Scientific Computing WS 2019/2020

Lecture 15

Jürgen Fuhrmann

juergen.fuhrmann@wias-berlin.de

Recap: the finite Volume method

Finite volumes: motivation

Regard stationary second order PDE with Robin boundary conditions as a system of two first order equations in a Lipschitz domain Ω:

$$
\begin{aligned}
\nabla \cdot \vec{j} &= f && \text{continuity equation in } \Omega\\
\vec{j} &= -\delta \vec{\nabla} u && \text{flux law in } \Omega\\
\vec{j} \cdot \vec{n} &= \alpha u - g && \text{on } \Gamma
\end{aligned}
$$

- Derivation of the continuity equation was based on the consideration of species balances of an representative elementary volume (REV)
- Why not just subdivide the computational domain into a finite number of REV's ?
  - Assign a value of u to each REV
  - Approximate $\vec{\nabla} u$ by finite differece of u values in neigboring REVs
  - ... call REVs "control volumes" or "finite volumes"

Constructing control volumes I

Assume $\Omega \subset \mathbb{R}^d$ is a polygonal domain such that $\partial\Omega = \bigcup_{m \in \mathcal{G}} \Gamma_m$, where Γ_m are planar such that $\vec{n}|_{\Gamma_m} = \vec{n}_m$.

Subdivide Ω into into a finite number of **control volumes** $\bar{\Omega} = \bigcup_{k \in \mathcal{N}} \bar{\omega}_k$ such that

- ω_k are open convex domains such that $\omega_k \cap \omega_l = \emptyset$ if $\omega_k \neq \omega_l$
- $\sigma_{kl} = \bar{\omega}_k \cap \bar{\omega}_l$ are either empty, points or straight lines.
  If $|\sigma_{kl}| > 0$ we say that ω_k, ω_l are neighbours.
- $\vec{\nu}_{kl} \perp \sigma_{kl}$: normal of $\partial\omega$ at σ_{kl}
- $\mathcal{N}_k = \{l \in \mathcal{N} : |\sigma_{kl}| > 0\}$: set of neighbours of ω_k
- $\gamma_{km} = \partial\omega_k \cap \Gamma_m$: domain boundary part of $\partial\omega_k$
- $\mathcal{G}_k = \{m \in \mathcal{G} : |\gamma_{km}| > 0\}$: set of non-empty boundary parts of ω_k.

$\Rightarrow \partial\omega_k = \left(\cup_{l \in \mathcal{N}_k} \sigma_{kl}\right) \bigcup \left(\cup_{m \in \mathcal{G}_k} \gamma_{km}\right)$

Constructing control volumes II

To each control volume ω_k assign a **collocation point**: $\vec{x}_k \in \bar{\omega}_k$ such that

- **Admissibility condition**:
  if $l \in \mathcal{N}_k$ then the line $\vec{x}_k\vec{x}_l$ is orthogonal to σ_{kl}
  - For a given function $u : \Omega \to \mathbb{R}$ this will allow to associate its value $u_k = u(\vec{x}_k)$ as the value of an unknown at $\vec{x}_k$.
  - For two neigboring control volumes ω_k, ω_l , this will allow to approximate $\vec{\nabla} u \cdot \vec{\nu}_{kl} \approx \frac{u_l - u_k}{h}$
- **Placement of boundary unknowns at the boundary**:
  if ω_k is situated at the boundary, i.e. for $|\partial\omega_k \cap \partial\Omega| > 0$, then $\vec{x}_k \in \partial\Omega$
  - This will allow to apply boundary conditions in a direct manner

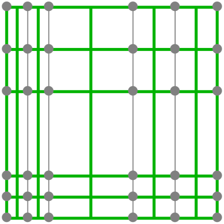
Constructing Control Volumes in 1D

Let $\Omega = (a, b)$ be subdivided into intervals by $x_1 = a < x_2 < x_3 < \cdots < x_{n-1} < x_n = b$. Then we set

$$\omega_k = \begin{cases} \left(x_1, \frac{x_1+x_2}{2}\right), & k = 1 \\ \left(\frac{x_{k-1}+x_k}{2}, \frac{x_k+x_{k+1}}{2}\right), & 1 < k < n \\ \left(\frac{x_{n-1}+x_n}{2}, x_n\right), & k = n \end{cases}$$

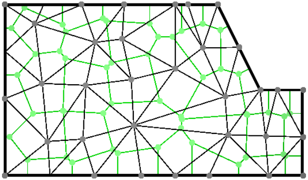
Control Volumes for a 2D tensor product mesh

