

Via numerical methods to simulations

Anna Trykozko
ICM, University of Warsaw

27-29 September 2011

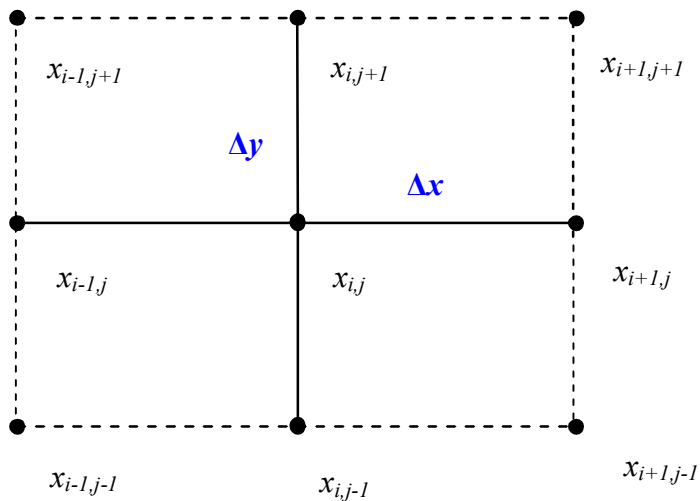
Computational approaches

- Molecular dynamics
- Cellular automata
- Models described by systems of ordinary differential equations
- Models described by systems of partial differential equations
- Hybrid models
- Coupled systems

Partial differential equations – computational approaches /1/

- Finite differences

Derivatives are replaced with finite difference schemes based on a (regular) grid of nodes and cells)



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

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

If $\Delta x = \Delta y = h$

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \approx \frac{u_{i+1,j} + u_{i-1,j} - 4u_{i,j} + u_{i,j+1} + u_{i,j-1}}{h^2}$$

Laplace's operator approximation in 2D case

Partial differential equations – computational approaches /2/

- Finite volumes

Computational domain divided into elements. Constructed based on balance equations.

- Galerkin-type methods (methods of Weighted Residuals)
 - **Finite element methods**
 - ...
- Discrete methods (particle methods, Lattice-Boltzmann method)

Outline

- Finite element method for elliptic problems.
- Technical remarks on discretization, mesh adaptation, etc.
- Surprising advective term.
- Parabolic equations – time discretization, implicit and explicit schemes.
- Multiscale problems – multiscale finite element method.

Galerkin Method –Metod of Weighted Residuals

- We consider:

$$\begin{cases} -\nabla \cdot K \nabla u = f & \text{in } \Omega \\ u = g & \text{in } \Gamma \end{cases}$$

- Approximate solution is represented by a linear combination of *basis (shape, interpolation, trial)* functions:

$$\hat{u} = \varphi_0 + \sum_{i=1}^n \alpha_i \varphi_i$$

φ_i linearly independent, vanishing at Γ , φ_0 satisfies nonzero boundary conditions.

- Substitution of \hat{u} into equation produces a nonzero *residuum*.

$$-\nabla \cdot K \nabla \hat{u} - f = R$$

Galerkin Method –Metod of Weighted Residuals

Aim:

- Determine coefficients α_i to minimize R in a ‘certain sense’.
- Determine coefficients α_i such that

$$\int_{\Omega} R w_i d\underline{x} = 0 \text{ for } i=1, \dots, n, \quad w_i - \text{weight (test) function}$$

Galerkin Method (1915)

Take *basis* functions φ_i as *weight* functions to get

$$\int_{\Omega} R \varphi_i d\underline{x} = 0 \text{ for } i=1, \dots, n$$

Galerkin Method – Green's theorem

- Green's theorem

$$\int_{\Omega} -\nabla \cdot K \nabla \hat{u} \varphi_i \, d\underline{x} = \int_{\Omega} f \varphi_i \, d\underline{x} \quad , i=1, \dots, n$$

Green's theorem:

$$\int_{\Omega} -\Delta u \, w \, d\underline{x} = \int_{\Omega} \nabla u \cdot \nabla w \, d\underline{x} - \int_{\Gamma} \frac{\partial u}{\partial n} \cdot w \, d\gamma$$

Or:

$$\int_{\Omega} -\nabla \cdot \nabla u \, w \, d\underline{x} = \int_{\Omega} \nabla u \cdot \nabla w \, d\underline{x} - \int_{\Gamma} \frac{\partial u}{\partial n} \cdot w \, d\gamma$$

Or:

$$\int_{\Omega} -\operatorname{div} \cdot \operatorname{grad} u \, w \, d\underline{x} = \int_{\Omega} \operatorname{grad} u \cdot \operatorname{grad} w \, d\underline{x} - \int_{\Gamma} \frac{\partial u}{\partial n} \cdot w \, d\gamma$$

$$\int_{\Omega} \nabla \hat{u} K \nabla \varphi_i \, d\underline{x} = \int_{\Omega} f \varphi_i \, d\underline{x} - \int_{\Gamma} \frac{\partial \hat{u}}{\partial n} \varphi_i \, d\gamma \quad , i=1, \dots, n.$$

Galerkin Method – substitution

- Substitution of

$$\hat{u} = \varphi_0 + \sum_{i=1}^n \alpha_i \varphi_i \text{ into } \int_{\Omega} \nabla \hat{u} \nabla \varphi_i \, d\underline{x} = \int_{\Omega} f \varphi_i \, d\underline{x};$$

- ...Results in a system of linear equations.

$$\int_{\Omega} \nabla (\varphi_0 + \sum_{i=1}^n \alpha_i \varphi_i) \nabla \varphi_j \, d\underline{x} = \int_{\Omega} f \varphi_j \, d\underline{x}, \quad j=1, \dots, n$$

$$\sum_{i=1}^n \alpha_i \int_{\Omega} \nabla \varphi_i \nabla \varphi_j \, d\underline{x} = \int_{\Omega} f \varphi_j \, d\underline{x} - \int_{\Omega} \nabla \varphi_0 \nabla \varphi_j \, d\underline{x}, \quad j=1, \dots, n.$$

Galerkin Method – example

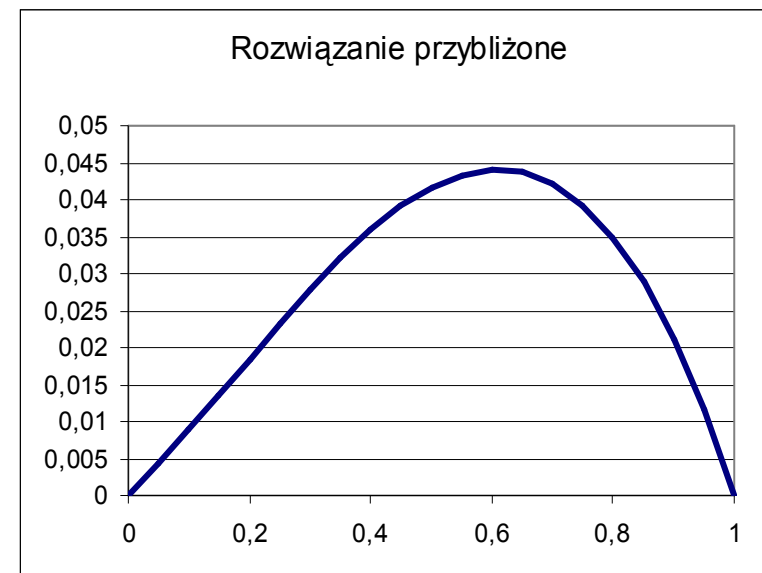
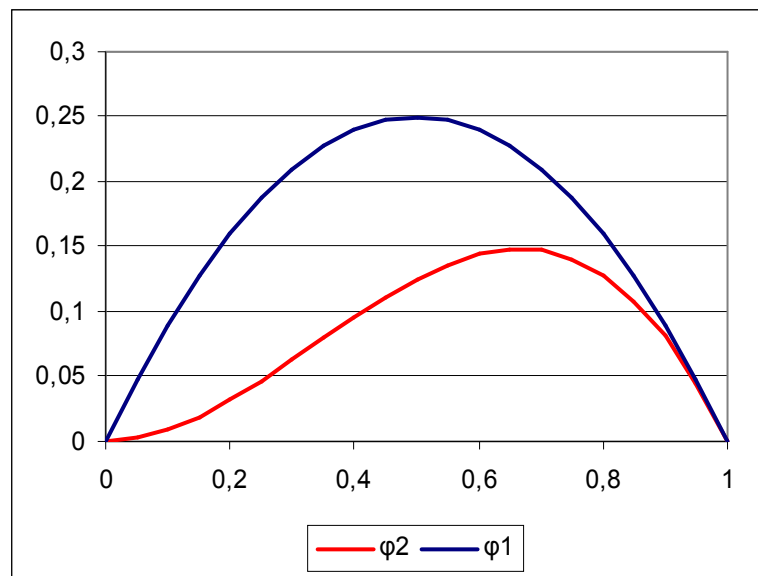
We consider:

$$y'' + y + x^2 = 0, \quad y(0) = y(1) = 0$$

As shape and test functions we take:

$$\varphi_1 = x(1-x), \quad \varphi_2 = x^2(1-x)$$

Approximated solution is represented as: $\bar{y} = C_1\varphi_1 + C_2\varphi_2$



Galerkin Method – summary

- Difficulty to apply over domains of complex geometries (test and basis functions should vanish at boundaries)
 - Difficulty to *exactly* match complex boundary conditions
-
- **These leads to the Finite Element formulation**

Finite Element Method

- Finite Element Method ([Courant 1943](#)) uses a special construction of spaces of approximated solutions.
- The construction of approximation spaces is based on a division of a computational domain into elements.
- Test functions are polynomials defined over elements.

