

TECHNISCHE UNIVERSITÄT DRESDEN

Skript:

FEM-Methode

Verfasser

Franziska Kühn

Daten

Prof. Dr. Axel Voigt  
Wintersemester 2012/13  
Hauptstudium

# Inhaltsverzeichnis

<b>1</b>	<b>Diskretisierung parabolischer Randwertaufgaben</b>	<b>3</b>
<b>2</b>	<b>Assemblierung von 1D-Problemen</b>	<b>10</b>
2.1	Randbedingungen . . . . .	16
<b>3</b>	<b>Mehrdimensionales FEM - Assemblierung</b>	<b>20</b>
3.1	Simplizes . . . . .	20
3.2	Quadraturformeln . . . . .	22
3.3	Randbedingungen . . . . .	27
<b>4</b>	<b>Gittergenerierung</b>	<b>29</b>
4.1	Verfeinerung . . . . .	29
4.2	Verbindung zwischen geometrischen und Finite Elemente Daten . . . . .	32
4.3	Generierung einer Makrotriangulierung . . . . .	32
<b>5</b>	<b>Differentialgleichungen auf gekrümmten Oberflächen</b>	<b>36</b>
5.1	Gittererzeugung . . . . .	37
<b>6</b>	<b>Adaptive Finite Elemente Methode</b>	<b>39</b>
6.1	Stationäre Probleme . . . . .	39
6.2	Zeitabhängige Probleme . . . . .	42
<b>7</b>	<b>Fehleranalysis</b>	<b>46</b>
7.1	A posteriori Fehlerschätzer . . . . .	47
7.2	Duale Fehlerschätzer (goal-oriented Fehlerschätzer) . . . . .	49
<b>8</b>	<b>Anwendungen</b>	<b>51</b>
8.1	Lineare Elastizitätstheorie . . . . .	51
8.2	Navier-Stokes-Gleichung - Inkompressible Flüssigkeiten . . . . .	53

# 1

## Diskretisierung parabolischer Randwertaufgaben

Betrachte die Randwertaufgabe

$$\left(\frac{d}{dt}u(t), v\right) + a(u(t), v) = (f(t), v) \quad u(0) = u_0 \in H \quad (1.1)$$

für  $v \in V := H_0^1(\Omega)$ ,  $H = L^2(\Omega)$ . Gesucht ist  $u_h(t)$  mit

$$\left(\frac{d}{dt}u_h(t), v\right) + a(u_h(t), v) = (f(t), v) \quad (1.2)$$

für  $v_h \in V_h$ . Seien  $\{\varphi_i\}_i$  Basisfunktionen von  $V_h$ . Ansatz:

$$u_h(t) = \sum_i u_i(t) \cdot \varphi_i$$

Damit wird (1.2) zu

$$\forall j : \sum_i (\varphi_i, \varphi_j) \cdot \frac{d}{dt}u_i(t) + \sum_i a(\varphi_i, \varphi_j) \cdot u_i(t) = (f(t), \varphi_j)$$

Kurz:

$$D_h U'(t) + A_h U(t) = F(t) \quad U(0) = U_0 \quad (1.3)$$

wobei  $U(t) := (u_1(t), \dots, u_N(t))^T$ .  $D_h$  heißt Massematrix,  $A_h$  Steifigkeitsmatrix.

**Verfahren** (Diskretisierung in der Zeit)

Sei  $0 < t < T$  und  $0 = t_0 < t_1 < \dots < t_L = T$  ein Zeitgitter. Weiterhin sei  $k_\ell := t_\ell - t_{\ell-1}$  für  $\ell = 1, \dots, L$ . Integriere Differentialgleichung (1.3) von  $t_{\ell-1}$  bis  $t_\ell$ :

$$\underbrace{D_h \int_{t_{\ell-1}}^{t_\ell} U'(t) dt + A_h \int_{t_{\ell-1}}^{t_\ell} U(t) dt}_{D_h(U(t_\ell) - U(t_{\ell-1}))} = \int_{t_{\ell-1}}^{t_\ell} F(t) dt$$

Berechne die andere Terme über Quadraturformeln:

(i). right-end-point Methode:

$$\begin{aligned} D_h(U(t_\ell) - U(t_{\ell-1})) + k_\ell \cdot A_h U(t_\ell) &= k_\ell \cdot F(t_\ell) \\ \Leftrightarrow (D_h + k_\ell A_h)U(t_\ell) &= D_h U(t_{\ell-1}) + k_\ell \cdot F(t_\ell) \\ \Leftrightarrow D_h \frac{U(t_\ell) - U(t_{\ell-1})}{k_\ell} + AU(t_\ell) &= F(t_\ell) \end{aligned}$$

(„implizites Euler-Verfahren“)

(ii). left-end-point Methode:

$$\begin{aligned} D_h(U(t_\ell) - U(t_{\ell-1})) + k_\ell \cdot A_h U(t_{\ell-1}) &= k_\ell \cdot F(t_{\ell-1}) \\ \Leftrightarrow D_h U(t_\ell) &= (D_h - k_\ell)U(t_{\ell-1}) + k_\ell \cdot F(t_{\ell-1}) \\ \Leftrightarrow D_h \frac{U(t_\ell) - U(t_{\ell-1})}{k_\ell} &= F(t_{\ell-1}) - A_h U(t_{\ell-1}) \end{aligned}$$

(„explizites Euler-Verfahren“)

(iii). Trapez-Regel:

$$\begin{aligned} D_h(U(t_\ell) - U(t_{\ell-1})) + k_\ell \cdot A_h \frac{U(t_\ell) - U(t_{\ell-1})}{2} &= k_\ell \cdot \frac{F(t_\ell) - F(t_{\ell-1})}{2} \\ \Leftrightarrow \left( D_h - \frac{k_\ell}{2} A_h \right) U(t_{\ell-1}) + \frac{k_\ell}{2} \cdot (F(t_\ell) - F(t_{\ell-1})) &= \left( D_h + \frac{k_\ell}{2} A_h \right) U(t_\ell) \\ \Leftrightarrow D_h \frac{U(t_\ell) - U(t_{\ell-1})}{k_\ell} + A_h \frac{U(t_\ell) - U(t_{\ell-1})}{2} &= \frac{F(t_\ell) - F(t_{\ell-1})}{2} \end{aligned}$$

(„Crank-Nicolson Methode“)

**Beispiel** (1D Wärmeleitungsgleichung)

$$u_t - u_{xx} = f$$

für  $x \in (0, 1)$  mit linearen finiten Elementen, äquidistantem Gitter. Dann

$$D_h = \frac{h}{6} \cdot (d_{ij})_{i,j} \qquad A_h := \frac{1}{h} \cdot (a_{ij})_{i,j}$$

mit

$$d_{ij} := \begin{cases} 4 & i = j \\ 1 & |i - j| = 1 \\ 0 & \text{sonst} \end{cases} \qquad a_{ij} := \begin{cases} 2 & i = j \\ -1 & |i - j| = 1 \\ 0 & \text{sonst} \end{cases}$$

Jedes der oben genannten Verfahren hat eine eigene Charakteristik:

- (i). implizites Euler-Verfahren: Ist stabil, Verfahren 1. Ordnung.
- (ii). explizites Euler-Verfahren: Ist schnell, Verfahren 1. Ordnung.
- (iii). Crank-Nicolson-Methode: Ist am genauesten, Verfahren 2. Ordnung.

**Bemerkung** Bisher: PDE  $\rightarrow$  Variationsformulierung  $\rightarrow$  Diskretisierung im Ort  $\rightarrow$  System von ODEs  $\rightarrow$  Diskretisierung in der Zeit  $\rightarrow$  Folge von linearen Problemen.

Jetzt: PDE  $\rightarrow$  Variationsformulierung  $\rightarrow$  Diskretisierung in der Zeit  $\rightarrow$  Folge von elliptischen PDE's in jedem Zeitschritt  $\rightarrow$  Diskretisierung im Ort  $\rightarrow$  Folge von linearen Problemen.

Beide Zugänge liefern das gleiche voll-diskrete Problem.

### Verfahren

Betrachte Randwertaufgabe

$$\left( \frac{d}{dt} u(t), v \right) + a(u(t), v) = (f(t), v) \qquad u(0) = u_0 \in H$$

Diskretisierung in der Zeit gibt

- (i). implizites Euler-Verfahren:

$$\left( \frac{u(t_\ell) - u(t_{\ell-1})}{k_\ell}, v \right) + a(u(t_\ell), v) = (f(t_\ell), v)$$

(ii). explizites Euler-Verfahren:

$$\left( \frac{u(t_\ell) - u(t_{\ell-1})}{k_\ell}, v \right) + a(u(t_{\ell-1}), v) = (f(t_{\ell-1}), v)$$

(iii). Crank-Nicolson Methode:

$$\left( \frac{u(t_\ell) - u(t_{\ell-1})}{k_\ell}, v \right) + a \left( \frac{u(t_\ell) + u(t_{\ell-1})}{2}, v \right) = \left( \frac{f(t_\ell) + f(t_{\ell-1})}{2}, v \right)$$

**Beispiel** (1D Wellengleichung)

$$\begin{aligned} u_{tt} - u_{xx} &= f \quad \text{in } (0, 1) \\ u(0, t) &= u(1, t) = 0 \\ u(x, 0) &= u_0(x) \\ u_t(x, 0) &= u_1(x) \end{aligned} \tag{1.4}$$

Formulierung als Variationsproblem:

$$\begin{aligned} \left( \frac{d^2}{dt^2} u(t), v \right) + a(u(t), v) &= (f(t), v) \quad (v \in V) \\ u(0) &= u_0 \in H \quad \frac{d}{dt} u(0) = u_1 \in H \end{aligned}$$

bzw.

$$D_h U''(t) + A_h U(t) = F(t) \quad U(0) = U_0 \quad U'(0) = U_1$$

d.h. Differentialgleichung 2. Ordnung. Umschreiben in System von Differentialgleichungen 1. Ordnung:

$$\begin{aligned} D_h U'(t) &= D_h V(t) \\ D_h V'(t) + A_h U(t) &= F(t) \end{aligned}$$

Zum Beispiel Crank-Nicolson-Verfahren:

$$\begin{aligned} D_h \left( \frac{u(t_\ell) - u(t_{\ell-1})}{k_\ell} \right) &= D_h \left( \frac{V(t_\ell) + V(t_{\ell-1})}{2} \right) \\ D_h \left( \frac{V(t_\ell) - V(t_{\ell-1})}{k_\ell} \right) + A_h \left( \frac{U(t_\ell) + U(t_{\ell-1})}{2} \right) &= \frac{F(t_\ell) + F(t_{\ell-1})}{2} \end{aligned}$$

Kurz:

$$\begin{pmatrix} D_h & -\frac{k_\ell}{2} D_h \\ \frac{k_\ell}{2} A_h & D_h \end{pmatrix} \begin{pmatrix} U(t_\ell) \\ V(t_\ell) \end{pmatrix} = \begin{pmatrix} D_h & \frac{k_\ell}{2} D_h \\ -\frac{k_\ell}{2} A_h & D_h \end{pmatrix} \begin{pmatrix} U(t_{\ell-1}) \\ V(t_{\ell-1}) \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{k_\ell}{2} \cdot (F(t_\ell) + F(t_{\ell-1})) \end{pmatrix}$$

### 1.1 Satz

In (1.4) sei  $f = 0$ . Für die Lösung  $u$  der Wellengleichung gilt

$$\|\dot{u}(\cdot, t)\|^2 + \|\nabla u(\cdot, t)\|^2 = c$$

Beweis: Wähle  $v := \dot{u}$  im zugehörigen Variationsproblem, dann

$$\begin{aligned} 0 &= \int_{\Omega} \ddot{u} \cdot \dot{u} \, dx + \int_{\Omega} \nabla u \cdot \nabla \dot{u} \, dx \\ &= \int_{\Omega} \frac{1}{2} \partial_t (\dot{u})^2 \, dx + \int_{\Omega} \frac{1}{2} \partial_t (\nabla u)^2 \, dx \\ &= \frac{1}{2} \partial_t (\|\dot{u}\|^2 + \|\nabla u\|^2) \end{aligned}$$

Integration über Zeit von 0 bis  $T$  gibt

$$\|\dot{u}(\cdot, T)\|^2 + \|\nabla u(\cdot, T)\|^2 = \|u_1\|^2 + \|\nabla u_0\|^2$$

□

**Verfahren** ( $\Theta$ -Methode (fractional-stop- $\Theta$ -Schema))

Abstrakte Formulierung der partiellen Differentialgleichung:

$$\partial_t u + F(u, x, t) = 0 \quad \text{in } \Omega \times [0, T]$$

Beispiele:

- (i). Wärmeleitungsgleichung:  $F(u, x, t) = -\Delta u - f$
- (ii). Konvektions-Diffusions-Gleichung:  $F(u, x, t) = -\Delta u + b \cdot \nabla u + c \cdot u - f$
- (iii). Allen-Cahn-Gleichung:  $F(u, x, t) = -\Delta u + f'(u)$  mit  $f(u) := \frac{1}{4}(u^2 - 1)^2$

Wähle

$$F(u, x, t) = F_1(u, x, t) + F_2(u, x, t)$$

Es sei

$$F^{(\ell)} := F(u(t_\ell), x, t_\ell) \quad u^{(\ell)} := u(t_\ell) \quad d_\Theta u^{(\ell)} := \frac{u^{(\ell)} - u^{(\ell-\theta)}}{\Theta \cdot k}$$

$\Theta$ -Methode: Sei  $\Theta \in (0, 1)$ ,  $\Theta \neq \frac{1}{2}$ .

- (i). Gegeben sei  $u^{(\ell)}$ . Berechne  $u^{(\ell+\Theta)}$  aus

$$d_\Theta u^{(\ell+\Theta)} + F_1^{(\ell+\Theta)} = -F_2^{(\ell)}$$

- (ii). Gegeben sei  $u^{(\ell+\Theta)}$ . Berechne  $u^{(\ell+1-\Theta)}$  als Lösung von

$$d_{(1-2\Theta)} u^{(\ell+1-\Theta)} + F_2^{(\ell+1-\Theta)} = -F_1^{(\ell+\Theta)}$$

- (iii). Gegeben sei  $u^{(\ell+1-\Theta)}$ . Berechne  $u^{(\ell+1)}$  als Lösung von

$$d_\Theta u^{(\ell+1)} + F_1^{(\ell+1)} = -F_2^{(\ell+1-\Theta)}$$

Zum Beispiel für Konvektions-Diffusions-Gleichungen:

$$F_1 := -\Delta u + \frac{c}{2} \cdot u - f$$

$$F_2 := b \cdot \nabla u + \frac{c}{2} \cdot u$$

**Definition** Ein Verfahren heißt

- (i). numerisch kontraktiv  $:\Leftrightarrow \exists \kappa \in (0, 1]$  sodass für alle Näherungslösungen  $\tilde{u}$ ,  $u$  für verschiedene Anfangswerte gilt

$$\forall \ell : \|\tilde{u}^{(\ell+1)} - u^{(\ell+1)}\| \leq \kappa \cdot \|\tilde{u}^{(\ell)} - u^{(\ell)}\|$$

- (ii).  $A$ -stabil  $:\Leftrightarrow$  Das Verfahren ist für das Testproblem  $u' = \lambda \cdot u$  mit  $\text{Re } \lambda \leq 0$  kontraktiv. Es gilt mit der Stabilitätsfunktion  $R$ , dass  $u^{(\ell+1)} = R(k \cdot \lambda) \cdot u^{(\ell)}$ , dies bedeutet  $|R(z)| \leq 1$  für alle  $z$  mit  $\text{Re } z \leq 0$ .

**Beispiel** (i). implizites Euler-Verfahren:

$$\frac{u^{(\ell+1)} - u^{(\ell)}}{k} + F(u^{(\ell+1)}, t_{\ell+1}) = 0$$

angewandt auf  $u' = \lambda \cdot u$  liefert

$$u^{(\ell+1)} = \frac{1}{1 - k \cdot \lambda} \cdot u^{(\ell)}$$

also  $R(z) := \frac{1}{1-z}$ . Also  $A$ -stabil.

(ii). explizites Euler-Verfahren:

$$\frac{u^{(\ell+1)} - u^{(\ell)}}{k} + F(u^{(\ell)}, t_{\ell}) = 0$$

angewandt auf  $u' = \lambda \cdot u$  liefert

$$u^{(\ell+1)} = (1 + k \cdot \lambda) \cdot u^{(\ell)}$$

somit  $R(z) := 1 + z$ , d.h. das Verfahren ist nicht  $A$ -stabil.

(iii). Crank-Nicolson-Verfahren:

$$\frac{u^{(\ell+1)} - u^{(\ell)}}{k} + F(u^{(\ell+1)}, u^{(\ell)}, t_{\ell+1}, t_{\ell}) = 0$$

angewandt auf  $u' = \lambda \cdot u$  liefert

$$u^{(\ell+1)} = \frac{2 + k \cdot \lambda}{2 - k \cdot \lambda} \cdot u^{(\ell)}$$

also  $R(z) := \frac{2+z}{2-z}$ , d.h. das Verfahren ist  $A$ -stabil.

Fazit: Implizite Verfahren sind zu favorisieren.

**Bemerkung** (Ausblick) Implizite Runge-Kutta-Verfahren liefern  $A$ -stabile höherer Ordnung, führen aber für nichtlineare Probleme zu nichtlinearen Gleichungssystemen. Mögliche linear implizite Verfahren sind Rosenbrock-Verfahren (siehe z.B. Vorlesung zu Modellierung+Simulation oder Zeitdiskretisierungsverfahren von Jörg Wensch).

**Definition** (Funktionalableitung) Sei  $(H, (\cdot, \cdot)_H, \|\cdot\|_H)$  ein Hilbertraum. Gegeben sei ein Funktional  $F \in H'$ ,  $\phi \in H$ .  $F$  heißt differenzierbar an  $\phi$ , genau dann wenn  $\frac{\delta F}{\delta \phi} \in H'$  existiert mit

$$\forall v \in H : \left\langle \frac{\delta F}{\delta \phi}, v \right\rangle = \left. \frac{d}{d\varepsilon} F(\phi + \varepsilon \cdot v) \right|_{\varepsilon=0}$$

$\frac{\delta F}{\delta \phi}$  wird Funktionalableitung von  $F$  bzgl.  $\phi$  genannt.

**Bemerkung** Wenn  $F$  differenzierbar an  $\phi \in H$  ist, dann existiert nach dem Darstellungssatz von Riesz ein eindeutiges Element  $w \in H$ , sodass

$$\forall v \in H : \left\langle \frac{\delta F}{\delta \phi}, v \right\rangle = (w, v)_H$$

Dieses  $w$  wird als  $\text{grad}_H F(\phi)$  bezeichnet (Gradient von  $F$  an  $\phi$ ).

**Beispiel** (i).  $F(u) := \int_{\Omega} f(u) dx$ , dann

$$\left. \frac{d}{d\varepsilon} F(u + \varepsilon \cdot h) \right|_{\varepsilon=0} = \left. \frac{d}{d\varepsilon} \int_{\Omega} f(u + \varepsilon \cdot h) dx \right|_{\varepsilon=0} = \int_{\Omega} \underbrace{f'(u)}_{\frac{\delta F[u]}{\delta u}} \cdot h dx$$

(ii).  $F(u) := \int_{\Omega} \frac{1}{2} |\nabla u|^2$ , dann

$$\begin{aligned} \left. \frac{d}{d\varepsilon} F(u + \varepsilon \cdot h) \right|_{\varepsilon=0} &= \left. \frac{d}{d\varepsilon} \int_{\Omega} \frac{1}{2} |\nabla(u + \varepsilon \cdot h)|^2 dx \right|_{\varepsilon=0} = \int_{\Omega} \nabla u \cdot \nabla h dx \\ &= \int_{\Omega} \underbrace{-\Delta u}_{\frac{\delta F[u]}{\delta u}} \cdot h dx \end{aligned}$$

**Definition** Sei  $H$  ein Hilbertraum,  $M > 0$  und  $F \in H'$  differenzierbar mit Gradient  $\text{grad}_H F$ . Eine Evolutionsgleichung

$$\partial_t \phi = -M \cdot \text{grad}_H F(\phi)$$

mit Startwert  $\phi(t=0) = \phi_0$ , zu verstehen als

$$(\partial_t \phi, v)_H = -M \cdot \left\langle \frac{\delta F}{\delta \phi}, v \right\rangle \quad (v \in H)$$

wird bezeichnet als Gradientenfluss.  $M$  heißt Mobilität des Flusses.

**Verfahren** (Convexity-Splitting (Eyers-Method))

Häufig entstehen zeitabhängige Probleme als Gradientenfluss zur Minimierung eines Energiefunktionals  $F(u)$ .

$$\partial_t u + \nabla_H F(u) = 0$$

$(\nabla_H F(u) \hat{=} \frac{\delta F[u]}{\delta u})$  Es gilt

$$\frac{d}{dt} F(u) = (\nabla_H F(u), u_t) = -\|\nabla_H F(u)\|^2$$

d.h.  $F(u(t)) \leq F(u(0))$  für  $t \geq 0$ . Falls  $F(u)$  konvex ist, existiert nur ein Gleichgewicht und die Evolution ist kontraktiv. Falls  $F(u)$  nicht konvex ist, wähle

$$F(u) = F_c(u) - F_e(u)$$

wobei  $F_c$  und  $F_e$  strikt konvex. Dann ist

$$\partial_t u = -\Delta_H F_c(u)$$

eine Kontraktion und

$$\partial_t u = \Delta_H F_e(u)$$

eine Expansion. Wähle

$$\frac{u^{(\ell+1)} - u^{(\ell)}}{k_\ell} = \nabla_H F_e(u^{(\ell)}) - \nabla_H F_c(u^{(\ell+1)})$$

(d.h. konvexer Anteil ist implizit, konkave Anteil ist explizit).

**Beispiel** Gegeben sei eine Energie  $F$ . Addiere und subtrahiere einen konvexen Term  $E$ , sodass

$$F = \underbrace{(F - c \cdot E)}_{=: -F_e} + \underbrace{c \cdot E}_{=: F_c}$$

mit  $F_e$  konvex (für  $c$  hinreichend groß). Wähle  $E$  so, dass Problem einfach invertierbar. Ziel: Methode ist stabil.



