

**Autorenadresse:**

Dr. rer. nat. Michael Jung  
Technische Universität Chemnitz–Zwickau  
Fakultät für Mathematik  
D – 09107 Chemnitz

Prof. Dr. rer. nat. habil. Ulrich Langer  
(O. Univ.–Professor)  
Johannes–Kepler Universität Linz  
Technisch–Naturwissenschaftliche Fakultät  
Institut für Mathematik  
Ordinariat „Numerische Mathematik“  
Altenberger Straße 69  
A – 4040 Linz Österreich

**Preview from Notesale.co.uk**  
**Page 2 of 188**

<b>4</b>	<b>FEM für mehrdimensionale Randwertprobleme 2. Ordnung</b>	<b>99</b>
4.1	Modellprobleme . . . . .	99
4.2	Herleitung der Variationsformulierung . . . . .	101
4.3	Galerkin–Ritz–FEM . . . . .	104
4.3.1	Galerkin–Verfahren . . . . .	104
4.3.2	Ritz–Verfahren . . . . .	106
4.3.3	FEM = Ritz–Galerkin–Verfahren mit speziellen Ansatzfunktionen . . . . .	107
4.4	FEM mit linearen Dreieckselementen . . . . .	111
4.4.1	Gebietsdiskretisierung (Triangularisierung) . . . . .	111
4.4.2	Definition der Ansatz- und Testfunktionen . . . . .	118
4.4.3	Aufbau des FE–Gleichungssystems . . . . .	121
4.4.4	Ein Konvergenzresultat . . . . .	135
4.4.5	Ein Beispiel . . . . .	136
4.4.6	Das Programm FEM2D . . . . .	142
<b>5</b>	<b>Au ösung von Finite–Elemente–Gleichungssystemen</b>	<b>147</b>
5.1	Direkte Verfahren . . . . .	148
5.2	Iterative Verfahren . . . . .	154
5.2.1	Das Jacobi– und das Gauß–Seidel–Verfahren . . . . .	155
5.2.2	Die Methode der konjugierten Gradienten ohne Vorkonditionierung . . . . .	156
5.2.3	Die Methode der konjugierten Gradienten mit Vorkonditionierung . . . . .	158
5.2.4	Mehrgitterverfahren . . . . .	161
5.2.5	Schlußbemerkungen . . . . .	166
5.3	Ein Vergleich der Lösungsverfahren . . . . .	168
<b>6</b>	<b>Galerkin–FEM für parabolische Anfangs–Randwertaufgaben</b>	<b>177</b>
6.1	Die stetige, die semidiskrete und die volldiskrete Aufgabe . . . . .	177
6.2	Konvergenz und Stabilität . . . . .	180
	<b>Literaturverzeichnis</b>	<b>183</b>

## 1.2 Zur Geschichte der FEM

### • Vorgeschichte

In der Arbeit [73] beschreibt SCHELLBACH die Lösung eines Minimalflächenproblems. Bei einem solchen Problem wird die kleinste Fläche gesucht, deren äußerer Rand eine im Raum gegebene geschlossene Kurve ist. Der von SCHELLBACH beschriebene Lösungsweg beinhaltet Teilschritte, wie sie für die Finite-Elemente-Methode charakteristisch sind. Seine Vorgehensweise kann als FEM mit linearen Dreieckselementen auf einem regelmäßigen Gitter interpretiert werden (siehe auch Abschnitt 4.3.3).

Weiterhin zählen die Arbeiten von RITZ und GALERKIN zur Vorgeschichte der Finite-Elemente-Methode. Die FEM ist nämlich ein spezielles Ritz- bzw. Galerkin-Verfahren (siehe auch die Abschnitte 4.2 und 4.3).

#### 1. W. RITZ (1908, siehe auch [68])

Es ist ein Funktional  $J(u)$  zu minimieren, wobei  $u$  alle zulässigen Funktionen durchläuft, z.B.:

Gesucht ist die Funktion  $u^*(x)$  aus der Menge aller über dem Intervall  $[0, 1]$  stetig differenzierbaren Funktionen  $u(x)$  mit  $u(0) = u(1) = 0$ , die das Funktional

$$J(u) = \int_0^1 [(u'(x))^2 - \sin \pi x u(x)] dx \quad (1.1)$$

minimiert.

Eine Näherungslösung für  $u^*$  wird beim Ritzschen Verfahren auf die folgende Weise bestimmt: Man wählt ein System linear unabhängiger Funktionen  $\varphi_i(x)$ ,  $i = 1, 2, \dots, n$ , und setzt den Ausdruck

$$u^{(n)}(x) = \sum_{i=1}^n u_i \varphi_i(x) \quad (1.2)$$

mit beliebigen reellen Zahlen  $u_i$  in das Funktional  $J(u)$  ein. Damit hängt das Funktional nur von diesen reellen Zahlen ab. Das Minimum des Funktionals über allen Funktionen, die sich gemäß der Beziehung (1.2) darstellen lassen, kann somit aus den Gleichungen

$$\frac{\partial J(u^{(n)})}{\partial u_1} = 0, \quad \frac{\partial J(u^{(n)})}{\partial u_2} = 0, \quad \dots, \quad \frac{\partial J(u^{(n)})}{\partial u_n} = 0$$

ermittelt werden. Die Güte der Näherung  $u^{(n)}$  für die Funktion  $u^*$  hängt von  $n$  ab.

Zur Bestimmung einer Näherung für das Minimum des Funktionals (1.1) kann man beispielsweise

$$\varphi_i(x) = (1-x)x^i \quad (1.3)$$

wählen.

#### 2. GALERKIN (1915)

Betrachtet wird eine Randwertaufgabe, z.B.: Gesucht ist die zweimal stetig differenzierbare Funktion  $u(x)$ , für die

$$-u''(x) = f(x) \quad \text{für alle } x \in (0, 1) \quad \text{und} \quad u(0) = u(1) = 0$$

Gesucht ist das elektrische Potential  $\varphi(x_1, x_2)$ , so daß

$$\begin{aligned} -\Delta\varphi &= 0 && \text{für alle } x \in \\ \varphi &= g_{11} && \text{für alle } x \in \Gamma_{11} \\ \varphi &= 0 && \text{für alle } x \in \partial \setminus \Gamma_{11} \end{aligned}$$

gilt.

Die Abbildung 1.4 zeigt die Äquipotentiallinien des elektrischen Potentials.

### 1.3.3 Berechnung elektromagnetischer Felder

Den Ausgangspunkt zur Aufstellung des mathematischen Modells bildet die *Maxwell'sche Gleichung*

$$\operatorname{rot} \vec{H} = \vec{S}, \quad (1.16)$$

wobei  $\vec{H}$  die magnetische Feldstärke und  $\vec{S}$  die Stromdichte bezeichnen (siehe auch [49]). Weiterhin führen wir das Vektorpotential  $\vec{A}$  ein, das durch die Gleichungen

$$\operatorname{rot} \vec{A} = \vec{B} \quad (1.17)$$

und

$$\operatorname{div} \vec{A} = 0 \quad (1.18)$$

definiert ist. Hierbei ist  $\vec{B}$  die magnetische Induktion. Der Zusammenhang zwischen der magnetischen Induktion  $\vec{B}$  und der magnetischen Feldstärke  $\vec{H}$  ist durch

$$\vec{B} = \mu_0 \mu_r (\vec{H} + \vec{H}_0) \quad (1.19)$$

mit der magnetischen Feldkonstante  $\mu_0$  und der materialabhängigen Permeabilitätszahl  $\mu_r$  gegeben. Für die nicht permanentmagnetischen Materialien wird  $\vec{H}_0 = \vec{0}$  vorausgesetzt. Bei Permanentmagneten bezeichnet  $-\vec{H}_0$  die magnetische Feldstärke, bei der die Induktion verschwindet (siehe auch [38]). Besteht für den Permanentmagneten ein linearer Zusammenhang zwischen  $\vec{B}$  und  $\vec{H}$ ,

$$\vec{B} = \mu_0 \mu_r \vec{H} + \vec{J}_0, \quad (1.20)$$

so ist

$$\vec{H}_0 = (\mu_0 \mu_r)^{-1} \vec{J}_0, \quad (1.21)$$

wobei  $\vec{J}_0$  die Permanentmagnetisierung bezeichnet. Für nicht ferromagnetische Materialien ist die relative Permeabilitätszahl  $\mu_r$  eine Konstante, und für Ferromagnetika besteht bei Vernachlässigung der Hysterese ein eindeutiger Zusammenhang zwischen  $\vec{B}$  und  $\vec{H}$ , so daß sich  $\mu_r$  als

$$\mu_r = \mu_r(|\vec{B}|) \quad (1.22)$$

aufschreiben läßt.

Der Querschnitt ist konstant in  $x_3$ -Richtung. Außerdem wirkt die Kraft  $F$  in Ebenen senkrecht zur  $x_3$ -Achse, und sie ist unabhängig von der  $x_3$ -Richtung. Deshalb können wir annehmen, daß

$$\frac{\partial u_1}{\partial x_3} = \frac{\partial u_2}{\partial x_3} = u_3 = 0 \quad (1.29)$$

gilt und folglich ist

$$\varepsilon_{13} = \varepsilon_{23} = \varepsilon_{33} = 0.$$

Unter Beachtung der Beziehung (1.29) erhalten wir aus der Randwertaufgabe (1.25) das folgende ebene lineare Elastizitätsproblem (*ebener Verzerrungszustand*):

Gesucht ist das Verschiebungsfeld  $\vec{u}(x) = (u_1(x), u_2(x))^T$ , für das

$$\begin{aligned} -\mu_e \left( \frac{\partial^2 u_1}{\partial x_1^2} + \frac{\partial^2 u_1}{\partial x_2^2} \right) - (\lambda_e + \mu_e) \frac{\partial}{\partial x_1} \left( \frac{\partial u_1}{\partial x_1} + \frac{\partial u_2}{\partial x_2} \right) &= 0 && \text{für alle } x \in \Gamma_1 \\ -\mu_e \left( \frac{\partial^2 u_2}{\partial x_1^2} + \frac{\partial^2 u_2}{\partial x_2^2} \right) - (\lambda_e + \mu_e) \frac{\partial}{\partial x_2} \left( \frac{\partial u_1}{\partial x_1} + \frac{\partial u_2}{\partial x_2} \right) &= 0 && \text{für alle } x \in \Gamma_2 \end{aligned} \quad (1.30)$$

$$\vec{u}(x) = \vec{0} \quad \text{für alle } x \in \Gamma_1$$

$$\sigma_{11}n_1 + \sigma_{12}n_2 = 0 \quad \text{für alle } x \in \Gamma_2$$

$$\sigma_{21}n_1 + \sigma_{22}n_2 = g_{22} \quad \text{für alle } x \in \Gamma_2$$

gilt. In unserem Beispiel ist  $E = 1,26 \cdot 10^{11} \text{ Nm}^{-2}$ ,  $\nu = 0,3$  und

$$g_{22} = \begin{cases} F = -10^6 x_1 - 4 \cdot 10^5 \text{ Nm}^{-2} & \text{auf der Oberseite des Trägers} \\ & (x_1 \in [0,0, 0,18], x_2 = 0,24) \\ 0 \text{ Nm}^{-2} & \text{sonst.} \end{cases}$$

Als zweites Beispiel betrachten wir ein dickwandiges Rohr unter Innendruck. Der Querschnitt des Rohres ist in der Abbildung 1.8 dargestellt. Auf Grund der Rotationssymmetrie des Rohres ist es sinnvoll, zur Bestimmung des Verschiebungsfeldes die Randwertaufgabe der linearen Elastizitätstheorie in Zylinderkoordinaten  $(r, \varphi, z)$  zu formulieren.