Finite Element Method

- Original problem:

$$\begin{cases} -\nabla \cdot K \nabla u = f & \text{in } \Omega \\ u = g & \text{in } \Gamma \end{cases}$$

- **Weak formulation** (variational problem)

Find $u \in H_g^1(\Omega)$ such that

$$\int_{\Omega} \nabla u \cdot K \nabla \omega \, d\underline{x} = \int_{\Omega} f \omega \, d\underline{x}, \quad \forall \omega \in H_0^1(\Omega)$$

Spaces:

$$H_g^1(\Omega) = \{\omega : \Omega \rightarrow R, \quad \omega \in H^1(\Omega) \text{ i } \forall x \in \Gamma \omega(\underline{x}) = g(\underline{x})\}$$

where

$$H^1(\Omega) = \{\omega \in L^2(\Omega), \quad \frac{\partial \omega}{\partial x}, \frac{\partial \omega}{\partial y}, \dots \in L^2(\Omega)\}, \quad H_0^1(\Omega) = \{\omega : \Omega \rightarrow R, \quad \omega \in H^1(\Omega) \text{ i } \forall x \in \Gamma \omega(\underline{x}) = 0\}.$$

Finite Element Method – discretization /1/

- Introduce a sequence of finite-dimensional spaces $H_h^1(\Omega)$ (V_h) approximating $H^1(\Omega)$. Dependence of $H_h^1(\Omega)$ on a domain discretization, h being a measure of a partitioning size.

We denote:

$$H_{gh}^1(\Omega_h) = \{\omega_h : \Omega_h \rightarrow R, \quad \omega_h \in H_h^1(\Omega_h) \text{ and } \forall x \in \Gamma_h \quad \omega_h(\underline{x}) = g_h(\underline{x})\}$$

$$H_{0h}^1(\Omega_h) = \{\omega_h : \Omega_h \rightarrow R, \quad \omega_h \in H_h^1(\Omega_h) \text{ and } \forall x \in \Gamma_h \quad \omega_h(\underline{x}) = 0\}$$

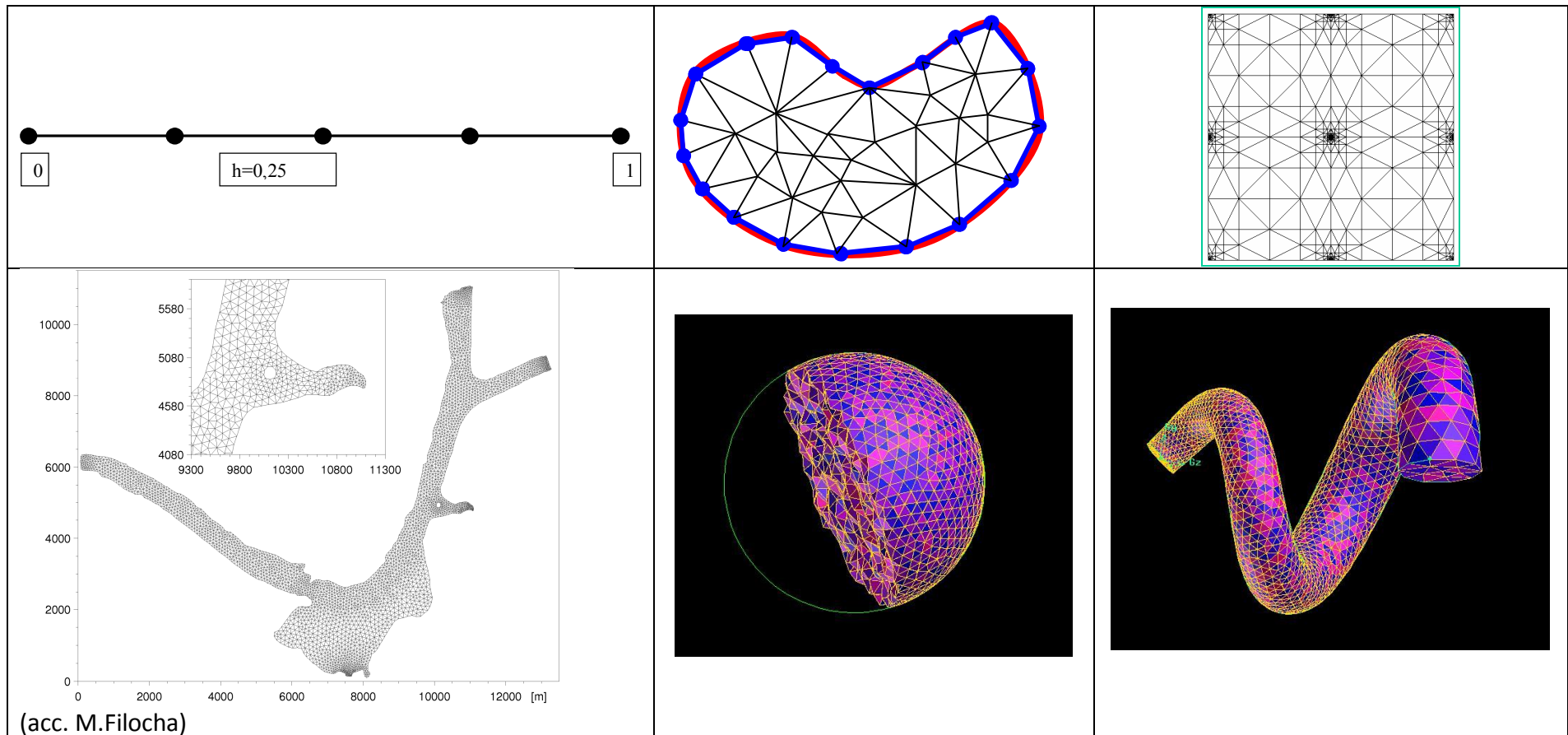
A space of approximate solutions is smaller than a space of the exact solution (internal approximation)

$$V_h \subset H^1(\Omega)$$

$$V_h \subset H^1(\Omega) \quad \Leftrightarrow \quad \omega_h \in V_h \text{ are continuous}$$

Finite Element Method – discretization /2/

- Construction is based on discretization (triangulation) T_h of a computational domain.



Finite Element Method – discretization /3/

- Linear approximation P1

$$H_h^1(\Omega_h) = \{\varphi_h : \Omega_h \rightarrow R, \varphi_h \text{ continuous on } \Omega_h \text{ and } \forall K \in T_h \varphi_h|_K \text{ linear}\}$$

(Piecewise linear functions)

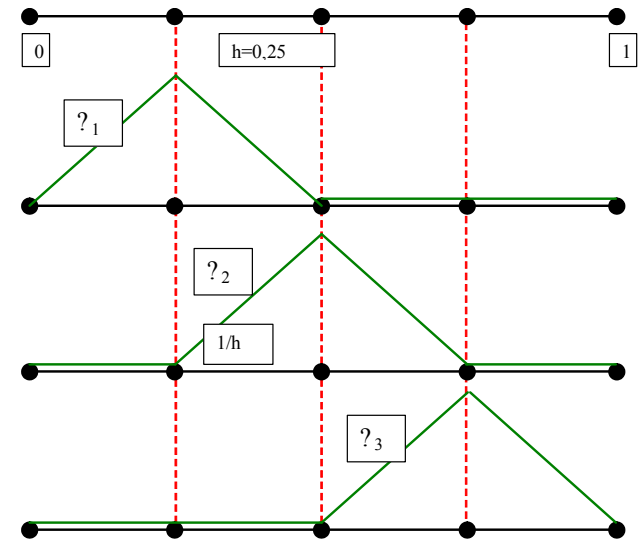
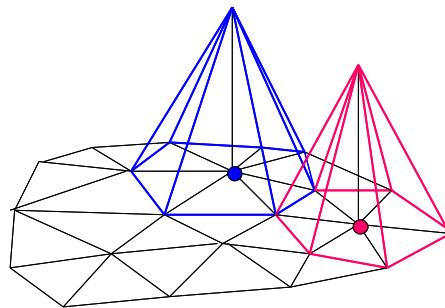
Any function in $H_h^1(\Omega_h)$ is uniquely defined by its values in nodes of discretization.