Beweis: Betrachte Taylor-Reihe für  $F_c$ , dann

$$\begin{aligned} F_c(u^{(\ell)}) &= F_c(u^{(\ell+1)}) - \left( \nabla_H F_c(u^{(\ell+1)}), u^{(\ell+1)} - u^{(\ell)} \right) + \underbrace{\frac{1}{2} (D^2 F_c \xi, \xi)}_{\geq 0} \\ &\geq F_c(u^{(\ell+1)}) - \left( \nabla_H F_c(u^{(\ell+1)}), u^{(\ell+1)} - u^{(\ell)} \right) \end{aligned}$$

wegen Konvexität von  $F_c$ . Betrachte Taylor-Reihe für  $F_e$ :

$$\begin{aligned} F_e(u^{(\ell+1)}) &= F_e(u^{(\ell)}) + \left( \nabla_H F_e(u^{(\ell)}), u^{(\ell+1)} - u^{(\ell)} \right) + \frac{1}{2} (D^2 F_e \xi, \xi) \\ &\geq F_e(u^{(\ell)}) + \left( \nabla_H F_e(u^{(\ell)}), u^{(\ell+1)} - u^{(\ell)} \right) \end{aligned}$$

wegen  $F_e$  konvex. Es gilt

$$\begin{aligned} F(u^{(\ell+1)}) &= F_c(u^{(\ell+1)}) - F_e(u^{(\ell+1)}) \\ &\leq F_c(u^{(\ell)}) + \left( \nabla_H F_c(u^{(\ell+1)}), u^{(\ell+1)} \right) \\ &\quad - F_e(u^{(\ell)}) - \left( \nabla_H F_e(u^{(\ell)}), u^{(\ell+1)} - u^{(\ell)} \right) \\ &= F(u^{(\ell)}) - k \cdot \|\nabla_H F_c(u^{(\ell+1)})\|^2 - k \cdot \|\nabla_H F_e(u^{(\ell)})\|^2 \\ &\quad - 2k \cdot \left( \nabla_H F_c(u^{(\ell+1)}), \nabla_H F_e(u^{(\ell)}) \right) \\ &\leq F(u^{(\ell)}) \end{aligned}$$

mit Cauchy-Schwartz für die letzte Abschätzung. □

**Beispiel** (Allen-Cahn-Gleichung)

$$\partial_t u = \Delta u - f'(u)$$

ist Gradientenfluß der Energie

$$F(u) := \int_{\Omega} \frac{1}{2} |\nabla u|^2 + f(u) \, dx$$

für  $f(u) := \frac{1}{4} \cdot (u^2 - 1)^2$ . Es gilt

$$\partial_t u = -\nabla_{L^2} F(u)$$

Wähle  $F = (F - c \cdot E) + c \cdot E$  mit geeigneten  $E$ , z.B.

$$E := \int_{\Omega} \frac{1}{2} (|\nabla u|^2 + u^2) \, dx$$

und damit  $F_c := c \cdot E$ ,  $F_e := F - c \cdot E$ . Es folgt

$$\frac{u^{(\ell+1)} - u^{(\ell)}}{k} = \Delta u^{(\ell)} - f'(u^{(\ell)}) - c \cdot \Delta u^{(\ell)} + c \cdot u^{(\ell)} + c \cdot \Delta u^{(\ell+1)} - c \cdot u^{(\ell+1)}$$

Lineares Problem, stabiles Verfahren.

# 2

## Assemblierung von 1D-Problemen

Sei  $\Omega = [a, b] \subseteq \mathbb{R}$  ein Intervall. Gesucht ist  $u \in V_g := \{v \in H^1(a, b); v(a) = g_a\}$ , sodass

$$a(u, v) = \langle F, v \rangle \quad (v \in V_0)$$

wobei  $V_0 := \{u \in H^1(a, b); v(a) = 0\}$  und

$$a(u, v) := \int_a^b (u'(x) \cdot v'(x) + c \cdot u(x) \cdot v(x)) dx + \alpha_b \cdot u(b) \cdot v(b)$$

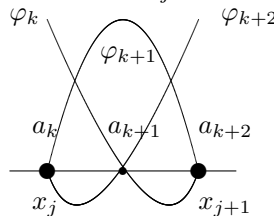
$$(F, v) := \int_a^b f'(x) \cdot v(x) dx + \alpha_b \cdot g_b \cdot v(b)$$

(die beiden zweiten Summanden entsprechen der Randbedingung  $u'(b) = \alpha_b \cdot (u(b) - g_b)$ ; Robin-Bedingung). Übergang zu endlich-dimensionalen Problem:

- Funktionenraum mit endlicher Dimension
- Basisfunktionen mit möglichst kleinem Träger

Dazu zerlegt man  $\Omega$  in  $n$  Teilintervalle,  $a = x_0 < \dots < x_n = b$ . Setze  $h_j := x_{j+1} - x_j$  und  $T_j := [x_j, x_{j+1}]$ . Jedem Knoten  $x_j$  (*besser: jedem Freiheitsgrad*) sei eine Funktion  $\varphi_j$  zugeordnet, die nur im Intervall  $[x_{j-1}, x_{j+1}]$  ungleich 0 ist, d.h.  $\text{spt } \varphi_j \subseteq [x_{j-1}, x_{j+1}]$  (z.B. Hütchenfunktion).

**Bemerkung** Unterscheidung zwischen Gitterknoten  $x_j$  und Freiheitsgraden  $a_k$ , zum Beispiel



Die Funktionen  $(\varphi_j)$  spannen einen linearen Raum  $V_h := \text{lin}\{\varphi_j; j\}$  auf.  $\{\varphi_j; j\}$  ist sogar Basis von  $V_h$ . Elemente von  $V_h$  lassen sich als lineare Kombination der Basiselemente zuordnen

$$V_h \ni u_h(x) = \sum_j u_j \cdot \varphi_j(x)$$

Im Beispiel gilt  $u_h(x_j) = u_j$  bzw.  $u_h(a_k) = u_k$ .

**Bemerkung** Betrachte die  $V$ -Projektion  $u_h := \Pi_h u$  von  $u \in V$ . Dann gilt im Allgemeinen nicht  $u_h(x_j) = u(x_j)$ .

Denn: Betrachte Variationsformel der  $V$ -Projektion  $(u_h, v_h) = (u, v_h)$  für  $v_h \in V_h$  mit  $u_h = \sum_j u_j \cdot \varphi_j$ ,  $v_h = \sum_j v_j \cdot \varphi_j$ . Man erhält ein lineares Gleichungssystem  $M \cdot U = b$  für  $U := (u_1, \dots, u_n)$  und

$$M := ((\varphi_i, \varphi_j))_{i,j} \quad b := \left( \int_a^b u(x) \cdot \varphi_i(x) dx \right)_i$$

Betrachte zum Beispiel  $\Omega = [0, 1]$  mit linearen finiten Elementen, d.h.  $\varphi_0 = 1 - x$ ,  $\varphi_1(x) = x$ . Sei  $u(x) := x^2$ . Dann gilt

$$M = \begin{pmatrix} \frac{1}{3} & \frac{1}{6} \\ \frac{1}{6} & \frac{1}{3} \end{pmatrix} \quad b = \begin{pmatrix} \frac{1}{12} \\ \frac{1}{4} \end{pmatrix}$$

Es ergibt sich  $U = (-1/6, 5/6) \neq (u(x_0), u(x_1)) = (0, 1)$ .

Zur Wärmeleitungsgleichung ergibt sich das folgende endlich-dimensionale Ersatzproblem: Gesucht ist  $u_h \in V_{gh} := \{v_h; v_h(x) = \sum_{j=1}^n v_j \cdot \varphi_j + g_a + \varphi_0(x)\}$ , sodass

$$a(u_h, v_h) = \langle F, v_h \rangle \quad (v_h \in V_{0h})$$

mit  $V_{0h} := \{v_h; v_h(x) = \sum_{j=1}^n v_j \cdot \varphi_j(x)\}$ . Äquivalent dazu:

$$\forall k \in \mathbb{N} : a(u_h, \varphi_k) = \langle F, \varphi_k \rangle - a(g_a \cdot \varphi_0, \varphi_k)$$

Betrachte  $k$ -te Gleichung:

$$\begin{aligned} a(u_h, \varphi_k) &= \sum_{j=1}^n u_j \cdot \int_a^b \varphi_j'(x) \cdot \varphi_k'(x) + c \cdot \varphi_j(x) \cdot \varphi_k(x) dx + \alpha_b \cdot u_n \cdot \varphi_k(b) \\ \langle F, \varphi_k \rangle &= \int_a^b f(x) \cdot \varphi_k(x) dx + \alpha_b \cdot g_b \cdot \varphi_k(b) \\ &\quad - g_a \cdot \int_a^b (\varphi_0'(x) \cdot \varphi_k'(x) + c \cdot \varphi_0(x) \cdot \varphi_k(x)) dx \end{aligned}$$

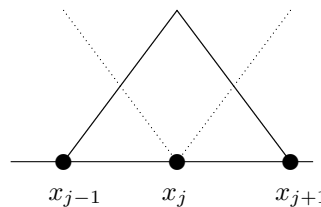
Mit  $\varphi_k(x_i) = \delta_{ik}$  und  $b = x_n$  ergibt sich das Gleichungssystem  $K_h \cdot U = f_h$  für  $U = (u_1, \dots, u_n)$  wobei

$$\begin{aligned} (K_h)_{k,j} &= \int_a^b (\varphi_j'(x) \cdot \varphi_k'(x) + c \cdot \varphi_j(x) \cdot \varphi_k(x)) dx + \delta_{jn} \cdot \delta_{kn} \cdot \alpha_b \\ (f_h)_k &= \int_a^b (f(x) \cdot \varphi_k(x) - g_a \cdot (\varphi_0'(x) \cdot \varphi_k'(x) + c \cdot \varphi_0(x) \cdot \varphi_k(x))) dx + \delta_{kn} \cdot \alpha_b \cdot g_b \end{aligned}$$

Matrix  $K_h$  ist dünn besetzt: Wegen lokalen Träger der Basisfunktionen gilt

$$\text{int}(\text{spt } \varphi_i) \cap \text{int}(\text{spt } \varphi_j) \neq \emptyset \Leftrightarrow |i - j| \leq 1$$

also  $(K_h)_{i,j} = 0$  für  $|i - j| \geq 2$ , d.h.  $K_h$  hat Tridiagonalgestalt.



(Matrix auch „dünn“ speichern. Für tridiagonale Matrizen gibt es spezielle Algorithmen zum Lösen von linearen Gleichungssystem, z.B. „Thomas-Algorithmus“.)

**Bemerkung** Besetztheitsstruktur der Matrix hängt direkt mit der Nummerierung der Knoten zusammen. Übungsaufgabe: Betrachte alternative Knotennummerierung,  $a = x_0 < x_n < x_1 < x_{n-1} < \dots$

Zur Berechnung der Matrixeinträge  $K_{ij}$  (nur die Nicht-Null-Einträge), zum Beispiel mit linearen finiten Elementen:

$$\varphi_j(x) := \begin{cases} 0 & x \in [a, x_{j-1}] \cup [x_{j+1}, b] \\ \frac{x-x_{j-1}}{x_j-x_{j-1}} & x \in [x_{j-1}, x_j] \\ \frac{x_{j+1}-x}{x_{j+1}-x_j} & x \in [x_j, x_{j+1}] \end{cases}$$

also

$$\varphi'_j(x) = \begin{cases} 0 & x \in [a, x_{j-1}] \cup [x_{j+1}, b] \\ \frac{1}{x_j-x_{j-1}} & x \in [x_{j-1}, x_j] \\ -\frac{1}{x_{j+1}-x_j} & x \in [x_j, x_{j+1}] \end{cases}$$

Damit ergibt sich für die Diagonaleinträge

$$\begin{aligned} K_{jj} &= \int_{x_{j-1}}^{x_{j+1}} \varphi'_j(x)^2 + c \cdot \varphi_j(x)^2 dx \\ &= \int_{x_{j-1}}^{x_j} \varphi'_j(x)^2 + c \cdot \varphi_j(x)^2 dx + \int_{x_j}^{x_{j+1}} \varphi'_j(x)^2 + c \cdot \varphi_j(x)^2 dx \\ &= \frac{1}{h_{j-1}} + c \cdot \frac{h_{j-1}}{3} + \frac{1}{h_j} \cdot c \cdot \frac{h_j}{3} \end{aligned}$$

für  $1 \leq j \leq n-1$ . Weiterhin

$$K_{nn} = \int_{x_{n-1}}^{x_n} \varphi'_n(x)^2 + c \cdot \varphi_n(x)^2 dx + \alpha_b = \frac{1}{h_{n-1}} + c \cdot \frac{h_{n-1}}{3} + \alpha_b$$

Nebendiagonaleinträge:

$$K_{j+1,j} = K_{j,j+1} = \int_{x_j}^{x_{j+1}} \varphi'_j(x) \cdot \varphi'_{j+1}(x) + c \cdot \varphi_j(x) \cdot \varphi_{j+1}(x) dx = \dots$$

(In der Praxis: Nutze Quadraturformeln um die Integrale zu berechnen. Hier z.B. Gauß-Quadratur möglich.)

**Bemerkung** Integrale gehen nur über einzelne Elemente, d.h. elementweiser Aufbau der Matrix und des Lastvektors möglich.

Definiere dazu Elementmatrix

$$K^{T_j} := \begin{pmatrix} K_{00}^{(j)} & K_{01}^{(j)} \\ K_{10}^{(j)} & K_{11}^{(j)} \end{pmatrix}$$

mit

$$K_{\beta\alpha}^{(j)} := \int_{T_j} \varphi'_{j+\alpha}(x) \cdot \varphi'_{j+\beta}(x) + c \cdot \varphi_{j+\alpha}(x) \cdot \varphi_{j+\beta}(x) dx$$

$K^{T_j}$  auf  $K_h$  aufaddieren gibt hier

$$K_h = \begin{pmatrix} K_{00}^{(1)} & K_{01}^{(1)} & 0 & \dots \\ K_{10}^{(1)} & K_{11}^{(1)} + K_{00}^{(2)} & K_{01}^{(2)} & \dots \\ 0 & K_{10}^{(2)} & K_{11}^{(2)} + K_{00}^{(3)} & \dots \end{pmatrix}$$

Randbedingungen im Nachhinein hinzufügen. Steifigkeitsmatrix:

$$a_{ij} = \begin{cases} -1 & |i-j| = 1 \\ 2 & 1 \leq i=j \leq n-1 \\ 1 & i=j \in \{1, n\} \end{cases}$$

**Bemerkung** (Übungsaufgabe) Warum Steifigkeitsmatrix singular bei homogenen Neumann-Randbedingungen?

Betrachte Referenzelement  $\hat{T}$ , Intervall  $[0, 1]$  mit Koordinate  $\xi$ , und darauf definieren wir „Formfunktionen“, zum Beispiel:

(i). lineare Lagrange-Formfunktionen:

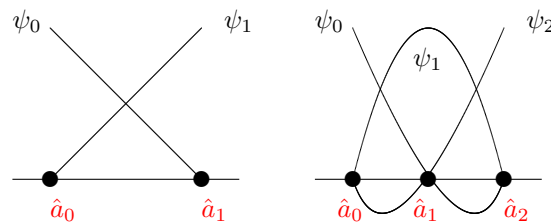
$$\psi_0(\xi) = 1 - \xi \quad \psi_1(\xi) = \xi$$

(ii). quadratische Lagrange-Formfunktionen:

$$\psi_0(\xi) = 1 - 3\xi^2 + \xi^3$$

$$\psi_1(\xi) = 4\xi - 3\xi^2$$

$$\psi_2(\xi) = -\xi + 2\xi^2$$



Intervalle in  $\mathbb{R}$  können affin linear aufeinander abgebildet werden:

$$F_{T_j}(\xi) = \underbrace{A_j}_{x_{j+1} - x_j} \cdot \xi + x_j =: x_{T_j}(\xi)$$

Weiterhin können auch die Freiheitsgrade aufeinander abgebildet werden:

$$F_{T_j}(\hat{\alpha}_\beta) = a_{k(j,\beta)}$$

wobei  $\beta$  lokaler Index und  $k(j, \beta)$  globaler Index. Hier kann leicht Inverse von  $F_{T_j}$  angegeben werden,

$$F_{T_j}^{-1}(x) = \frac{x - x_j}{x_{j+1} - x_j} =: \xi_{T_j}(x)$$

**Bemerkung** Sei  $T_j$  ein Geradenstück in  $\mathbb{R}^d$ , d.h.  $x_j, x_{j+1} \in \mathbb{R}^d$ , dann

$$F_{T_j}(\xi) = (x_{j+1} - x_j) \cdot \xi + x_j$$

mit Inverse

$$F_{T_j}^{-1}(x) = \frac{\|x - x_j\|_2}{\|x_{j+1} - x_j\|_2}$$

für  $x = (x_{j+1} - x_j) \cdot \xi + x_j$  mit  $\xi \in [0, 1]$ .

Damit können globale Ansatzfunktionen definiert werden:

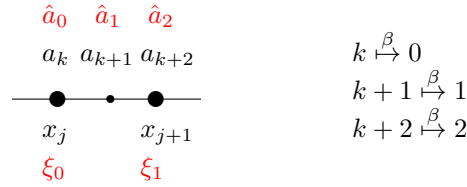
$$\varphi_k := \begin{cases} \psi_\beta(\xi_{T_j}(x)) & x \in T_j, j \in B_k \\ 0 & \text{sonst} \end{cases}$$

wobei

- $T_j$ : globales Element, das den Knoten  $a_k$  enthält

- $B_k$ : Indizes der Elemente, die den Knoten  $a_k$  enthalten
- $\beta = \beta(j, k)$ : lokale Knotennummer, die zu globaler Knotennummer  $k$  im Element  $T_j$  gehört

Zum Beispiel für quadratische finite Elemente:



Betrachte die Elementmatrizen  $K_{\beta\alpha}^{(j)}$ :

$$\begin{aligned} K_{\beta\alpha}^{(j)} &= \int_{T_j} \varphi'_{j+\alpha}(x) \cdot \bar{\varphi}'_{j+\beta}(x) + c \cdot \varphi_{j+\alpha}(x) \cdot \bar{\varphi}_{j+\beta}(x) dx \\ &= \int_{T_j} \psi'_\alpha(\xi_{T_j}(x)) \cdot \bar{\psi}'_\beta(\xi_{T_j}(x)) + c \cdot \psi_\alpha(\xi_{T_j}(x)) \cdot \bar{\psi}_\beta(\xi_{T_j}(x)) dx \\ &= \int_{\hat{T}} \left( \frac{1}{(x_{j+1} - x_j)^2} \cdot \frac{d\psi_\alpha(\xi)}{d\xi} \cdot \frac{d\bar{\psi}_\beta(\xi)}{d\xi} + c \cdot \psi_\alpha(\xi) \cdot \bar{\psi}_\beta(\xi) \right) \cdot (x_{j+1} - x_j) d\xi \end{aligned}$$

( $\bar{\varphi}$  bzw.  $\bar{\psi}$ : Die Ansatzfunktion muss nicht notwendigerweise aus dem gleichen Ansatzraum stammen wie  $\varphi$  bzw.  $\psi$ .) Dabei genutzt:

$$\begin{aligned} \frac{dx}{d\xi} &= x_{j+1} - x_j \\ \psi'_\alpha(\xi_{T_j}(x)) &= \frac{d\psi_\alpha(\xi(T_j(x)))}{d\xi} \cdot \underbrace{\frac{d\xi_{T_j}(x)}{dx}}_{\frac{1}{x_{j+1} - x_j}} \end{aligned}$$

und  $\psi_\alpha(\xi_{T_j}(x_{T_j}(\xi))) = \psi_\alpha(\xi)$ .

**Bemerkung** (i). Wie sieht Transformation für  $T_j$  Geradenstück in  $\mathbb{R}^d$  aus?  $\xi \mapsto x_{T_j}(\xi)$  ist Parametrisierung von  $T_j$ ,

$$\int_{T_j} f(x) dx = \int_0^1 f(x_{T_j}(\xi)) \cdot \left| \frac{dx_{T_j}}{d\xi} \right|_2 d\xi$$

(Kurvenintegral)

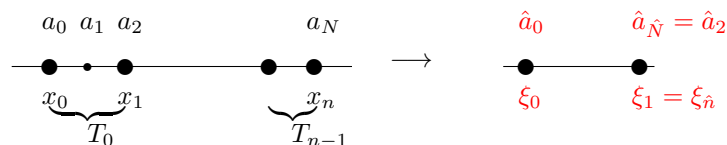
(ii). Auch Integration auf (gekrümmten) Kurven ist durchführbar, z.B.

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \cos\left(\frac{\xi \cdot \pi}{2}\right) \\ \sin\left(\frac{\xi \cdot \pi}{2}\right) \end{pmatrix} =: \gamma(\xi) \quad (\xi \in [0, 1])$$

Dann

$$\frac{d\gamma}{d\xi} = \begin{pmatrix} -\frac{\pi}{2} \cdot \sin\left(\frac{\xi \cdot \pi}{2}\right) \\ \frac{\pi}{2} \cdot \cos\left(\frac{\xi \cdot \pi}{2}\right) \end{pmatrix} \quad \left| \frac{d\gamma}{d\xi} \right|_2 = \frac{\pi}{2}$$

Beziehung zwischen lokalen und globalen Indizes kann mittels Elementzusammenhangstabelle bereitgestellt werden. Zum Beispiel für quadratische Elemente:



Elementnummer	globale Knotennummer $k(j, \beta)$ des Knotens mit lokaler Nummer $\beta$		
	0	1	2
0	0	1	2
1	2	3	4
2	4	5	6
$\vdots$	$\vdots$	$\vdots$	$\vdots$
n-1	2n-2	2n-1	2n=N

Es lässt sich also ablesen:

$$k(1, 2) = 4 \qquad \beta(1, 4) = 2$$

also lokal  $\rightarrow$  global und global  $\rightarrow$  lokal.

Aufdatierung der Elementmatrizen zu  $K_{\beta\alpha}^{(j)}$ : Sei  $U := (u_0, \dots, u_N)^T$ ,  $V := (v_0, \dots, v_N)^T$ . Definiere Matrix  $C^{(j)} := (c_{\alpha k}^{(j)})_{\alpha=0, \dots, \hat{N}, k=0, \dots, N}$  durch

$$c_{\alpha k}^{(j)} = \begin{cases} 1 & k \text{ ist die globale Knotennummer des Knotens mit lokaler Knotennummer } \alpha \text{ im } j\text{-ten Element} \\ 0 & \text{sonst} \end{cases}$$

$$= \begin{cases} 1 & k = k(j, \alpha) \\ 0 & \text{sonst} \end{cases}$$

Damit

$$C^{(j)} \cdot U = \begin{pmatrix} u_{k(j,0)} \\ \vdots \\ u_{k(j, \hat{N})} \end{pmatrix}$$

**Beispiel** (Lineare finite Elemente) Für lineare finite Elemente ergibt sich

$$C^{(j)} U = \begin{pmatrix} u_j \\ u_{j+1} \end{pmatrix}$$

Damit

$$\begin{aligned} v_h K_h u_h &= \int_a^b (u_h'(x) \cdot v_h'(x) + c \cdot u_h(x) \cdot v_h(x)) dx \\ &= \sum_{j=0}^{n-1} (v_j, v_{j+1}) \cdot K^{T_j} \begin{pmatrix} u_j \\ u_{j+1} \end{pmatrix} \\ &= \sum_{j=0}^{n-1} V \cdot (C^{(j)})^T \cdot K^{T_j} \cdot C^{(j)} \cdot U \\ &= V \cdot \left( \sum_{j=0}^{n-1} (C^{(j)})^T \cdot K^{T_j} \cdot C^{(j)} \right) \cdot U \\ \Rightarrow K_h &= \sum_{j=0}^{n-1} (C^{(j)})^T \cdot K^{T_j} \cdot C^{(j)} \end{aligned}$$

Analoges Vorgehen für Lastvektor:

$$f_h = \sum_{j=0}^{n-1} (C^{(j)})^T \cdot f^{T_j}$$

## 2.1 Algorithmus (Assemblierungsalgorithmus)

$K_h = \text{zeros}(0 \dots N, 0 \dots N)$   
 $f_h = \text{zeros}(0 \dots N, 1)$

```

FOR  $j = 0 \dots n - 1$ 
     $E = \text{zeros}(0 \dots \hat{N}, 0 \dots \hat{N})$ 
     $e = \text{zeros}(0 \dots \hat{N}, 1)$ 
     $h = x_{j+1} - x_j$ 

    FOR  $\beta = 0 \dots \hat{N}$ 
        FOR  $\alpha = 0 \dots \hat{N}$ 
             $E_{\beta\alpha} = \int_{\hat{T}} \left( \frac{1}{h^2} \psi'_\alpha(\xi) \cdot \psi'_\beta(\xi) + c \cdot \psi_\alpha(\xi) \cdot \psi_\beta(\xi) \right) h d\xi$ 
        END  $\alpha$ 
         $e_\beta = \int_{\hat{T}} f(x_{T_j}(\xi)) \cdot \psi_\beta(\xi) \cdot h d\xi$ 
    END  $\beta$ 

    FOR  $\beta = 0, \dots, \hat{N}$ 
        FOR  $\alpha = 0, \dots, \hat{N}$ 
             $K_{k(j,\beta), k(j,\alpha)} += E_{\beta\alpha}$ 
        END  $\alpha$ 
         $f_{k(j,\beta)} += e_\beta$ 
    END  $\beta$ 
END  $j$ 
    
```

Dann SOLVE  $K_h U = f_h$ . Bestimmt damit  $u_h = \sum_j u_i \cdot \varphi_i$ .