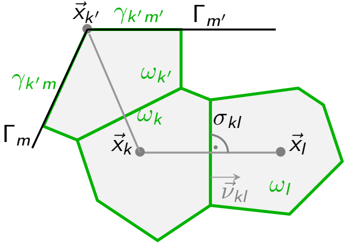
- Let $\Omega = (a, b) \times (c, d) \subset \mathbb{R}^2$.
- Assume subdivisions $x_1 = a < x_2 < x_3 < \dots < x_{n-1} < x_n = b$ and $y_1 = c < y_2 < y_3 < \dots < y_{n-1} < y_n = d$
- $\Rightarrow$ 1D control volumes ω_k^x and ω_k^y
- Set $\vec{x}_{kl} = (x_k, y_l)$ and $\omega_{kl} = \omega_k^x \times \omega_l^y$.



- Gray: original grid lines and points
- Green: boundaries of control volumes

Control volumes on boundary conforming Delaunay mesh





- Obtain a boundary conforming Delaunay triangulation with vertices $\vec{x}_k$
- Construct restricted Voronoi cells ω_k with $\vec{x}_k \in \omega_k$
  - Corners of Voronoi cells are either cell circumcenters of midpoints of boundary edges
  - Admissibility condition $\vec{x}_k\vec{x}_l \perp \sigma_{kl} = \bar{\omega}_k \cap \bar{\omega}_l$ fulfilled in a natural way
- Triangulation edges: connected neigborhood graph of Voronoi cells
- Boundary placement of collocation points of boundary control volumes

Discretization of continuity equation

- Stationary continuity equation: $\nabla \cdot \vec{j} = f$
- Integrate over control volume ω_k:

$$
\begin{aligned}
0 &= \int_{\omega_k} \nabla \cdot \vec{j}\, d\omega - \int_{\omega_k} f\ d\omega \\
&= \int_{\partial\omega_k} \vec{j} \cdot \vec{n}_\omega\ ds - \int_{\omega_k} f\ d\omega \\
&= \sum_{l \in \mathcal{N}_k} \int_{\sigma_{kl}} \vec{j} \cdot \vec{\nu}_{kl}\ ds + \sum_{m \in \mathcal{G}_k} \int_{\gamma_{km}} \vec{j} \cdot \vec{n}_m\ ds - \int_{\omega_k} f d\omega
\end{aligned}
$$

Approximation of flux between control volumes

Utilize flux law: $\vec{j} = -\delta\vec{\nabla}u$

- Admissibility condition $\Rightarrow \vec{x}_k\vec{x}_l \parallel \nu_{kl}$
- Let $u_k = u(\vec{x}_k)$, $u_l = u(\vec{x}_l)$
- $h_{kl} = |\vec{x}_k - \vec{x}_l|$: distance between neigboring collocation points
- Finite difference approximation of normal derivative:

$$\vec{\nabla}u \cdot \vec{\nu}_{kl} \approx \frac{u_l - u_k}{h_{kl}}$$

$\Rightarrow$ flux between neigboring control volumes:

$$\begin{aligned}\int_{\sigma_{kl}} \vec{j} \cdot \vec{\nu}_{kl}\, ds &\approx \frac{|\sigma_{kl}|}{h_{kl}}\delta(u_k - u_l) \\ &=: \frac{|\sigma_{kl}|}{h_{kl}} g(u_k, u_l)\end{aligned}$$

where $g(\cdot,\cdot)$ is called flux function

Approximation of Robin boundary conditions

Utilize boundary condition $\vec{j} \cdot \vec{n} = \alpha u - g$

- Assume $\alpha|_{\Gamma_m} = \alpha_m$
- Approximation of $\vec{j} \cdot \vec{n}_m$ at the boundary of ω_k:

$$\vec{j} \cdot \vec{n}_m \approx \alpha_m u_k - g$$

- Approximation of flux from ω_k through Γ_m:

$$\int_{\gamma_{km}} \vec{j} \cdot \vec{n}_m \, ds \approx |\gamma_{km}|(\alpha_m u_k - g)$$