- Construction of basis in $H_h^1(\Omega_h)$

Basis function φ_i , connected to a node i of a triangulation T_h is:

- continuous function $\Omega_h \rightarrow R$
- linear on element
- for any $\varphi_i, \varphi_i(\underline{x}_j) = \delta_{ij} = \begin{cases} 1 & , i = j \\ 0 & , i \neq j \end{cases}, \underline{x}_j - \text{node.}$

$$\sum_{i=1}^n \varphi_i = 1 \text{ for } \forall \underline{x} \in \Omega_h$$

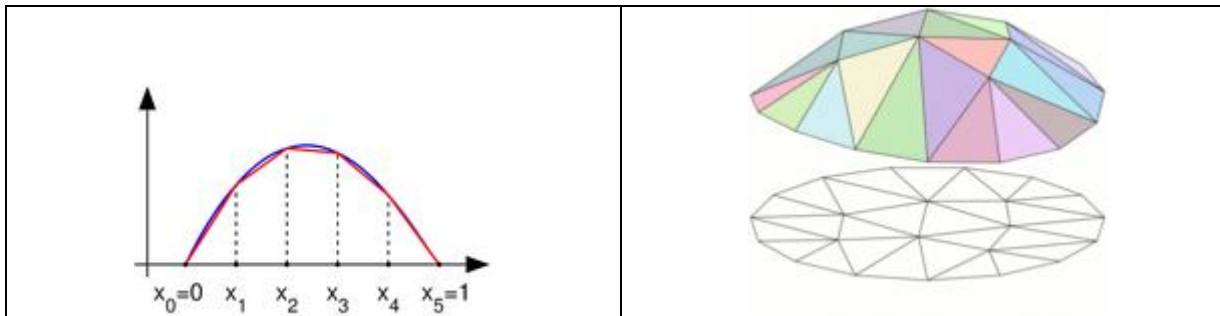


Finite Element Method – discretization /4/

Approximated solution $u_h \in H_{gh}^1(\Omega_h)$ is represented as a linear combination of basis functions:

$$u_h = \sum_{i=1}^{n_i} \alpha_i \varphi_i(\underline{x}) + \sum_{i=n_i+1}^{n_i} g(\underline{x}_i) \varphi_i(\underline{x}).$$

(a second term due to non-zero Dirichlet conditions)



Finite Element Method – towards a system of linear equations

- A weak formulation in approximation spaces

Find $u_h \in H_{gh}^1(\Omega_h)$ such that

$$\int_{\Omega} \nabla u_h \cdot K \nabla \omega_h \, d\underline{x} = \int_{\Omega} f \omega_h \, d\underline{x} \quad \forall \omega_h \in H_{oh}^1(\Omega_h)$$

- Substitution of representation of u_h

$$u_h = \sum_{i=1}^{n_i} \alpha_i \varphi_i(\underline{x}) + \sum_{i=n_i+1}^{n_t} g(\underline{x}_i) \varphi_i(\underline{x}).$$

- Following Galerkin method we set $\omega_h = \varphi_i$, $i = 1, \dots, n$

$$\int_{\Omega} \nabla \left(\sum_{j=1}^{n_i} \alpha_j \varphi_j(\underline{x}) + \sum_{j=n_i+1}^{n_t} g(\underline{x}_j) \varphi_j(\underline{x}) \right) \cdot K \nabla \varphi_i(\underline{x}) \, d\underline{x} = \int_{\Omega} f(\underline{x}) \varphi_i(\underline{x}) \, d\underline{x}, \quad i = 1, \dots, n_i$$

- Finally:

$$\sum_{j=1}^{n_i} \alpha_j \int_{\Omega} \nabla \varphi_j(\underline{x}) \cdot K \nabla \varphi_i(\underline{x}) \, d\underline{x} = \int_{\Omega} f(\underline{x}) \varphi_i(\underline{x}) \, d\underline{x} - \sum_{j=n_i+1}^{n_t} g(\underline{x}_j) \int_{\Omega} \nabla \varphi_j(\underline{x}) \cdot K \nabla \varphi_i(\underline{x}) \, d\underline{x}, \quad i = 1, \dots, n_i$$

It is a system of linear equations $\mathbf{A} \boldsymbol{\alpha} = \mathbf{F}$ with unknowns α_i , $i = 1, \dots, n_i$.

Forming this system is called *assembling*.

Finite Element Method – a system of linear equations

Matrix elements:

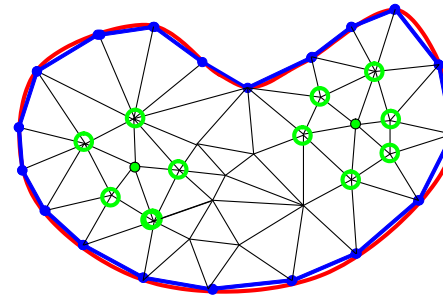
$$\mathbf{A}_{ij} = \int_{\Omega} \nabla \varphi_i(\underline{x}) \cdot K \nabla \varphi_j(\underline{x}) d\underline{x}, \quad i, j = 1, \dots, n_i$$

RHS vector:

$$\mathbf{F}_i = \int_{\Omega} f(\underline{x}) \varphi_i(\underline{x}) d\underline{x} - \sum_{j=n_i+1}^{n_i} g(\underline{x}_j) \int_{\Omega} \nabla \varphi_j(\underline{x}) \cdot K \nabla \varphi_i(\underline{x}) d\underline{x}$$

Computational advantages of \mathbf{A} :

- \mathbf{A} is **positive definite**, in particular it is not singular, what guarantees a uniqueness of a solution of a weak problem.
- \mathbf{A} is **symmetric**. There exists many robust methods for solving systems of symmetric linear equations (conjugate gradients).
- \mathbf{A} is **sparse**.

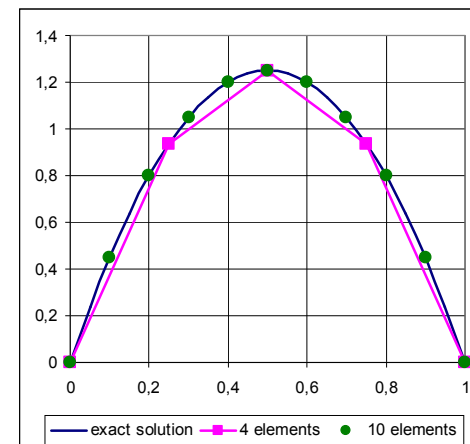
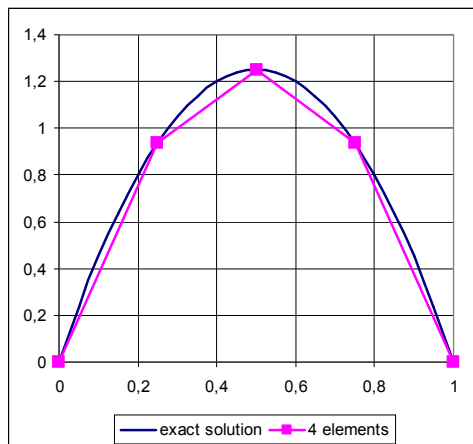
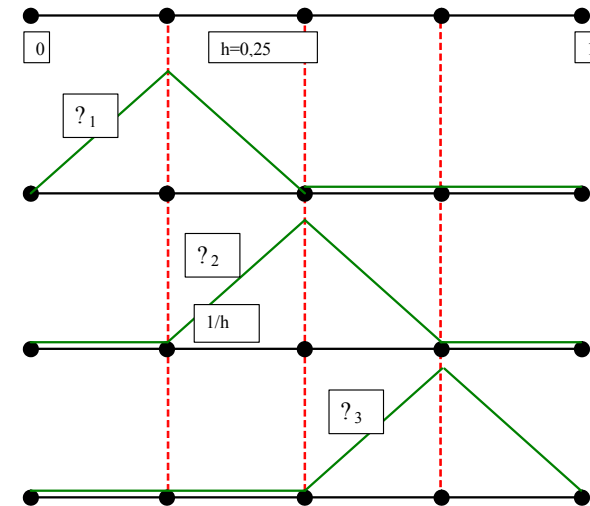


Finite Element Method – a simple example

$$-\frac{d^2u}{dx^2} = 10 \quad \text{in } (0,1)$$

$$u(0) = u(1) = 0$$

Exact solution: $u(x) = -5x^2 + 5x$



Finite Element Method – remarks

- For practical problems matrices are large (3D!) and sparse, thanks to a special construction of basis and test functions.
- Linear basis are the most often used, however it is possible to apply elements of higher orders
- Piecewise constant derivatives, what may cause problems while solving transport equation (mass conservation does not hold within an element).
- Mixed methods (or, mixed-hybrid), give mass-conservative flux approximations
- (Adaptive) mesh refinement
- Growing complexity of a problem may be due to: rapidly varying parameters, tensorial form of parameter, nonlinearities,, convective term (to be continued).