In diesem Koordinatensystem sind die Komponenten des Verzerrungstensors durch die Beziehungen

$$\varepsilon_{rr} = \frac{\partial u_r}{\partial r}, \quad \varepsilon_{\varphi\varphi} = \frac{1}{r} \left( \frac{\partial u_\varphi}{\partial \varphi} + u_r \right), \quad \varepsilon_{zz} = \frac{\partial u_z}{\partial z}, \quad (1.31)$$

$$\varepsilon_{r\varphi} = \frac{1}{2} \left( \frac{1}{r} \frac{\partial u_r}{\partial \varphi} + \frac{\partial u_\varphi}{\partial r} - \frac{1}{r} u_\varphi \right), \quad \varepsilon_{rz} = \frac{1}{2} \left( \frac{\partial u_r}{\partial z} + \frac{\partial u_z}{\partial r} \right), \quad \varepsilon_{\varphi z} = \frac{1}{2} \left( \frac{\partial u_\varphi}{\partial z} + \frac{1}{r} \frac{\partial u_z}{\partial \varphi} \right)$$

- *Übergang zur differentiellen Form*

Multiplizieren wir die Gleichung (2.11) mit  $1/(\Delta t \Delta x)$  und lassen  $\Delta x$  sowie  $\Delta t$  gegen Null streben, dann erhalten wir bei glatten Eingangsdaten (z.B. homogenes Material, stetig verteilte Wärmequelle, siehe auch die Voraussetzungen (2.3)) die differentielle Form der instationären Wärmeleitgleichung.

Gesucht ist  $u(x, t) \in C^{2,1}(Q_T) \cap C(\overline{Q_T})$ , so daß

$$c(x, t) \rho(x, t) \frac{\partial u}{\partial t} - \frac{\partial}{\partial x} \left( \lambda(x, t) \frac{\partial u}{\partial x} \right) + \bar{q}(x, t) u(x, t) = f(x, t) + \bar{q}(x, t) u_A(x, t) \quad (2.12)$$

für alle  $(x, t) \in Q_T$

gilt und die Randbedingungen (z.B. 1. Art)

$$\left. \begin{aligned} u(a, t) &= g_a(t) \\ u(b, t) &= g_b(t) \end{aligned} \right\} \text{ für alle } t \in [0, T]$$

sowie die Anfangsbedingung

$$u(x, 0) = u_0(x) \quad \text{für alle } x \in (a, b)$$

erfüllt werden.

Preview from Notesale.co.uk  
Page 48 of 188

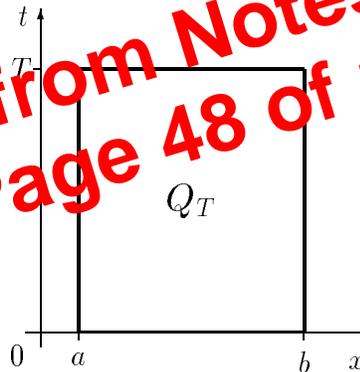


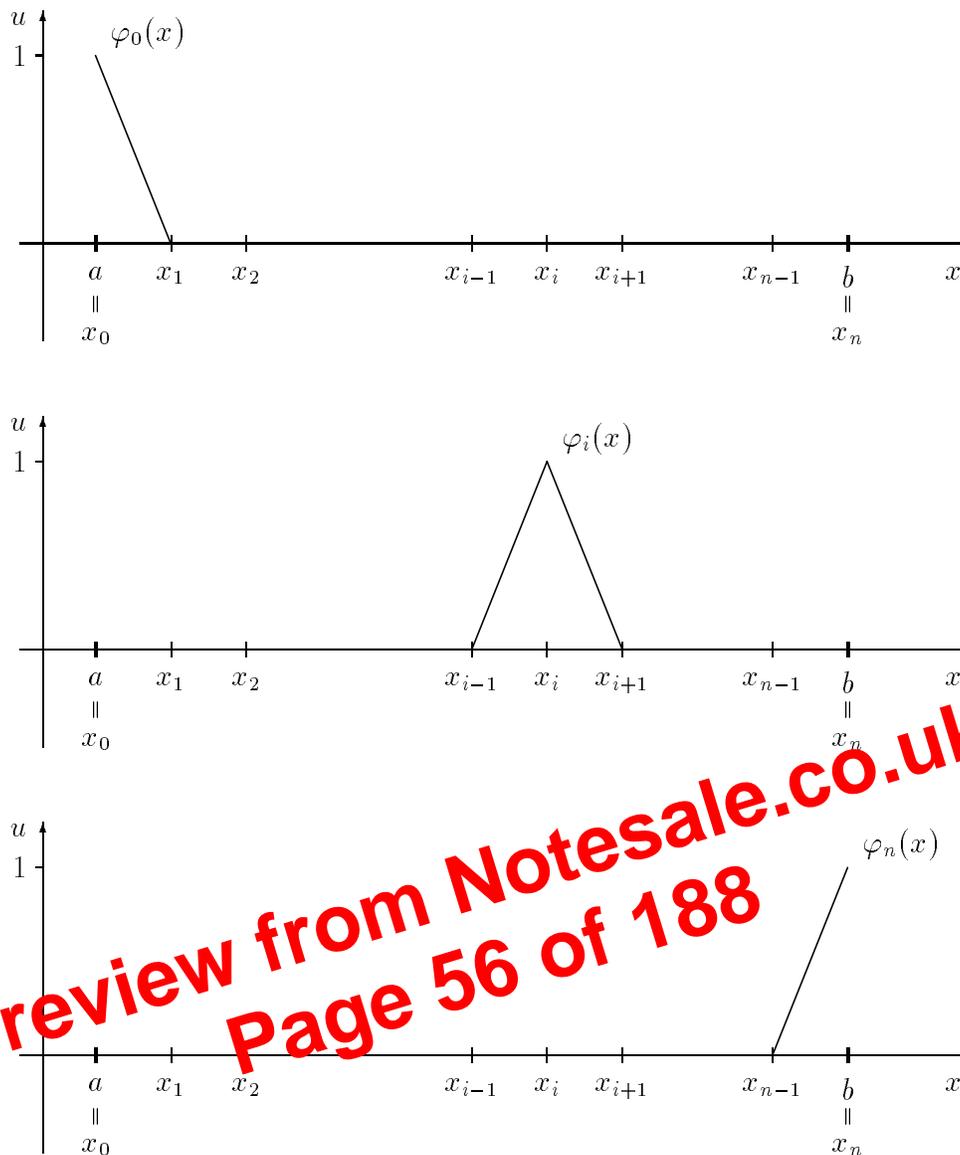
Abbildung 2.13: Raum-Zeit-Zylinder  $Q_T = (a, b) \times (0, T)$

**Bemerkung 2.8** Der Raum  $C^{2,1}(Q_T)$  umfaßt alle Funktionen, die zweimal stetig differenzierbar bezüglich des Ortes  $x \in (a, b)$  und einmal stetig differenzierbar bezüglich der Zeit  $t \in (0, T)$  sind.

**Bemerkung 2.9** Analog zu der stationären Wärmeleitgleichung können auch bei der instationären Wärmeleitgleichung Randbedingungen 2. Art oder 3. Art bzw. gemischte Randbedingungen gestellt werden.

**Bemerkung 2.10** Für die Existenz einer *klassischen* Lösung, d.h. Lösung der Aufgabe (2.12), ist die *Kompatibilität* zwischen der Anfangsbedingung und den Randbedingungen notwendig, d.h.

$$\lim_{t \rightarrow +0} g_a(t) = u_0(a) \quad , \quad \lim_{t \rightarrow +0} g_b(t) = u_0(b)$$

Abbildung 3.5: Stückweise lineare Funktionen  $\varphi_i(x)$ 

Für die Funktionen  $\varphi_i(x)$ ,  $i = 1, 2, \dots, n - 1$ , gilt

$$\varphi_i(x) = \begin{cases} 0 & a \leq x \leq x_{i-1} \\ \frac{x - x_{i-1}}{h} & x_{i-1} < x \leq x_i \\ -\frac{x - x_{i+1}}{h} & x_i < x \leq x_{i+1} \\ 0 & x_{i+1} < x \leq b \end{cases} \quad (3.7)$$

und für die Funktion  $\varphi_0(x)$  bzw.  $\varphi_n(x)$

$$\varphi_0(x) = \begin{cases} -\frac{x - x_1}{h} & a < x \leq x_1 \\ 0 & x_1 < x \leq b \end{cases}, \quad \varphi_n(x) = \begin{cases} 0 & a \leq x \leq x_{n-1} \\ \frac{x - x_{n-1}}{h} & x_{n-1} < x \leq b \end{cases}. \quad (3.8)$$

Die Matrizen  $K^{(i)}$ ,  $i = 1, 2, \dots, n$ ,

$$\begin{aligned}
 K^{(i)} &= \begin{pmatrix} K_{11}^{(i)} & K_{12}^{(i)} \\ K_{21}^{(i)} & K_{22}^{(i)} \end{pmatrix} \\
 &= \begin{pmatrix} \int_{x_{i-1}}^{x_i} \varphi'_{i-1}(x) \varphi'_{i-1}(x) dx & \int_{x_{i-1}}^{x_i} \varphi'_i(x) \varphi'_{i-1}(x) dx \\ \int_{x_{i-1}}^{x_i} \varphi'_{i-1}(x) \varphi'_i(x) dx & \int_{x_{i-1}}^{x_i} \varphi'_i(x) \varphi'_i(x) dx \end{pmatrix}.
 \end{aligned} \tag{3.15}$$

werden als *Elementsteifigkeitsmatrizen* bezeichnet.

Auf analoge Weise erhalten wir für die rechte Seite

$$\begin{aligned}
 \int_a^b f(x) v_h(x) dx &= \sum_{i=1}^n \int_{x_{i-1}}^{x_i} f(x) v_h(x) dx \\
 &= \sum_{i=1}^n \int_{x_{i-1}}^{x_i} f(x) (v_{i-1} \varphi_{i-1}(x) + v_i \varphi_i(x)) dx \\
 &= \sum_{i=1}^n (v_{i-1} v_i) \underline{f}^{(i)}
 \end{aligned} \tag{3.16}$$

Die *Elementlastvektoren*  $\underline{f}^{(i)}$ ,  $i = 1, 2, \dots, n$ , sind durch die Beziehung

$$\underline{f}^{(i)} = \begin{pmatrix} \int_{x_{i-1}}^{x_i} f(x) \varphi_{i-1}(x) dx \\ \int_{x_{i-1}}^{x_i} f(x) \varphi_i(x) dx \end{pmatrix}. \tag{3.17}$$

definiert.