Discrete system of equations

- Let $f_k = f(\vec{x}_k)$ (or $f_k = \frac{1}{|\omega_k|}\int_{\omega_k} f(\vec{x}_k)\, d\omega$)
- The discrete system of equations then writes for $k \in \mathcal{N}$:

$$\sum_{l\in\mathcal{N}_k} \frac{|\sigma_{kl}|}{h_{kl}}\delta(u_k - u_l) + \sum_{m\in\mathcal{G}_k} |\gamma_{km}|\alpha_m u_k = |\omega_k| f_k + \sum_{m\in\mathcal{G}_k} |\gamma_{km}| g$$

$$u_k\left(\delta\sum_{l\in\mathcal{N}_k}\frac{|\sigma_{kl}|}{h_{kl}} + \alpha_m\sum_{m\in\mathcal{G}_k}|\gamma_{km}|\right) - \delta\sum_{l\in\mathcal{N}_k}\frac{|\sigma_{kl}|}{h_{kl}}u_l = |\omega_k| f_k + \sum_{m\in\mathcal{G}_k}|\gamma_{km}| g$$

$$a_{kk}u_k + \sum_{l=1\dots|\mathcal{N}|, l\neq k} a_{kl}u_l = b_k$$

with $b_k = |\omega_k| f_k + \sum_{m\in\mathcal{G}_k}|\gamma_{km}| g$ and

$$a_{kl} = \begin{cases} \sum_{l'\in\mathcal{N}_k}\delta\frac{|\sigma_{kl'}|}{h_{kl'}} + \sum_{m\in\mathcal{G}_k}|\gamma_{km}|\alpha_m, & l = k \\ -\delta\frac{\sigma_{kl}}{h_{kl}}, & l \in \mathcal{N}_k \\ 0, & else \end{cases}$$

Discretization matrix properties

- $N = |\mathcal{N}|$ equations (one for each control volume ω_k)
- $N = |\mathcal{N}|$ unknowns (one for each collocation point $x_k \in \omega_k$)
- Matrix is sparse: nonzero entries only for neighboring control volumes
- Matrix graph is connected: nonzero entries correspond to edges in Delaunay triangulation $\Rightarrow$ irreducible
- A is irreducibly diagonally dominant if at least for one i, $|\gamma_{i,k}|\alpha_i > 0$
- Main diagonal entries are positive, off diagonal entries are non-positive
- $\Rightarrow$ A has the M-property.
- A is symmetric $\Rightarrow$ A is positive definite

Matrix assembly algorithm

- Due to the connection between Voronoi diagram and Delaunay triangulation, one can assemble the discrete system based on the triangulation
- Necessary information:
  - List of point coordinates $\vec{x}_K$
  - List of triangles which for each triangle describes indices of points belonging to triangle
    - This induces a mapping of local node numbers of a triangle T to the global ones: $\{1, 2, 3\} \to \{k_{T,1}, k_{T,2}, k_{T,3}\}$
  - List of (boundary) segments which for each segment describes indices of points belonging to segment
- Assembly in two loops:
  - Loop over all triangles, calculate triangle contribution to matrix entries
  - Loop over all boundary segments, calculate contribution to matrix entries

Matrix assembly – main part

- Loop over all triangles $T \in \mathcal{T}$, add up edge contributions

for $k, l = 1 \dots N$ **do**
  set $a_{kl} = 0$
end
for $T \in \mathcal{T}$ **do**
  for $i, j = 1 \dots 3, i \neq j$ **do**

$$\sigma = \sigma_{k_{T,j},k_{T,i}} \cap T$$

$$s = \frac{|\sigma|}{h_{k_{T,j},k_{T,i}}}$$

$$a_{k_{T,j},k_{T,j}} += \delta\sigma_h$$

$$a_{k_{T,j},k_{T,i}} -= \delta\sigma_h$$

$$a_{k_{T,i},k_{T,j}} -= \delta\sigma_h$$

$$a_{k_{T,i},k_{T,n}} += \delta\sigma_h$$

  end
end

Matrix assembly – boundary part

- Keep list of global node numbers per boundary element γ mapping local node element to the global node numbers: $\{1,2\} \to \{k_{\gamma,1}, k_{\gamma,2}\}$
- Keep list of boundary part numbers m_γ per boundary element
- Loop over all boundary elements $\gamma \in \mathcal{G}$ of the discretization, add up contributions

for $\gamma \in \mathcal{G}$ **do**
  for $i = 1,2$ **do**
    $a_{k_{\gamma_i},k_{\gamma_i}} += \alpha_{m_\gamma} |\gamma \cap \partial\omega_{k_{\gamma_i}}|$
  end
end