Solving systems of linear equations

Solving systems of linear equations is a 'bottle-neck'.

- **Direct methods:**

Gauss elimination, Cholesky's method – successive elimination of unknowns. Lead to exact solution (up to roundoff errors). Band matrices, bandwidth depends on numbering of nodes.

- **Iterative methods:**

an approximate solution is generated, meeting a given convergence criterion. Advantage: sparse structure is easily taken into account.

Sparse matrices representation based on *Compressed Row Storage*

Classification:

- **decomposition methods (classical approach)** (Jacobi method, Gauss-Seidel method, SOR method)
- **projection methods** (conjugate gradient method, Arnoldi method, GMRES method)

Preconditioning – to improve a system's conditioning. A strong relationship between quality of discretization and the conditioning of a matrix.

J. Shewchuk, An Introduction to the Conjugate Gradient Method Without the Agonizing Pain

<http://www.cs.cmu.edu/~quake-papers/painless-conjugate-gradient.pdf>

J. Shewchuk, What is a Good Linear Finite Element? Interpolation, Conditioning, Anisotropy, and Quality Measures,

<http://www.cs.berkeley.edu/~jrs/papers/elemj.pdf>

Nonsymmetric PDEs

Nonsymmetric partial differential equations

Advection-diffusion equation:

$$(*) \quad -\nabla \cdot (k \nabla u) + \underline{\beta} \cdot \nabla u + a_0 u = f \quad \text{in } \Omega$$

with a (general) boundary condition

$$K \frac{\partial u}{\partial n} + a_1 u = g \quad \text{on } \Gamma .$$

$k(x)$ – diffusion coefficient;

$$\underline{\beta} = (\beta_1, \beta_2, \beta_3) \in (L^\infty(\Omega))^3$$

$a_0 \in L^\infty(\Omega)$ - scalar function

$$f \in L^2(\Omega)$$

a_1, g - defined over Γ ; $a_1 \in L^\infty(\Gamma)$, $g \in L^2(\Gamma)$.

Boundary condition is defined depending on velocity field:

$$u = g_- \quad \text{on } \Gamma_-$$

$$u = g_+ \quad (\text{or } \partial_n u = g_+) \quad \text{on } \Gamma_+ .$$

Γ_- inflow boundary, Γ_+ outflow boundary: $\Gamma_- = \{\underline{x} \in \Gamma : \underline{\beta}(\underline{x}) \cdot \underline{n}(\underline{x}) < 0\}$, $\Gamma_+ = \{\underline{x} \in \Gamma : \underline{\beta}(\underline{x}) \cdot \underline{n}(\underline{x}) \geq 0\}$.

Nonsymmetric PDEs

Variational formulation: (in this example ε stands for diffusion k)

Find $u \in H^1(\Omega)$ such that:

$$\int_{\Omega} (\varepsilon \nabla u)^T \cdot \nabla \omega \, d\underline{x} + \int_{\Omega} (\beta \cdot \nabla u) \omega \, d\underline{x} = \int_{\Omega} 0 \, \omega \, d\underline{x} \quad \forall \omega \in H^1(\Omega)$$

If:

- we apply elements P1 (piecewise linear),
- use equally distributed nodes, which are numbered in a ‘natural way’
- then we get a system of linear equations

$$A \xi = F,$$

matrix A:

$$\begin{bmatrix} \frac{2\varepsilon}{h} & -\frac{\varepsilon}{h} + \frac{1}{2} & 0 & 0 \\ -\frac{\varepsilon}{h} - \frac{1}{2} & \frac{2\varepsilon}{h} & -\frac{\varepsilon}{h} + \frac{1}{2} & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & -\frac{\varepsilon}{h} - \frac{1}{2} & \frac{2\varepsilon}{h} \end{bmatrix},$$

h denotes size of an element.

Matrix A is not symmetric !!! A smaller class of methods to solve a system of linear equations (CG methods do not apply).

Nonsymmetric PDEs

We consider:

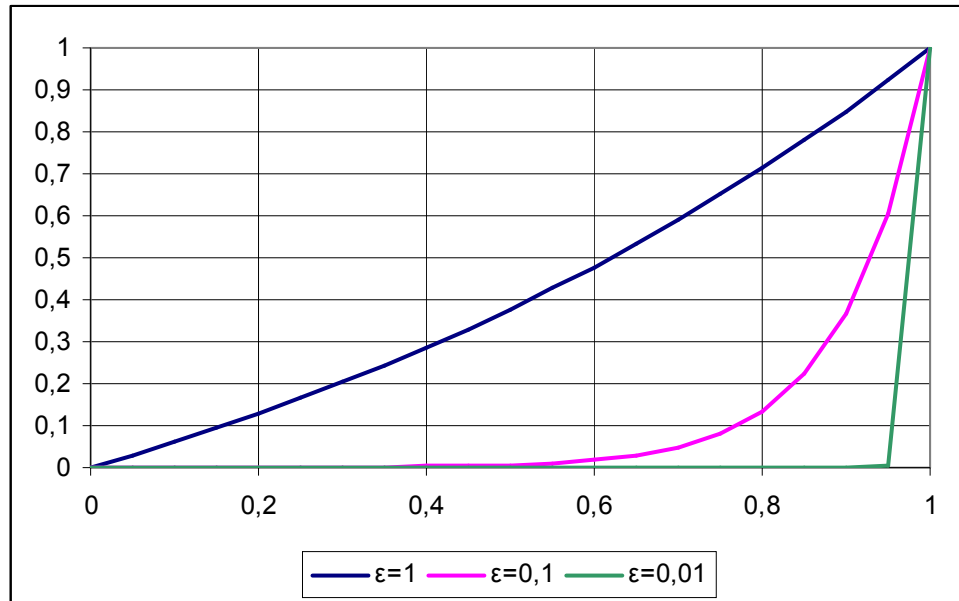
$$(*) \quad -\varepsilon \frac{d^2 u}{dx^2} + \frac{du}{dx} = 0 \quad w(0, 1)$$
$$u(0) = 0, \quad u(1) = 1$$

Elliptic equation, advective term of a general form $\underline{\beta} \cdot \nabla u$; here $\beta = 1$.

Exact solution:

$$u(x) = \frac{e^{x/\varepsilon} - 1}{e^{1/\varepsilon} - 1}$$

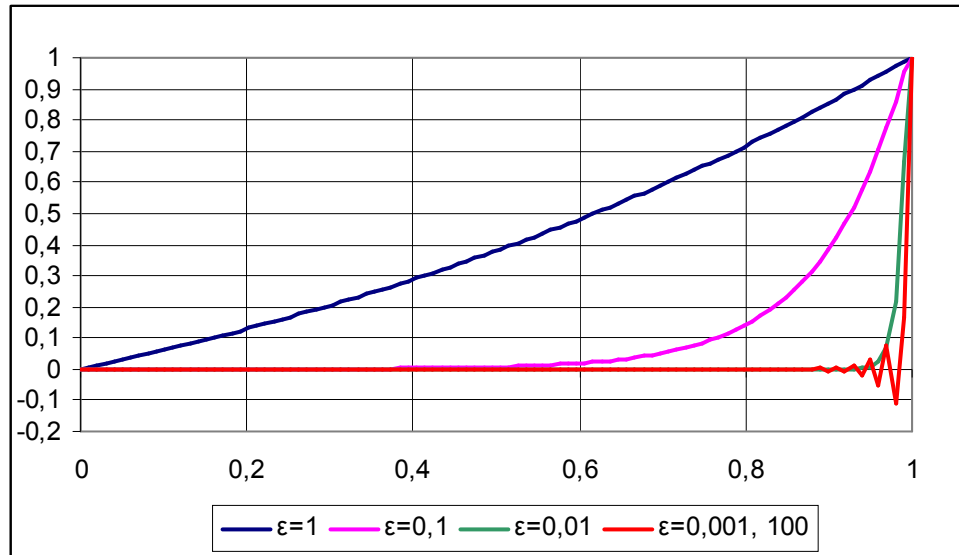
Nonsymmetric PDEs - example



Boundary layer ε/β .