- *Assemblierung des FEM Gleichungssystems*, d.h. Zusammensetzen der Elementsteifigkeitsmatrizen und Elementlastvektoren zur globalen Steifigkeitsmatrix bzw. zum globalen Lastvektor.

Wir schreiben die Beziehungen (3.14) und (3.16) nochmals ausführlich auf. Es gilt

$$\int_a^b u'_h(x) v'_h(x) dx = \sum_{i=1}^n (v_{i-1} \ v_i) K^{(i)} \begin{pmatrix} u_{i-1} \\ u_i \end{pmatrix} = \underline{\underline{v}}_h^T \bar{K}_h \underline{\underline{u}}_h =$$

Der Einbau von  $K^{(2)}$  und  $f^{(2)}$  liefert

$$\begin{pmatrix} K_{11}^{(1)} & K_{12}^{(1)} & 0 & 0 & \cdots & 0 \\ K_{21}^{(1)} & K_{22}^{(1)} + K_{11}^{(2)} & K_{12}^{(2)} & 0 & \cdots & 0 \\ 0 & K_{21}^{(2)} & K_{22}^{(2)} & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 0 \end{pmatrix} \quad \text{und} \quad \begin{pmatrix} f_1^{(1)} \\ f_2^{(1)} + f_1^{(2)} \\ f_2^{(2)} \\ 0 \\ \vdots \\ 0 \end{pmatrix}.$$

Nach der Assemblierung aller  $K^{(i)}$  bzw.  $\underline{f}^{(i)}$ ,  $i = 1, 2, \dots, n$ , entstehen die globale Steifigkeitsmatrix

$$\bar{K}_h = \begin{pmatrix} K_{11}^{(1)} & K_{12}^{(1)} & 0 & 0 & 0 & \cdots & 0 \\ K_{21}^{(1)} & K_{22}^{(1)} + K_{11}^{(2)} & K_{12}^{(2)} & 0 & 0 & \cdots & 0 \\ 0 & K_{21}^{(2)} & K_{22}^{(2)} + K_{11}^{(3)} & K_{12}^{(3)} & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & K_{21}^{(n-2)} & K_{22}^{(n-2)} + K_{11}^{(n-1)} & K_{12}^{(n-1)} & 0 \\ 0 & \cdots & 0 & 0 & K_{21}^{(n-1)} & K_{22}^{(n-1)} + K_{11}^{(n)} & K_{12}^{(n)} \\ 0 & \cdots & 0 & 0 & 0 & K_{21}^{(n)} & K_{22}^{(n)} \end{pmatrix}$$

sowie der globale Lastvektor

$$\underline{\bar{f}}_h = \begin{pmatrix} f_1^{(1)} \\ f_2^{(1)} + f_1^{(2)} \\ f_2^{(2)} + f_1^{(3)} \\ \vdots \\ f_2^{(n-2)} + f_1^{(n-1)} \\ f_2^{(n-1)} + f_1^{(n)} \\ f_2^{(n)} \end{pmatrix},$$

wobei die Randbedingungen noch nicht berücksichtigt sind.

- *Einbau der Randbedingungen*

Im Beispiel aus dem Abschnitt 3.2 sind die folgenden Randbedingungen vorgegeben:

- Randbedingungen 1. Art bei  $x = a$ :  $u(a) = g_a$
- Randbedingungen 3. Art bei  $x = b$ :  $-u'(b) = \alpha_b(u(b) - g_b)$

Die folgenden Normen sind interessant:

1.  $C$ -Norm  $\|\cdot\|_C$

$$\|u - u_h\|_C := \max_{x \in [a,b]} |u(x) - u_h(x)|$$

2.  $L_2$ -Norm  $\|\cdot\|_0 = \|\cdot\|_{0,2,(a,b)}$

$$\|u - u_h\|_0 := \sqrt{\int_a^b (u(x) - u_h(x))^2 dx}$$

3.  $H^1$ -Norm  $\|\cdot\|_1 = \|\cdot\|_{1,2,(a,b)}$

$$\|u - u_h\|_1 := \sqrt{\int_a^b [(u(x) - u_h(x))^2 + ((u(x) - u_h(x)))']^2 dx}$$

Falls  $u \in V_g$  quadratisch integrierbare Ableitungen bis zur Ordnung  $k+1$  (zumindest elementweise) besitzt, d.h. wenn  $u', u'', \dots, u^{(k+1)} \in L_2(x_{j-1}, x_j)$  gilt und finite Elemente  $p$ -ter Ordnung ( $p = 1$  linear,  $p = 2$  quadratisch,  $p = 3$  kubisch) verwendet werden, dann erhalten wir folgende Diskretisierungsfehlerabschätzungen:

$$\begin{aligned} \|u - u_h\|_s &\leq c_s h^{\min\{k,p\}-s}, & s = 0, 1 \\ \|u - u_h\|_C &\leq c_\infty h^{\min\{k,p\}+0.5}. \end{aligned} \quad (3.31)$$

Dabei sind  $c_s$  und  $c_\infty$  nichtnegative Konstanten, die von der exakten Lösung  $u$  abhängen und vom Diskretisierungsparameter  $h$  unabhängig sind (siehe auch [13, 30]).

**Bemerkung 3.9** Um derartige Fehlerabschätzungen beweisen zu können, müssen wir voraussetzen, daß die Bilinearform  $a(\cdot, \cdot)$   $V_0$ -elliptisch und  $V_0$ -beschränkt ist, d.h.:

$$\begin{aligned} \text{Es existiert ein } \mu_1 > 0 : \quad a(v, v) &\geq \mu_1 \|v\|_1^2 && \text{für alle } v \in V_0 \quad \text{und} \\ \text{es existiert ein } \mu_2 > 0 : \quad |a(u, v)| &\leq \mu_2 \|u\|_1 \|v\|_1 && \text{für alle } u, v \in V_0. \end{aligned} \quad (3.32)$$

Im weiteren wollen wir die Abschätzung (3.31) für  $s = 1$  ( $\|\cdot\|_1$ ),  $p = 1$  (lineare Ansatzfunktionen) und die Modellaufgabe

$$\begin{aligned} -u''(x) &= f(x) && \text{in } (0, 1) \\ u(0) &= g_0 \\ u'(1) &= 0 \end{aligned} \quad (3.33)$$

beweisen.

$$\int_0^{0.5} 50u'(x)v'(x) dx + \int_{0.5}^1 371u'(x)v'(x) dx - 50u'(0.5)v(0.5) + 50u'(0)v(0) - 371u'(1)v(1) + 371u'(0.5)v(0.5) = 0.$$

Beachten wir noch die Interfacebedingung  $-50u'(0.5 - 0) = -371u'(0.5 + 0)$  sowie die Bedingung  $v(0) = v(1) = 0$ , so erhalten wir die Variationsformulierung:

Gesucht ist  $u(x) \in V_g = \{u(x) \in H^1(0,1) : u(0) = 0, u(1) = 50\}$ , so daß

$$\int_0^1 \lambda(x)u'(x)v'(x) dx = 0 \quad \text{für alle } v \in V_0 = \{v(x) \in H^1(0,1) : v(0) = v(1) = 0\} \quad (3.53)$$

mit

$$\lambda(x) = \begin{cases} 50 & \text{für } x < 0.5 \\ 371 & \text{für } x > 0.5 \end{cases}$$

gilt.

Diese Variationsformulierung ist der Ausgangspunkt einer Finite-Elemente-Diskretisierung. Zur Demonstration des Algorithmus der Methode der finiten Elemente zerlegen wir das Intervall  $(0,1)$  in 4 Teilintervalle. Um auch den Fall einer nicht äquidistanten Unterteilung erläutern zu können, wählen wir die folgende Verlegung:

$$(0,1) = \bigcup_{i=1}^4 [x_{i-1}, x_i] \quad (3.54)$$

mit

$$(x_0, x_1) = (0, 0.3), \quad (x_1, x_2) = (0.3, 0.5), \quad (x_2, x_3) = (0.5, 0.7), \quad (x_3, x_4) = (0.7, 1).$$

Wesentlich ist dabei, daß die Materialgrenze, d.h. der Punkt, in dem die Interfacebedingung gegeben ist, als Knoten und damit als End- bzw. Anfangspunkt zweier Teilintervalle gewählt wird.

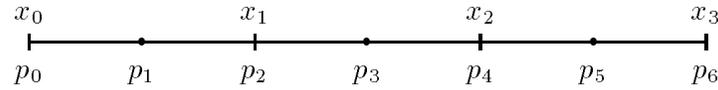
Wir wissen, daß die Lösung des Randwertproblems linear in den Teilgebieten  $\Omega_1 = (0, 0.5)$  und  $\Omega_2 = (0.5, 1)$  ist. Deshalb würde bereits bei einer FE-Diskretisierung mit den zwei finiten Elementen  $(0, 0.5)$  und  $(0.5, 1)$  sowie stückweise linearen Ansatzfunktionen die Lösung des Randwertproblems exakt approximiert. Wir wählen die obenbeschriebene Diskretisierung nur, um den FE-Algorithmus nochmals ausführlich demonstrieren zu können.