RHS assembly: calculate control volumes

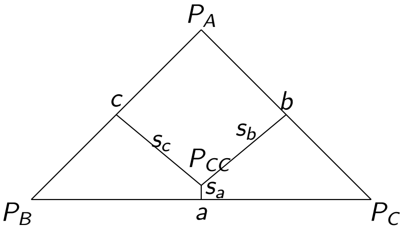
- Denote $w_k = |\omega_k|$
- Loop over triangles, add up contributions

for $k \dots N$ **do**
  set $w_k = 0$
end
for $\tau \in \mathcal{T}$ **do**
  for $n = \dots 3$ **do**
    $w_k += |\omega_{k_{\tau,j}} \cap \tau|$
  end
end

Matrix assembly: summary

- Sufficient to keep list of triangles, boundary segments – they typically come out of the mesh generator
- Be able to calculate triangular contributions to form factors: $|\omega_k \cap \tau|$, $|\sigma_{kl} \cap \tau|$ – we need only the numbers, and not the construction of the geometrical objects
- $O(N)$ operation, one loop over triangles, one loop over boundary elements

Finite volume local stiffness matrix calculation I



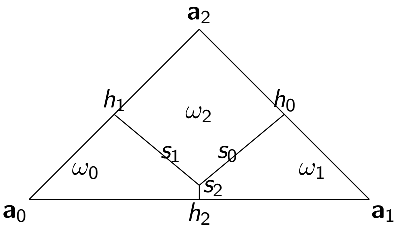
- Need to calculate s_a, s_b, s_c
- Triangle edge lengths: a, b, c
- Semiperimeter: $s = \frac{a}{2} + \frac{b}{2} + \frac{c}{2}$
- Square area (from Heron's formula):
  $16A^2 = 16s(s-a)(s-b)(s-c) =$
  $(-a+b+c)\,(a-b+c)\,(a+b-c)\,(a+b+c)$
- Square circumradius: $R^2 = \frac{a^2b^2c^2}{(-a+b+c)(a-b+c)(a+b-c)(a+b+c)} = \frac{a^2b^2c^2}{16A^2}$

Finite volume local stiffness matrix calculation II

- Square of the Voronoi surface contribution via Pythagoras:
  $s_a^2 = R^2 - \left(\frac{1}{2}a\right)^2 = -\frac{a^2\left(a^2-b^2-c^2\right)^2}{4(a-b-c)(a-b+c)(a+b-c)(a+b+c)}$
- Square of edge contribution in the finite volume method:
  $e_a^2 = \frac{s_a^2}{a^2} = -\frac{\left(a^2-b^2-c^2\right)^2}{4(a-b-c)(a-b+c)(a+b-c)(a+b+c)} = \frac{(b^2+c^2-a^2)^2}{64A^2}$
- Edge contribution. $e_a = \frac{s_a}{a} = \frac{b^2+c^2-a^2}{8A}$
- The sign chosen implies a positive value if the angle $\alpha < \frac{\pi}{2}$, and a negative value if it is obtuse. In the latter case, this corresponds to the negative length of the line between edge midpoint and circumcenter, which is exactly the value which needs to be added to the corresponding amount from the opposite triangle in order to obtain the measure of the Voronoi face.

Finite volume local stiffness matrix calculation III

$a_0 = (x_0, y_0) \dots a_d = (x_2, y_2)$: vertices of the simplex K Calculate the contribution from triangle to $\frac{\sigma_{kl}}{h_{kl}}$ in the finite volume discretization



Let $h_i = |a_{i+1} - a_{i+2}|$ (i counting modulo 2) be the lengths of the discretization edges. Let A be the area of the triangle. Then for the contribution from the triangle to the form factor one has

$$\frac{|s_i|}{h_i} = \frac{1}{8A}(h_{i+1}^2 + h_{i+2}^2 - h_i^2)$$

$$|\omega_i| = (|s_{i+1}|h_{i+1} + |s_{i+2}|h_{i+2})/4$$

Variations of the discretization ansatz

- 3D: tetrahedron based
- $\delta = \delta(x) \Rightarrow \delta(x)\nabla u \approx \delta_{kl}\frac{u_l - u_k}{h_{kl}}$
- Non-constant α_i, g
- Nonlinear dependencies ...