**Bemerkung** (Abstraktion des Assemblierungsprozesses) (i). Initialisierung

(ii). Gitter traversieren

(iii).  $\forall T \in \mathcal{T}$ : Elementmatrix(/-vektor) assemblieren. Bereitstellen:

- $F_T / X_T(\xi)$
- $k(j, \beta)$
- Jacobimatrix  $JF_T$

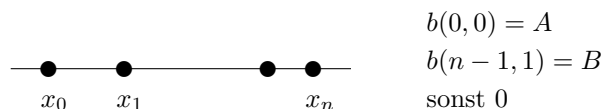
Quadratur (Terme/Ableitungen von Termen, Formfunktionen)

## 2.1 Randbedingungen

- Dirichlet Randbedingungen (wesentliche Randbedingung)
- Robin-Randbedingung/Neumann-Randbedingung (natürliche Randbedingung)
- periodische Randbedingungen

Es gebe eine Abbildung  $b : \{0, \dots, n-1\} \times \{0, \dots, \hat{n}\} \rightarrow \mathbb{Z}$ ,  $(j, \beta) \mapsto b(j, \gamma)$ , die einer Facette eines Elements  $T_j$  eine Randnummer zuweist. Dabei hat die Facette am Knoten  $\beta$  die Nummer  $\beta$ . Ist eine Facette kein Rand, dann gilt  $b(j, \beta) = 0$ .

**Beispiel**





**Beispiel** (Robin-Randbedingung) Gesucht  $u \in V_{gh}$  mit

$$a(u, v) = \langle F, v \rangle \quad (v \in V_{0h})$$

mit

$$a(u, v) = \int_{\Omega} u' \cdot v' + c \cdot u \cdot v + \int_{\Gamma_b} \alpha_b \cdot u \cdot v$$

$$\langle F, v \rangle = \int_{\Omega} f \cdot v + \int_{\Gamma_b} \alpha_b \cdot g_b \cdot v$$

Ziel: Randintegrale mit in System einfügen.

## 2.2 Algorithmus (Robin-Randbedingung)

Annahme: Assemblierungsalgorithmus 2.1 schon durchlaufen. Dann

```

FOR j = 0, ..., n - 1
  FOR γ = 0, ..., n̂
    IF b(j, γ) = B THEN
      Kk(j,β(γ)),k(j,β(γ))+ = αb
      fk(j,β(γ))+ = αb · gb
    END IF
  END(γ)
END(j)
    
```

**Bemerkung** Hier nur einfacher Fall. Kann eine Kante (hier: Knoten) mehr als einen Freiheitsgrad, benötigt man eine weitere Schleife. Im mehrdimensionalen Fall hätte man zudem eigentlich Integrale über den Rand (keine Punktauswertung).

**Beispiel** (Dirichlet-Randbedingung) Ziel:  $u(a) = g_a$ . Dazu: Füge Einheitszeile in Systemmatrix ein.

## 2.3 Algorithmus (Dirichlet-Randbedingung V1)

Annahme: Assemblierungsalgorithmus 2.1 schon durchlaufen. Dann

```

FOR j = 0, ..., n - 1
  FOR γ = 0, ..., n̂
    IF b(j, γ) = A THEN
      Kk(j,β(γ)),* = 0
      Kk(j,β(γ)),k(j,β(γ)) = 1
      fk(j,β(γ)) = ga
    END IF
  END(γ)
END(j)
    
```

Alternativ: Betrachte Optimierungsaufgabe

$$J := \frac{1}{2}a(u, u) - (f, u) \rightarrow \min \quad \text{mit } u(a) = g_a$$

Definiere

$$J_1 := \frac{1}{2}a(u, u) - (f, u) + \frac{1}{2\varepsilon} \cdot (u(a) - g_a)^2$$

mit  $0 < \varepsilon \ll 1$  (Bestrafung der Nichterfüllung der Randbedingungen). Damit

$$a(u, v) + \frac{1}{\varepsilon} \cdot (u(a) - g_a) \cdot v(a) - (f, v) = 0$$

Statt einen Strafterm einzufügen könnte man auch einen Lagrange-Multiplikator nutzen, um die Nebenbedingung zu implementieren ( $\rightarrow$  Lagrangefunktion). Aber eher unüblich.

**Bemerkung** Bei Systemen von PDE's können unterschiedliche Randbedingungen für unterschiedliche Komponenten gesetzt werden. Zum Beispiel

$$\partial_t c - \Delta \mu = 0 \qquad \mu + \varepsilon^2 \cdot \Delta c - f'(c) = 0$$

mit  $c|_\Gamma = 1$ ,  $\frac{\partial \mu}{\partial n} \Big|_\Gamma = g$ . Gesucht sind also  $c \in V_1$ ,  $\mu \in V_2$  mit

$$\begin{aligned} (\partial_t c, v) + (\nabla \mu, \nabla v) - (g, v)_\Gamma &= 0 \\ (\mu, w) - \varepsilon^2 \cdot (\nabla c, \nabla w) - (f'(c), w) &= 0 \end{aligned}$$

für  $v \in V$ ,  $w \in V_0$ . Man erhält als Systemmatrix (*Rückwärts-Euler-Zeitdiskretisierung*)

$$\begin{pmatrix} \frac{1}{h} D_h & A_h \\ -\varepsilon^2 \cdot A_h - F & D_h \end{pmatrix} \cdot \begin{pmatrix} c^{k+1} \\ \mu^{k+1} \end{pmatrix} = \begin{pmatrix} \frac{1}{h} D_h \cdot c^k + B \cdot C \\ \bar{F} \cdot C^k \end{pmatrix}$$

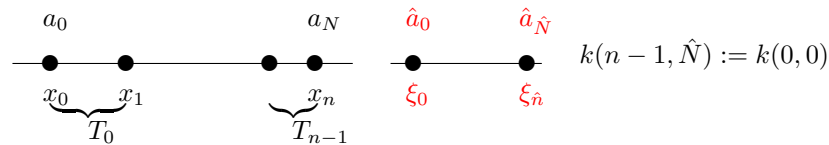
(Neumann-Randbedingung: 1. Zeile, Dirichlet-Randbedingung: 2. Zeile)

**Beispiel** (Periodische Randbedingung) Gesucht ist  $u \in V$  mit

$$u(a) = u(b) \qquad u'(a) = u'(b)$$

(So nicht ganz richtig!) Verschiedene Implementierungsmöglichkeiten:

- (i). Man kann die (Rand)Knoten miteinander identifizieren, d.h. in diesem Fall muss die Abbildung  $(j, \beta) \rightarrow k(j, \beta)$  modifiziert werden.



- (ii). Nach dem Assemblieren wird die Systemmatrix modifiziert. Der Knoten  $k \xrightarrow{p} k_p$  wird den Knoten  $k_p$  zugeordnet. Hier zum Beispiel

$$p(j) = \begin{cases} N & j = 0 \\ 0 & j = N \\ j & j \notin \{0, N\} \end{cases}$$

## 2.4 Algorithmus (Periodische Randbedingungen)

Annahme: Assemblierungsalgorithmus 2.1 schon durchlaufen. Dann

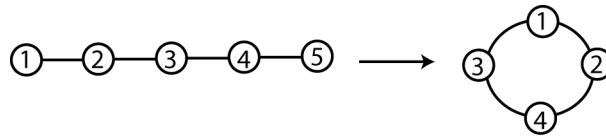
```
FOR k = 0, ..., N
  k_p := p(k)
  FOR l = 0, ..., N
    l_p := p(l)
    IF ((k < k_p) OR (k = k_p AND l < l_p)) THEN
      m := 1/2 * (K_kl + K_k_p l_p)
      K_kl := K_k_p l_p = m
    END IF
  END(l)
END(k)
```

//Analog Lastvektor

```
FOR k = 0, ..., N
  k_p := p(k)
  IF (k < k_p) THEN
    m := 1/2 * (F_k + F_k_p)
    F_k := F_k_p := m
  END IF
END(k)
```

(iii). Fordere  $u_N - u_0 = 0$ . Aber auch Ableitungen müssen übereinstimmen... s. Übung

**Beispiel** (Poisson-Gleichung mit periodischen Randbedingungen)



Betrachte Poisson-Gleichung  $-\Delta u = f$ , Assemblierung mit linearen finiten Elementen.

(i). Durch Identifizierung der Knoten hat man nur noch 4 Freiheitsgrade, man erhält als Systemmatrix dann

$$\begin{pmatrix} 2 & -1 & 0 & -1 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ -1 & 0 & -1 & 2 \end{pmatrix} \cdot \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix} = \begin{pmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{pmatrix}$$

(ii). Assemblierung ohne Randbedingungen gibt

$$\begin{pmatrix} 1 & -1 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & -1 & 1 \end{pmatrix} \cdot \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{pmatrix} = \begin{pmatrix} F'_1 \\ F_2 \\ F_3 \\ F_4 \\ F_5 \end{pmatrix}$$

Nach Algorithmus 2.4 ergibt sich

$$\begin{pmatrix} 1 & -\frac{1}{2} & 0 & -\frac{1}{2} & 0 \\ -\frac{1}{2} & 2 & -1 & 0 & -\frac{1}{2} \\ 0 & -1 & 2 & -1 & 0 \\ -\frac{1}{2} & 0 & -1 & 2 & -\frac{1}{2} \\ 0 & -\frac{1}{2} & 0 & -\frac{1}{2} & 1 \end{pmatrix} \cdot \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{pmatrix} = \begin{pmatrix} \frac{1}{2} \cdot (F'_1 + F_5) \\ F_2 \\ F_3 \\ F_4 \\ \frac{1}{2} \cdot (F'_1 + F_5) \end{pmatrix}$$

Symmetrie wurde erhalten. Ziehe erste Zeile von letzter Zeile ab, dann erhält man gerade die Bedingung  $u_5 - u_1 = 0$ .

Man kann zeigen, dass beide Varianten identische Ergebnisse liefern.

# 3

## Mehrdimensionales FEM - Assemblierung

### 3.1 Simplizes

**Definition** (i). Seien  $x_0, \dots, x_d \in \mathbb{R}^n$ , sodass  $x_1 - x_0, \dots, x_d - x_0$  linear unabhängig sind. Die konvexe Menge  $T := \text{co}\{x_0, \dots, x_d\}$  wird  $d$ -Simplex in  $\mathbb{R}^n$  genannt.

Für  $k < d$  sei  $T' := \text{co}\{x'_0, \dots, x'_k\} \subseteq \partial T$  ein  $k$ -Simplex mit  $\{x_0, \dots, x'_k\} \subseteq \{x_0, \dots, x_d\}$ , dann wird  $T$  ein  $k$ -Sub-Simplex (Facette) von  $T$  genannt.

(ii). Ein 0-Sub-Simplex heißt Knoten. Ein 1-Sub-Simplex heißt Kanten. Ein 2-Sub-Simplex heißt Fläche.

(iii). Ein „Standard-Simplex“ in  $\mathbb{R}^d$  ist definiert als

$$\hat{T} := \text{co}\{\xi_0 := 0, \xi_1 := e_1, \dots, \xi_d := e_d\}$$

mit  $e_i$  Einheitsvektoren des  $\mathbb{R}^d$ .

(iv). Sei  $F_T : \hat{T} \rightarrow T \subseteq \mathbb{R}^n$  eine invertierbare differenzierbare Abbildung, dann wird  $T$  auch parametrischer  $d$ -Simplex in  $\mathbb{R}^n$  genannt.

**Beispiel** (i). Ein Tetraeder ist ein 3-Simplex in  $\mathbb{R}^3$ . Die Kanten sind jeweils 1-Sub-Simplizes.

(ii). Ein Rechteck ist ein 3-Simplex in  $\mathbb{R}^2$ . Die Ränder sind Sub-Simplizes.

**Bemerkung** (i). Sei  $T$  ein parametrischer  $d$ -Simplex in  $\mathbb{R}^n$ . Die  $k$ -Sub-Simplizes  $T'$  von  $T$  ergeben sich aus den Bildern der  $k$ -Sub-Simplizes  $\hat{T}'$  von  $\hat{T}$ . Damit sind insbesondere die Knoten  $x_0, \dots, x_d$  von  $T$  gegeben durch die Punkte  $F_T(\xi_0), \dots, F_T(\xi_d)$ .

(ii). Jeder  $d$ -Simplex in  $\mathbb{R}^n$  ist ein parametrischer Simplex. Definiere dazu die Matrix  $A_T \in \mathbb{R}^{n \times d}$  als

$$A_T := (x_1 - x_0 \quad \dots \quad x_d - x_0)$$

dann ist  $A_T$  invertierbar und es ergibt sich die Parametrisierung  $F_T$  durch

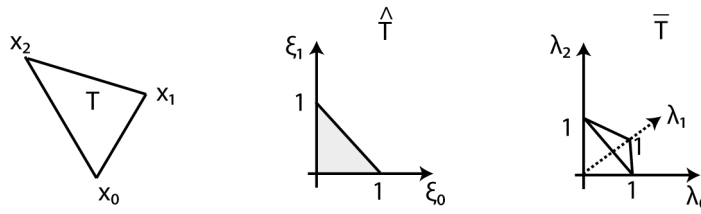
$$F_T(\xi) := A_T \cdot \xi + x_0$$

Offenbar ist  $F_T$  differenzierbar (da affin-linear). Man kann leicht zeigen, dass  $F_T$  auch invertierbar ist und dass  $F_T(e_i) = x_i$  für  $i = 0, \dots, d$ .

**Bemerkung** Sei  $\Omega \subseteq \mathbb{R}^d$  trianguliert mit Elementen aus  $\mathcal{T}$ . Um finite Elemente auf Gittern aus Simplizes zu beschreiben, werden  $d + 1$  baryzentrische Koordinaten als lokales Koordinatensystem auf den Elementen der Triangulierung verwendet.

**Definition** Die Menge  $\bar{T} := \{(\lambda_0, \dots, \lambda_d) \in \mathbb{R}^{d+1}; \forall k : \lambda_k \geq 0, \sum_{k=0}^d \lambda_k = 1\}$  heißt Referenz-Simplex und beschreibt die Menge der baryzentrischen Koordinaten.

**Bemerkung** (Zusammenhang zwischen  $T, \hat{T}, \bar{T}$ )



$T$  sei ein Element der Triangulierung  $\mathcal{T}$  mit Knoten  $x_0, \dots, x_d$ .  $F_T : \hat{T} \rightarrow T$  sei eine Parametrisierung von  $T$  über  $\hat{T}$  mit invertierbarer Jacobi-Matrix  $DF_T$ , sodass  $F(\xi_k) = x_k$  gilt. Für ein Simplex  $T$  sind die baryzentrischen Koordinaten  $\lambda(x) = (\lambda_0, \dots, \lambda_d)(x) \in \mathbb{R}^{d+1}$  für einen Punkt  $x \in \mathbb{R}^d$  eindeutig bestimmt durch die  $(d+1)$  Gleichungen

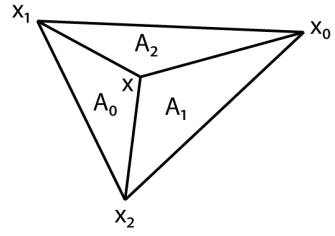
$$\sum_{k=0}^d \lambda_k(x) \cdot x_k = x \qquad \sum_{k=0}^d \lambda_k(x) = 1$$

Es gilt

$$x \in T \Leftrightarrow \forall k = 0, \dots, d : \lambda_k(x) \in [0, 1] \Leftrightarrow \lambda(x) \in \bar{T}$$

Was, wenn  $x \in \mathbb{R}^n$ ,  $x_i \in \mathbb{R}^n$  mit  $n > d$ ? Dann Bestimmung der  $\lambda_k$  durch Flächenverhältnisse: Sei  $A := |T|$  das Volumen von  $T$ , dann ergeben sich die  $\lambda_k$  als

$$\lambda_k = \frac{A_k}{A} \quad (k = 0, \dots, d)$$



(Dann gilt offenbar  $\sum_{k=0}^d \lambda_k = 1$ ). Andererseits definiert jedes  $\lambda \in \bar{T}$  einen eindeutig bestimmten Punkt  $x(\lambda) \in T$  durch

$$x(\lambda) := \sum_{k=0}^d \lambda_k \cdot x_k$$

(Ist invertierbar.) Damit folgt: Jede Abbildung  $f : T \rightarrow V$  kann auf zweierlei Weise parametrisiert werden:

$$\bar{f} : \bar{T} \rightarrow V, \lambda \mapsto f(x(\lambda)) \qquad \hat{f} : \hat{T} \rightarrow V, \xi \mapsto f(F_T(\xi))$$

Somit können auch Funktionenräume auf  $T$ ,  $\hat{T}$ ,  $\bar{T}$  definiert werden, wenn für ein Simplex aus  $\{T, \hat{T}, \bar{T}\}$  gegeben.

**Beispiel** (Polynome vom Grad  $\leq r$ ) Sei  $\bar{\mathbb{P}} := \bar{\mathbb{P}}_r(\bar{T})$  der Raum der Polynome vom Grad  $\leq r$  auf  $\bar{T}$ . Sei nun  $\{\psi_0, \dots, \psi_{\hat{N}}\}$  eine Basis vom Funktionenraum  $\bar{\mathbb{P}}$  und sei  $\{\varphi_0, \dots, \varphi_N\}$  eine Basis von  $V_h$  mit  $\dim V_h = N+1$ , sodass für alle  $T \in \mathcal{T}$  und  $\varphi_k$  mit  $\text{spt } \varphi_k \cap T \neq \emptyset$  gilt:

$$\forall \lambda \in \bar{T} : \varphi_k|_T(x(\lambda)) = \psi_\beta(\lambda)$$

für ein  $\beta = \beta(T, k) \in \{0, \dots, \hat{N}\}$ . Die Umkehrabbildung von  $\beta$  sei  $k : \mathcal{T} \times \{0, \dots, \hat{N}\} \rightarrow \{0, \dots, N\}$ . Dann gilt

$$\begin{aligned} \forall \lambda \in \bar{T}, k \in \{0, \dots, \hat{N}\} : \varphi_k(x(\lambda)) &= \psi_{\beta(T,k)}(\lambda) \\ \forall \lambda \in \bar{T}, \beta \in \{0, \dots, \hat{N}\} : \varphi_{k(T,\beta)}(x(\lambda)) &= \psi_\beta(\lambda) \end{aligned}$$

Darstellung von  $u_h \in V_h$  als

$$u_h(x) = \sum_{k=0}^N U_h \cdot \varphi_k(x) \quad (x \in \Omega)$$

bzw.

$$u_h(x(\lambda)) = \sum_{\beta=0}^{\hat{N}} U_{\beta}^T \cdot \psi_{\beta}(\lambda) \quad (\lambda \in \bar{T})$$

mit  $U_{\beta}^T = U_{k(T,\beta)}$ .

**Bemerkung** (Transformation der Ableitungen) Zur Transformation der (Bi-)Linearform  $a(\cdot, \cdot)$  bzw.  $\langle F, \cdot \rangle$  werden Ableitungen benötigt bzgl. eines Koordinatensystems, d.h. Koordinatenwechsel und Transformation der Ableitungen notwendig. Dazu wird die Jacobimatrix  $\Lambda = \Lambda_T \in \mathbb{R}^{(d+1) \times n}$  der Transformationsabbildung  $T \xrightarrow{\lambda} \hat{T}$  der baryzentrischen Koordinaten benötigt. Es gilt

$$\Lambda(x) = \begin{pmatrix} (\nabla \lambda_0(x))^T \\ \vdots \\ (\nabla \lambda_d(x))^T \end{pmatrix}$$

Für jede Funktion  $\varphi \in \mathbb{P}(T)$  ergibt sich

$$\begin{aligned} \nabla \varphi(x) &= \nabla(\psi(\lambda(x))) = \sum_{\ell=0}^d \frac{d}{d\lambda_{\ell}} \psi(\lambda(x)) \cdot \nabla \lambda_{\ell}(x) \\ &= \Lambda^T(x) \cdot \nabla_{\lambda} \psi(\lambda(x)) \end{aligned}$$

für  $x \in T$ . Analog folgt für die Hesse-Matrix:

$$D^2 \varphi(x) = \Lambda^T \cdot D_{\lambda}^2(\psi(\lambda(x))) \cdot \Lambda(x) + \sum_{\ell=0}^d D_x^2(\lambda_{\ell}(x)) \cdot \frac{d}{d\lambda_{\ell}} \psi(\lambda(x))$$

für  $x \in T$ . Damit ergibt sich

$$\nabla u_h(x) = \Lambda^T(x) \cdot \sum_{\beta=0}^{\hat{N}} U_{\beta}^T \cdot \nabla_{\lambda} \psi_{\beta}(\lambda)$$

für  $x \in T$  bzw. im Umkehrschluss

$$\nabla u_h(\lambda(x)) = \Lambda^T(x(\lambda)) \cdot \sum_{\beta=0}^{\hat{N}} U_{\beta}^T \cdot \nabla_{\lambda} \psi_{\beta}(\lambda)$$

für  $\lambda \in \bar{T}$ . Analog für  $D^2 u_h$ .

### 3.2 Quadraturformeln

Betrachte das Problem

$$a(u, v) = \langle f, v \rangle \quad (v \in V)$$

mit

$$\begin{aligned} a(u, v) &:= \int_{\Omega} \nabla u(x) \cdot A(x) \cdot \nabla v(x) \, dx + \int_{\Omega} b(x) \cdot \nabla u(x) \cdot v(x) + \int_{\Omega} c(x) \cdot u(x) \cdot v(x) \, dx \\ \langle f, v \rangle &:= \int_{\Omega} f(x) \cdot v(x) \, dx \end{aligned}$$

mit  $A(x) \in \mathbb{R}^{n \times n}$ ,  $b(x) \in \mathbb{R}^n$ .

**Bemerkung** (i). Integrale über Polynome können einfach ausgewertet werden, zum Beispiel symbolisch (symbolic C++) oder generisch (d.h. zur Compilezeit, beispielsweise GSSE).