Die stückweise linearen Ansatzfunktionen sind durch

$$\varphi_i(x) = \begin{cases} 0 & x \leq x_{i-1} \\ \frac{x - x_{i-1}}{x_i - x_{i-1}} & x_{i-1} < x \leq x_i \\ -\frac{x - x_{i+1}}{x_{i+1} - x_i} & x_i < x \leq x_{i+1} \\ 0 & x_{i+1} < x \end{cases} \quad (3.55)$$

Bei der FE-Diskretisierung der Aufgabe (3.61) wollen wir die im Abschnitt 3.5 definierten stückweise quadratischen Ansatzfunktionen nutzen. Wir zerlegen zuerst das Intervall  $(0, 2)$  in 3 gleichgroße Teilintervalle, d.h.

$$[0, 2] = \bigcup_{i=1}^3 [x_{i-1}, x_i] \quad \text{mit} \quad (x_0, x_1) = (0, \frac{2}{3}), \quad (x_1, x_2) = (\frac{2}{3}, \frac{4}{3}) \quad \text{und} \quad (x_2, x_3) = (\frac{4}{3}, 2).$$



globale Knotennummern:

$$0 \quad 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6$$

lokale Knotennummern:

$$1 \quad 2 \quad 3 \quad 1 \quad 2 \quad 3 \quad 1 \quad 2 \quad 3$$

Abbildung 3.15: Zerlegung des Intervalls  $(0, 2)$  und die Zuordnung zwischen globaler und lokaler Knotennumerierung

Die stückweise quadratischen Ansatzfunktionen  $\varphi_j(x)$ ,  $j = 0, 1, \dots, 6$ , definieren wir, indem wir die auf dem Element  $\hat{\Omega} = (0, 1)$  definierten Formfunktionen (siehe Abschnitt 3.5) auf das jeweilige Element transformieren. Dazu verwenden wir die Transformationsvorschrift

$$\xi(x) = \frac{x - x_{i-1}}{x_i - x_{i-1}} = \frac{1}{h_i} (x - x_{i-1}), \quad x(\xi) = h_i \xi + x_{i-1}$$

und die Formfunktionen

$$\hat{\Phi}_1(\xi) = 2\xi^2 - 3\xi + 1, \quad \hat{\Phi}_2(\xi) = -4\xi^2 + 4\xi, \quad \hat{\Phi}_3(\xi) = 2\xi^2 - \xi.$$

Wir erhalten

$$\varphi_0(x) = \begin{cases} \hat{\Phi}_1(\xi(x)) = \hat{\Phi}_1(\frac{1}{h_1}(x)) & \text{für } 0 \leq x < \frac{2}{3} \\ 0 & \text{für } \frac{2}{3} \leq x \end{cases}$$

$$\varphi_{2i}(x) = \begin{cases} 0 & \text{für } x \leq x_{i-1} \\ \hat{\Phi}_3(\frac{1}{h_i}(x - x_{i-1})) & \text{für } x_{i-1} \leq x < x_i \\ \hat{\Phi}_1(\frac{1}{h_{i+1}}(x - x_i)) & \text{für } x_i \leq x < x_{i+1} \\ 0 & \text{für } x_{i+1} \leq x \end{cases}, \quad i = 1, 2$$

- 5. Variante:  $p = 70$ , 10 Elemente mit linearen Ansatzfunktionen, wobei  $[0, 1] = [0, 0.5] \cup [0.5, 0.8] \cup [0.8, 0.85] \cup [0.85, 0.9] \cup [0.9, 0.925] \cup [0.925, 0.95] \cup [0.95, 0.9625] \cup [0.9625, 0.975] \cup [0.975, 0.9875] \cup [0.9875, 1]$

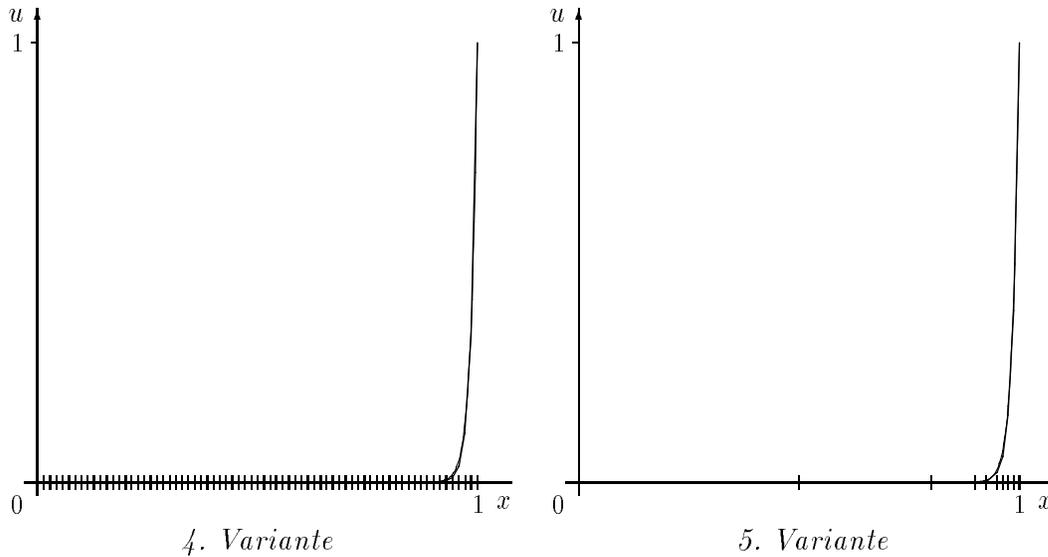


Abbildung 3.19: FE-Diskretisierungen mit gleichmäßiger und lokaler Netzverfeinerung

Eine weitere Möglichkeit zur Anpassung der FE-Diskretisierung an das Lösungsverhalten ist die Erhöhung des Polynomgrades der Ansatzfunktionen. Die folgende Diskretisierung zeigt, daß auf diese Weise ebenfalls eine Näherungslösung mit einem kleinen Diskretisierungsfehler erzeugt werden kann.

- 6. Variante:  $p = 70$ , Element  $[0, 0.5]$  mit linearen Ansatzfunktionen, Elemente  $[0.5, 0.8]$ ,  $[0.8, 0.9]$  und  $[0.9, 0.95]$  mit quadratischen Ansatzfunktionen sowie das Element  $[0.95, 1]$  mit kubischen Ansatzfunktionen

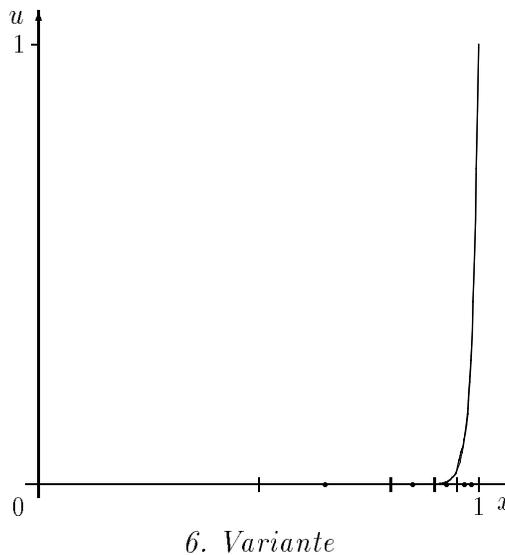


Abbildung 3.20: FE-Diskretisierung mit linearen, quadratischen und kubischen Ansatzfunktionen

### 4.3.3 FEM = Ritz–Galerkin–Verfahren mit speziellen Ansatzfunktionen

Die grundlegende Idee, Ansatzfunktionen (Testfunktionen)  $p_i = p_i(x)$  mit *lokalem Träger* ( *inite Funktionen*) im Ritz– bzw. Galerkin–Verfahren zu verwenden, stammt von R. COURANT (1943, [14]) und wurde Mitte der 50er Jahre von Ingenieuren (z.B. von TURNER 1956, [81]) neu entdeckt.

Bei der Finite–Elemente–Methode wird das Gebiet in „kleine“, nichtüberlappende Teilgebiete  $\delta^{(r)}$  (Dreiecke oder Vierecke) zerlegt. Die Eckpunkte dieser Teilgebiete bezeichnen wir als *Knoten*. Als Ansatzfunktionen wählen wir Funktionen, die nur über sehr wenigen Dreiecken bzw. Vierecken von Null verschieden sind. Eine derartige Ansatzfunktion ist in der Abbildung 4.2 dargestellt.

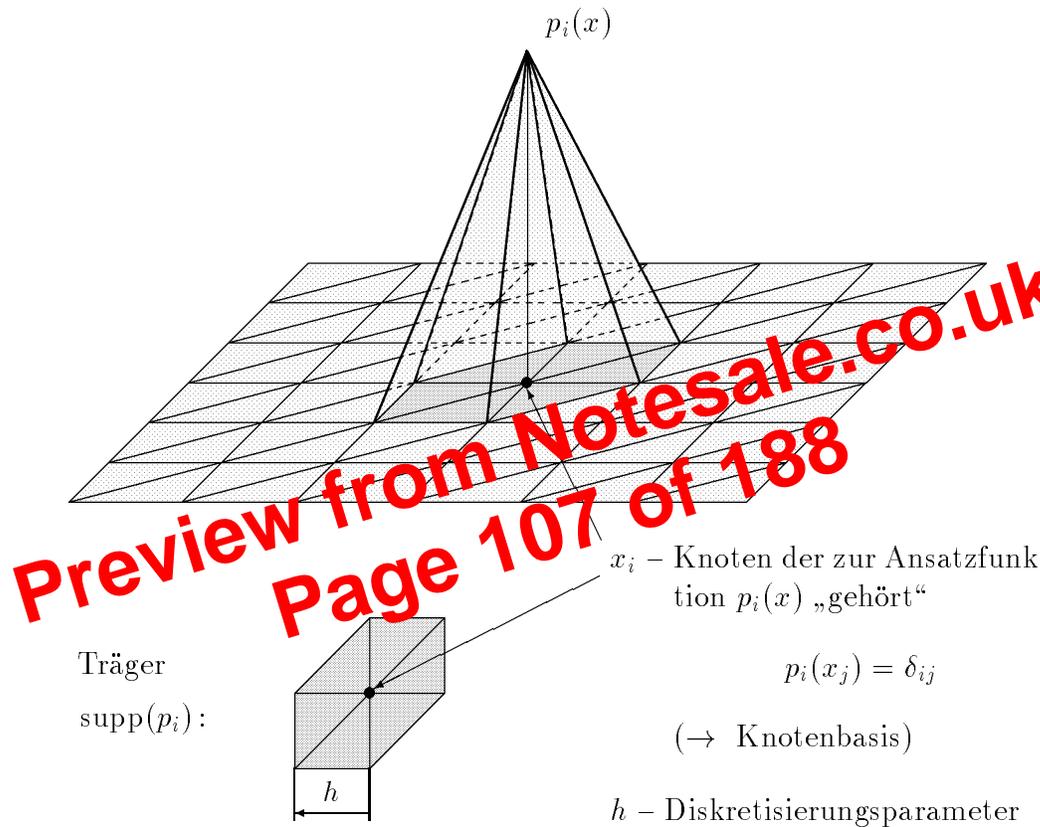


Abbildung 4.2: Darstellung einer Ansatzfunktion mit lokalem Träger

Unter Verwendung derartiger Ansatzfunktionen wird die Näherungslösung in der Form

$$u_h(x) = \sum_{i \in \bar{\omega}_h} u_i p_i(x) \in V_h$$

gesucht. Dabei ist  $u_h(x)$  eine stetige, aber keine stetig differenzierbare Funktion. Bei einer Knotenbasis gilt

$$u_h(x_i) = u_i$$

für alle Knoten  $x_i \in \bar{\omega}_h$ .