Interpretation of results

- One solution value per control volume ω_k allocated to the collocation point $x_k \Rightarrow$ piecewise constant function on collection of control volumes
- But: x_k are at the same time nodes of the corresponding Delaunay mesh $\Rightarrow$ representation as piecewise linear function on triangles

The problem with Dirichlet boundary conditions

- Eliminate Dirichlet BC algebraically after building of the matrix, i.e. fix "known unknowns" at the Dirichlet boundary $\Rightarrow$ highly technical
- Modifiy matrix such that equations at boundary exactly result in Dirichlet values $\Rightarrow$ loss of symmetry of the matrix
- Penalty method

Dirichlet BC: Algebraic manipulation

- Assume 1D situation with BC $u_1 = g$
- From handling of control volumes without regard of boundary values:

$$AU = \begin{pmatrix} \frac{1}{h} & -\frac{1}{h} & & & \\ -\frac{1}{h} & \frac{2}{h} & -\frac{1}{h} & & \\ & -\frac{1}{h} & \frac{2}{h} & -\frac{1}{h} & \\ & & \ddots & \ddots & \ddots \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \end{pmatrix} = \begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ \vdots \end{pmatrix}$$

- Fix u_1 and eliminate:

$$A'U = \begin{pmatrix} \frac{2}{h} & -\frac{1}{h} & & \\ -\frac{1}{h} & \frac{2}{h} & -\frac{1}{h} & \\ & \ddots & \ddots & \ddots \end{pmatrix} \begin{pmatrix} u_2 \\ u_3 \\ \vdots \end{pmatrix} = \begin{pmatrix} f_2 + \frac{1}{h}g \\ f_3 \\ \vdots \end{pmatrix}$$

- A' becomes idd and stays symmetric
- operation is quite technical

Dirichlet BC: Modify boundary equations

- From handling of control volumes without regard of boundary values:

$$AU = \begin{pmatrix} \frac{1}{h} & -\frac{1}{h} & & & \\ -\frac{1}{h} & \frac{2}{h} & -\frac{1}{h} & & \\ & -\frac{1}{h} & \frac{2}{h} & -\frac{1}{h} & \\ & & \ddots & \ddots & \ddots \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \end{pmatrix} = \begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ \vdots \end{pmatrix}$$

- Modify equation at boundary to exactly represent Dirichlet values

$$A'U = \begin{pmatrix} \frac{1}{h} & 0 & & & \\ -\frac{1}{h} & \frac{2}{h} & -\frac{1}{h} & & \\ & -\frac{1}{h} & \frac{2}{h} & -\frac{1}{h} & \\ & & \ddots & \ddots & \ddots \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \end{pmatrix} = \begin{pmatrix} \frac{1}{h}g \\ f_2 \\ f_3 \\ \vdots \end{pmatrix}$$

- A' becomes idd
- loses symmetry $\Rightarrow$ problem e.g. with CG method

Dirichlet BC: Discrete penalty trick

- From handling of control volumes without regard of boundary values:

$$AU = \begin{pmatrix} \frac{1}{h} & -\frac{1}{h} & & & \\ -\frac{1}{h} & \frac{2}{h} & -\frac{1}{h} & & \\ & -\frac{1}{h} & \frac{2}{h} & -\frac{1}{h} & \\ & & \ddots & \ddots & \ddots \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \end{pmatrix} = \begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ \vdots \end{pmatrix}$$

- Add penalty terms

$$A'U = \begin{pmatrix} \frac{1}{\varepsilon} + \frac{1}{h} & -\frac{1}{h} & & & \\ -\frac{1}{h} & \frac{2}{h} & -\frac{1}{h} & & \\ & -\frac{1}{h} & \frac{2}{h} & -\frac{1}{h} & \\ & & \ddots & \ddots & \ddots \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \end{pmatrix} = \begin{pmatrix} f_1 + \frac{1}{\varepsilon} g \\ f_2 \\ f_3 \\ \vdots \end{pmatrix}$$

- A' becomes idd, keeps symmetry, and the realization is technically easy.
- If ε is small enough, $u_1 = g$ will be satisfied exactly within floating point accuracy.
- Iterative methods should be initialized with Dirichlet values.
- Works for nonlinear problems, finite volume methods

Dirichlet penalty trick, general formulation

- Dirichlet boundary value problem

$$-\nabla \cdot \delta \nabla u = f \quad \text{in } \Omega$$
$$u|_{\Gamma_m} = g_m$$

- Approximate Dirichlet boundary condition by

$$\delta \nabla u \cdot \vec{n}_m + \frac{1}{\epsilon} u|_{\Gamma_m} = \frac{1}{\epsilon} g_m$$