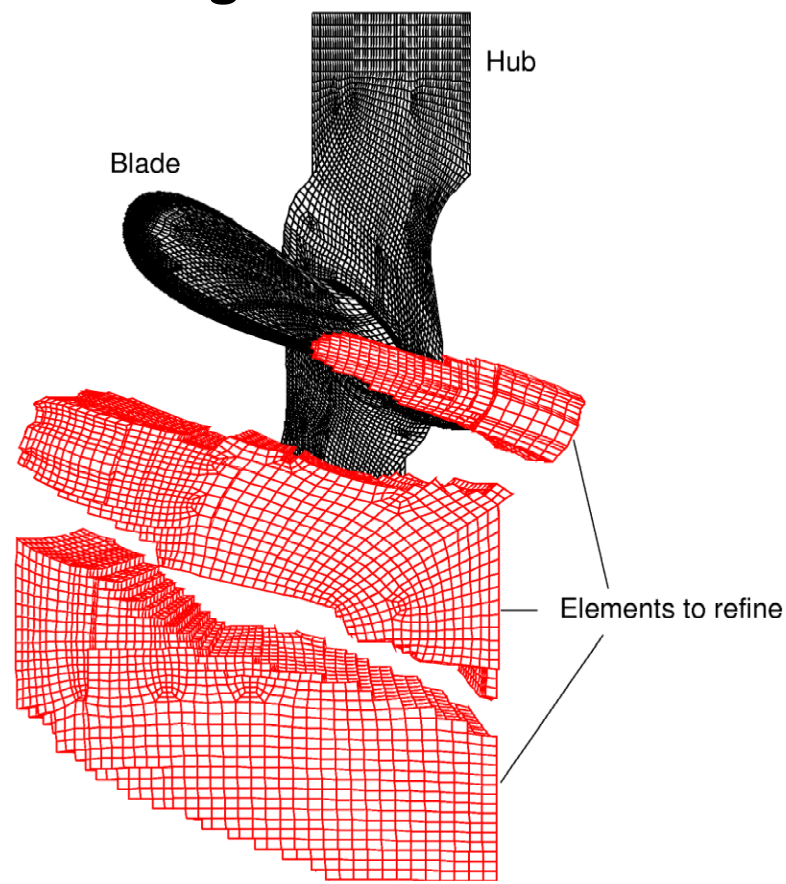
Prediction of vortex cavitation

Example ship propeller

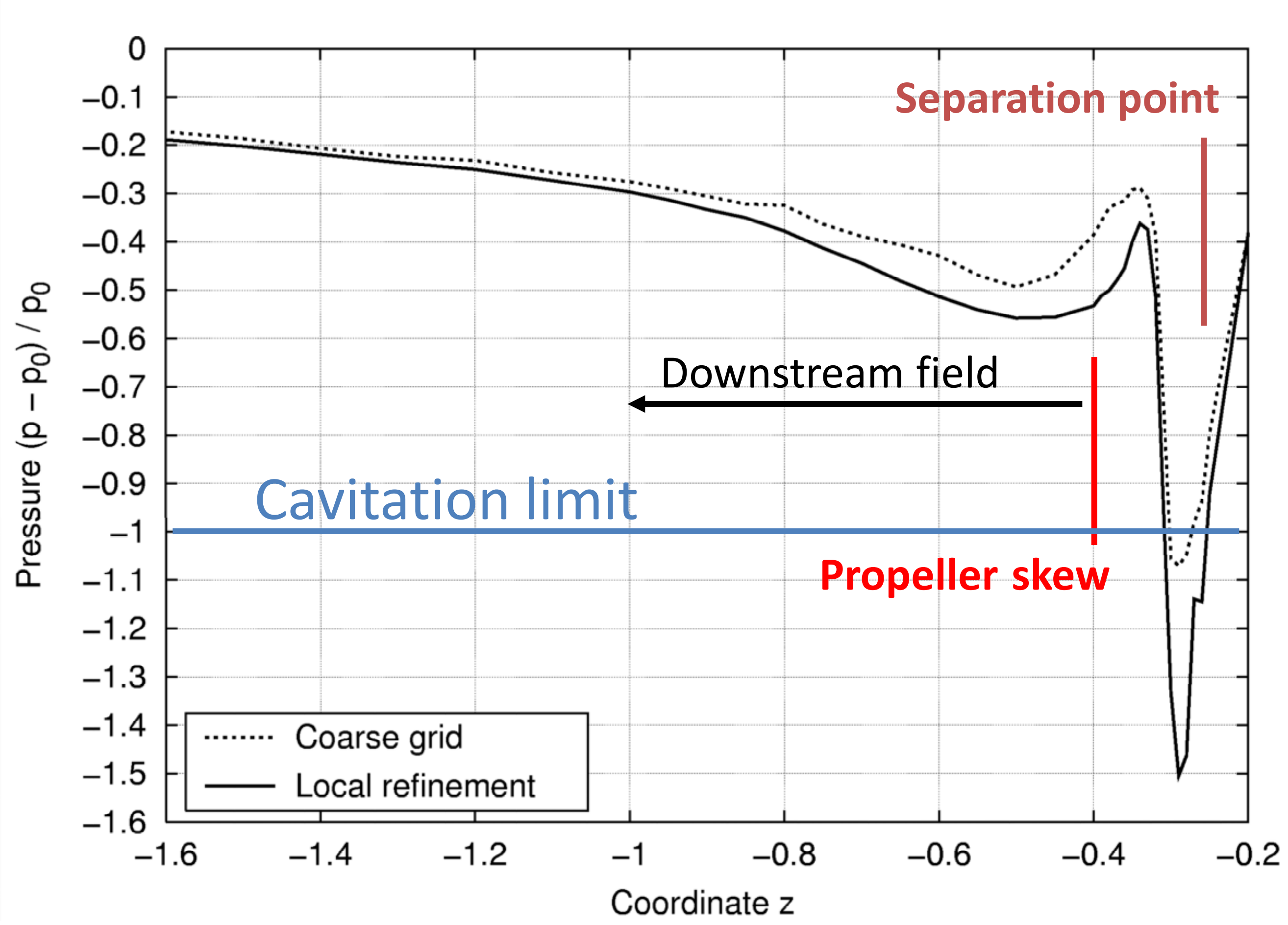
Tip vortex center



Local grid refinement



Example ship propeller



Summary

- Finite Element Method is very flexible
  - Different element types (tetraether, hexaeder, curvilinear elements, …..)
  - Different approximation (linear, quadratic …..)
- Unstructured grids
- Mostly used: Galerkin method
  - Green-Gauss theorem
  - Reduction of continuity requirements
- Streamline upwind Petrov Galerkin method for convection dominated flows
  - Skew-symmetric weighting function according to flow direction

---

Error estimation possible

Adaptive grid refinement possible