- Vermeidung zu spitzer und zu stumpfer Dreiecke, insbesondere von Dreiecken der Gestalt

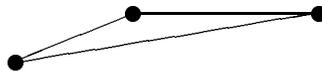


Abbildung 4.13: Dreieck mit einem sehr stumpfen Innenwinkel

- Generierung einer regulären Triangularisierung, d.h.

für alle  $r \in \mathbb{R}_h$  und für alle Diskretisierungsschrittweiten  $h$  existieren Konstanten  $\alpha_0$  und  $\theta_0$  mit  $\alpha_0 > 0$  und  $\theta_0 > 0$ , so daß

$$\alpha_0 h \leq h_1^{(r)}, h_2^{(r)}, h_3^{(r)} \leq h$$

$$0 < \theta_0 \leq \theta_1^{(r)}, \theta_2^{(r)}, \theta_3^{(r)} \leq \pi - \theta_0$$

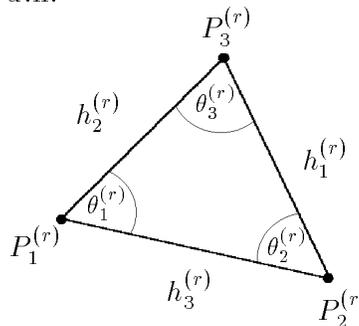
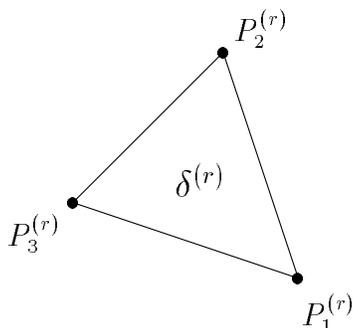


Abbildung 4.14: Darstellung eines Dreiecks einer regulären Vernetzung

Die Vernetzung des Gebietes  $\bar{\omega}$  wird bei der algorithmischen Realisierung durch die Festlegung einer *globalen* und einer *lokalen* Nummerierung einschließlich deren *Verknüpfung* beschrieben.

global: Numerierung aller Knoten und Elemente  
 $\bar{\omega}_h = \{1, 2, \dots, \bar{N}_h\}$ ,  $\bar{N}_h$  die Anzahl der Knoten  
 $\mathbb{R}_h = \{1, 2, \dots, \bar{R}_h\}$ ,  $\bar{R}_h$  die Anzahl der Elemente  
 Festlegung der Knotenkoordinaten  $x_i = (x_{1,i}, x_{2,i})$ ,  
 $i = 1, 2, \dots, \bar{N}_h$

lokal: Numerierung der Knoten in jedem Dreieck



Knotenkoordinaten der Knoten  $P_\alpha^{(r)}$ :

$$x_\alpha^{(r)} = (x_{1,\alpha}^{(r)}, x_{2,\alpha}^{(r)})$$

$$\alpha \in A = A^{(r)} = \{1, 2, 3\}$$

Abbildung 4.15: Lokale Knotennummerierung eines Dreiecks

Verknüpfung: Festlegung der Zuordnungsvorschrift zwischen lokaler und globaler Knotennummerierung für jedes Dreieck  $\delta^{(r)}$ ,  $r \in \mathbb{R}_h$

$$r : \alpha \leftrightarrow i = i(r, \alpha), \quad \alpha \in A^{(r)}, \quad i \in \bar{\omega}_h$$

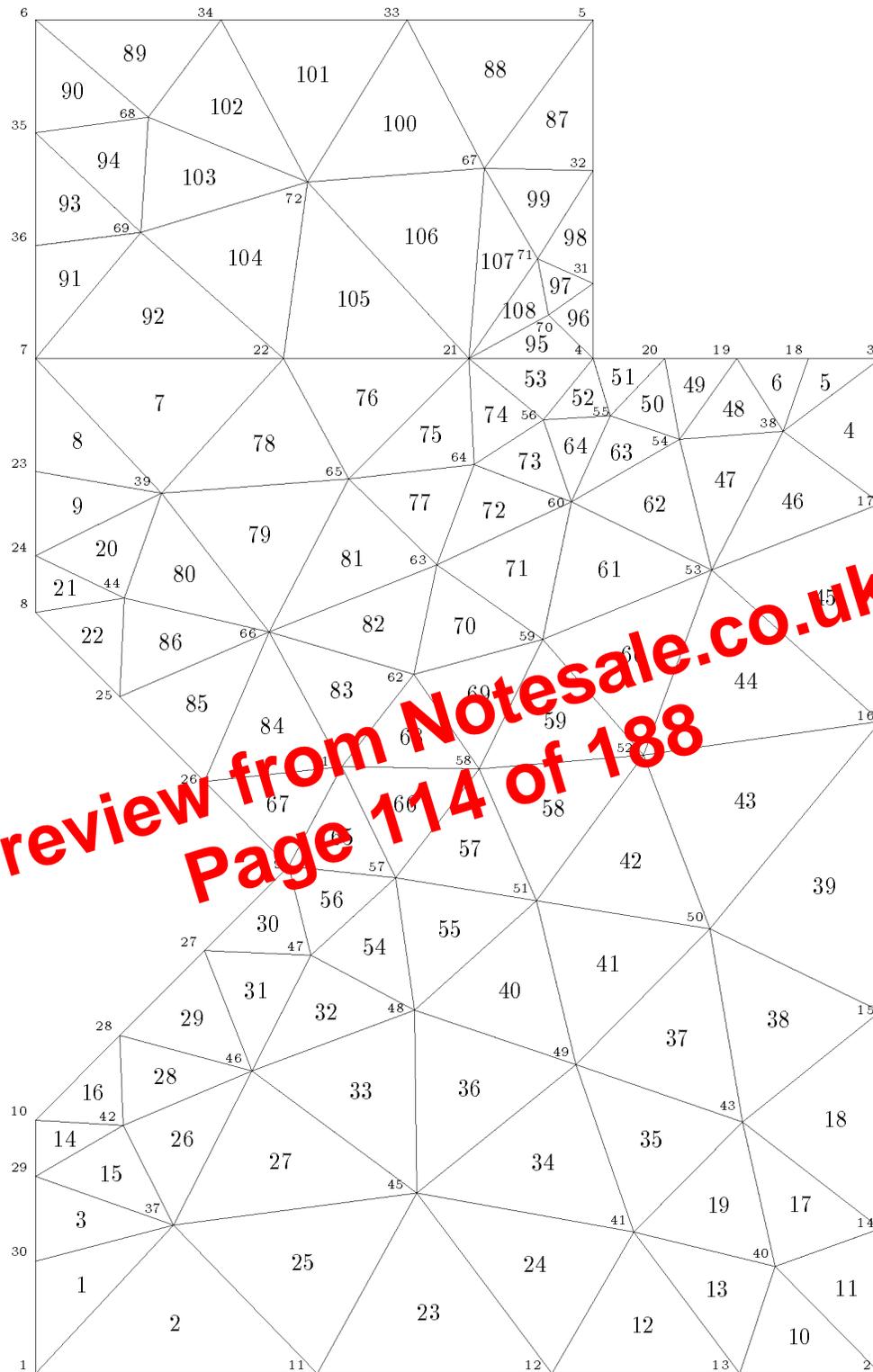


Abbildung 4.16: Beispiel einer Vernetzung mit Numerierung der Knoten und Elemente

Alle FE-Programme benötigen deshalb mindestens die folgenden zwei Felder, deren Aufbau wir am Beispiel der in der Abbildung 4.16 dargestellten Vernetzung des Gebietes (siehe das Beispiel im Abschnitt 4.1) erläutern.

- $r : \alpha \leftrightarrow i, r \in \mathbb{R}_h, \alpha \in A^{(r)} = A, i \in \bar{\omega}_h$

Elementnummer	globale Knotennummern der lokalen Knoten			Elementkennzahl z.B. Materialbereichsnummer
	$P_1^{(r)}$	$P_2^{(r)}$	$P_3^{(r)}$	
1	1	37	30	1
2	1	11	37	1
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
86	25	66	44	1
87	67	32	5	2
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$R_h = 108$	21	70	71	2

Tabelle 4.1: Zuordnungstabelle zwischen lokaler und globaler Knotennumerierung

- $i : (x_{1,i}, x_{2,i})$

$i$	1	2	3	$\dots$	$\bar{N}_h = 72$
$x_{1,i}$	0.5	1.0	1.0	$\dots$	0.66
$x_{2,i}$	0.0	0.0	0.6	$\dots$	0.7

Tabelle 4.2: Tabelle der Knotenkoordinaten

**Bemerkung 4.11** Weitere Felder zur Charakterisierung von Knoten sind möglich, z.B. zur Randbeschreibung und zur Randbedingungskodierung.

- *Methoden zur Generierung der Netzdaten*

1. Erzeugung der Felder für die Zuordnung zwischen der lokalen und der globalen Knotennumerierung ( $r : \alpha \leftrightarrow i$ ) und der Knotenkoordinaten ( $i : (x_{1,i}, x_{2,i})$ ) von „Hand“.
2. Halbautomatische Erzeugung der Daten, d.h. zum Beispiel Zerlegung des Gebietes in Teilgebiete. Für die Vernetzung der Teilgebiete werden in Programm Bibliotheken vorhandene Vernetzungen von Standardgebieten genutzt. Die Vernetzungen der Standardgebiete werden auf die konkreten Teilgebiete abgebildet. Hierbei muß beachtet werden, daß die gesamte Vernetzung des Gebietes eine zulässige Vernetzung ist.

### 4.4.3 Aufbau des FE-Gleichungssystems

Wir beschreiben im weiteren den elementweisen Aufbau des Finite-Elemente-Gleichungssystems, d.h. des Ritz-Galerkin-Systems (siehe auch Abschnitt 4.3):

Gesucht ist  $\underline{u}_h \in \mathbb{R}^{N_h}$ , so daß

$$K_h \underline{u}_h = \underline{f}_h \quad (4.14)$$

mit

$$K_h = [a(p_i, p_k)]_{i,k \in \omega_h}$$

$$\underline{f}_h = [\langle F, p_k \rangle - \sum_{i \in \omega_h} u_{*,i} a(p_i, p_k)]_{k \in \omega_h}$$

gilt, wobei die Bilinearform  $a(\cdot, \cdot)$  und die Linearform  $\langle F, \cdot \rangle$  wie folgt definiert sind :

$$a(p_i, p_k) = \int \left[ \lambda_1 \frac{\partial p_i}{\partial x_1} \frac{\partial p_k}{\partial x_1} + \lambda_2 \frac{\partial p_i}{\partial x_2} \frac{\partial p_k}{\partial x_2} \right] dx + \int_{\Gamma_3} \alpha(x) p_i(x) p_k(x) ds$$

$$\langle F, p_k \rangle = \int f(x) p_k(x) dx + \int_{\Gamma_2} g_2(x) p_k(x) ds + \int_{\Gamma_3} \alpha(x) g_3(x) p_k(x) ds.$$

Zuerst generieren wir eine Steifigkeitsmatrix und einen Lastvektor, in denen die Randterme noch nicht berücksichtigt sind.