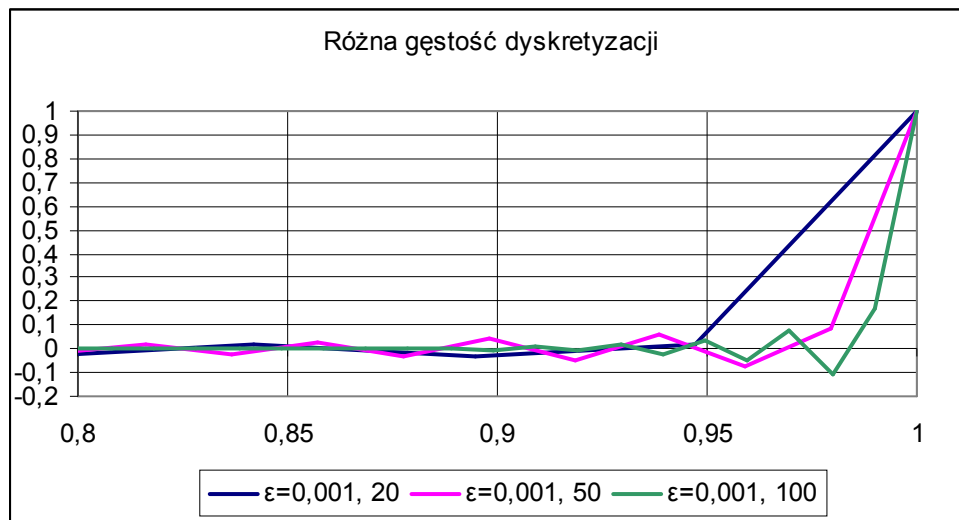
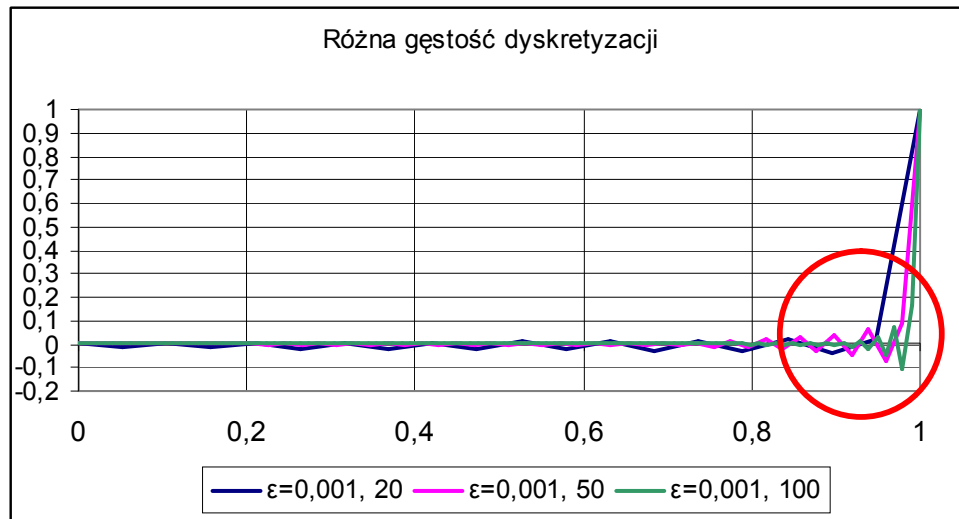
We solve the equation with the standard FEM ... to obtain:

Nonsymmetric PDEs



Oscillations appear for $\epsilon \ll \beta$ ($\epsilon \ll 1$ in our case).

Nonsymmetric PDEs



Nonsymmetric PDEs

Along with decreasing ε , the equation ‘almost’ degenerates to a first order equation, Dirichlet boundary condition in $x=1$ becomes inadequate. Numerical solution becomes ‘polluted’ with this boundary condition, moreover, this pollution propagates.

Remark:

No oscillations appear when Neumann BC is imposed in $x=1$.

Oscillations appear when $\frac{\beta h}{\varepsilon} \geq 2$, where h denotes element’s size.

$\frac{\beta h}{\varepsilon}$ cell Reynolds number.

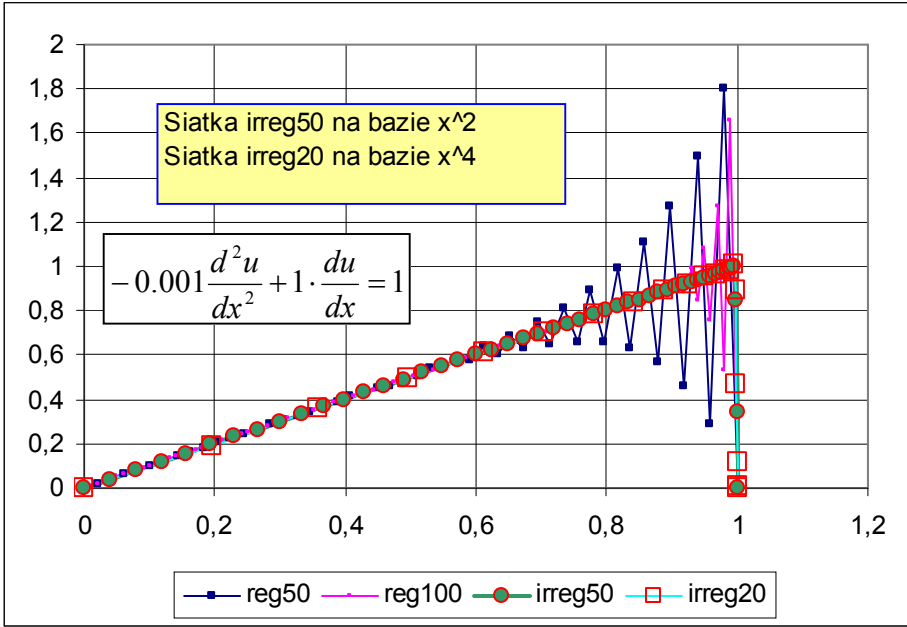
Nonsymmetric PDEs

Fighting with oscillations

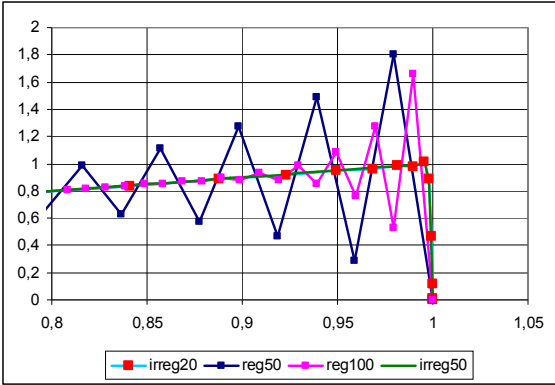
- Increasing number of nodes, to get $\frac{\beta h}{\varepsilon} < 2$.
- Local grid refinement in vicinity of fronts.

In our example we expect that mesh refinement close to $x=1$ should improve the approximate solution.

Nonsymmetric PDEs



Details:



Nonsymmetric PDEs

- **Upwind methods –Petrov-Galerkin method**

The idea is to:

destroy symmetry of test functions in a way to 'promote' a flow direction.