(ii). Wenn  $A$ ,  $b$  oder  $c$  projiziert sind, dann wie oben; sonst numerische Quadratur.

**Definition** Eine numerische Quadratur  $\hat{Q}$  auf  $\hat{T}$  ist eine Menge von Paaren  $\{(\omega_k, \lambda_k) \in \mathbb{R} \times \mathbb{R}^{d+1}; k = 0, \dots, n_Q - 1\}$  von Gewichten  $\omega_k$  und Quadraturpunkten  $\lambda_k \in \hat{T}$  (gegeben in baryzentrischen Koordinaten), sodass

$$\int_{\hat{T}} f(\xi) d\xi \approx \hat{Q}(f) := \sum_{k=0}^{n_Q-1} \omega_k \cdot f(\xi(\lambda_k))$$

**Definition** Eine numerische Quadratur  $\hat{Q}$  heißt

(i). exakt vom Grad  $p$  (für  $p \in \mathbb{N}$ )

$$:\Leftrightarrow \forall q \in \mathbb{P}_p(\hat{T}) : \int_{\hat{T}} q(\xi) d\xi = \hat{Q}(q)$$

(ii). stabil  $:\Leftrightarrow \forall k = 0, \dots, n_Q - 1 : \omega_k > 0$ .

**Bemerkung** Eine gegebene Quadratur  $\hat{Q}$  auf  $\hat{T}$  definiert für alle  $T \in \mathcal{T}$  eine Quadratur  $Q_T$ . Unter Verwendung der Transformationsregel definiert man  $Q_T$  auf einem Element  $T$ , parametrisiert durch  $F_T : \hat{T} \rightarrow T$ , für eine Funktion  $f : T \rightarrow \mathbb{R}$  als

$$\int_T f(x) dx \approx Q_T(f) := \hat{Q}((f \circ F_T) \cdot |\det DF_T|) = \sum_{k=0}^{n_Q-1} \omega_k \cdot f(x(\lambda_k)) \cdot |\det DF_T(\xi(\lambda_k))|$$

Allgemeiner gilt für  $\hat{T} \subseteq \mathbb{R}^d$ ,  $T \subseteq \mathbb{R}^n$  mit  $n \geq d$  die Transformation

$$Q_T(f) := \hat{Q}((f \circ F_T) \cdot \sqrt{g}) = \sum_{k=0}^{n_Q-1} \omega_k \cdot f(x(\lambda_k)) \cdot \sqrt{g(\xi(\lambda_k))}$$

mit der Gramschen Determinante

$$g(\xi) := \det(DF_T(\xi)^T \cdot DF_T(\xi))$$

Für einen Simplex  $T$  mit  $\hat{T} \subseteq \mathbb{R}^d$  ergibt sich

$$Q_T(f) = d! \cdot |T| \cdot \sum_{k=0}^{n_Q-1} \omega_k \cdot f(x(\lambda_k))$$

**Beispiel (Gauß-Quadratur (1D))** Es gibt genau eine Quadraturformel zu  $n_Q$  paarweise verschiedenen Stützstellen über dem Intervall  $[-1, 1]$ , die exakt ist vom Grad  $2 \cdot n_Q$ . Ihre Stützstellen sind gerade die Nullstellen  $\xi_0, \dots, \xi_{n_Q-1} \in (-1, 1)$  des  $n_Q$ -ten Legendre-Polynoms  $L_{n_Q} \in \mathbb{P}_{n_Q}$ . Damit lassen sich die Gewichte  $\omega_k$  ableiten,

$$\omega_k := \int_{-1}^1 \prod_{\substack{j=0 \\ j \neq k}}^{n_Q-1} \left( \frac{x - \xi_j}{\xi_k - \xi_j} \right)^2 dx > 0$$

(Es handelt sich also um eine stabile Quadratur.) Alternativ:

$$\omega_k = \frac{2}{1 - \xi_k^2} \cdot \left( \frac{1}{L'_{n_Q}(\xi_k)} \right)^2$$

**Bemerkung** Hier Stützstellen von Gewichten für „Standardintervall“  $[-1, 1]$ , eventuell Transformation auf  $[0, 1]$  nötig.

**Bemerkung** (Legendre-Polynome) Die Legendre-Polynome sind gegeben durch

$$L_n(x) := \frac{1}{2^n \cdot n!} \cdot \frac{d^n}{dx^n} ((x^2 - 1)^n)$$

(Rodrigues-Formel) und genügen den Rekursionsformeln

$$(n + 1) \cdot L_{n+1}(x) = (2n + 1) \cdot x \cdot L_n(x) - n \cdot L_{n-1}(x)$$

mit  $L_0 := 0$ ,  $L_1(x) := x$

$$(x^2 - 1) \cdot \frac{d}{dx} L_n(x) = n \cdot x \cdot L_n(x) - n \cdot L_{n-1}(x)$$

Die Nullstellen lassen sich mit Newton-Verfahren und den beiden Rekursionsformeln für Wert von Ableitungen bestimmen. Dazu kann man als Startwert für die  $k$ -te Nullstelle von  $L_n$  den Wert

$$x_k := \cos \left( \pi \cdot \frac{4k - 1}{4n + 2} \right)$$

verwenden.

**Beispiel**

$$\int_0^1 f(\xi) d\xi \approx \hat{Q}^{(i)}(f)$$

mit

$$\begin{aligned} \hat{Q}^{(1)}(f) &:= f \left( \frac{1}{2} \right) \\ \hat{Q}^{(2)}(f) &:= \frac{1}{2} \cdot \left( f \left( \frac{1}{2} \cdot \left( 1 - \frac{1}{\sqrt{3}} \right) \right) + f \left( \frac{1}{2} \cdot \left( 1 + \frac{1}{\sqrt{3}} \right) \right) \right) \\ \hat{Q}^{(3)}(f) &:= \frac{1}{2} \cdot \left( \frac{5}{9} \cdot f \left( \frac{1}{2} \cdot \left( 1 - \frac{\sqrt{3}}{\sqrt{5}} \right) \right) + \frac{8}{9} \cdot f \left( \frac{1}{2} \right) + \frac{5}{9} \cdot f \left( \frac{1}{2} \cdot \left( 1 + \frac{\sqrt{3}}{\sqrt{5}} \right) \right) \right) \end{aligned}$$

In baryzentrischen Koordinaten kann man Symmetrie der Formel ausgenutzt bzw. erzwungen werden. Baryzentrische Koordinaten sind zudem invariant unter affinen Transformationen (Übung DIY).

**Definition** Eine Quadratur  $\hat{Q}$  wird symmetrisch genannt, wenn sie invariant unter Permutationen der baryzentrischen Koordinaten ist, d.h. für jede Quadraturstelle  $\lambda = (\lambda_0, \dots, \lambda_d)$  von  $\hat{Q}$  bzgl. des Gewichtes  $\omega$  ist für jede Permutation  $\pi$  der Indizes auch  $(\lambda_{\pi(0)}, \dots, \lambda_{\pi(d)})$  auch eine Quadraturstelle von  $\hat{Q}$  bzgl. des Gewichtes  $\omega$ .

**Beispiel**

$$\hat{Q}^{(1)}(f) = 1 \cdot f \left( \frac{1}{2}, \frac{1}{2} \right)$$

mit Permutationsstern  $S_2 \left( \frac{1}{2} \right) = \left( \frac{1}{2}, \frac{1}{2} \right)$ .

$$\hat{Q}^{(2)}(f) = \frac{1}{2} \cdot (f(a, b) + f(b, a))$$

mit  $a := \frac{1}{2} \cdot \left( 1 + \frac{1}{\sqrt{3}} \right)$ ,  $b := 1 - a$ , Permutationsstern  $S_{11}(a) = (a, 1 - a)$ .

$$\hat{Q}^{(3)}(f) = \frac{5}{18} \cdot (f(a', b') + f(b', a')) + \frac{8}{18} \cdot f \left( \frac{1}{2}, \frac{1}{2} \right)$$

mit Permutationssternen  $S_2 \left( \frac{1}{2} \right)$  und  $S_{11}(a') = (a', 1 - a')$  wobei  $a' = \frac{1}{2} \cdot \left( 1 + \frac{\sqrt{3}}{\sqrt{5}} \right)$ ,  $b' = 1 - a'$ .



**Bemerkung** Unter Ausnutzung folgender Beziehung lassen sich die Formeln für Dreiecke und Tetraeder herleiten:

$$\int_{\hat{T}} \xi_0^{k_0} \cdot \xi_1^{k_1} d(\xi_0, \xi_1) = \frac{k_0! \cdot k_1!}{(2 + k_0 + k_1)!}$$

für Standarddreieck mit  $k_0 \leq k_1$ ,

$$\int_{\hat{T}} \xi_0^{k_0} \cdot \xi_1^{k_1} \cdot \xi_2^{k_2} d(\xi_0, \xi_1, \xi_2) = \frac{k_0! \cdot k_1! \cdot k_2!}{(3 + k_0 + k_1 + k_2)!}$$

für Standardtetraeder wobei  $k_0 \leq k_1 \leq k_2$ . Für Dreiecke ergeben sich folgende Formeln:

$n_Q$	Ordnung	Gewichte	Permutationssterne
1	1	1	$S_3\left(\frac{1}{3}\right) = \left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right)$
3	2	$\frac{1}{3}$	$S_{21}(a) = (a, a, 1 - 2a)$ , $a = \frac{1}{6}$
6	4	$\frac{620 \pm \sqrt{213125} - 53320 \cdot \sqrt{10}}{3720}$	$S_{21}(a_{\pm})$ mit $a_{\pm} = \frac{8 - \sqrt{10} \pm \sqrt{38 - 44 \cdot \sqrt{\frac{2}{5}}}}{18}$

Für Tetraeder ergeben sich folgende Formeln:

$n_Q$	Ordnung	Gewichte	Permutationssterne
1	1	1	$S_4\left(\frac{1}{4}\right)$
4	2	$\frac{1}{4}$	$S_{31}(a) = (a, a, a, 1 - 3a)$ , $a = \frac{5 - \sqrt{5}}{20}$

Für Rechtecke/Quader: Tensorproduktformeln. Beispielsweise

$$\begin{aligned} \int_{\hat{T}} f(\xi_0, \xi_1) d(\xi_0, \xi_1) &= \int_{-1}^1 \int_{-1}^1 f(\xi_0, \xi_1) d\xi_0 d\xi_1 \\ &\approx \int_{-1}^1 \sum_{k=0}^{n_Q^{(0)}-1} \omega_k^{(0)} \cdot f(x_k, \xi_1) d\xi_1 \\ &\approx \sum_{\ell=0}^{n_Q^{(1)}-1} \sum_{k=0}^{n_Q^{(0)}-1} \omega_{\ell}^{(1)} \cdot \omega_k^{(0)} \cdot f(x_k, y_{\ell}) \\ &=: \sum_{k=0}^{n_Q-1} \omega_k \cdot f(x_k^{(0)}, x_k^{(1)}) \end{aligned}$$

für  $\hat{T} = [-1, 1] \times [-1, 1]$ .

Literatur:

- „A set of symmetric quadrature rules on triangles and tetraedra“, L. Zhang, T. Lui, A. Liu, Journal of Computational Mathematics 2009
- „A compendium of FEM integration rules“, Felippa

### 3.2.1 Assemblierung

Zur Assemblierung des Problems

$$\bar{a}(u_h, v_h) = \langle \bar{f}, v_h \rangle$$

mit

$$\begin{aligned} \bar{a}(u_h, v_h) &= (A \nabla u_h, v_h) + (b \cdot \nabla u_h, v_h) + (c \cdot u_h, v_h) \\ \langle \bar{f}, v_h \rangle &= (f, v_h) \end{aligned}$$

werden analog zu 1D (siehe Algorithmus 2.1) die Elementmatrix-/vektor aufgebaut:

$$E(\beta, \alpha) = \int_{T_j} (A \cdot \nabla \varphi_{k(j,\beta)})^T \cdot \nabla \varphi_{k(j,\alpha)} + b \cdot \nabla \varphi_{k(j,\alpha)} \cdot \varphi_{k(j,\beta)} + c \cdot \varphi_{k(j,\beta)} \cdot \varphi_{k(j,\alpha)} dx$$

$$e(\beta) = \int_{T_j} f(x) \cdot \varphi_{k(j,\beta)}(x) dx$$

Einsetzen der Transformationen ergibt

$$E(\beta, \alpha) = \int_{T_j} \nabla_\lambda \psi_\beta(\lambda(x)) \cdot \Lambda(x) \cdot A(x) \cdot \Lambda^T(x) \cdot \nabla_\lambda \psi_\alpha(\lambda(x)) dx$$

$$+ \int_{T_j} \psi_\beta(\lambda(x)) \cdot (\Lambda(x) \cdot b(x)) \cdot \nabla_\lambda \psi_\alpha(x) dx + \int_{T_j} c(x) \cdot \psi_\beta(\lambda(x)) \cdot \psi_\alpha(\lambda(x)) dx$$

$$= \int_{\hat{T}} \nabla_\lambda \psi_\beta(\lambda(\xi)) \cdot \bar{A}(\lambda(\xi)) \cdot \nabla_\lambda \psi_\alpha(\lambda(\xi)) d\xi$$

$$+ \int_{\hat{T}} \psi_\beta(\lambda(\xi)) \cdot \bar{b}(\lambda(\xi)) \cdot \nabla_\lambda \psi_\alpha(\lambda(\xi)) d\xi + \int_{\hat{T}} \bar{c}(\lambda(\xi)) \cdot \psi_\beta(\lambda(\xi)) \cdot \psi_\alpha(\lambda(\xi)) d\xi$$

mit

$$\bar{A}(\lambda) = |\det JF_T(\xi(\lambda))| \cdot \Lambda(x(\lambda)) \cdot A(x(\lambda)) \cdot \Lambda^T(x(\lambda))$$

$$\bar{b}(\lambda) = |\det JF_T(\xi(\lambda))| \cdot \Lambda(x(\lambda)) \cdot b(x(\lambda))$$

$$\bar{c}(\lambda) = |\det JF_T(\xi(\lambda))| \cdot c(x(\lambda))$$

Einsetzen der Quadraturformeln

$$E(\beta, \alpha) = \hat{Q}_2(\nabla_\lambda \psi_\beta(\lambda) \cdot \bar{A}(\lambda) \cdot \nabla_\lambda \psi_\alpha(\lambda))$$

$$+ \hat{Q}_1(\psi_\beta(\lambda) \cdot \bar{b}(\lambda) \cdot \nabla_\lambda \psi_\alpha(\lambda))$$

$$+ \hat{Q}_0(\bar{c}(\lambda) \cdot \psi_\beta(\lambda) \cdot \psi_\alpha(\lambda))$$

wobei  $\hat{Q}_i$  Quadratur exakt für Terme  $i$ -ter Ordnung. Analog für  $e$ .

### 3.2.2 Mass-Lumping

Für einige Probleme ist es vorteilhaft, wenn die Massematrix eine Diagonalmatrix ist, zum Beispiel im Hinblick auf

- explizites Euler-Verfahren für parabolische Differentialgleichungen

$$D_h U^{\ell+1} = (D_h - k \cdot A_h) U^\ell$$

(Diagonalmatrix ermöglicht einfaches Invertieren.)

- insbesondere ist  $D_h$  i.A. keine M-Matrix, auch  $(D_h + k \cdot A_h)$  nicht zwingend M-Matrix.

Mass-Lumping: „Approximation von  $D_h$  durch Diagonalmatrix“. Idee: Approximiere  $(\cdot, \cdot)$  so, dass Basis  $\{\varphi_j\}_j$  von  $V_h$  Orthogonalsystem bildet bzgl. neuem Skalarprodukt  $(\cdot, \cdot)_h$ ,

$$(u, v)_h := \sum_{\ell} \omega_{\ell,h} \cdot u(\hat{x}_\ell) \cdot v(\hat{x}_\ell) \quad (3.1)$$

mit  $\hat{x}_\ell$  Quadraturstellen,  $\omega_{\ell,h}$  Quadraturgewichte.

#### 3.1 Lemma

Für Lagrange-FE gilt: Wenn die Knoten  $a_k$  des FE-Raumes  $V_h$  und die Quadraturstellen  $\hat{x}_\ell$  zusammenfallen (d.h.  $\{\hat{x}_\ell; \ell\} = \{a_k; k\}$ ), dann handelt es sich um ein Orthogonalsystem bzgl.  $(\cdot, \cdot)_h$  aus (3.1).

**Bemerkung** Damit feste Quadraturstellen. Gewichte  $\omega_{\ell,h}$  können so bestimmt werden, dass möglichst hohe Quadraturordnung erreicht wird.

**Beispiel** (Reguläre Dreiecksgitter mit  $P^1$ -Elementen)

$$Q^{\text{Lump}}(f) := \frac{|T|}{3} \cdot \sum_{i=0}^2 f(\hat{a}_i)$$

Entspricht Trapezregel. Integrationsfehler  $\mathcal{O}(h^2)$ .

**Bemerkung** Problem: Bei einigen FE-Räumen führt Quadraturansatz zu nicht positiven-definiten Matrizen oder negativen Quadraturgewichten. Idee: Hinzunahme einer zusätzlicher Basisfunktion (z.B. Bubble-Funktion).

Literatur: „Higher Order Triangular Finite Elements with Mass-Lumping for the Wave equation“, S. Cohen, P. Joly, E. Roberts, N. Tordjman (2001)

### 3.3 Randbedingungen

Betrachte Variationsproblem

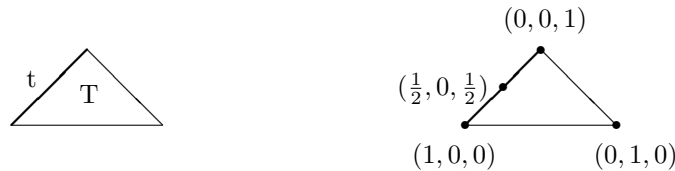
$$a(u, v) = \langle f, v \rangle \quad (v \in V)$$

mit Randtermen:

$$a(u, v) = \bar{a}(u, v) + \int_{\Gamma_B} \mu \cdot u \cdot v \, dS$$

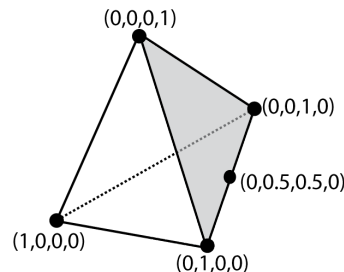
$$\langle f, v \rangle = \langle \bar{f}, v \rangle + \int_{\Gamma_B} \mu \cdot g_B \cdot v \, dS$$

Betrachte zunächst einen Sub-Simplex  $t$  eines Simplexes  $T$ .



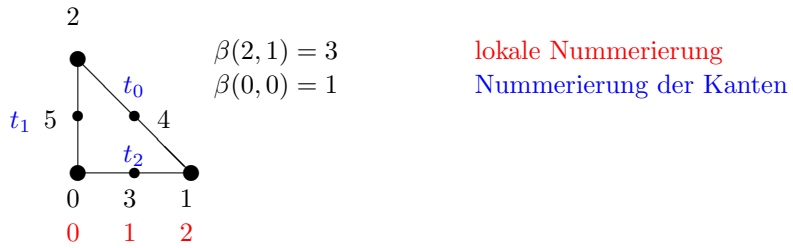
Jeder Randkante wird eine Nummer zugeordnet. Vorteilhaft: Verwendung von baryzentrischen Koordinaten. Baryzentrische Koordinaten haben am Rand eine Komponente = 0, diese gehöre zu dem gegenüberliegenden Gitterknoten.

**Beispiel** (Tetraeder)



Betrachte Standardelement  $\hat{T}$  mit (vorgegebener) Standardnummerierung der Freiheitsgrade. Analog wie  $k(\cdot, \cdot)$  einer lokalen Knotennummer auf  $T$  eine globale Nummer in  $\Omega$  zuweist, definiert man  $\beta : \{0, \dots, \hat{n}\} \times \{0, \dots, \hat{N}'\} \rightarrow \{0, \dots, \hat{N}\}$  als Abbildung, die einen Knoten aus  $\{0, \dots, \hat{N}'\}$  auf  $t_i$  mit  $i \in \{0, \dots, \hat{n}\}$  auf eine Knotennummer  $\{0, \dots, \hat{N}\}$  in  $T$  abbildet, d.h.  $\beta(\gamma, \delta)$  ist die lokale Freiheitsgradnummer des  $\delta$ -ten Knotens auf der Kante  $t_\gamma$ .

**Beispiel**



Analog wie in 1D wird auch eine Randbedingungsnummer zugewiesen, mittels der Abbildung

$$b : \mathcal{T} \times \{0, \dots, \hat{n}\} \rightarrow \mathbb{Z}, (T, \gamma) \mapsto \begin{cases} 0 & t_\gamma \text{ keine Randkante} \\ b & \text{Randbedingungsnummer von Kante } t_\gamma \end{cases}$$

**3.2 Algorithmus**

```

FOR j = 0, ..., n - 1 (Elemente traversieren)
  FOR gamma = 0, ..., n-hat (Elementkanten durchlaufen)
    IF b(T_j, gamma) = B THEN
      E=zeros(0...N-hat', 0...N-hat')
      e=zeros(0...N-hat', 1)

      FOR beta' = 0, ..., N-hat'
        e(beta') = integral_{t_gamma} mu * g_B(x) * phi_{k(j, beta(gamma, beta'))}(x) dS(x)

        FOR alpha' = 0, ..., N-hat'
          E(beta', alpha') = integral_{t_gamma} mu * phi_{k(j, beta(gamma, alpha'))}(x) * phi_{k(j, beta(gamma, beta'))}(x) dS(x)
        END(alpha')
      END(beta')
    END IF
  END(gamma)
END(j)

```

Analog wie  $F_T$  kann  $F_t : \mathbb{R}^d \rightarrow \mathbb{R}^{d-1}$  definiert werden, für Transformation der Randkante  $t$  auf das Standardelement bzw. Referenzelement (d.h. Transformation des Integrals mittels Gramscher Determinante).

**Bemerkung** Bei polygonal (oder polyhedral) berandeten Gebiet  $\Omega$  sind die Randelemente  $t$  Simplex,  $|\det JF_t|$  beschreibt das Volumen des Simplex  $t$  (siehe Übung 10, Aufgabe 2).

$$\int_{t_\gamma} f(x) dS(x) = \int_{\hat{t}} \mu \cdot \psi_{\beta(\gamma, \alpha')}(\lambda(\xi)) \cdot \psi_{\beta(\gamma, \beta')}(\lambda(\xi)) \cdot f(x(\xi)) \cdot \sqrt{g_t(\xi)} d\xi$$

$$= \hat{Q}_\Gamma(\mu \cdot \psi_{\beta(\gamma, \alpha')}(\lambda) \cdot \psi_{\beta(\gamma, \beta')}(\lambda) \cdot f(x) \cdot \sqrt{g_t})$$

mit  $\hat{Q}_\Gamma$  Quadraturformel auf  $\hat{t}$  exakt für Randterme.