#### Aufbau des FE-Gleichungssystems ohne Berücksichtigung der Randbedingungen

Beim Aufbau des FE-Gleichungssystems ohne Berücksichtigung der Randbedingungen werden die Matrix  $\bar{K}_h$

$$\bar{K}_h = [\bar{a}(p_i, p_k)]_{i,k \in \bar{\omega}_h} = \left[ \int \left[ \lambda_1 \frac{\partial p_i}{\partial x_1} \frac{\partial p_k}{\partial x_1} + \lambda_2 \frac{\partial p_i}{\partial x_2} \frac{\partial p_k}{\partial x_2} \right] dx \right]_{i,k \in \bar{\omega}_h}$$

und die rechte Seite  $\bar{f}_h$

$$\bar{f}_h = [\langle \bar{F}, p_k \rangle]_{k \in \bar{\omega}_h} = \left[ \int f(x) p_k(x) dx \right]_{k \in \bar{\omega}_h}$$

berechnet. Den Ausgangspunkt unserer weiteren Überlegungen bilden die Beziehungen

$$(\bar{K}_h \underline{u}_h, \underline{v}_h) = \bar{a}(u_h, v_h) \quad \text{für alle } u_h \leftrightarrow \underline{u}_h, v_h \leftrightarrow \underline{v}_h, u_h, v_h \in V_h, \underline{u}_h, \underline{v}_h \in \mathbb{R}^{N_h}$$

und

$$(\bar{f}_h, \underline{v}_h) = \langle \bar{F}, v_h \rangle \quad \text{für alle } v_h \leftrightarrow \underline{v}_h, v_h \in V_h, \underline{v}_h \in \mathbb{R}^{N_h}.$$



Werden auf analoge Weise alle weiteren Elementsteifigkeitsmatrizen eingebaut, dann entsteht die globale Steifigkeitsmatrix  $\bar{K}_h$  ohne Berücksichtigung der Randbedingungen.

Für die Assemblierung der rechten Seite  $\bar{f}_h$  nutzen wir den gleichen Algorithmus, d.h. zunächst setzen wir  $\bar{f}_h$  identisch Null. Nach dem Einbau von  $\underline{f}^{(1)}$  erhalten wir

$$\bar{f}_h = \begin{pmatrix} f_1^{(1)} & 0 & \cdots & 0 & f_3^{(1)} & 0 & \cdots & 0 & f_2^{(1)} & 0 & \cdots & 0 \end{pmatrix}^T$$

1    2    ... 29 30 31    ... 36 37 38    ... 72

und nach der Addition von  $\underline{f}^{(2)}$

$$\bar{f}_h = \begin{pmatrix} f_1^{(1)} + f_1^{(2)} & 0 & \cdots & 0 & f_2^{(2)} & 0 & \cdots & 0 & f_3^{(1)} & 0 & \cdots & 0 & f_2^{(1)} + f_3^{(2)} & 0 & \cdots & 0 \end{pmatrix}^T .$$

1        2    ... 10 11 12    ... 29 30 31    ... 36        37        38    ... 72

Die Fortsetzung dieses Assemblierungsprozesses für alle  $r = 3, 4, \dots, R_h$  liefert den globalen Lastvektor  $\bar{f}_h$ . Für das Element  $f_{72}$  gilt z.B.

$$\bar{f}_{72} = f_1^{(100)} + f_1^{(101)} + f_1^{(102)} + f_1^{(103)} + f_1^{(104)} + f_1^{(105)} + f_1^{(106)} .$$

**Bemerkung 4.14** Zur Darstellung der Matrix  $\bar{K}_h$  und der rechten Seite  $\bar{f}_h$  können auch die Connectivity-Matrizen  $C^{(r)}$  genutzt werden. Dabei ist  $C^{(r)}$  wie folgt definiert

$$C_{ij}^{(r)} = \begin{cases} 1 & i \text{ globale Knotennummer eines Eckknotens des Dreiecks } \delta^{(r)} \\ & \text{und } j \text{ die zugehörige lokale Knotennummer} \\ 0 & \text{sonst,} \end{cases}$$

$i = 1, 2, \dots, \bar{N}_h$ ,  $j = 1, 2, 3$ . Damit gilt

$$\bar{K}_h = \sum_{r \in \mathbb{R}_h} C^{(r)} K^{(r)} (C^{(r)})^T \quad \text{und} \quad \bar{f}_h = \sum_{r \in \mathbb{R}_h} C^{(r)} \underline{f}^{(r)} .$$

Die Matrix  $C^{(1)}$  hat beispielsweise die Gestalt

$$(C^{(1)})^T = \begin{pmatrix} 1 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 1 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \end{pmatrix}$$

1 2    ... 29 30 31    ... 36 37 38    ... 72

Die Matrizen  $K^{(e_3)}$  sowie die Vektoren  $\underline{f}^{(e_3)}$  und  $\underline{f}^{(e_2)}$ ,  $e_3 = 1, 2, \dots, N_h^{(3)}$ ,  $e_2 = 1, 2, \dots, N_h^{(2)}$ , werden genauso wie die Elementsteifigkeitsmatrizen  $K^{(r)}$  und die Elementlastvektoren  $\underline{f}^{(r)}$  assembliert.

Abschließend sind noch die Randbedingungen 1. Art in das FE-Gleichungssystem einzuarbeiten.

- *Berücksichtigung der Randbedingungen 1. Art im FE-Gleichungssystem*

1. Variante : Homogenisieren im Diskreten

Wir setzen  $u_j = g_1(x_j)$  für alle  $j \in \bar{\omega}_h \setminus \omega_h$ , d.h. für alle Indizes  $j$ , die zu Knoten auf dem Rand  $\Gamma_1$  gehören, und korrigieren die rechte Seite durch

$$f_i := f_i - \sum_{j \in \bar{\omega}_h} K_{ij} u_j \quad \text{für alle } i \in \omega_h,$$

wobei  $K_{ij}$  die Matrixelemente der globalen Steifigkeitsmatrix  $K_h$  nach Einbau der Elementsteifigkeitsmatrizen  $K^{(r)}$  und der Matrizen  $K^{(e_3)}$  sowie  $f_i$  die Komponenten des globalen Lastvektors nach Einbau der Elementlastvektoren  $\underline{f}^{(r)}$  und der Vektoren  $\underline{f}^{(e_2)}$  bzw.  $\underline{f}^{(e_3)}$  sind. Anschließend streichen wir die Spalten  $j$ ,  $j \in \bar{\omega}_h$ , und die Zeilen  $i$ ,  $i \in \bar{\omega}_h$ , aus der Steifigkeitsmatrix sowie die Zeilen  $i$ ,  $i \in \bar{\omega}_h$ , aus dem Lastvektor.

2. Variante:

Wir gehen analog wie in der Variante 1 vor, und anstelle des Streichens der Spalten  $j$  und der Zeilen  $i$  mit  $i, j \in \bar{\omega}_h$  setzen wir

$$K_{ij} = \begin{cases} 1 & \text{für } i = j \\ 0 & \text{für } i \neq j \end{cases} \quad \text{sowie} \quad f_i = g_1(x_i)$$

für alle  $i, j \in \bar{\omega}_h$ .

3. Variante: Straftechnik (siehe auch [44])

Für alle  $i \in \bar{\omega}_h$  setzen wir

$$K_{ii} := K_{ii} + \hat{K}, \quad \hat{K} = 10^p \cdot \max_{j=1,2,\dots,\bar{N}_h} |K_{ij}| \cdot \bar{N}_h, \quad p \text{ hinreichend groß}$$

$$f_i := f_i + \hat{K} g_1(x_i).$$

## Zusammenfassung zum Aufbau des FE-Gleichungssystems

1. Triangularisierung des Gebietes  $\bar{\omega}_h$ , d.h.

$$\bar{\omega}_h = \bigcup_{r \in \mathbb{R}_h} \bar{\delta}^{(r)} \quad \text{bzw.} \quad \bar{\omega}_h \approx \bar{\omega}_h = \bigcup_{r \in \mathbb{R}_h} \bar{\delta}^{(r)}$$

#### 4.4.5 Ein Beispiel

Wir betrachten folgendes Randwertproblem:

Gesucht ist das Temperaturfeld  $u(x) \in C^2(\Omega) \cap C^1(\partial\Omega) \cap C(\bar{\Omega})$ , so daß

$$\begin{aligned}
 -\Delta u &= 0 & x \in \Omega &= (0,1) \times (0,1) \\
 u(x_1, x_2) &= x_2 & x \in \Gamma_{11} &= \{x \in \partial\Omega : x_1 = 1, 0 < x_2 \leq 1\} \\
 u(x_1, x_2) &= 0 & x \in \Gamma_{12} &= \{x \in \partial\Omega : 0 \leq x_1 \leq 1, x_2 = 0\} \\
 \frac{\partial u}{\partial n} &= -x_2 & x \in \Gamma_{21} &= \{x \in \partial\Omega : x_1 = 0, 0 < x_2 < 1\} \\
 \frac{\partial u}{\partial n} &= x_1 & x \in \Gamma_{22} &= \{x \in \partial\Omega : 0 < x_1 < 1, x_2 = 1\}
 \end{aligned} \tag{4.16}$$

gilt.

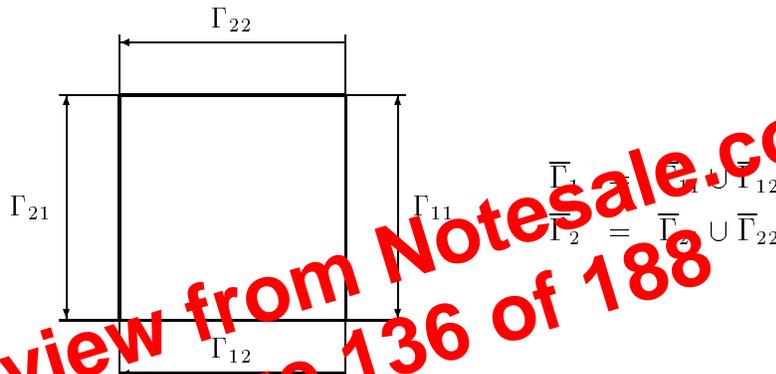


Abbildung 4.26: Darstellung des Gebietes

Die exakte Lösung dieser Aufgabe ist  $u(x_1, x_2) = x_1 x_2$ , denn

$$\frac{\partial u}{\partial x_1} = x_2, \quad \frac{\partial^2 u}{\partial x_1^2} = 0,$$

$$\frac{\partial u}{\partial x_2} = x_1, \quad \frac{\partial^2 u}{\partial x_2^2} = 0,$$

$$\frac{\partial u}{\partial n} = \frac{\partial u}{\partial x_1} n_1 + \frac{\partial u}{\partial x_2} n_2 = -\frac{\partial u}{\partial x_1} = -x_2 \quad \text{auf } \Gamma_{21},$$

$$\frac{\partial u}{\partial n} = \frac{\partial u}{\partial x_1} n_1 + \frac{\partial u}{\partial x_2} n_2 = \frac{\partial u}{\partial x_2} = x_1 \quad \text{auf } \Gamma_{22}.$$

- Bei Vorhandensein von Randbedingungen 2. Art
  - Wieviele Randkanten mit Randbedingungen 2. Art ?
  - Für jede Randkante Eingabe von Anfangsknoten, Endknoten und  $g_2$  aus der Randbedingung

$$k_{xx} \frac{\partial u}{\partial x} n_x + k_{yy} \frac{\partial u}{\partial y} n_y = g_2$$

- Bei Vorhandensein von Randbedingungen 3. Art
  - Wieviele Randkanten mit Randbedingungen 3. Art ?
  - Für jede Randkante Eingabe von Anfangsknoten, Endknoten,  $k_0$  und  $g_3$  aus der Randbedingung

$$k_{xx} \frac{\partial u}{\partial x} n_x + k_{yy} \frac{\partial u}{\partial y} n_y + k_0 u = k_0 g_3$$

- Ausgabe der Ergebnisse über Drucker (J N) ?
- Speichern der Ergebnisse auf Diskette (J N) ?  
Falls (J), Eingabe eines Filenamens
- Ausgabe der Eingangsdaten (J N) ?
- Falls (J), Drucken der Eingangsdaten (J N) ?