Petrov-Galerkin approximation for our example reads:

Find $u_h \in H_h^1(\Omega)$ such that

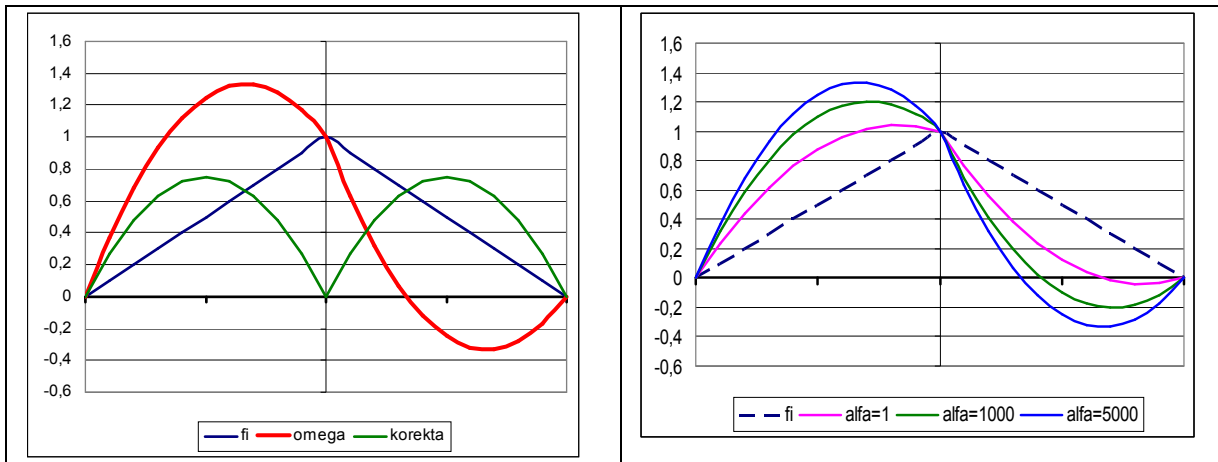
$$\int_{\Omega} \varepsilon \frac{du_h}{dx} \frac{d\omega_h}{dx} dx + \int_{\Omega} \frac{du_h}{dx} \omega_h dx = \int_{\Omega} 1 \cdot \omega_h dx \text{ for every } \omega_h \in \widehat{H}_h^1(\Omega)$$

Spaces $H_h^1(\Omega)$ and $\widehat{H}_h^1(\Omega)$ are different, but both belong to $H^1(\Omega)$.

Nonsymmetric PDEs

Construction of spaces $H_h^1(\Omega)$ and $\widehat{H}_h^1(\Omega)$

- Functions $\varphi_i \in H_h^1(\Omega)$ are standard 'hat' functions.
- $\omega_i \in \widehat{H}_h^1(\Omega)$: φ_i +/- quadratic function (vanishing in nodes):

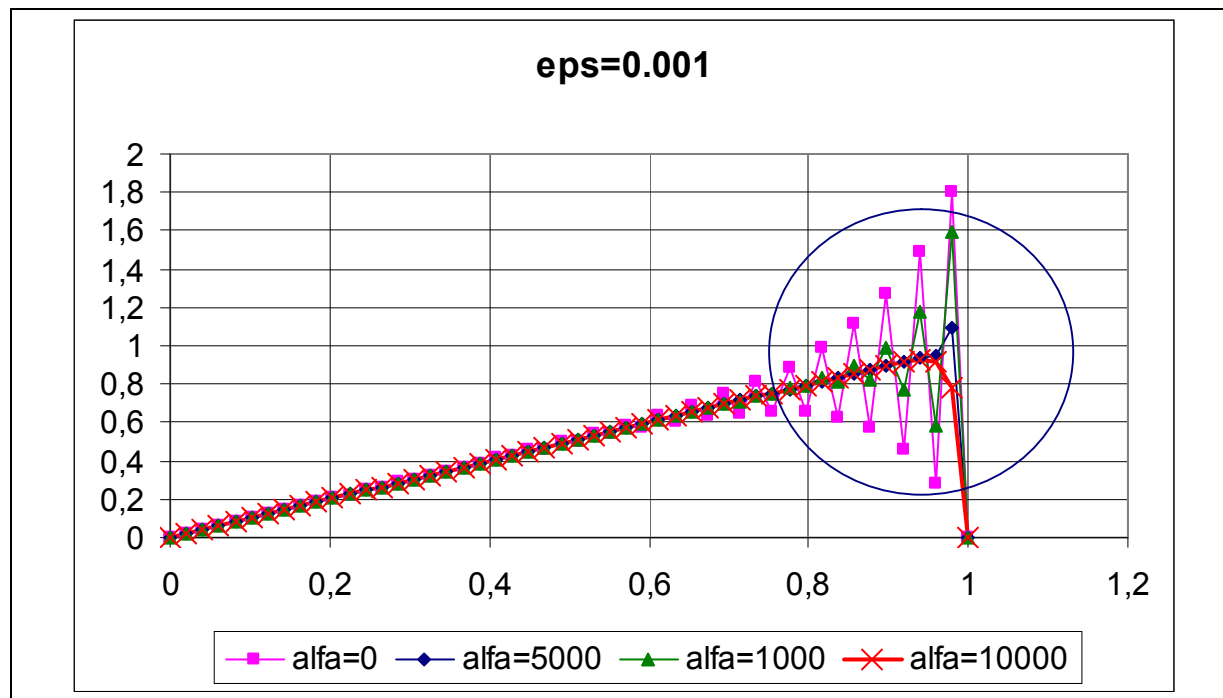


$$\omega_i(x) = \begin{cases} \varphi_i(x) + \frac{\alpha}{h} N\left(\frac{x - x_{i-1}}{h}\right) \\ \varphi_i(x) - \frac{\alpha}{h} N\left(\frac{x_{i+1} - x}{h}\right) \end{cases} \text{ if } \begin{cases} x_{i-1} \leq x \leq x_i \\ x_i \leq x < x_{i+1} \end{cases}, \quad N(\xi) = 3\xi(1-\xi).$$

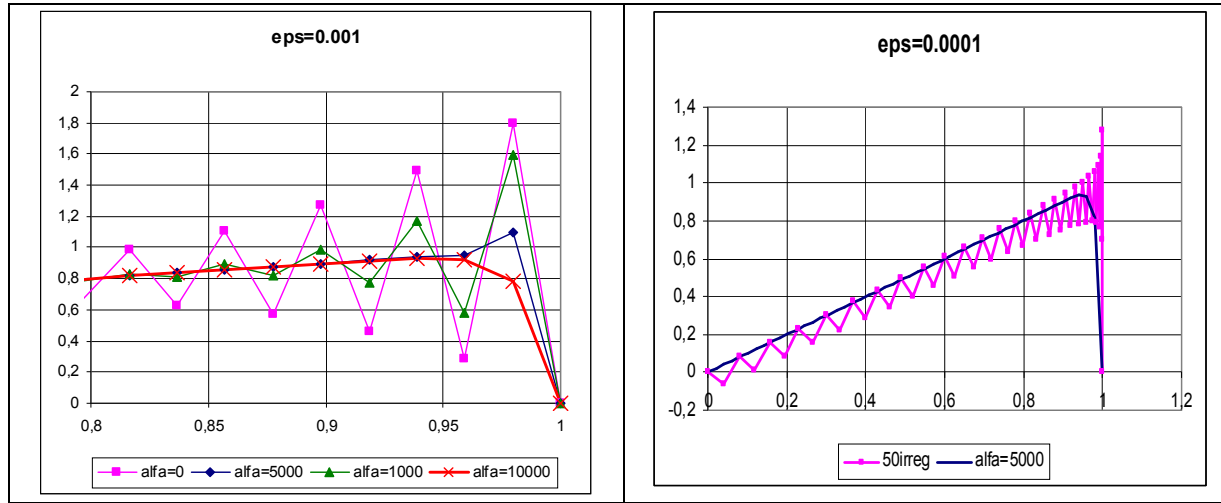
Nonsymmetric PDEs

We get an *upwind* orientation of test functions.

Results obtained with a Petrov-Galerkin approximation:



Nonsymmetric PDEs



Price for improvement:

Artificial diffusion (numerical viscosity) influences a solution.

Nonsymmetric PDEs – summary

Discretization of a convective term $\underline{\beta} \cdot \nabla u$

- nonsymmetric system of linear equations;
- nonphysical *oscillations*, or
- *numerical diffusion* causing *smearing out* sharp fronts.
- Attention when velocity field is obtained computationally by numerical differentiation.....

Parabolic problems

We consider:

$$\frac{\partial u}{\partial t} - \nabla \cdot (K(x, y, z) \nabla u) = f, \text{ in } \Omega \times (0, T]$$

boundary conditions:

$$u = 0 \text{ on } \Gamma_1 \times (0, T], \quad K \frac{\partial u}{\partial n} = 0 \text{ on } \Gamma_2 \times (0, T]$$

initial condition:

$$u(\underline{x}, 0) = u^0(\underline{x}), \quad \underline{x} \in \Omega.$$

Discretization consists of two steps:

- [Space discretization](#) (Finite Element Method).
- [Time discretization](#) with finite differences schemes.