**Beispiel** Für Seite eines Dreiecks/Vierecks ergibt sich die Transformation

$$x_t(\xi) = \begin{pmatrix} x_{1,0}^\perp - x_{0,0}^\perp \\ x_{1,1}^\perp - x_{0,1}^\perp \end{pmatrix} \cdot \xi + \begin{pmatrix} x_{0,0}^\perp \\ x_{0,1}^\perp \end{pmatrix}$$

wobei  $x_0^\perp, x_1^\perp$  Begrenzungspunkte von  $t$ . Weiterhin gilt

$$\sqrt{g} = \sqrt{(x_{1,0}^\perp - x_{0,0}^\perp)^2 + (x_{1,1}^\perp - x_{0,1}^\perp)^2}$$

# 4

## Gittergenerierung

### 4.1 Verfeinerung

Idee: Nutze Simplizes. Einfacher Geometrietyp, geeignet für komplizierte Geometrien. (*Wiederholung Definition Simplex, Untersimplex, Standardelement*)

**Bemerkung** (Notationen)

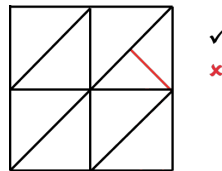
$$h_S := \text{diam}(S) \qquad \varrho_S := \sup\{2r; \exists x \in S : B(x, r) \subseteq S\}$$

**Definition** Sei  $\Omega \subseteq \mathbb{R}^n$  offen. Eine Triangulierung ist eine Menge  $\mathcal{S}$  von  $d$ -Simplizes mit

$$\Omega = \text{int} \left( \bigcup_{S \in \mathcal{S}} S \right)$$

$\mathcal{S}$  heißt eine konforme Triangulierung, wenn für  $S_1, S_2 \in \mathcal{S}$  mit  $S_1 \neq S_2$  gilt  $S_1 \cap S_2 = \emptyset$  oder  $S_1 \cap S_2$  ist ein  $k$ -Sub-Simplex für beide Simplizes.

**Beispiel**



**Definition** Sei  $(\mathcal{S}_k)_{k \geq 0}$  eine Folge von konformen Triangulierungen. Sie heißt regulär, falls

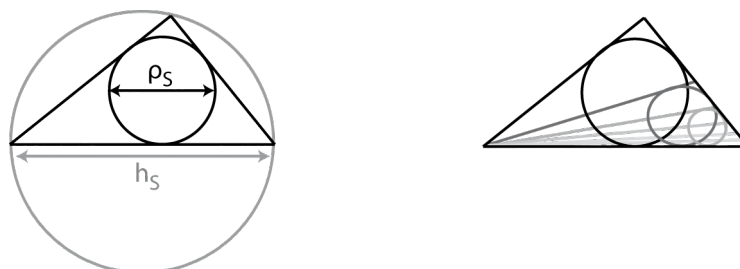
$$\sup_{k \in \mathbb{N}_0} \sup_{S \in \mathcal{S}_k} \max_{\hat{x} \in \hat{S}} \text{cond}(DF_S^T(\hat{x}) \cdot DF_S(\hat{x})) < \infty$$

wobei  $DF_S$  die Jacobi-Matrix von  $F_S : \hat{S} \rightarrow S$  mit  $F_S(\hat{x}) = A_S \cdot \hat{x} + a_0$  und  $\text{cond}(A) = \|A\| \cdot \|A^{-1}\|$ .

**Bemerkung** Dies ist äquivalent zu

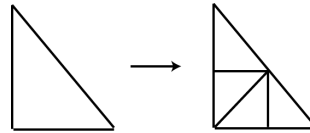
$$\sup_{k \in \mathbb{N}_0} \sup_{S \in \mathcal{S}_k} \frac{h_S}{\varrho_S} < \infty$$

**Beispiel** (Nichtkonforme Triangulierung) Immer weitere Verfeinerung der einen Kante führt zu  $\frac{h_S}{\varrho_S} \rightarrow \infty$ .

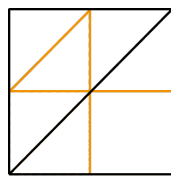


**Bemerkung** Eine Folge von Triangulierungen kann aus Grobgitter durch Verfeinerung erzeugt werden. Es gibt unterschiedliche Verfeinerungsstrategien:

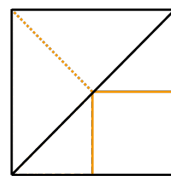
- reguläre Verfeinerung („red“-Refinement): zum Beispiel 2D gibt 4 neue Teildreiecke, 3D 8 neue Teiltetraeder.



Vorteil: Elemente sind ähnlich. Nachteil: keine einfache lokale Verfeinerung möglich. Entstehende hängende Knoten müssen durch Bisektion aufgelöst werden („green“-Refinement).

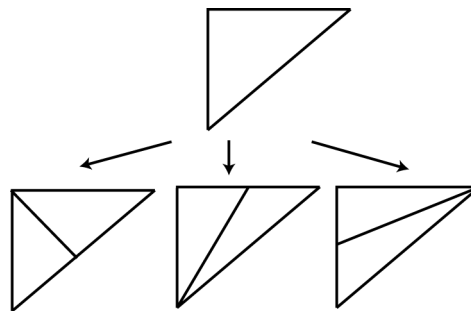


Red refinement ✓



Red green refinement

- Bisektion: Benötigt wird eine „Verfeinerungskante“,

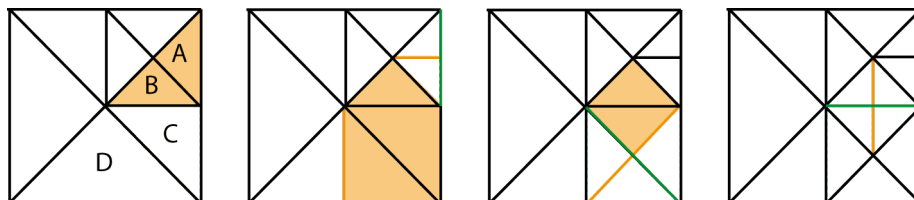


zum Beispiel

- (i). längste Kante
- (ii). neueste Kante

Alle Elemente, die auch die Verfeinerungskante beinhalten, müssen auch verfeinert werden.

**Beispiel** Verfeinerung mit Bisektion, wähle längste Kante als Verfeinerungskante. Ziel:  $A, B$  verfeinern.  $A$  unproblematisch. um  $B$  zu verfeinern, muss zunächst  $C$  und  $D$  verfeinert werden. (Orange Simpleses sollen/müssen verfeinert werden, die grüne Kante ist die Verfeinerungskante.)



#### 4.1 Algorithmus (Verfeinerung)

```

subroutine recursive_refine (S, S)
  A = {S' ∈ S; S' kann nicht gemeinsam mit S geteilt werden}
  do

```

```

for all  $S' \in \mathcal{A}$  do
    recursive_refine ( $S', \mathcal{S}$ )
end for

 $\mathcal{A} := \{S' \in \mathcal{S}; S' \text{ kann nicht gemeinsam mit } S \text{ geteilt werden}\}$ 
until  $\mathcal{A} = \emptyset$ 

 $\mathcal{A} = \{S' \in \mathcal{S}; S' \text{ ist Element an Verfeinerungskante von } S\}$ 

for all  $S' \in \mathcal{A}$ 
    bisect  $S'$  into  $S'_0, S'_1$ 
     $\mathcal{S} = \mathcal{S} \setminus \{S'\} \cup \{S'_0, S'_1\}$ 
end for

subroutine refine ( $\mathcal{S}$ )
    for all  $S \in \mathcal{S}$  do
        if  $S$  ist markiert fuer Verfeinerung
            recursive_refine ( $S, \mathcal{S}$ )
        end if
    end for
end subroutine

```

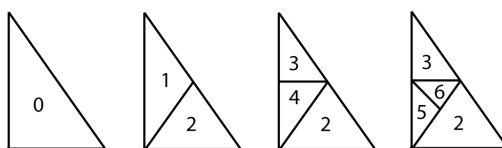
Was muss gemacht werden bei Verfeinerung?

- Verfeinerung des Gitters
- Administration der Freiheitsgrade (an Ecken, Kanten, Seiten,...)
- geometrische Information für neue Freiheitsgrade
- Transformation der Finite Elemente Informationen auf neue Gitter, z.B. durch Interpolation

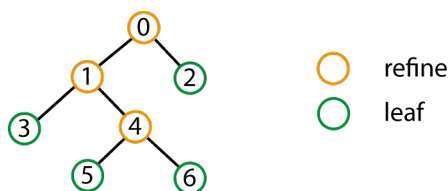
$$u_h(\text{midpoint}) = \frac{1}{2} \cdot (u_h(\text{vertex } 0) + u_h(\text{vertex } 1))$$

- Bemerkung** (Wie speichert man Gitter?)
- (i). Elemente als Vektoren, alle Informationen im Element gespeichert
  - (ii). Informationen über Hierarchie der Gitterverfeinerung, geometrische Informationen nur auf Grobgitter. Speziell für Bisektion erhält man einen Binärbaum. Speicherung z.B. mit pre-order traverse oder mesh-structure-code.

**Beispiel**



Der Binärbaum ist hier gegeben durch



Speicherung:

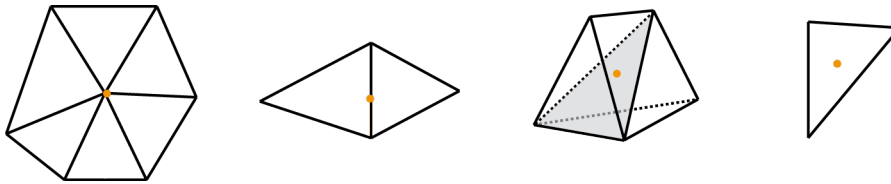
- (i). pre-oder traverse: 0134562
- (ii). mesh-structure-code (leaf  $\hat{=}$  0, refine  $\hat{=}$  1): 1101000. Entspricht dem Dezimalwert 104.

**Bemerkung** Weiterer Ansatz: Setze beliebig viele Punkte in die gegebene Menge. Diese seien über Federn miteinander verbunden. Finde physikalisches Gleichgewicht. Führt zu einem Gitter. Matlab: Dist Mesh. „A simple Mesh Generator in Matlab“, P.-O. Persson, G. Strang, SIAM Review 46 (2004) 329

## 4.2 Verbindung zwischen geometrischen und Finite Elemente Daten

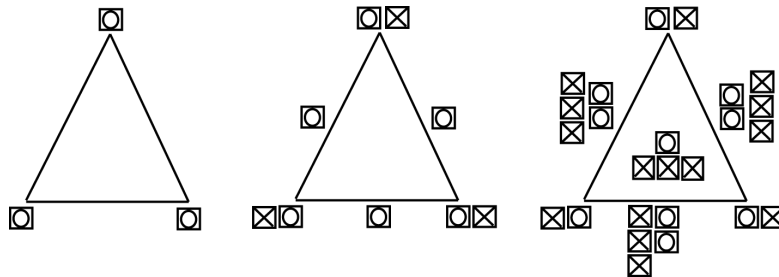
**Bemerkung** Geometrische Informationen (Koordinaten) sind nur auf Makro-Ebene (größtes Gitter) gespeichert und werden für verfeinerte Elemente berechnet. Vorteil: wenig Speicherbedarf. Nachteil: größerer Rechenaufwand, um Nachbarn eines Elementes zu ermitteln.

**Beispiel** Stetige stückweise lineare FE Funktionen sind durch Wert in den Knoten beschrieben. Freiheitsgrade können sich aber auch auf Kanten, auf Flächen oder im Inneren der Elemente definiert werden. Freiheitsgrade werden somit zum Teil zwischen mehreren Elementen geteilt.



**Bemerkung** Eventuell sind unterschiedliche Ansatzfunktionen auf einer Triangulierung zu verwenden, zum Beispiel Navier-Stokes-Gleichung: stückweise quadratische Elemente für Geschwindigkeit, stückweise lineare Elemente für Druck.

- Beispiel (Speicherbedarf)**
- (i). stückweise quadratische Elemente (links im Bild)
  - (ii). stückweise quadratisch und stückweise linear (Mitte)
  - (iii). stückweise kubisch und stückweise 4. Ordnung (rechts im Bild)

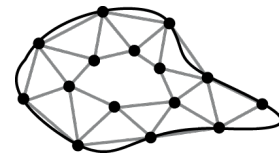


## 4.3 Generierung einer Makrotriangulierung

Gegeben sei ein Gebiet  $\Omega$ , gesucht geeignete Makrotriangulierung.

**Beispiel**

- (i). Diskretisiere Rand.
- (ii). Füge weitere Punkte ins Innere (oder verbinde Punkte) hinzu, so dass die Abstände zwischen den Punkten ungefähr gleich groß sind und die Winkel  $\approx 60^\circ$  sind.
- (iii). Wiederhole den Vorgang bis Gebiet vollständig trianguliert. (Fasse dazu den entstandenen Rand als neuen äußeren Rand auf.)

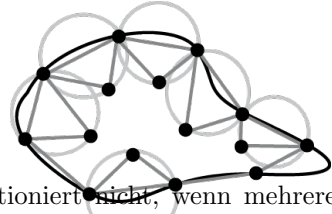


Problem: Approximation von krummlinigen Rändern. Statt Dreiecken können deshalb zum Beispiel auch isoparametrische Dreiecke zur Triangulierung verwendet werden ( $\rightarrow$  isogeometrische Analysis). Basisfunktionen sind NURBS (non-uniform rational B-splines).

**Beispiel (Delauney-Triangulierung)**



- (i). Diskretisiere den Rand.
- (ii). Gegeben sei eine Menge von Punkten im Inneren. Wähle Triangulierung so, dass kein Punkt innerhalb der einhüllenden Kreise liegt.



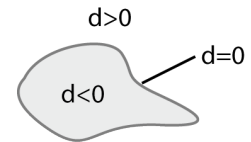
Damit kleinste Winkel in den Dreieck maximiert. Probleme: Funktioniert nicht, wenn mehrere Punkte auf einer Linie liegen bzw. wenn 4 oder mehr Punkte auf einem Kreis liegen. Die Triangulierung ist nicht eindeutig.

**Beispiel** (Dist Mesh) (i). Definiere Geometrie durch signierte Distanzfunktion.

- (ii). Verteile Punkte im Inneren.
- (iii). Bewege Punkte so, dass Kräftegleichgewicht vorliegt.
- (iv). Abbruchkriterium

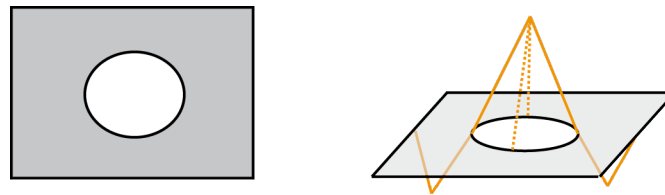
**Definition** (signierte Distanzfunktion)

$$d(x) \begin{cases} < 0 & x \in \Omega \\ = 0 & x \in \partial\Omega \\ > 0 & x \in \Omega^c \end{cases}$$

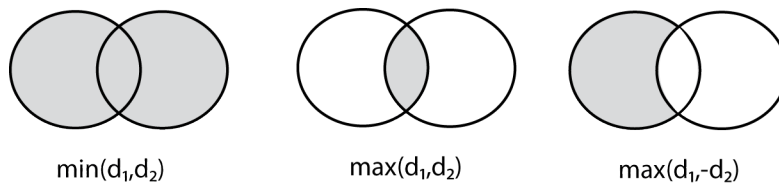


mit Metrik in geeigneter Norm, z.B. euklidische Norm.

**Beispiel**



**Bemerkung** Geometrie wird implizit durch  $d$  beschrieben. Es gilt z.B.



**Bemerkung** Idee: Bette (zu triangulierendes) Gebiet  $\Omega$  in „einfaches“ Gebiet  $\bar{\Omega}$  ein. Definiere Grobgitter-Triangulierung für  $\bar{\Omega}$  und verfeinere lokal, sodass  $\partial\Omega$  möglichst gut approximiert wird. Gegebenfalls ist eine Anpassung der Differentialgleichung auf  $\bar{\Omega}$  notwendig.

**Beispiel**

$$\Delta u = f \quad \text{in } \Omega$$

mit Randbedingungen

- (i).  $u = g$  auf  $\partial\Omega$
- (ii).  $\nabla u \cdot n = g$  auf  $\partial\Omega$
- (iii).  $\nabla u \cdot n = k(u - g)$  auf  $\partial\Omega$

Dann entsprechende Anpassung:

(ii) Die Gleichung

$$-\int_{\Omega} \nabla u \cdot \nabla \psi + \int_{\partial\Omega} g \cdot \psi \, ds = \int_{\Omega} f \cdot \psi$$

wird zu

$$-\int_{\bar{\Omega}} 1_{\Omega} \cdot \nabla u \cdot \nabla \psi + \int_{\bar{\Omega}} \delta_{\partial\Omega} \cdot g \cdot \psi = \int_{\bar{\Omega}} 1_{\Omega} \cdot f \cdot \psi$$

wobei  $\delta_{\partial\Omega}$  die Deltadistribution bezeichnet. Partielle Integration gibt (Voraussetzungen eigentlich nicht erfüllt...)

$$\int_{\bar{\Omega}} (\nabla(1_{\Omega} \cdot \nabla u) + \delta_{\partial\Omega} g - 1_{\Omega} f) \cdot \psi = 0$$

also starke Formulierung

$$\nabla(1_{\Omega} \cdot \nabla u) + \delta_{\partial\Omega} g = 1_{\Omega} f \quad \text{in } \bar{\Omega}$$

(iii) Analog.

(i) Schwache Formulierung ist gegeben durch

$$-\int_{\Omega} \nabla u \cdot \nabla \psi + \int_{\partial\Omega} \frac{\partial u}{\partial n} \cdot \psi = \int_{\Omega} f \cdot \psi$$

Erneute partielle Integration (vorausgesetzt alles hinreichend glatt) gibt

$$\int_{\Omega} u \cdot \Delta \psi + \int_{\partial\Omega} \left( \frac{\partial u}{\partial n} \cdot \psi - g \cdot \frac{\partial \psi}{\partial n} \right) = \int_{\Omega} f \cdot \psi$$

Analoges Vorgehen wie in (ii):

$$\int_{\bar{\Omega}} 1_{\Omega} \cdot u \cdot \Delta \psi + \int_{\bar{\Omega}} \delta_{\partial\Omega} \left( \frac{\partial u}{\partial n} \psi - g \cdot \frac{\partial \psi}{\partial n} \right) = \int_{\bar{\Omega}} 1_{\Omega} \cdot f \cdot \psi$$

Es gilt  $\nabla 1_{\Omega} = -\delta_{\partial\Omega} \vec{n}$ . Annahme:  $g$  ist konstant in Normalenrichtung. Damit

$$\begin{aligned} \int_{\bar{\Omega}} \delta_{\partial\Omega} \frac{\partial u}{\partial n} \cdot \psi &= - \int_{\bar{\Omega}} \psi \cdot \nabla 1_{\Omega} \cdot \nabla u \\ \int_{\bar{\Omega}} \delta_{\partial\Omega} g \cdot \frac{\partial \psi}{\partial n} &= \int_{\bar{\Omega}} \psi \cdot g \cdot \nabla \nabla 1_{\Omega} \end{aligned}$$

Man erhält somit

$$\int_{\bar{\Omega}} \psi (\nabla(1_{\Omega} \nabla u) + (u - g) \nabla \nabla 1_{\Omega} - 1_{\Omega} \cdot f) = 0$$

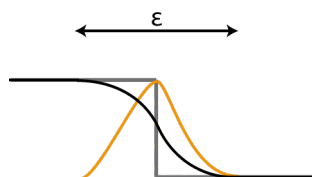
und damit die starke Formulierung

$$\nabla(1_{\Omega} \cdot \nabla u) + (u - g) \nabla \nabla 1_{\Omega} = f \cdot 1_{\Omega}$$

Approximiere  $1_{\Omega}$  und  $\delta_{\partial\Omega}$  durch glatte Funktionen

$$\varphi(x) := \frac{1}{2} \cdot \left( 1 - \tanh \frac{3 \cdot d(x)}{\varepsilon} \right)$$

mit  $d$  signierte Abstandfunktion (und  $\varepsilon$  klein).



Es gilt  $\varphi \sim 1_\Omega$ ,  $|\nabla\varphi| \sim \delta_{\partial\Omega}$ . (siehe auch Li et al. „Solving PDE’s in complex geometrics: a diffuse domain approach“ Comm. Math. Sei 7(2009) 81)

**Bemerkung** (Algorithmen zur Berechnung von Distanzfunktionen) Gegeben: Interface. Gesucht: für Punkt  $x \in \bar{\Omega}$  Abstand zum Interface. Das Interface sei gegeben durch eine Funktion  $\psi$ , sodass  $\partial\Omega = \{x \in \bar{\Omega}; \psi(x) = 0\}$ . Löse Level-Set-Gleichung

$$\partial_t\psi + |\nabla\psi| = 1 \quad (*)$$

bis stationärer Zustand erreicht ist (dann  $|\nabla\psi| = 1$ ). Problem hierbei:  $\{x \in \bar{\Omega}; \psi(x) = 0\}$  ändert sich. Dieses Problem kann durch eine Modifikation von (\*) behoben werden:

$$\partial_t\psi + S(\psi_0) \cdot (|\nabla\psi| - 1) = 0$$

mit

$$S(\psi_0)(x) := \begin{cases} 1 & x \in \Omega \\ 0 & x \in \partial\Omega \\ -1 & \bar{\Omega} \setminus \Omega \end{cases}$$

Numerisch besser ist

$$S(\psi_0) = \frac{\psi_0}{\sqrt{\psi_0^2 + h^2}} \quad S(\psi) = \frac{\psi}{\sqrt{\psi^2 + |\nabla\psi|^2 \cdot h^2}}$$

Allgemein lautet die Level-Set-Gleichung

$$\partial_t\psi + \vec{v} \cdot \nabla\psi = f$$

Euler-Formulierung für die Bewegung eines Interfaces  $\psi(x) = 0$  mit der Geschwindigkeit  $\vec{v}$ . Gilt  $\vec{v} = a \cdot \vec{n}$ , dann

$$\vec{n} \cdot \nabla\psi = \frac{\nabla\psi}{|\nabla\psi|} \cdot \nabla\psi = |\nabla\psi|$$

Der gleiche Algorithmus kann auch benutzt werden, um  $g$  auf  $\bar{\Omega}$  fortzusetzen. Löse dazu

$$\partial_t g + S(\psi) \cdot \vec{n} \cdot \nabla g = 0$$

Dies extrapoliert  $g$  in die Normalenrichtung.

# 5

## Differentialgleichungen auf gekrümmten Oberflächen

**Beispiel** (i). 1D: Übertrage die 1D-Gleichung

$$\partial_{xx}u = f \quad \text{in } [0, 1]$$

auf 1D-Oberfläche  $\Gamma$

$$\partial_{ss}u = f \quad \text{auf } \Gamma$$

wobei  $s$  die Bogenlänge bezeichnet.