**Bemerkung 4.16** Bei der Anforderung der Eingabe der Randbedingungen 1., 2. bzw. 3. Art wird nach einer Option für die Eingabe gefragt. Es stehen zur Verfügung:

B – Blockeingabe oder E – Einzeleingabe.

Sind die Randbedingungsdaten über mehrere Randkanten gleich, dann ist die Eingabeform „B“ empfehlenswert, da nur der Anfangsknoten der ersten Randkante und der Endknoten der letzten Randkante sowie die entsprechenden Daten eingegeben werden müssen.

Preview from Notesale.co.uk  
Page 146 of 188

eine Matrix \* Vektor-Multiplikation, drei Vektoradditionen und zwei Skalarprodukte zu berechnen sind, ist der Aufwand an arithmetischen Operationen pro Iterationsschritt proportional zur Anzahl der Unbekannten  $N$ . Damit verhält sich der Gesamtaufwand bei 2D-Problemen wie  $h^{-3} \ln \varepsilon^{-1} (N^{1.5} \ln \varepsilon^{-1})$ .

Eine schnellere Konvergenz würde man erhalten, wenn die Konditionszahl  $\kappa(K)$  schwächer von  $h$  abhängen würde. Zur Verbesserung der Kondition betrachten wir im nächsten Abschnitt die Methode der konjugierten Gradienten mit Vorkonditionierung.

### 5.2.3 Die Methode der konjugierten Gradienten mit Vorkonditionierung

Wir betrachten anstelle des zu lösenden Gleichungssystems  $K\underline{u} = \underline{f}$  das äquivalente Gleichungssystem

$$\tilde{K}\tilde{\underline{u}} = \tilde{\underline{f}} \quad (5.11)$$

mit  $\tilde{K} = L^{-1}KL^{-T}$ ,  $\tilde{\underline{u}} = L^T\underline{u}$  und  $\tilde{\underline{f}} = L^{-1}\underline{f}$ , wobei  $C = LL^T$  eine symmetrische, positiv definite Matrix ist. Dabei wird die *Vorkonditionierungsmatrix*  $C$  so gewählt, daß  $\kappa(\tilde{K}) \ll \kappa(K)$  gilt. Das Gleichungssystem (5.11) lösen wir mittels des im Abschnitt 5.2.2 beschriebenen CG-Verfahrens. Wir formulieren die Teilschritte aber so um, daß mit der Matrix  $K$  und der rechten Seite  $\underline{f}$  gerechnet werden kann. Damit ergibt sich der folgende Algorithmus der Methode der konjugierten Gradienten mit Vorkonditionierung.

*PCCG-Verfahren (preconditioned conjugate gradient method)*

Start: Wahl von  $\underline{u}^{(0)}$

$$\underline{r}^{(0)} = K\underline{u}^{(0)} - \underline{f}$$

$$C\underline{w}^{(0)} = \underline{r}^{(0)}$$

$$\underline{d}^{(1)} = -\underline{w}^{(0)}$$

Iteration: ( $k = 1, 2, \dots$ )

$$\left. \begin{aligned} \tilde{\alpha}^{(k-1)} &= (\underline{w}^{(k-1)}, \underline{r}^{(k-1)}) / (\underline{w}^{(k-2)}, \underline{r}^{(k-2)}) \\ \underline{d}^{(k)} &= -\underline{w}^{(k-1)} + \tilde{\alpha}^{(k-1)} \underline{d}^{(k-1)} \end{aligned} \right\} k \geq 2$$

$$\tilde{\alpha}^{(k)} = (\underline{w}^{(k-1)}, \underline{r}^{(k-1)}) / (K\underline{d}^{(k)}, \underline{d}^{(k)})$$

$$\underline{u}^{(k)} = \underline{u}^{(k-1)} + \tilde{\alpha}^{(k)} \underline{d}^{(k)}$$

$$\underline{r}^{(k)} = \underline{r}^{(k-1)} + \tilde{\alpha}^{(k)} K\underline{d}^{(k)}$$

$$C\underline{w}^{(k)} = \underline{r}^{(k)}$$

$$(\underline{w}^{(k)}, \underline{r}^{(k)}) \leq \varepsilon^2 (\underline{w}^{(0)}, \underline{r}^{(0)}) \rightarrow \text{STOP}$$

( $\varepsilon$  – vorgegebene Abbruchschranke)

Wie können wir die Startnäherung für die Iterationsverfahren wählen?

Wenn wir eine Folge von FE-Diskretisierungen erzeugt haben, dann bietet sich der folgende Algorithmus zur Bestimmung einer Startnäherung an.

1. Löse das Gleichungssystem

$$K_1 \underline{u}_1 = \underline{f}_1$$

mittels eines direkten oder iterativen Verfahrens. Setze  $q = 1$ ,  $k_q = 1$  und  $\underline{u}_q^{(k_q)} = \underline{u}_1$ .

2. Interpoliere die Lösung  $\underline{u}_q^{(k_q)}$  auf das Netz  $\mathcal{T}_{q+1}$ , d.h.

$$\underline{u}_{q+1}^{(0)} = I_q^{q+1} \underline{u}_q^{(k_q)}.$$

Setze  $q = q + 1$  und löse das Gleichungssystem

$$K_q \underline{u}_q = \underline{f}_q$$

mittels  $k_q$  Schritten eines Iterationsverfahrens. Starte dieses mit  $\underline{u}_q^{(0)}$ .

Falls  $q < l$ , gehe zu Schritt 2, sonst STOP.

Setzen wir im Schritt 2 zur näherungsweise Lösung des Gleichungssystems ein Mehrgitter- oder MG(m)-PCCG-Verfahren ein, dann kann man zeigen, daß wir mit einem zur Anzahl der Unbekannten  $\bar{N}_l$  proportionalen Aufwand an arithmetischen Operationen eine Näherungslösung  $\underline{u}_l^{(k_l)}$  erhalten, für die der Fehler  $\|\underline{u}_l - \underline{u}_l^{(k_l)}\|_{K_l}$  in der Größenordnung des Diskretisierungsfehlers liegt ([3, 78]).

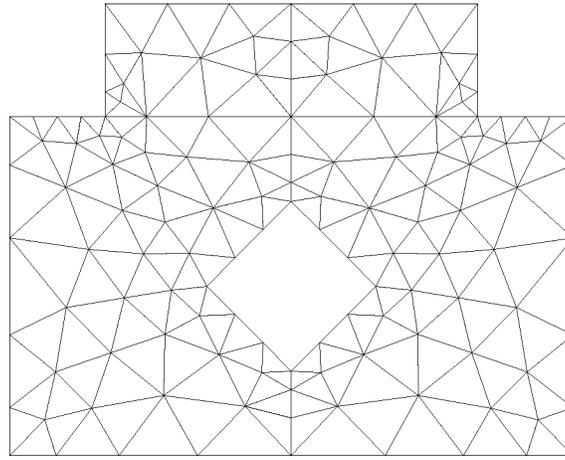
### 5.3 Ein Vergleich der Auflösungsverfahren

Wir wollen das Cholesky-Verfahren und die vorgestellten Iterationsverfahren hinsichtlich des Aufwandes an notwendigen arithmetischen Operationen sowie ihres Speicherplatzbedarfs vergleichen. Dabei werden wir demonstrieren, wie sich für die jeweiligen Verfahren der Aufwand an arithmetischen Operationen und der Speicherplatzbedarf bei wachsender Dimension des FE-Gleichungssystems verhält. Den Vergleich führen wir anhand des Modellbeispiels 1 aus dem Abschnitt 4.1 durch. Als Maß für den Aufwand an arithmetischen Operationen nutzen wir die benötigte CPU-Zeit. Alle Rechnungen wurden mit dem Programmsystem FEMGP ([42, 66, 76]) durchgeführt. Dieses Programmsystem kann zur Lösung von linearen und nichtlinearen Randwertproblemen in 2D-Gebieten eingesetzt werden. Die Tests wurden auf einem PC 80486 (33 MHz) unter Nutzung des LAHEY-FORTAN-Compilers ausgeführt.

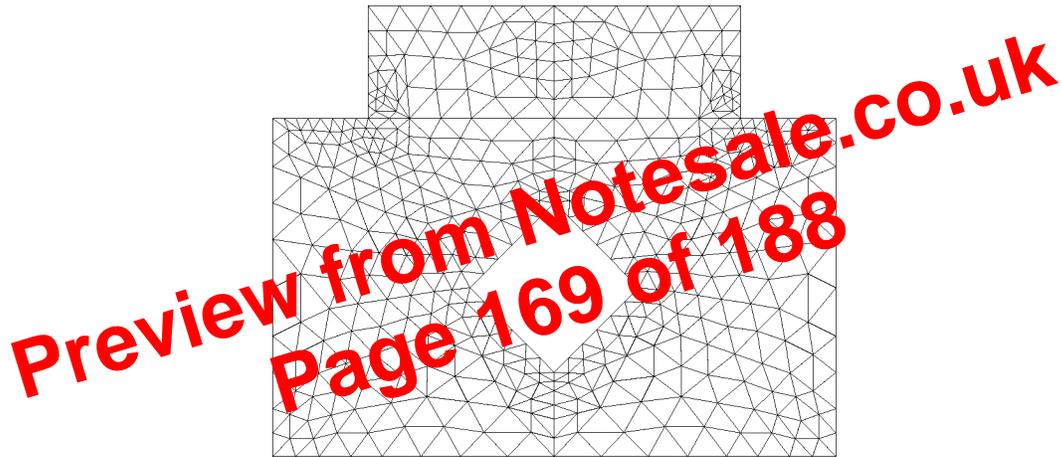
Wir erzeugen eine Folge von vier Vernetzungen  $\mathcal{T}_q$ ,  $q = 1, 2, 3, 4$ , mit den Diskretisierungsparametern  $h_q$ ,  $h_{q+1} = h_q/2$ ,  $q = 1, 2, 3$ . Zu jeder Vernetzung wird das entsprechende FE-Gleichungssystem

$$K_q \underline{u}_q = \underline{f}_q$$

generiert und gelöst. Als größte Vernetzung wählen wir die in der Abbildung 5.7 dargestellte Triangulation.

Abbildung 5.7: Darstellung der Vernetzung  $\mathcal{T}_1$ 

Die feineren Vernetzungen erhalten wir, indem wir jeweils alle Dreiecke vierteln, d.h. wir halbieren alle Dreiecksseiten. Als zweite Vernetzung entsteht somit das folgende Netz.