-

Parabolic problems – space discretization

Step 1: towards a semi-discrete scheme:

We follow FEM steps:

Variational problem:

Find $u \in V$, $t \in (0, T]$ such that

$$u(\underline{x}, 0) = u^0(\underline{x})$$

$$\int_{\Omega} \frac{\partial u(\underline{x}, t)}{\partial t} v(\underline{x}) d\underline{x} + \int_{\Omega} \nabla u(\underline{x}, t) K \nabla v(\underline{x}) d\underline{x} = \int_{\Omega} f(\underline{x}, t) v(\underline{x}) d\underline{x} \quad \text{for any } \forall v \in H_0^1(\Omega)$$

Approximate solution belongs to $V_h \subset V$, with a basis $\{\varphi_1, \varphi_2, \dots, \varphi_n\}$.

We use P1 (piecewise constant) elements.

Find $u_h \in V_h$, $t \in (0, T]$ such that:

$$\int_{\Omega_h} \frac{\partial u_h(\underline{x}, t)}{\partial t} v(\underline{x}) d\underline{x} + \int_{\Omega_h} \nabla u_h(\underline{x}, t) K \nabla v(\underline{x}) d\underline{x} = \int_{\Omega_h} f(\underline{x}, t) v(\underline{x}) d\underline{x} \quad \text{for any } \forall v \in V_h(\Omega)$$

$$\int_{\Omega_h} u_h(\underline{x}, 0) v(\underline{x}) d\underline{x} = \int_{\Omega_h} u^0(\underline{x}) v(\underline{x}) d\underline{x}$$

Parabolic problems – space discretization

$$u_h(\underline{x}, t) = \sum_{i=1}^n \alpha_i(t) \varphi_i(\underline{x})$$

Coefficients α_i depends on t , $\alpha_i = \alpha_i(t)$.

Substitution to variational formulation, taking $\varphi_1, \varphi_2, \dots, \varphi_n$ as $v \in V_h(\Omega)$:

$$\int_{\Omega_h} \frac{\partial \left(\sum_{i=1}^n \alpha_i(t) \varphi_i(\underline{x}) \right)}{\partial t} \varphi_j(\underline{x}) d\underline{x} + \int_{\Omega_h} \nabla \left(\sum_{i=1}^n \alpha_i(t) \varphi_i(\underline{x}) \right) K \nabla \varphi_j(\underline{x}) d\underline{x} = \int_{\Omega_h} f(\underline{x}, t) \varphi_j(\underline{x}) d\underline{x}$$
$$\int_{\Omega_h} \left(\sum_{i=1}^n \alpha_i(0) \varphi_i(\underline{x}) \right) \varphi_j(\underline{x}) d\underline{x} = \int_{\Omega_h} u^0(\underline{x}) \varphi_j(\underline{x}) d\underline{x} \quad j = 1, \dots, n$$

Finally:

$$\sum_{i=1}^n \frac{d\alpha_i(t)}{dt} \int_{\Omega_h} \varphi_i(\underline{x}) \varphi_j(\underline{x}) d\underline{x} + \sum_{i=1}^n \alpha_i(t) \int_{\Omega_h} \nabla \varphi_i(\underline{x}) K \nabla \varphi_j(\underline{x}) d\underline{x} = \int_{\Omega_h} f(\underline{x}, t) \varphi_j(\underline{x}) d\underline{x}$$
$$\sum_{i=1}^n \alpha_i(0) \int_{\Omega_h} \varphi_i(\underline{x}) \varphi_j(\underline{x}) d\underline{x} = \int_{\Omega_h} u^0(\underline{x}) \varphi_j(\underline{x}) d\underline{x} \quad j = 1, \dots, n$$

Parabolic problems – time discretization

In a matrix form:

$$B \frac{d\alpha(t)}{dt} + A\alpha(t) = F(t), \quad t \in (0, T)$$

$$B\alpha(0) = U^0$$

$$B_{ij} = \int_{\Omega} \varphi_i \varphi_j \, d\underline{x} \quad (\text{mass matrix})$$

$$A_{ij} = \int_{\Omega} \nabla \varphi_i \cdot K \nabla \varphi_j \, d\underline{x} \quad (\text{stiffness matrix})$$

$$F_i(t) = \int_{\Omega} f(x, t) \varphi_i \, d\underline{x}$$

A and B symmetric.

Step 2: Discretization in time

Derivative in time is replaced by a finite difference:

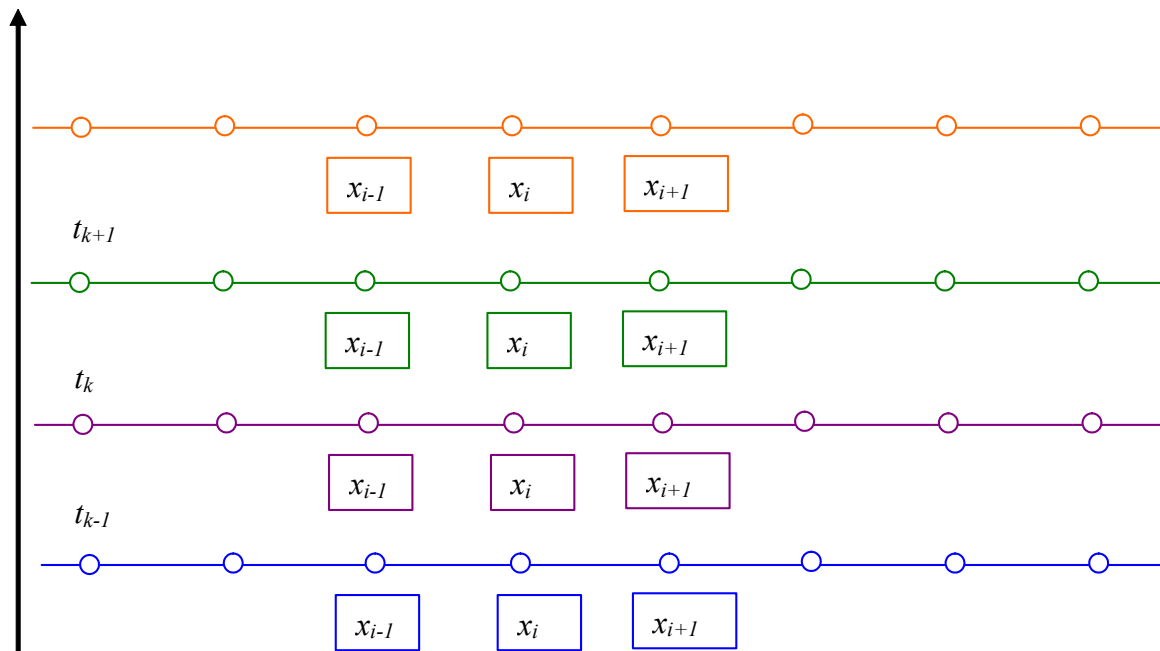
$$\frac{d\alpha}{dt} \cong \frac{\alpha^{n+1} - \alpha^n}{\Delta t}$$

Parabolic problems – time discretization

$$B \frac{\alpha^{n+1} - \alpha^n}{\Delta t} + A\alpha^{n+1} = F^{n+1}$$

Δt - time step.

α^n are known, α^{n+1} - *still* unknown. Depending on ??? we get *implicit* or *explicit* schemes.