(ii). 2D-Oberfläche (z.B. Sphäre):

$$\Delta u = f \quad \text{in } \Omega$$

wird zu

$$\Delta_{\Gamma} u = f \quad \text{auf } \Gamma$$

wobei  $\Delta_{\Gamma}$  den Laplace-Beltrami-Operator bezeichnet. Es gilt

$$\Delta_{\Gamma} f := \nabla_{\Gamma} \cdot \nabla_{\Gamma} f := \frac{1}{\sqrt{|g|}} \cdot \partial_i (\sqrt{|g|} \cdot g^{ij} \cdot \partial_j f)$$

wobei  $|g| = |\det(g_{ij})|$ ,

$$g_{ij} \cdot g^{jk} = \delta_{ik} \qquad \partial_i = \frac{\partial}{\partial x^i}$$

(FE-Diskretisierung)

Annahme: alle bekannten Sätze gelten auch hier.

$$-\int_{\Gamma} \nabla_{\Gamma} u \cdot \nabla_{\Gamma} \psi = \int_{\Gamma} f \cdot \psi$$

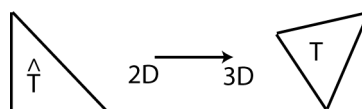
Diskretisiere  $\Gamma$  durch Polygonzug  $\Gamma_h$ . Damit FE-Formulierung:

$$-\int_{\Gamma_h} \nabla_{\Gamma_h} u_h \cdot \nabla_{\Gamma_h} \psi_h = \int_{\Gamma_h} f \cdot \psi_h$$

für  $\psi_h \in V_h$ . Summation gibt

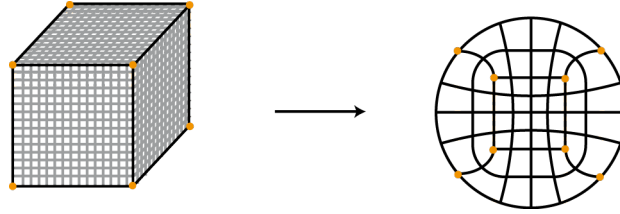
$$-\sum_{T \in \Gamma_h} \int_T \nabla_{\Gamma_h} u_h \cdot \nabla_{\Gamma_h} \psi_h = \sum_{T \in \Gamma_h} \int_T f \cdot \psi_h$$

und  $u_h = \sum_{i=1}^N U_i \cdot \psi_i$ . Assemblierung erfolgt analog zu 1D- bzw. 2D-Problem mit der folgenden Modifikation:



## 5.1 Gittererzeugung

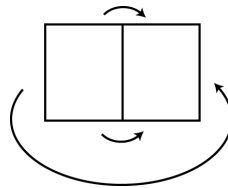
**Beispiel (Sphäre)** (i). Würfel mit Rechteckgitter  $\rightarrow$  Gitter auf Sphäre durch Aufblasen des Würfels  
 - Entartete Vierecke



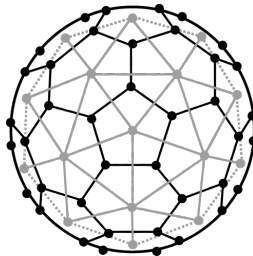
(ii). weiteres Vierecksgitter - Singularitäten in den Polen



(iii). geeignete Transformation



(iv). Dreiecksgitter, vgl. Fußball



12 Dreiecke mit 5 Nachbarn,  $N - 12$  Dreiecke mit 6 Nachbarn. Theorem von Euler:  $N$  Anzahl der Punkte (Knoten),  $E$  Anzahl der Kanten,  $E = \frac{1}{2} \sum_{i=1}^N c_i$  wobei  $c_i$  die Koordinationszahl (Anzahl der Nachbarn),  $F$  Anzahl der Dreiecke,  $F = \frac{2}{3} E$ . Es gilt

$$F - E + V = 2$$

Für Sphäre ergibt sich durch Einsetzen

$$\begin{aligned} \frac{\chi}{3} \frac{1}{\chi} \sum_{i=1}^N c_i - \frac{1}{2} \sum_{i=1}^N c_i + N &= 2 \\ \Rightarrow -\frac{1}{6} \sum_{i=1}^N c_i + N &= 2 \\ \Rightarrow \sum_{i=1}^N (6 - c_i) &= 12 \end{aligned}$$

**Bemerkung** Offenes Problem (eines der Hilbert-Probleme des 21. Jahrhunderts): Gegeben seien  $N$  Punkte auf Sphäre. Verteile Punkte so, dass

$$\sum_{\substack{i,j=1 \\ i \neq j}}^N \frac{1}{|r_i - r_j|^\beta} \rightarrow \min \quad (\beta \geq 1)$$

(In der Physik: Thomson Problem,  $\beta = 2$ )

(Implizite Beschreibung der Geometrie)

Ein anderer Zugang beschreibt Oberfläche  $\Gamma$  implizit (bilde  $\Gamma$  in ein 3d-Gebiet  $\bar{\Omega}$  ein):

$$\nabla_\Gamma u = P \nabla \bar{u} \qquad \Delta_\Gamma u = \nabla(P \nabla \bar{u})$$

mit  $u : \Gamma \rightarrow \mathbb{R}$ ,  $\bar{u} : \bar{\Omega} \rightarrow \mathbb{R}$ ,  $\bar{u}|_\Gamma = u$ ,  $P := I - \frac{\nabla \varphi \otimes \nabla \varphi}{|\nabla \varphi|^2}$ ,  $\Gamma = \{x \in \bar{\Omega}; \varphi(x) = 0\}$  für ein  $\varphi : \bar{\Omega} \rightarrow \mathbb{R}$ . Analog gilt auch

$$\begin{aligned} -\Delta_\Gamma u &= f \\ \Rightarrow -\int_\Gamma \nabla_\Gamma u \cdot \nabla_\Gamma \psi &= \int_\Gamma f \cdot \psi \\ -\int_\Omega \delta_\Gamma \nabla \bar{u} \cdot \nabla \bar{\psi} &= \int_\Omega \delta_\Gamma \cdot \bar{f} \cdot \bar{\psi} \\ \Rightarrow \int_\Omega (\nabla(\delta_\Gamma \nabla \bar{u}) - \delta_\Gamma \bar{f}) \cdot \bar{\psi} &= 0 \\ \Rightarrow \nabla(\delta_\Gamma \nabla \bar{u}) &= \delta_\Gamma \bar{f} \quad \text{in } \bar{\Omega} \end{aligned}$$

Approximiere  $\delta_\Gamma$  durch Phasenfeldapproximation  $\delta_\Gamma \approx |\nabla \eta|$  mit

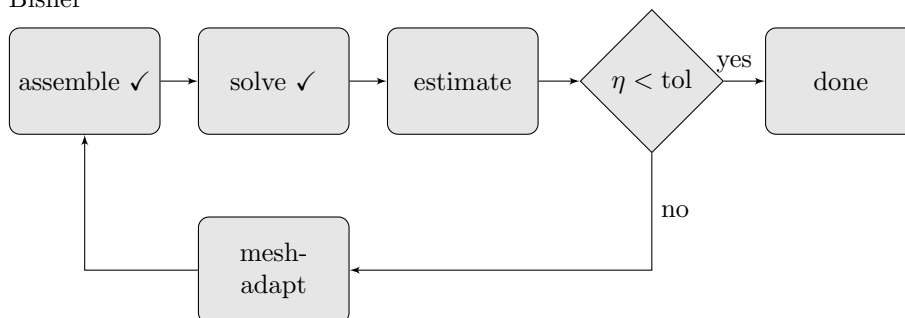
$$\eta(x) := \frac{1}{2} \cdot \tanh\left(1 - \frac{3d(x)}{\varepsilon}\right)$$

Wähle für Numerik  $|\nabla \eta| + \delta$  mit  $0 < \delta \ll 1$ .

# 6

## Adaptive Finite Elemente Methode

Bisher



Noch offen:  $h$ -Adaptivität. Ziel: „optimales“ Gitter für Problem, um gegebenes Toleranzkriterium mit minimalem Aufwand zu erfüllen. Alternativen zu  $h$ -Adaptivität:

- (i).  $p$ -Adaptivität (lokal Polynomgrad der Ansatzfunktionen erhöhen). Kombination:  $hp$ -Adaptivität.
- (ii). Modell-Adaptivität (lokal genaueres Modell wählen)

### 6.1 Stationäre Probleme

Es sei  $\mathcal{S}$  eine Triangulierung von  $\Omega$ ,  $u_h \in V_h$  eine finite Elemente Lösung. Weiterhin sei ein a-posteriori Fehlerschätzer gegeben, d.h.

$$\|u - u_h\| \leq \eta(u_h) = \left( \sum_{S \in \mathcal{S}} (\eta_S(u_h))^p \right)^{\frac{1}{p}}$$

für ein  $p \in [1, \infty)$ .  $\text{tol}$  sei die gegebene Fehlertoleranz. Falls  $\eta(u_h) > \text{tol}$  sind die folgenden Fragen zu klären:

- Wo muss das Gitter verfeinert werden, um den Fehler zu reduzieren?
- Wie kann das realisiert werden, ohne die Anzahl der Freiheitsgrade drastisch zu erhöhen?

Die Strategie der globalen Verfeinerung ist beste Lösung im Hinblick auf Fehlerreduktion, aber zu teuer. Auch ein „optimales Gitter“ ist ein sehr schwieriges Optimierungsproblem und häufig teurer als Lösung des Ausgangsproblems.

Folgerung: Wähle heuristische Argumente, sodass Gitter nah an „optimalem“ Gitter und Aufwand gering. Alle Strategien haben eine Gleichverteilung des Fehlers zum Ziel.

### 6.1 Algorithmus

Start mit  $\mathcal{S}_0$  und  $\text{tol}$

$k = 0$

Loese auf  $S_k$

Berechne  $\eta$  und  $\eta_S$  fuer  $S \in \mathcal{S}_k$

```

while  $\eta > \text{tol}$  do
  mark element fuer Verfeinerung/Vergroeberung
  adapt Gitter  $S_k \rightarrow S_{k+1}$ 
   $k \rightarrow k + 1$ 
  Loese auf  $S_k$ 
  Berechne  $\eta$  und  $\eta_S$  fuer  $S \in S_k$ 
end while

```

Verschiedene Strategien:

- (i). globale Verfeinerung: Strategie konvergiert falls a-priori Abschätzung mit Gitterweite  $h$  existiert.
- (ii). Maximumstrategie:  $\gamma \in (0, 1)$ , alle Elemente mit  $\eta_S > \gamma \cdot \max_{S' \in S_k} \eta_{S'}$  werden verfeinert. Je kleiner  $\gamma$  gewählt wird, desto mehr Verfeinerungen werden durchgeführt. Typischer Wert:  $\gamma = \frac{1}{2}$ .

**6.2 Algorithmus** (Maximumstrategie)

```

 $\gamma \in (0, 1)$ 
 $\eta_{\max} = \max(\eta_S; S \in S_k)$ 

for all  $S \in S_k$  do
  if  $\eta_S > \gamma \cdot \eta_{\max}$  then
    mark  $S$  for refinement
  end if
end for

```

- (iii). Gleichverteilungsstrategie: Sei  $N_k$  die Anzahl der Elemente in  $S_k$ . Annahme:  $\eta_S = \eta_{S'}$  für alle  $S, S' \in S_k$ . Dann gilt

$$\eta = \left( \sum_{S \in S_k} \eta_S^p \right)^{\frac{1}{p}} = N_k^{\frac{1}{p}} \cdot \eta_S \stackrel{!}{=} \text{tol}$$

also  $\eta_S = \frac{\text{tol}}{N_k^{\frac{1}{p}}}$ . Versuche dies zu erreichen durch Verfeinerung aller Elemente für die  $\eta_S > \frac{\text{tol}}{N_k^{\frac{1}{p}}}$  gilt.

Stabiler ist Variante mit  $\eta_S > \Theta \cdot \frac{\text{tol}}{N_k^{\frac{1}{p}}}$  mit  $\Theta \in (0, 1)$ , typisch  $\Theta = 0.9$ .

**6.3 Algorithmus** (Gleichverteilungsstrategie)

```

 $\Theta \in (0, 1), \Theta \approx 1$ 
for all  $S \in S_k$  do
  if  $\eta_S > \Theta \cdot \frac{\text{tol}}{N_k^{\frac{1}{p}}}$  then
    mark for refinement
  end if
end for

```

Problem: Eine Fehlerreduktion kann nicht garantiert werden (auch bei Maximumstrategie).

- (iv). Fehlerreduktionsstrategie: Die folgenden Annahmen seien erfüllt:
  - Daten (z.B. die rechte Seite) sind durch  $S_k$  hinreichend gut aufgelöst, d.h. „Fehler durch Quadraturformel ist vernachlässigbar“.



- Alle Kanten der markierten Elemente werden verfeinert.

Idee: Verfeinere den Teil der Triangulierung, deren Elemente einen bestimmten (fixen) Teil des Fehlers ausmachen. Wähle  $\Theta^* \in (0, 1)$ . Wähle eine Menge  $\mathcal{A} \subseteq \mathcal{S}_k$ , sodass

$$\sum_{S \in \mathcal{A}} \eta_S^p \geq (1 - \Theta^*)^p \cdot \eta^p$$

Fehler wird um Faktor  $\kappa < 1$  reduziert. Um  $\mathcal{A}$  zu bestimmen, wähle  $\gamma$  in der Maximumstrategie und setze  $\gamma = \gamma - \nu$  für  $\nu \in (0, 1)$ ,  $\nu \ll 1$ , bis genügend große Menge erreicht ist.

#### 6.4 Algorithmus

$\Theta^* \in (0, 1)$ ,  $\nu \in (0, 1)$   
 $\eta_{\max} = \max(\eta_S; S \in \mathcal{S}_k)$   
 sum = 0,  $\gamma = 1$

```

while sum < (1 - Θ*)p · ηp do
  γ = γ - ν
  for all S ∈ Sk do
    if S is not marked
      if ηS > γ · ηmax
        mark S for refinement
        sum = sum + ηSp
      end if
    end if
  end for
end while

```

**Bemerkung (Modifikation des Algorithmus)** Erlaube auch Verfeinerung, falls Daten nicht genügend aufgelöst (Morin, Nochetto, Siebert: SIAM Review 44(2002) 631).

**Bemerkung** Für lineare Probleme (elliptisch) ist Verfeinerung ausreichend, für nichtlineare und zeitabhängige Probleme wird auch Vergrößerung benötigt.

Verwende ähnliche Strategien wie für Verfeinerung. Einschränkung: Elemente werden nur dann vergrößert, wenn alle Nachbarelemente, die beeinträchtigt sind, auch vergrößert werden sollen. Idee ist auch Gleichverteilung des Fehlers.

- Maximumstrategie: Wähle  $\gamma > \gamma_c$ . Verfeinere  $S$ , falls  $\eta_S^p > \gamma \cdot \max_{S' \in \mathcal{S}_k} \eta_{S'}^p$ . Vergrößere  $S$ , falls  $\eta_S^p + \eta_{c,S}^p \leq \gamma_c \cdot \max_{S' \in \mathcal{S}_k} \eta_{S'}^p$ .
- Gleichverteilungsstrategie: Annahme  $\eta_s \approx \frac{\text{tol}}{N_k^{\frac{1}{p}}}$ . Verfeinere  $S$ , falls  $\eta_S > \Theta \cdot \frac{\text{tol}}{N_k^{\frac{1}{p}}}$ . Vergrößere  $S$ , falls  $\eta_S + \eta_{c,S} \leq \Theta_c \cdot \frac{\text{tol}}{N_k^{\frac{1}{p}}}$ .
- garantierte Fehlerreduktion: Markiere Menge  $\mathcal{B} \subseteq \mathcal{S}_k$  so, dass

$$\sum_{S \in \mathcal{B}} \eta_S^p + \eta_{c,S}^p \leq \Theta_c^p \cdot \eta^p$$

d.h. wähle Menge  $B$  so, dass Fehler nicht größer wird als fester Anteil der Fehlertoleranz.

**Bemerkung** Information geht durch Vergrößerung verloren. Mögliche Strategien:

- erst am Ende vergrößern
- nur dann Informationen löschen, wenn Fehler nicht größer wird

## 6.2 Zeitabhängige Probleme

Benutze adaptive Gitteranpassung in jedem Zeitschritt, kombiniert mit einer adaptiven Zeitschrittweitensteuerung. Mögliche Strategien:

- (i). explizite Strategie: Zeitschritt wird einmal auf Gitter des alten Zeitschritts gelöst ( $\rightarrow$  FE-Lösung  $u_h$ ). Berechne Fehlerindikatoren  $\eta_S$  und adaptiere das Gitter. Keine erneute Berechnung von  $u_h$ ! Notwendige Voraussetzungen:
  - kleine Zeitschritte
  - nah an stationärer Lösung
- (ii). semi-implizite Strategie: Zeitschritt wird auf Gitter des alten Zeitschritts gelöst ( $\rightarrow$  FE-Lösung  $\tilde{u}_h$ ). Berechne Fehlerindikatoren  $\eta_S$  und adaptiere das Gitter. Löse erneut auf neuem Gitter ( $\rightarrow u_h$ ). Notwendige Voraussetzungen:
  - kleine Zeitschritte
  - funktioniert i.A. gut
- (iii). implizite Strategie A: adaptive Strategie in jedem Zeitschritt bis Fehlertoleranz erreicht (beginne jeweils mit Gitter des letzten Zeitschrittes). Für kleine Zeitschritte werden nur wenige Iterationen benötigt.
- (iv). implizite Strategie B: adaptive Strategie in jedem Zeitschritt bis Fehlertoleranz erreicht (beginne jeweils mit Makrotriangulierung). Mehr Iterationen notwendig, eventuell konvergiert iterativer Löser besser, da Lösung auf Grobgitter häufig bessere Startlösungen liefern.

### 6.2.1 Strategien für adaptive Zeitschrittweitensteuerung

Gegeben sei eine Toleranz  $\text{tol}$  (z.B. zum Zeitpunkt  $T$  im Intervall  $[0, T]$  oder in geeigneter Norm über  $[0, T]$ ).

Annahme:  $\Gamma_h \cdot \text{tol}$  ist Fehlertoleranz für Ort,  $\Gamma_\tau \cdot \text{tol}$  ist Fehlertoleranz für Zeit mit  $\Gamma_h + \Gamma_\tau \leq 1$ . Wähle Zeitschritt  $\tau$  so, dass  $\eta_\tau \approx \Gamma_\tau \cdot \text{tol}$  und Ortsauflösung  $h$  so, dass  $\eta_h + \eta_c \approx \Gamma_h \cdot \text{tol}$ . Zeitschritt kann wie bei gewöhnlichen Differentialgleichungen gesteuert werden, z.B. Extrapolationsverfahren.

**Bemerkung** Vermeide zuviele Lösungsschritte, z.B. erst Zeitschrittweitensteuerung, dann adaptive Gitteranpassung, am Ende überprüfen, ob Zeitschritt okay.

### 6.5 Algorithmus (implizit A)

$\text{tol}, \delta_1 \in (0, 1), \delta_2 > 1, \Theta_1 \in (0, 1), \Theta_2 \in (0, \Theta_1)$   
 $u_n$  auf  $\mathcal{S}_n$  zum Zeitpunkt  $t_n$  bekannt,  $\tau_n$  war letzte Zeitschritt.

$\mathcal{S}_{n+1} = \mathcal{S}_n$

$\tau_{n+1} = \tau_n$

$t_{n+1} = t_n + \tau_{n+1}$

solve  $u_{n+1}$  auf  $\mathcal{S}_{n+1}$  mit  $u_n$

berechne Fehlerschaetzer auf  $\mathcal{S}_{n+1}$

while  $\eta_\tau > \Theta_1 \cdot \Gamma_\tau \cdot \text{tol}$

$\tau_{n+1} = \delta_1 \cdot \tau_{n+1}$

$t_{n+1} = t_n + \tau_{n+1}$

solve  $u_{n+1}$  auf  $\mathcal{S}_{n+1}$  mit  $u_n$

berechne Fehlerschaetzer auf  $\mathcal{S}_{n+1}$

end while

do

mark elements for refinement/coarsening

if elements are marked then

```

    adapt mesh  $\mathcal{S}_{n+1}$ 
    solve  $u_{n+1}$  auf  $\mathcal{S}_{n+1}$  mit  $u_n$ 
    berechne Fehlerschaetzer auf  $\mathcal{S}_{n+1}$ 
end if

while  $\tau_\tau > \Theta_1 \cdot \Gamma_\tau \cdot \text{tol}$ 
     $\tau_{n+1} = \delta_1 \cdot \tau_{n+1}$ 
     $t_{n+1} = t_n + \tau_{n+1}$ 
    solve  $u_{n+1}$  auf  $\mathcal{S}_{n+1}$  mit  $u_n$ 
    berechne Fehlerschaetzer auf  $\mathcal{S}_{n+1}$ 
end while
while  $\eta_h > \Gamma_h \cdot \text{tol}$ 
    if  $\eta_\tau < \Theta_2 \cdot \Gamma_\tau \cdot \text{tol}$  then
         $\tau_{n+1} = \delta_2 \cdot \tau_{n+1}$ 
    end if
end if

```

**Bemerkung** Gegebenenfalls muss im Algorithmus die Lösung  $u_n$  vom Gitter  $\mathcal{S}_n$  auf das Gitter  $\mathcal{S}_{n+1}$  interpoliert werden.

**Bemerkung (Multi-Mesh Methode)** Oft ist es notwendig, Systeme von Differentialgleichungen zu lösen („multiphysics“). Das geschieht häufig auf einem adaptiv angepassten Gitter. Dies ist ineffizient. „Optimales“ Gitter für eine Komponente muss nicht „optimal“ für die anderen Komponenten sein. Wähle deshalb unabhängig verfeinerte Gitter für jede Komponente. (A. Voigt, T. Witkowski, J. Comput. Science (in press)).

#### Beispiel

$$\Delta^2 u = 0 \qquad u \Big|_{\partial\Omega} = \frac{\partial u}{\partial n} \Big|_{\partial\Omega} = 0$$

Gemischte Formulierung: Gesucht ist  $u, v \in H_0^1(\Omega) \times H^1(\Omega)$  mit

$$\int_{\Omega} \nabla u \cdot \nabla \varphi \, dx + \int_{\Omega} v \cdot \varphi \, dx = 0 \quad (\varphi \in H^1(\Omega))$$

$$\int_{\Omega} \nabla v \cdot \nabla \psi \, dx = 0 \quad (\psi \in H_0^1(\Omega))$$

$\mathcal{S}_h^\circ$  und  $\mathcal{S}'_h$  seien unterschiedliche Triangulierungen für  $\Omega$ , die aus der gleichen Makrotriangulierung resultieren. Es seien

$$V_h^\circ := \{v_h \in H^1; \forall T \in \mathcal{S}_h^\circ : v_h|_T \in \mathbb{P}^n\}$$

$$V'_h := \{v_h \in H_0^1; \forall T \in \mathcal{S}'_h : v_h|_T \in \mathbb{P}^n\}$$

Gesucht ist  $(u_h, v_h)$  mit

$$\int_{\Omega} \nabla u_h \cdot \nabla \varphi + \int_{\Omega} v_h \cdot \varphi = 0 \quad (\varphi \in V_h^\circ)$$

$$\int_{\Omega} \nabla v_h \cdot \nabla \psi = 0 \quad (\psi \in V'_h)$$

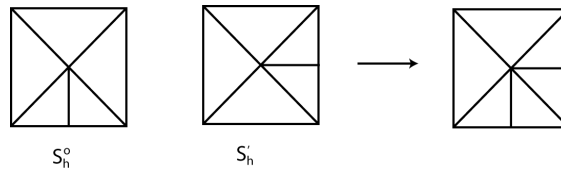
Es sei  $u_h = \sum_{i=1}^n u_i \cdot \varphi_i$ ,  $v_h = \sum_{i=1}^m v_i \cdot \psi_i$ . Dann

$$\sum_{j=1}^n u_j \cdot \left( \sum_{T \in \mathcal{S}_h^\circ} \int_T \nabla \varphi_j \cdot \nabla \varphi_i \right) + \sum_{j=1}^n v_j \cdot \left( \sum_{T \in \mathcal{S}_h^\circ \cup \mathcal{S}'_h} \int_T \psi_j \cdot \varphi_i \right) = 0$$

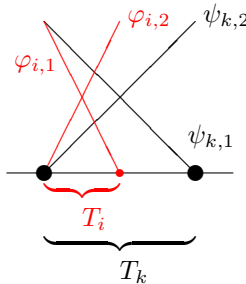
$$\sum_{j=1}^m v_j \cdot \left( \sum_{T \in \mathcal{S}'_h} \int_T \nabla \psi_j \cdot \nabla \psi_i \right) = 0$$

für  $i = 1, \dots, n$  bzw.  $i = 1, \dots, m$ .

Für Kopplungsterm muss Vereinigung der Triangulierungen betrachtet werden. Es gilt:  $T^\circ \in \mathcal{S}_h^\circ$  ist Subelement von  $T' \in \mathcal{S}'_h$  oder umgekehrt (da beide Triangulierungen aus der gleichen Makrotriangulierung resultieren).  $\mathcal{S}_h^\circ \cup \mathcal{S}'_h$  bezeichne die Triangulierung der lokal feinsten Elemente.



Idee: Definiere lokale Basisfunktionen  $\psi_{k,\ell}$  als Linearkombinationen der lokalen Basisfunktionen  $\varphi_{i,j}$ , wobei  $T_i \in \mathcal{S}_h^\circ$  und  $T_k \in \mathcal{S}'_h$  mit  $T_i \subset T_k$ ,  $\psi_{k,\ell}$  lokale Basisfunktion des Elementes  $T_k$  und  $\varphi_{i,j}$  des Elementes  $T_i$ .



Damit:

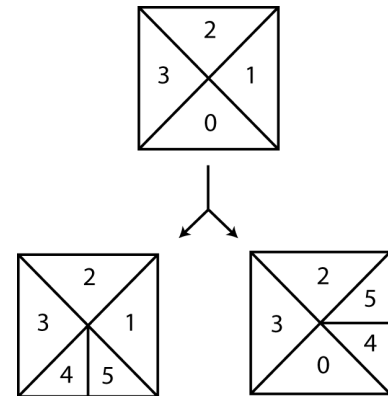
$$\int_{T_i \in \mathcal{S}_h^\circ} \psi_{k,\ell} \cdot \varphi_{i,j} = \sum_{\alpha=1}^m \int_{T_i \in \mathcal{S}_h^\circ} (c_{k,\alpha} \cdot \varphi_{i,\alpha}) \cdot \varphi_{i,j}$$

Die Berechnung von  $\mathcal{S}_h^\circ \cup \mathcal{S}'_h$  erfolgt durch eine gemeinsame Traversierung der Gitter, zum Beispiel: Traverse:

- (1) 4,0
- (2) 5,0
- (3) 1,4
- (4) 1,5
- (5) 2,2
- (6) 3,3

Unterscheide die folgenden Fälle:

- (i).  $T' \subset T$ , kleineres Element ist in Gitter für Testfunktion.



$$\begin{aligned} M_{T'} &= \begin{pmatrix} \int_{T'} \psi_0 \cdot \varphi_0 & \dots & \int_{T'} \psi_0 \cdot \varphi_n \\ \vdots & & \vdots \\ \int_{T'} \psi_n \cdot \varphi_0 & \dots & \int_{T'} \psi_n \cdot \varphi_n \end{pmatrix} \\ &= \begin{pmatrix} \int_{T'} \sum_i (c_{0i} \cdot \varphi_i) \cdot \varphi_0 & \dots & \int_{T'} \sum_i (c_{0i} \cdot \varphi_i) \cdot \varphi_n \\ \vdots & & \vdots \\ \int_{T'} \sum_i (c_{ni} \cdot \varphi_i) \cdot \varphi_0 & \dots & \int_{T'} \sum_i (c_{ni} \cdot \varphi_i) \cdot \varphi_n \end{pmatrix} \\ &= C \cdot \begin{pmatrix} \int_{T'} \varphi_0 \cdot \varphi_0 & \dots & \int_{T'} \varphi_0 \cdot \varphi_n \\ \vdots & & \vdots \\ \int_{T'} \varphi_n \cdot \varphi_0 & \dots & \int_{T'} \varphi_n \cdot \varphi_n \end{pmatrix} \end{aligned}$$

wobei  $C$  Transformationsmatrix.

(ii).  $T \subset T'$ , das kleinere Element ist in Gitter für Ansatzfunktionen. Dann folgt analog

$$M_{T'} = \dots = \begin{pmatrix} \int_{T'} \varphi_0 \cdot \varphi_0 & \dots & \int_{T'} \varphi_0 \cdot \varphi_n \\ \vdots & & \vdots \\ \int_{T'} \varphi_n \cdot \varphi_0 & \dots & \int_{T'} \varphi_n \cdot \varphi_n \end{pmatrix} \cdot C^T$$

wobei  $C$  Transformationsmatrix, die aus Verfeinerungssequenz resultiert, vgl. Mesh Structure Code.

Damit geringer Implementierungsaufwand, Gleichungssysteme werden kleiner. Zeitersparnis bis zu 50 %.

# 7

## Fehleranalyse

Sei  $\Omega \subseteq \mathbb{R}^d$  ein polynomial berandetes Gebiet. Betrachte das Problem

$$\begin{aligned} Lu := -\nabla(A \cdot \nabla u) + b \cdot \nabla u + c \cdot u &= f \quad \text{in } \Omega \\ u|_{\partial\Omega} &= 0 \end{aligned}$$

wobei  $A$  symmetrische positiv definite Matrix,  $b$  divergenzfrei ( $\operatorname{div} b = 0$ ) und  $c \geq 0$ . (Bisher meist  $A = I$ ,  $c = b = 0$ .) Die schwache Formulierung ist gegeben durch

$$a(u, v) := \langle A \nabla u, \nabla v \rangle + \langle b \cdot \nabla u + c \cdot u, v \rangle = \langle f, v \rangle$$

für  $v \in V$ . Ziel:  $\|u - u_h\|$  abschätzen.

**Bemerkung** (Analogie zu linearen Gleichungssystemen) Seien  $A, A_h \in \mathbb{R}^{n \times n}$  regulär,  $b, b_h \in \mathbb{R}^n$ . Gesucht sind  $x, x_h \in \mathbb{R}^n$  mit

$$Ax = b \qquad A_h x_h = b_h$$

Es gelte  $A_h \rightarrow A$ ,  $b_h \rightarrow b$  für  $h \rightarrow 0$ .

- Approximationsfehler  $e_h := x - x_h$
- „truncation error“  $\tau_h := A_h x - b_h$
- Residuum  $\varrho_h := b - Ax_h$

Verschiedene Fehlerschätzer:

- a priori Fehleranalyse: Basiert auf truncation error.

$$\begin{aligned} A_h e_h &= A_h x - A_h x_h = A_h x - b_h = \tau_h \\ \Rightarrow \|e_h\| &\leq c_h \cdot \|\tau_h\| \end{aligned}$$

mit  $c_h := \|A_h^{-1}\|$  („diskrete Stabilitätskonstante“).

- a posteriori Fehleranalyse: Basiert auf dem Residuum.

$$\begin{aligned} A e_h &= Ax - Ax_h = b - Ax_h = \varrho_h \\ \Rightarrow \|e_h\| &\leq d_h \cdot \|\varrho_h\| \end{aligned}$$

mit  $d_h := \|A^{-1}\|$  („kontinuierliche Stabilitätskonstante“).

Ziel: Übertragung dieser Überlegungen auf Finite Elemente Lösungen. Abschätzung des Fehlers  $u - u_h$  in der Energienorm.

### 7.1 Lemma (A priori Fehlerschätzer)

$$\inf_{v \in V_h} \|u - v\|_V \leq \|u - u_h\|_V \leq \frac{c_1}{c_2} \cdot \inf_{v_h \in V_h} \|u - v_h\|_V$$

Beweis: Für  $v \in V_h$  gilt auf Grund der Koerzitivität

$$c_2 \cdot \|u - u_h\|_V^2 \leq a(u - u_h, u - u_h) \stackrel{*}{=} a(u - u_h, u - v) \leq c_1 \cdot \|u - u_h\|_V \cdot \|u - v\|_V$$

wobei in (\*) die Galerkin-Orthogonalität benutzt wurde.  $\square$

**Bemerkung** FE-Lösung ist beste Approximation an  $u$  in  $V_h$ . Liefert aber keine quantitative Information über  $\|u - u_h\|$ .

**Definition (Residuum)** Definiere das Residuum  $\varrho(u_h) \in V'$  durch

$$\langle \varrho(u_h), v \rangle := \langle f, v \rangle - a(u_h, v)$$

**Bemerkung** (i). Es gilt

$$\|\varrho(u_h)\|_{V'} = \sup_{v \in V'} \frac{\langle \varrho(u_h), v \rangle}{\|v\|_V}$$

(ii).  $\varrho(u_h)$  hängt nur von den Daten und  $u_h$  ab und ist somit eine berechenbare Größe.

## 7.1 A posteriori Fehlerschätzer

### 7.2 Lemma (A posteriori Fehlerschätzer)

$$c_2 \cdot \|u - u_h\|_V \leq \|\varrho(u_h)\|_{V'} \leq c_1 \cdot \|u - u_h\|_V$$

Beweis: Nach Definition gilt

$$a(u - u_h, v) = \langle f, v \rangle - a(u_h, v) = \langle \varrho(u_h), v \rangle$$

Nutze Koerzitivität und Stetigkeit.  $\square$

**Bemerkung** Idee: Schätze  $\|\varrho(u_h)\|_{V'}$  statt  $\|u - u_h\|_V$  ab. Dafür benötigen wir eine praktische Möglichkeit um die Norm in  $V'$  zu berechnen. (Zum Beispiel  $V = H_0^1$ ,  $V' = H^{-1}$ .)

**Bemerkung (Interpolation in Sobolev -Räumen - Clement-Interpolation)** Sei  $k \geq 1$  der Polynomgrad. Sei  $v \in W_p^m(\Omega)$  und  $S \in \mathcal{S}_h$  Element der konformen Triangulierung  $\mathcal{S}_h$ , dann gilt

$$\|D^t(v - I_h v)\|_{L^p(S)} \leq c \cdot h_S^{s-t} \cdot \|D^s v\|_{L^p(N(S))}$$

für  $0 \leq t \leq s \leq \min\{m, 1 + k\}$  wobei  $N(S)$  die Menge der Nachbarelemente von  $S$  in  $\mathcal{S}_h$  mit nichtleerer Schnittmenge mit  $S$  bezeichnet. Somit gilt für  $p = 2$ :

$$\begin{aligned} \langle f, v \rangle &= \langle f, v - I_h v \rangle \leq \|h \cdot f\|_{L^2(\Omega)} \cdot \|h^{-1} \cdot (v - I_h v)\|_{L^2(\Omega)} \\ &\leq c \cdot \|h \cdot f\|_{L^2(\Omega)} \cdot \|Dv\|_{L^2(\Omega)} \end{aligned}$$

**Beispiel** Glatte Lösung  $u \in H^2(\Omega)$ .  $\mathcal{S}_h$  sei uniform mit  $h_S = h$  für alle  $S \in \mathcal{S}_h$ . Dann für  $d = 2$ ,  $N = h^2$  Freiheitsgrade,

$$\|u - u_h\|_V \leq c \cdot h \cdot \|D^2 u_h\|_{L^2(\Omega)} = c \cdot N^{-\frac{1}{2}}$$

ist optimale Abklingrate für stückweise lineare Ansatzfunktionen.

(A posteriori Fehlerschätzer)

Es sei  $\mathcal{S}_h$  ein Gitter und  $\mathcal{U}_h$  die Menge aller inneren Kanten und Seiten. Es sei  $e_h := u - u_h$ . Berechne  $a(e_h, v)$  elementweise und integriere partiell:

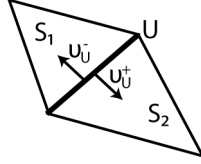
$$a(e_h, v) = \langle \varrho(u_h), v \rangle = \sum_{S \in \mathcal{S}_h} \int_S R_S(u_h) \cdot v + \sum_{U \in \mathcal{U}_h} \int_U J_u(u_h) \cdot v$$

wobei

$$R_S(u_h) := f + \nabla(A \cdot \nabla u_h) - b \cdot \nabla u_h - c \cdot u_h$$

$$J_U(u_h) := -A \nabla u_h^+ \cdot \nu_U^+ - A \cdot \nabla u_h^- \cdot \nu_U^- =: [[A \cdot \nabla u_h]] \cdot \nu_U$$

$R_S$  heißt Elementresiduum und  $J_U$  Sprungresiduum.



Sei

$$\eta_h(S)^2 := h_S^2 \cdot \|R_S(u_h)\|^2 + \sum_{U \in \partial S} h_U \cdot \|J_U(u_h)\|_{L^2(U)}^2$$

Für  $\omega \subseteq \Omega$  definiere

$$\eta_h(\omega)^2 = \sum_{\substack{S \in \mathcal{S}_h \\ S \subseteq \omega}} \eta_h(S)^2$$

### 7.3 Lemma

Es existiert  $c > 0$  abhängig nur von der Regularität de Gitters und  $c_1, c_2$  sodass

$$\|u - u_h\|^2 \leq c \cdot \eta_h(\Omega)^2$$

Beweis: Sei  $\varphi := e_h - I_h e_h$ . Dann

$$\begin{aligned} \|e_h\|^2 &= a(e_h, e_h) = \langle \varrho(u_h), e_h \rangle \stackrel{*}{=} \langle \varrho(u_h), \varphi \rangle \\ &= \sum_{S \in \mathcal{S}_h} \int_S R_S \varphi + \sum_{U \in \mathcal{U}_h} \int_U J_U \varphi \\ &\stackrel{\text{CSU}}{\leq} \sum_{S \in \mathcal{S}_h} \|R_S\|_{L^2(S)} \cdot \|\varphi\|_{L^2(S)} + \sum_{U \in \mathcal{U}_h} \|J_U\|_{L^2(U)} \cdot \|\varphi\|_{L^2(U)} \end{aligned}$$

In (\*): Galerkin-Orthogonalität. Es gilt aus Interpolationsabschätzung, dass

$$\begin{aligned} \sum_{S \in \mathcal{S}_h} \|R_S\|_{L^2(S)} \cdot \|\varphi\|_{L^2(S)} &\leq c \cdot \sum_{S \in \mathcal{S}_h} \|h \cdot R_S\|_{L^2(S)} \cdot \|\nabla e_h\|_{L^2(N(S))} \\ &\leq c \cdot \left( \sum_{S \in \mathcal{S}_h} \|h \cdot R_S\|_{L^2(S)}^2 \right)^{\frac{1}{2}} \cdot \|e_h\| \end{aligned}$$

Aus dem Spursatz folgt für  $U \in \mathcal{U}_h, S_U \in \mathcal{S}_h$ :

$$\|\varphi\|_{L^2(U)} \leq c \cdot h_U^{\frac{1}{2}} \|\nabla \varphi\|_{L^2(S_U)} + c \cdot h_U^{-\frac{1}{2}} \cdot \|\varphi\|_{L^2(S_U)}$$

Damit

$$\begin{aligned} \sum_{U \in \mathcal{U}_h} \|J_U\|_{L^2(U)} \cdot \|\varphi\|_{L^2(U)} &\leq c \cdot \sum_{U \in \mathcal{U}_h} \|h^{\frac{1}{2}} \cdot J_U\|_{L^2(U)} \cdot \|\nabla \varphi_h\|_{L^2(N(U))} \\ &\leq c \cdot \left( \sum_{U \in \mathcal{U}_h} \|h^{\frac{1}{2}} \cdot J_U\|_{L^2(U)}^2 \right)^{\frac{1}{2}} \cdot \|e_h\| \end{aligned}$$

□

**Bemerkung** Dies ist der einfachste Fehlerschätzer, für lineare Probleme. Schranken sind nicht optimal. Bemerke: Die Abschätzung gilt nicht elementweise!



## 7.2 Duale Fehlerschätzer (goal-oriented Fehlerschätzer)

**Bemerkung (Motivation)** (i). lineare Gleichungssysteme: Sei  $Ax = b$ ,  $A_h x_h = b_h$ . Definiere  $\varrho_h := b - Ax_h$ ,  $e_h := x - x_h$ . Für  $j \in \mathbb{R}^n$  sei  $J(y) := (e, j)$ . Dann

$$(e_h, j) = J(e_h) = J(x) - J(x_h)$$

Betrachte Lösung  $z \in \mathbb{R}^n$  Lösung der dualen Gleichung  $A^* z = j$ . Damit

$$\begin{aligned} J(e_h) &= (e_h, j) = (e_h, A^* z) = (Ae_h, z) = (\varrho_h, z) \\ \Rightarrow \|J(e_h)\| &\leq \sum_{i=1}^n |\varrho_{h,i}| \cdot |z_i| \end{aligned}$$

Um die Gewichte  $z_i$  zu bestimmen, muss das adjungierte Problem gelöst werden. Sie beschreiben den Einfluss der „lokalen“ Residuen auf die Größe  $J(e_h)$ .

(ii). nichtlineare Probleme: Das Konzept kann auf nichtlineare Probleme übertragen werden. Seien dazu  $A, A_h : \mathbb{R}^n \rightarrow \mathbb{R}^n$  differenzierbar und betrachte

$$A(x) = b \qquad A_h(x_h) = b_h$$

$A'(x)$  sei die Jacobi-Matrix in  $x$  und  $\varrho_h := b - A(x_h)$  das (nichtlinear) Residuum.

$$(A(x) - A(x_h), y) = \int_0^1 (A'(x_h + s \cdot e_h) \cdot e_h, y) ds = (Be_h, y)$$

mit  $B = B(x, x_h) := \int_0^1 A'(x_h + s \cdot e_h) ds$ . Sei  $J := (\cdot, j)$ . Gesucht ist  $z \in \mathbb{R}^n$  mit  $B^* z = j$ .

$$\begin{aligned} J(e_h) &= (e_h, j) = (e_h, B^* z) = (Be_h, z) = (A(x) - A(x_h), z) = (\varrho_h, z) \\ \Rightarrow |J(e_h)| &\leq \sum_{i=1}^n |\varrho_{h,i}| \cdot |z_i| \end{aligned}$$

Problem: Berechnung von  $z$  hängt von  $e_h$  ab. Ausweg: Approximiere  $B$  durch

$$\tilde{B} := B(x_h, x_h) = \int_0^1 A'(x_h) ds = A'(x_h)$$

Löse approximiertes duales Problem  $(A'(x_h))^* \tilde{z} = j$ . Der hierdurch entstandene zusätzliche Fehler kann abgeschätzt werden.

**Beispiel (Übertragung auf FE-Methode)** Betrachte

$$-\Delta u = f \qquad u|_{\partial\Omega} = 0$$

Fehlerfunktional  $J(u)$  ist die gesuchte Größe. Ziel: Wähle Gitter so, dass  $J(u) - J(u_h) \leq \text{tol}$ . Zum Beispiel:

(i).  $J(u) := u(a)$  oder  $J(u) := \partial_i u(a)$  mit  $a \in \Omega$  („Punktfehler“)

(ii).  $J(u) := \int_{\partial\Omega} \partial_n u dS$  („gemittelter Normalenfluss“)

Für ein lineares Funktional  $J$  gilt  $J(e_h) = J(u) - J(u_h)$ . Das duale Problem ist gegeben durch

$$\forall \varphi \in V, \varphi_h \in V_h : a(\varphi, z) = J(\varphi), a(\varphi_h, z) = J(\varphi_h)$$

Damit

$$\begin{aligned} J(e_h) &= a(e_h, z) = a(e_h, z - \psi_h) = (f, z - \psi_h) - a(u_h, z - \psi_h) \\ &= \varrho(u_h)(z - \psi_h) \end{aligned}$$

für  $\psi_h \in V_h$  wobei  $\varrho(u_h)$  das Residuum der FE-Approximation  $u_h$  bezeichnet. Analoges Vorgehen wie bei bereits bekanntem a posteriori Fehlerschätzer: Elementweise partielle Integration gibt

$$\begin{aligned} \varrho(u_h)(z - \psi_h) &= \sum_{S \in \mathcal{S}_h} ((f + \Delta u_h, z - \psi_h)_S - (\nabla u_h \cdot \nu, z - \psi_h)_{\partial S}) \\ &= \sum_{S \in \mathcal{S}_h} \left( \underbrace{(f + \Delta u_h, z - \psi_h)_S}_{=: R_{h,S}} + \frac{1}{2} \underbrace{([\nabla u_h \cdot \nu], z - \psi_h)_{\partial S \setminus \partial \Omega}}_{=: J_{h,\partial S}} \right) \end{aligned}$$

$R_{h,S}$  heißt Elementresiduum,  $J_{h,\partial S}$  Sprungresiduum.

#### 7.4 Lemma

Aus

$$J(e_h) = \sum_{S \in \mathcal{S}_h} ((R_{h,S}, z - \psi_h)_S + (J_{h,\partial S}, z - \psi_h)_{\partial S})$$

folgt

$$|J(e_h)| \leq \sum_{S \in \mathcal{S}_h} \varrho_S \cdot \omega_S$$

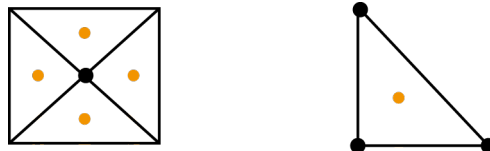
mit

$$\begin{aligned} \varrho_S &= (\|R_{h,S}\|_S^2 + h_S^{-1} \cdot \|J_{h,\partial S}\|_{\partial S}^2)^{\frac{1}{2}} \\ \omega_S &= (\|z - \psi_h\|_S^2 + h_S \cdot \|z - \psi_h\|_{\partial S}^2)^{\frac{1}{2}} \end{aligned}$$

**Bemerkung** Dies benötigt  $z$  als Lösung des dualen Problems. Funktioniert gut für elliptische Probleme.

**Bemerkung (Zienkiewicz-Zhu-Fehlerschätzer)** Idee: Berechne in jedem Knoten eine verbesserte Lösung und vergleiche mit ursprünglicher FE-Lösung, zum Beispiel durch geeignete Mittelung über alle Nachbarelemente in Quadraturpunkten mit Superkonvergenzeigenschaften.

**Beispiel** Stückweise lineare Elemente (orange: Quadraturpunkt mit Superkonvergenzeigenschaft)



Links: Mittelung über Lösung in diesen Punkten liefert „bessere Lösung“ als in Knoten. Nutze dies, um Entscheidung zu treffen, ob Gitter an der Stelle verfeinert werden muss.

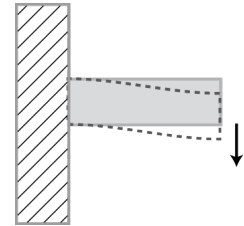
# 8

## Anwendungen

### 8.1 Lineare Elastizitätstheorie

Hier liegt Ursprung der FE Theorie. Man ist an relativen Deformationen interessiert. Deformationen können durch einen Verschiebungsvektor  $u = (u_x, u_y, u_z)^T$  beschrieben werden. Eine weitere wichtige Größe ist die Verzerrung

$$\varepsilon := \frac{1}{2} \cdot ((\nabla \circ u)^T + (\nabla \circ u))$$



wobei  $\circ$  das dyadische Produkt bezeichnet.