Parabolic problems – time discretization

- Euler's scheme (explicit)

??? refers to a former time step:

$$B \frac{\alpha^{n+1} - \alpha^n}{\Delta t} + A\alpha^n = F^n,$$

Linear system to solve is:

$$B\alpha^{n+1} = (B - \Delta t A) \alpha^n + \Delta t F^n$$

Remark:

The scheme is *conditionally stable*, time step Δt must be sufficiently small:

$$\Delta t \leq Ch^2$$

- Euler's scheme (implicit)

??? refers to a current time step:

$$B \frac{\alpha^{n+1} - \alpha^n}{\Delta t} + A\alpha^{n+1} = F^{n+1},$$

Unconditionally stable

Parabolic problems – time discretization

- Crank-Nicholson scheme

A combination of explicit and implicit schemes:

$$B \frac{\alpha^{n+1} - \alpha^n}{\Delta t} + A \left(\frac{1}{2} \alpha^n + \frac{1}{2} \alpha^{n+1} \right) = \frac{1}{2} (F^n + F^{n+1})$$

$$\left(B + \frac{1}{2} \Delta t A \right) \alpha^{n+1} = \left(B - \frac{1}{2} \Delta t A \right) \alpha^n + \frac{1}{2} \Delta t (F^n + F^{n+1}).$$

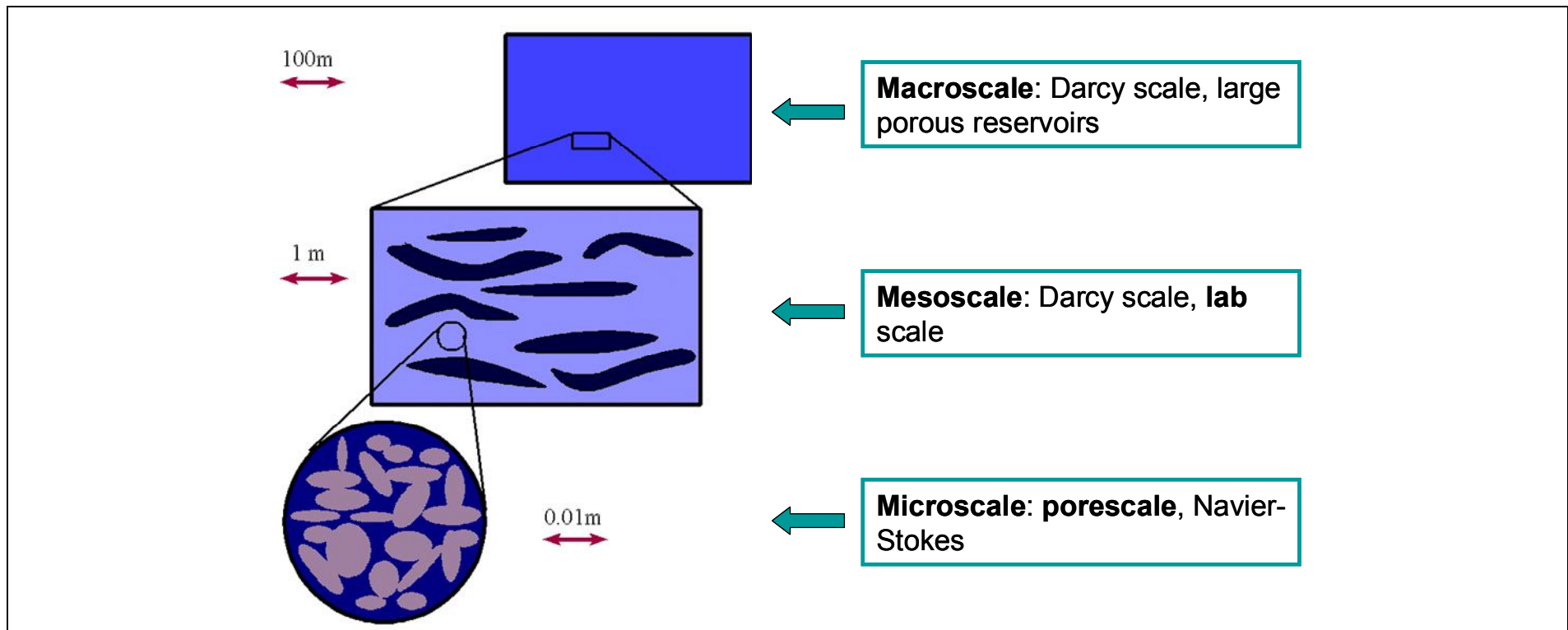
Remarks:

Unconditionally stable. Discretization error is $O((\Delta t)^2)$.

Multiscale approach

- Multiple scale problems: major difficulty of direct solutions is the size of the computation;
- The aim is to develop a method that captures the small-scale effects on the large scales, but does not require resolving all small-scale features;
- The practical need to coarsening the model. But it is important to preserve multiscale features of physical processes across scales;
-

Multiscale approach



Multiscale approach

We consider:

$$\{-\nabla \cdot K(x) \nabla u = f \quad \text{in } \Omega + \text{boundary conditions}$$

$K(x)$ – spatial field varying over multiple scales

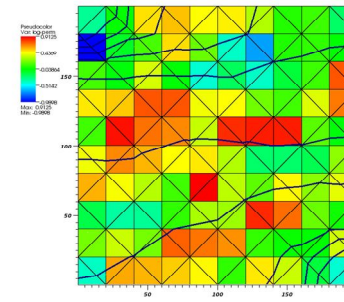
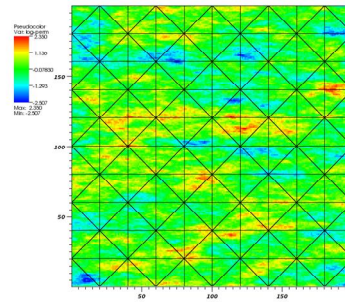
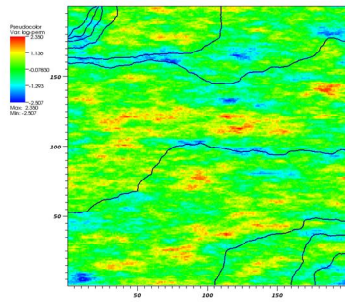
In porous media applications $K(x)$ represents *permeability*.

Computational Multiscale Methods

- Upscaling and downscaling
- Multiscale methods

Upscaling concept

- Computation of coarse-scale quantities
- Coupling of multiscale parameters into coarse-scale problems
- Simulation results (at coarse-scale)



A domain with fine-scale permeability field defined...

...is covered with a coarse mesh, and

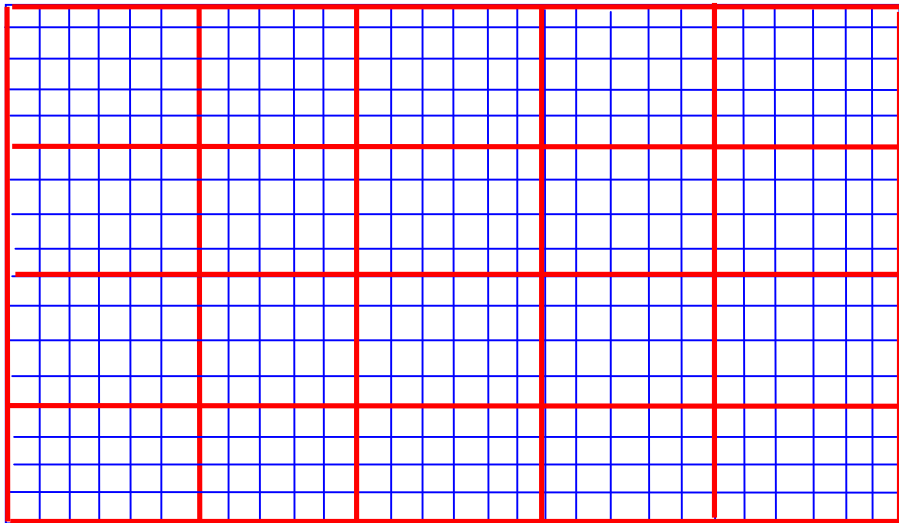
...fine-scale permeabilities are replaced with 'upscaled' values (effective permeabilities)

Multiscale Finite Element Method

- MFEM attempts to capture the multiscale structure of the solution via special basis functions.
- Basis functions contain essential multiscale information.

Two main ingredients of MFEM:

- construction of basis functions
- global formulation at a coarse-scale using multiscale basis functions.



T_h – partition of Ω (coarse grid)

φ_i^0 - standard basis functions on T_h , $S_i = \text{supp } \varphi_i^0$

Multiscale basis function is defined as a solution of

$$\begin{cases} -\nabla \cdot K(x) \nabla \varphi_i = 0 & \text{in } K \\ \varphi_i = \varphi_i^0 & \text{on } \partial K, \forall K \in T_h, K \subset S_i \end{cases}$$

φ_i are oscillatory in the interior of coarse-grid block

Further reading

Finite Element Method

C.A.J. Fletcher, Computational Galerkin Methods, Springer-Verlag, 1984

K. Eriksson, D. Estep, P. Hansbo, C. Johnson, Computational Differential Equations, Cambridge University Press, 1996

B. Lucquin, O. Pironneau, Introduction to Scientific Computing, John Wiley & Sons, 1998

Multiscale finite element methods

Y. Efendiev, T. Hou, Multiscale finite element methods. Theory and applications. Surveys and tutorials in the Applied Mathematical Sciences 4, Springer, 2009.