Für den Gleichgewichtszustand gilt (wegen Erhaltung der linearen Momente)

$$-\nabla \cdot \sigma := -\sigma \cdot \nabla = f$$

mit Spannung  $\sigma$  und den Volumenkräften  $f$ . Weiterhin gilt  $\sigma = \sigma^T$ , dies folgt aus der Drehmomentenerhaltung.

Konstitutive Gleichung zur Beschreibung des Materials liefern Zusammenhang zwischen  $\sigma$  und  $\varepsilon$ , z.B. durch Hook'sches Gesetz  $\sigma = \mathcal{E} : \varepsilon$  mit  $\mathcal{E}$  Elastizitätstensor.  $\mathcal{E}$  ist Tensor 4. Ordnung, z.B. für ein isotropes Material

$$\mathcal{E}_{ijkl} = \lambda \cdot \delta_{ij} \cdot \delta_{kl} + \mu \cdot (\delta_{ik} \cdot \delta_{jl} + \delta_{il} \cdot \delta_{jk})$$

mit  $\lambda, \mu$  Lamé-Koeffizienten. Dann gilt für

$$\begin{aligned} \sigma_{ij} &= \mathcal{E}_{ijkl} \cdot \varepsilon_{kl} \\ \Rightarrow \sigma_{ij} &= \lambda \cdot \delta_{ij} \cdot \delta_{kl} \cdot \varepsilon_{kl} + \mu \cdot (\delta_{ik} \cdot \delta_{jl} + \delta_{il} \cdot \delta_{jk}) \cdot \varepsilon_{kl} \\ &= \lambda \cdot \varepsilon_{kk} \cdot \delta_{ij} + 2\mu \varepsilon_{ij} \end{aligned}$$

und somit

$$\sigma = \lambda \cdot \text{trace}(\varepsilon) \cdot I + 2\mu \cdot \varepsilon$$

Alles zusammen liefert eine elliptische Differentialgleichung für  $u$ ,

$$-\lambda \cdot \nabla(\nabla \cdot u) - \mu \cdot \nabla \cdot ((\nabla \circ u)^T + \nabla \circ u) = f \quad \text{in } \Omega$$

mit Randbedingungen

$$u|_{\Gamma_D} = u_0 \qquad \sigma \cdot n|_{\Gamma_N} = g$$

Komponentenweise liefert das

$$\begin{aligned} f_x &= -(\lambda + 2\mu)\delta_{xx}u_x - \lambda \cdot (\partial_{xy}u_y + \partial_{xz}u_z) - \mu \cdot (\delta_{xx}u_y + \delta_{xy}u_x + \delta_{xx}u_z + \partial_{xz}u_x) \\ f_y &= -(\lambda + 2\mu)\partial_{yy}u_y - \lambda \cdot (\partial_{yx}u_x + \partial_{yz}u_z) - \mu \cdot (\partial_{yy}u_x + \partial_{yx}u_y + \partial_{yy}u_z + \partial_{yz}u_y) \\ f_z &= -(\lambda + 2\mu)\partial_{zz}u_z - \lambda \cdot (\partial_{zx}u_x + \partial_{zy}u_y) - \mu \cdot (\partial_{zz}u_x + \partial_{zx}u_z + \partial_{zz}u_y + \partial_{zy}u_z) \end{aligned}$$

**Bemerkung** (Diffuse-Domain Ansatz) (vgl. Kaktus-Beispiel)

$$-\lambda \cdot \nabla(\varphi(x) \cdot \nabla u) - \mu \cdot \nabla \cdot (\varphi(x) \cdot ((\nabla \circ u)^T + \nabla \circ u)) = \varphi(x) \cdot f(x)$$

in  $\bar{\Omega}$ , ggf. noch Randbedingungen, wobei  $\varphi$  die Phasenfunktion für die Beschreibung der Geometrie bezeichnet.

**Bemerkung** Häufig ist man nicht an  $u$ , sondern an  $\sigma$  interessiert. Wähle deshalb eine Formulierung, die beides direkt berechnet.

Problem kann als Variationsproblem beschrieben werden. Sei

$$\mathcal{F} := \int_{\Omega} \left( \frac{1}{2} \cdot \varepsilon : \sigma - f \cdot u \right) + \int_{\Gamma_N} g \cdot u$$

mit

$$\varepsilon : \sigma := \sum_{i,k} \varepsilon_{ik} \cdot \sigma_{ik}$$

Unbekannte sind nun  $u, \varepsilon, \sigma$ . Verschiedene Ansätze:

- (i). Verschiebungsansatz: Eliminiere  $\varepsilon, \sigma$ .

$$\mathcal{F} = \int_{\Omega} \left( \mu \cdot \varepsilon(u) : \varepsilon(u) + \frac{\lambda}{2} (\operatorname{div} u)^2 - f \cdot u \right) + \int_{\Gamma_N} g \cdot u$$

für  $u_0 = 0$  auf  $\Gamma_D$ . Sei

$$H_{\Gamma_N}^1 := \{v \in (H^1(\Omega))^3; v|_{\Gamma_D} = 0\}$$

Schwache Formulierung: Gesucht ist  $u \in H_{\Gamma_N}^1$  sodass

$$\int_{\Omega} (\mu \cdot \varepsilon(u) : \varepsilon(v) + \lambda \cdot \operatorname{div} u \cdot \operatorname{div} v) = \int_{\Omega} f \cdot v - \int_{\Gamma_N} g \cdot v$$

für alle  $v \in H_{\Gamma_N}^1$ . Die klassische Formulierung lautet

$$-\lambda \cdot \operatorname{grad} \operatorname{div} u - \mu \cdot \operatorname{div} \varepsilon(u) = f$$

in  $\Omega$  mit Randbedingungen

$$u|_{\Gamma_D} = 0 \qquad \sigma(u) \cdot n|_{\Gamma_N} = g$$

- (ii). gemischte Methode nach Hellinger und Reissner: Gesucht sind Verschiebung  $u$  und Spannung  $\sigma$ , Verzerrung  $\varepsilon$  wird eliminiert.

$$\begin{aligned} -\operatorname{div} \sigma &= f & \sigma &= \mathcal{E} : \varepsilon(u) \quad \text{in } \Omega \\ u|_{\Gamma_D} &= 0 & \sigma \cdot n|_{\Gamma_N} &= g \end{aligned}$$

Es sei  $X := L^2(\Omega)$ ,  $M := H_{\Gamma_N}^1(\Omega)$ . Seien dazu

$$a(\sigma, \tau) := (\mathcal{E}^{-1} \sigma, \tau) \qquad b(\tau, v) := -(\tau, \varepsilon(v))$$

**Bemerkung** Einfache stabile Elemente für die gemischte Formulierung sind schwierig. Zum Beispiel möglich (2005):

- stückweise Polynome 4. Ordnung für Spannung  $\sigma$  ( $\rightarrow$  162 Freiheitsgrade pro Element)
- stückweise lineare Ansatzfunktionen für Verschiebung  $u$

(iii). gemischte Formulierung nach Hu und Washizu: Hier werden alle drei Größen  $u, \varepsilon, \sigma$  erhalten.

$$\begin{aligned}(\mathcal{E}\varepsilon - \sigma, \eta) &= 0 & (\eta \in L^2(\Omega)) \\(\varepsilon - \varepsilon(u), \tau) &= 0 & (\tau \in L^2(\Omega)) \\-(\sigma, \varepsilon(v)) &= -(f, v) + \int_{\Gamma_N} g \cdot v & (v \in H_{\Gamma_N}^1(\Omega))\end{aligned}$$

Sei  $X := L^2(\Omega) \times L^2(\Omega)$ ,  $M := H_{\Gamma_N}^1(\Omega)$ .

$$a(\varepsilon, \sigma, \eta, \tau) := (\mathcal{E}\varepsilon, \eta) \qquad b(\eta, \tau, v) := (\tau, \varepsilon(v) - \varepsilon)$$

**Bemerkung** Numerische Umsetzung ist noch komplizierter ...

### 8.1 Satz

Sei  $\Omega \subseteq \mathbb{R}^3$  ein Gebiet mit stückweise glattem Rand. Der Dirichletrand  $\Gamma_D$  habe ein positives (zweidimensionales) Maß. Dann hat die Variationsaufgabe der linearen Elastizitätstheorie genau eine Lösung.

Beweis: Anwenden des Lax-Milgram-Lemmas. Zu zeigen ist dazu, dass das Variationsproblem elliptisch ist. Dies folgt aus der Kornschen Ungleichung

$$\int_{\Omega} \varepsilon(v) : \varepsilon(v) \geq c \cdot \|v\|_1^2 \quad (v \in H_{\Gamma_N}^1(\Omega))$$

mit  $c = c(\Omega, \Gamma_D) > 0$ . □

### 8.2 Navier-Stokes-Gleichung - Inkompressible Flüssigkeiten

Es gilt  $\frac{d}{dt}V(t) = 0$  wobei  $V(t)$  das Volumen bezeichnet, d.h.  $V = V(t)$  ist konstant. Aus dem Transporttheorem folgt dann

$$\int_V \nabla \cdot \vec{v} = 0$$

mit  $\vec{v}$  Geschwindigkeit. Somit  $\nabla \cdot \vec{v} = 0$ . Außerdem gelten

- Kontinuitätsgleichung (Massenerhaltung):

$$\partial_t \varrho + v \cdot \nabla \varrho = 0$$

mit  $\varrho$  Dichte

- Impulsgleichung:

$$\varrho \cdot \partial_t v + \varrho \cdot v \cdot \nabla v - \mu \cdot \Delta v + \nabla p = \varrho \cdot f$$

wobei  $p$  Druck,  $\mu$  Viskosität,  $f$  Kraft.

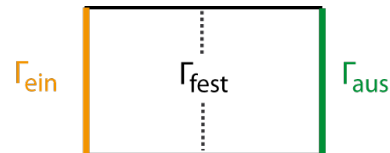
In homogenen Medien ist die Dichte  $\varrho$  konstant, damit erhält man die Gleichungen

$$\partial_t v + v \cdot \nabla v - \nu \Delta v + \frac{1}{\varrho} \cdot \nabla p = f \tag{8.1}$$

$$\operatorname{div} v = 0 \tag{8.2}$$

in  $\Omega$  mit  $\nu := \frac{\mu}{\varrho}$ . Im weiteren sei  $\varrho = 1$ . Es gelte

- (i).  $\partial\Omega = \Gamma_{\text{ein}} \cup \Gamma_{\text{fest}} \cup \Gamma_{\text{aus}}$   
 (ii).  $v|_{\Gamma_{\text{fest}}} = 0, v|_{\Gamma_{\text{ein}}} = v^e$   
 (iii).  $v(t=0) = v_0$



Spezialfälle:

- $\mu \gg 1$  („Honig“): Vernachlässige die ersten beiden Termine in (8.1). Dann

$$-\nu \cdot \Delta v + \nabla p = f \qquad \operatorname{div} v = 0$$

(Stokes-Problem)

- $\mu = 0$  (ideale Flüssigkeit):

$$\partial_t v + v \cdot \nabla v + \nabla p = f \qquad \operatorname{div} v = 0$$

(inkompressible Euler-Gleichungen)

**Bemerkung** (Theorie für stationäre Navier-Stokes-Gleichung ( $\partial_t v = 0$ )) Es seien

$$H := (H_0^1(\Omega, \Gamma_D))^d := \{v \in H_0^1(\Omega); v|_{\partial D} = 0\}$$

$$L := L_0^2(\Omega) := \{q \in L^2(\Omega); (q, 1)_\Omega = 0\}$$

Schwache Formulierung von (8.1) ist dann gegeben durch

$$\begin{aligned} \nu \cdot (\nabla v, \nabla \varphi) + (v \cdot \nabla v, \varphi) - (p, \operatorname{div} \varphi) &= (f, \varphi) \\ (\operatorname{div} v, \chi) &= 0 \end{aligned}$$

für  $\varphi \in H, \chi \in L$ . Unter der Annahme genügender Regularität (an die schwache Lösung) erhält man mit partieller Integration

$$\int_{\Omega} (-\nu \cdot \Delta v + v \cdot \nabla v + \nabla p - f) \cdot \varphi + \int_{\Gamma_{\text{aus}}} (\nu \cdot \partial_n v - p \cdot n) \cdot \varphi \, dS$$

Natürliche Randbedingung an  $\Gamma_{\text{aus}}$  ist daher  $\nu \cdot \partial_n v - p \cdot n|_{\Gamma_{\text{aus}}} = 0$ .

## 8.2 Satz

Das stationäre Navier-Stokes-Problem besitzt im Feld  $\partial\Omega = \Gamma_{\text{fest}}$  eine Lösung  $(v, p) \in H \times L$ . Für hinreichend glatte Daten  $c \cdot \nu^{-2} \cdot \|f\|_{-1} < 1$  ist die Lösung auch eindeutig. (*c ist Konstante abhängig von gegebenen Daten. Nicht frei wählbar ...*)

**Bemerkung** Für die instationäre Navier-Stokes-Gleichung (8.1), (8.2) mit Randbedingungen

$$v|_{\Gamma_{\text{fest}}} = 0 \quad v|_{\Gamma_{\text{ein}}} = v^e \quad (\nu \cdot \partial_n v - p \cdot n)|_{\Gamma_{\text{aus}}} = 0 \quad v(t=0) = v_0 \quad (8.3)$$

erhält man analoge Ergebnisse. Existenz für  $\Gamma_{\text{aus}} = \emptyset$ . Eindeutigkeit erfordert strengere Bedingungen an Problem Daten.

- 2D: Schwache Lösung ist eindeutig, für hinreichend glatte Daten auch klassische Lösung.
- 3D: Hier ist die Regularität der Lösung ein offenes Problem.

Für numerische Lösung wähle Stromfunktionsformulierung (in 2D). Seien  $\Psi$  Stromfunktion und  $\omega$  Wirbelstärke definiert durch

$$\begin{aligned} v &= \operatorname{rot}(\Psi) := \begin{pmatrix} \partial_2 \Psi \\ -\partial_1 \Psi \end{pmatrix} \\ \omega &:= \operatorname{rot}(v) := \partial_1 v_2 - \partial_2 v_1 = -\Delta \Psi \end{aligned}$$

Es gilt

$$(\operatorname{rot}(v), \Phi) = (v, \operatorname{rot}(\Phi))$$

für  $v \in (H_0^1(\Omega))^2$ ,  $\Phi \in H_0^1(\Omega)$ . Einsetzen in Navier-Stokes-Gleichung (8.1) mit  $\partial_t v = 0$  (d.h. stationäre Navier-Stokes-Gleichung) und Anwenden des Rotationsoperators gibt

$$-\nu \cdot \operatorname{rot}(\Delta v) + \operatorname{rot}(v \cdot \nabla v) + \operatorname{rot}(\nabla p) = \operatorname{rot}(f) \quad (8.4)$$

Es gilt

$$\operatorname{rot}(v \cdot \nabla v) = \operatorname{rot} v \cdot \operatorname{div} v + (v \cdot \nabla) \cdot \operatorname{rot} v$$

Damit folgt in (8.4):

$$-\nu \cdot \Delta \omega + (\operatorname{rot}(\Psi) \cdot \nabla) \cdot \omega = \operatorname{rot}(f)$$

Analoges Vorgehen durch Anwenden des Divergenzoperators gibt

$$\Delta p = \nabla \cdot f - 2 \cdot ((\partial_1 \partial_2 \Psi)^2 - \partial_1^2 \Psi \cdot \partial_2^2 \Psi)$$

Für  $\Psi$  ergibt sich somit

$$\Delta^2 \Psi - (\operatorname{rot}(\Psi) \cdot \nabla) \Delta \Psi = \nu \cdot \Delta^2 \Psi + \partial_2(\partial_1 \Psi \cdot \Delta \Psi) - \partial_1(\partial_2 \Psi \cdot \Delta \Psi) = \operatorname{rot}(f)$$

d.h. skalare Gleichung 4. Ordnung. Mit  $v = \operatorname{rot} \Psi$  erhält man hieraus auch wieder Geschwindigkeit. Schwache Formulierung dieser Gleichung: Finde  $\Psi \in H_0^2(\Omega)$  mit

$$\nu \cdot (\Delta \Psi, \Delta \varphi) - (\partial_1 \Psi \cdot \Delta \Psi, \partial_2 \varphi) - (\partial_2 \Psi \cdot \Delta \Psi, \partial_1 \varphi) = (\operatorname{rot}(f), \varphi)$$

für  $\varphi \in H_0^2(\Omega)$ .

**Bemerkung** Dies erfordert Elemente, deren 2. Ableitung stetig ist. Für Lagrange-Elemente nicht erfüllt. Daher neue Formulierung.

Gesucht ist  $(\omega, \Psi) \in L^2(\Omega) \times H_0^2(\Omega)$  mit

$$(\omega, \psi) + (\Delta \Psi, \psi) = 0 \quad (8.5)$$

$$\nu \cdot (\nabla \omega, \nabla \varphi) + (\partial_1 \Psi \cdot \omega, \partial_2 \varphi) - (\partial_2 \Psi \cdot \omega, \partial_1 \varphi) = (\operatorname{rot}(f), \varphi) \quad (8.6)$$

für  $\psi \in L^2(\Omega)$ ,  $\varphi \in H_0^1(\Omega)$ . Diese Formulierung ist ebenfalls nicht brauchbar, da  $\Psi \in H_0^2(\Omega)$ . Aber: Partielle Integration in (8.5) gibt

$$(\omega, \psi) - (\nabla \Psi, \nabla \psi) = 0 \quad (\psi \in H^1(\Omega))$$

Gesucht ist also nun  $(\omega, \Psi) \in H^1(\Omega) \times H_0^1(\Omega)$ . Wähle

$$V_h := \{\psi_h \in H^1(\Omega); \forall T \in \mathcal{T}_h : \psi_h|_T \in \mathcal{P}(T)\}$$

$$V_h^0 := \{\psi_h \in H_0^1(\Omega); \forall T \in \mathcal{T}_h : \psi|_T \in \mathcal{P}^0(T)\}$$

z.B.  $\mathcal{P}(T) = \mathcal{P}^0(T) = \mathcal{P}_1(T)$  lineare Dreieckselemente.

Betrachte nun den linearen Fall:

$$\begin{aligned} (\omega_h, \psi_h) - (\nabla \Psi_h, \nabla \psi_h) &= 0 \\ \nu \cdot (\nabla \omega_h, \nabla \varphi_h) &= (f, \operatorname{rot}(\varphi_h)) \end{aligned}$$

für  $\psi_h \in V_h$ ,  $\varphi_h \in V_h^0$ . Dies gibt

$$\begin{pmatrix} M & B \\ -B^T & 0 \end{pmatrix} \cdot \begin{pmatrix} W_h \\ P_h \end{pmatrix} = \begin{pmatrix} 0 \\ b \end{pmatrix}$$

mit

$$\begin{aligned}
 M &:= ((\psi_h^i, \psi_h^j))_{i,j=1,\dots,N} \\
 B &:= ((\nabla \varphi_h^i, \nabla \varphi_h^j))_{i=1,\dots,N^0, j=1,\dots,N} \\
 b &:= ((f, \text{rot}(\varphi_i)))_{i=1,\dots,N^0} \\
 \omega_h &= \sum_{i=1}^N W_h^i \cdot \psi_h^i \\
 \Psi_h &= \sum_{i=1}^{N^0} P_h^i \cdot \varphi_h^i
 \end{aligned}$$

$M$  ist regulär, somit  $P_h = -M^{-1} \cdot B \cdot W_h$  und

$$\underbrace{B^T \cdot M^{-1} \cdot B}_{=: \Sigma} \cdot W_h = b$$

$\Sigma$  heißt Schur-Komplement. Damit

$$\begin{pmatrix} M & B^T \\ -B^T & 0 \end{pmatrix} = \begin{pmatrix} M & 0 \\ B^T & \Sigma \end{pmatrix} \cdot \begin{pmatrix} I & M^{-1} \cdot B \\ 0 & I \end{pmatrix}$$

Die Matrix auf der linken Seite ist genau dann regulär, wenn  $\Sigma$  regulär ist bzw. wenn  $B$  injektiv ist. Dies ist erfüllt, falls

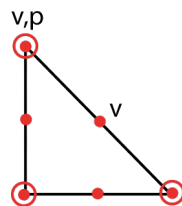
$$\inf_{\varphi_h \in V_h^0} \sup_{\psi_h \in V_h} \frac{(\nabla \varphi_h, \nabla \psi_h)}{\|\nabla \varphi_h\| \cdot \|\nabla \psi_h\|} > 0$$

(Babuska-Brezzi-Bedingung).

Anwendet auf stationäres Stokes-Problem:

$$\begin{aligned}
 \nu \cdot (\nabla v, \nabla \varphi) - (p, \text{div} \varphi) &= (f, \varphi) \\
 (\chi, \text{div} v) &= 0
 \end{aligned}$$

für  $\varphi \in H = (H_0^1(\Omega))^d$ ,  $\chi \in L = L^2(\Omega)$ . Wähle  $H_h \subseteq H$ ,  $L_h \subseteq L$ , sodass Babuska-Brezzi-Bedingung erfüllt ist, zum Beispiel Taylor-Hood-Element.



Nun Anwendung auf stationäre Navier-Stokes-Gleichung:

$$\begin{aligned}
 \nu \cdot (\nabla v, \nabla \varphi) + (v \cdot \nabla v, \varphi) - (\varphi, \text{div} \varphi) &= (f, \varphi) \\
 (\chi, \text{div} v) &= 0
 \end{aligned}$$

für  $\varphi \in H$ ,  $\chi \in L$ . Erfordert iterative Lösung wobei der Term  $(v \cdot \nabla v, \varphi)$  linearisiert wird.

(i). Stokes-Linearisierung für  $\frac{1}{\nu} \ll 1$ :

$$\begin{aligned}
 -\nu \cdot \Delta v^\ell + \nabla p^\ell &= f - v^{\ell-1} \cdot \nabla v^{\ell-1} \\
 \nabla \cdot v^\ell &= 0
 \end{aligned}$$



(ii). Oseen-Linearisierung:

$$\begin{aligned} -\nu \cdot \Delta v^\ell + \bar{v} \cdot \nabla v^\ell + \nabla p^\ell &= f \\ \nabla \cdot v^\ell &= 0 \end{aligned}$$

mit z.B.  $\bar{v} = v_0$ ,  $\bar{v} = v^{\ell-1}$  oder  $\bar{v} = 2v^{\ell-1} - v^{\ell-2}$  (Extrapolation)

(iii). Newton-Linearisierung:

$$\begin{aligned} -\nu \cdot \Delta v^\ell + v^{\ell-1} \cdot \nabla v^\ell + v^\ell \cdot \nabla v^{\ell-1} + \nabla p^\ell &= f + v^{\ell-1} \cdot \nabla v^{\ell-1} \\ \nabla \cdot v^\ell &= 0 \end{aligned}$$

Dies entspricht Newton-Verfahren zur Berechnung einer Nullstelle in der stationären Navier-Stokes-Gleichung.