Abbildung 5.8: Darstellung der Vernetzung  $\mathcal{T}_2$ 

Zuerst lösen wir die FE-Gleichungssysteme mittels des Cholesky-Verfahrens. Die Steifigkeitsmatrizen werden dafür in der Speicherform „variable Bandweite spaltenweise“ abgespeichert. In der Tabelle 5.1 sind der Speicherplatzbedarf und die benötigten CPU-Zeiten für die Cholesky-Zerlegung sowie für das Vorwärts- und Rückwärts einsetzen zusammengestellt.

Anzahl der Unbekannten	133	482	1828	7112
Speicherplatzbedarf	13.66 kB	92.00 kB	661.61 kB	5039.52 kB
CPU-Zeit für die Zerlegung	0.01 s	0.33 s	4.18 s	60.31 s
CPU-Zeit für das Vorwärts- und Rückwärtseinsetzen	0.01 s	0.05 s	0.38 s	2.97 s

Tabelle 5.1: Rechenzeit und Speicherplatzbedarf beim Cholesky-Verfahren

Aus dieser Tabelle können wir ablesen, daß bei jeweiliger Halbierung des Diskretisierungsparameters

- sich die Anzahl der Unbekannten etwa vervierfacht, d.h. wie  $h^{-2}$  verhält,
- der Speicherplatzbedarf sich wie  $h^{-3}$  ( $N^{1.5}$ ) verhält, d.h. mit dem Faktor 8 wächst,
- die benötigte CPU-Zeit für die Zerlegung ungefähr wie  $h^{-4}$  ( $N^2$ ) wächst, d.h. um das 16fache zunimmt,
- die benötigte CPU-Zeit für das Vorwärts- und Rückwärtseinsetzen wie  $h^{-3}$  ( $N^{1.5}$ ) ansteigt, d.h. sich verachtfacht.

Somit bestätigen die erzielten Resultate die im Abschnitt 5.1 getroffenen theoretischen Aussagen hinsichtlich Speicherplatzbedarf und arithmetischem Aufwand des Cholesky-Verfahrens.

Im weiteren untersuchen wir verschiedene Iterationsverfahren. Bei allen Iterationsverfahren wurde als Startvektor der Vektor  $\underline{u}_4^{(0)} = [500, 500, \dots, 500]^T$  verwendet. Die Iteration wurde beendet, wenn das Abbruchkriterium  $\|\underline{r}^{(k)}\| \leq \varepsilon \|\underline{r}^{(0)}\|$  mit dem Defekt  $\underline{r}^{(k)} = K_4 \underline{u}_4^{(k)} - \underline{f}_4$  für  $\varepsilon = 10^{-4}$  erfüllt war.

Da bei allen untersuchten Iterationsverfahren die Steifigkeitsmatrizen in der Speicherform „Kompaktliste zeilenweise“ abgespeichert werden können, verhält sich der Speicherplatzbedarf wie  $h^{-2}$ , d.h. der Speicherplatzbedarf ist proportional zur Anzahl der Unbekannten des zu lösenden Gleichungssystems.

Wir betrachten zunächst zwei klassische Iterationsverfahren, das Jacobi-Verfahren und das Gauß-Seidel-Verfahren. Beim Jacobi-Verfahren wurden folgende Ressourcen an Speicherplatz und CPU-Zeit benötigt:

Anzahl der Unbekannten	133	482
Speicherplatzbedarf	7.08 kB	26.87 kB
Anzahl der durchgeführten Iterationen	58226	226747
CPU-Zeit für die durchgeführten Iterationen	250.90 s	3841.70 s

Tabelle 5.2: Rechenzeit und Speicherplatzbedarf beim Jacobi-Verfahren

Die Anwendung des Gauß-Seidel-Verfahrens erforderte die in der Tabelle 5.3 angegebenen Rechenzeiten und Speicherplätze.

Anzahl der Unbekannten	133	482
Speicherplatzbedarf	7.08 kB	26.87 kB
Anzahl der durchgeführten Iterationen	29269	114319
CPU-Zeit für die durchgeführten Iterationen	133.08 s	1990.88 s

Tabelle 5.3: Rechenzeit und Speicherplatzbedarf beim Gauß-Seidel-Verfahren

Bei beiden Verfahren wächst die Anzahl an notwendigen Iterationen bei einem vorgegebenen  $\varepsilon$  wie  $h^{-2} \ln \varepsilon^{-1}$  ( $N \ln \varepsilon^{-1}$ ), d.h. sie vervierfacht sich. Da der Aufwand an arithmetischen Operationen pro Iterationsschritt sich wie  $h^{-2}$  ( $N$ ) verhält, wächst somit der Gesamtaufwand an arithmetischen Operationen wie  $h^{-4} \ln \varepsilon^{-1}$  ( $N^2 \ln \varepsilon^{-1}$ ), d.h. mit

dem Faktor 16. Die Tabellen 5.2 und 5.3 zeigen, daß tatsächlich die benötigte CPU-Zeit um das 16fache anwächst. Offenbar konvergieren sowohl das Jacobi-Verfahren als auch das Gauß-Seidel-Verfahren nur sehr langsam. Damit sind diese Verfahren als eigenständige Algorithmen zur Lösung der FE-Gleichungssysteme unbrauchbar. Hätten wir nämlich beispielsweise das FE-Gleichungssystem mit 7112 Unbekannten mittels des Gauß-Seidel-Verfahrens gelöst, dann wäre eine CPU-Zeit von etwa 124 Stunden, also von rund 5 Tagen, notwendig gewesen.

Als nächsten Algorithmus untersuchen wir die Methode der konjugierten Gradienten ohne Vorkonditionierung (CG-Verfahren). Theoretisch müßte sich die Iterationszahl bei einem vorgegebenem  $\varepsilon$  wie  $h^{-1} \ln \varepsilon^{-1}$  verhalten, d.h. sich jeweils verdoppeln. In der Tabelle 5.4 ist ersichtlich, daß sich zunächst die Iterationszahl um das 3.5fache, dann um das 3.1fache und schließlich um das 2.8fache erhöht. Man kann erwarten, daß sich bei weiteren Netzverfeinerungen tatsächlich das theoretisch ermittelte Wachstum der Iterationszahl einstellt. Auch beim CG-Verfahren ist der Aufwand an arithmetischen Operationen pro Iterationsschritt proportional zur Anzahl der Unbekannten, d.h. er verhält sich wie  $h^{-2}$  ( $N$ ). Asymptotisch nimmt also der Gesamtaufwand an arithmetischen Operationen wie  $h^{-3} \ln \varepsilon^{-1}$  ( $N^{1.5} \ln \varepsilon^{-1}$ ) zu. Zusätzlich zu den Speicherplätzen für das FE-Gleichungssystem werden beim CG-Verfahren noch drei Hilfsvektoren der Länge  $N$  benötigt.

Anzahl der Unbekannten	133	482	1828	7112
Speicherplatzbedarf	9.16 kB	34.40 kB	132.95 kB	522.44 kB
Anzahl der durchgeführten Iterationen	125	432	1377	3871
CPU-Zeit für die durchgeführten Iterationen	0.55 s	5.98 s	75.75 s	1016.23 s

Tabelle 5.4: Rechenzeit und Speicherplatzbedarf beim CG-Verfahren

Eine Reduzierung der notwendigen Anzahl an Iterationen und damit auch ein Rechenzeitgewinn wird bereits durch die einfache Vorkonditionierung mit der Hauptdiagonale der Systemmatrix erreicht (siehe Tabelle 5.5).

Anzahl der Unbekannten	133	482	1828	7112
Speicherplatzbedarf	9.16 kB	34.40 kB	132.95 kB	522.44 kB
Anzahl der durchgeführten Iterationen	34	71	146	291
CPU-Zeit für die durchgeführten Iterationen	0.22 s	1.15 s	8.95 s	84.64 s

Tabelle 5.5: Rechenzeit und Speicherplatzbedarf beim DIAG-PCCG-Verfahren

Aus der Tabelle 5.5 wird deutlich, daß die Iterationszahl wie  $h^{-1} \ln \varepsilon^{-1}$  wächst. Daraus resultiert ein Anwachsen der Anzahl an notwendigen arithmetischen Operationen wie  $h^{-3} \ln \varepsilon^{-1}$ , was sich in der Zunahme der CPU-Zeit um das 8fache widerspiegelt.

Die Tabelle 5.11 zeigt, daß das Cholesky–Verfahren für Gleichungssysteme mit wenigen Unbekannten bezüglich der benötigten CPU–Zeit am effektivsten ist. Zur Lösung von großdimensionierten Gleichungssystemen sollten nur Mehrgitterverfahren oder PCCG–Verfahren eingesetzt werden. Dabei sind die Mehrgitterverfahren und die MG(m)–PCCG–Verfahren die effektivsten Verfahren, da bei ihnen die Iterationszahl unabhängig vom Diskretisierungsparameter ist.

In der Abbildung 5.9 ist der Niveaulinienverlauf des berechneten Temperaturfeldes dargestellt.

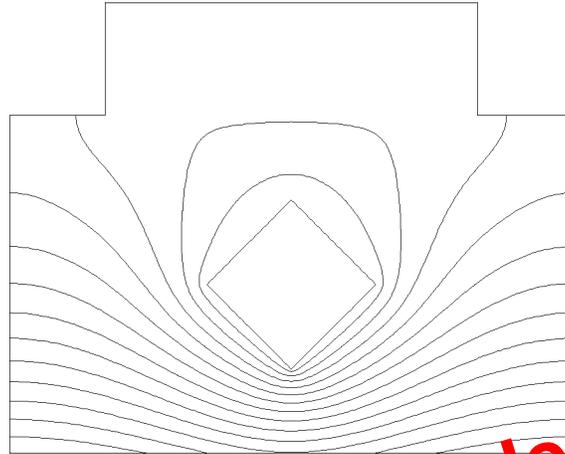


Abbildung 5.9: Darstellung des Niveaulinienverlaufs des Temperaturfeldes

**Preview from Notesale.co.uk**  
**Page 175 of 188**

# Literaturverzeichnis

- [1] R. A. Adams. *Sobolev Spaces*. Academic Press, New York, 1975.
- [2] Th. Apel. *Finite-Elemente-Methoden über lokal verfeinerten Netzen für elliptische Probleme in Gebieten mit Kanten*. Dissertation A, Technische Universität Chemnitz, 1991.
- [3] J. H. Argyris. Energy theorems and structural analysis. *Aircraft Engineering*, 27:125–154, 1955.
- [4] O. Axelsson und V. A. Barker. *Finite Element Solution of Boundary Value Problems: Theory and Computation*. Academic Press, Orlando Fla., 1984.
- [5] I. Babuška und W. C. Rheinboldt. Error estimates for adaptive finite element computations. *SIAM J. Num. Anal.*, 15(4):736–754, 1978.
- [6] P. K. Banerjee und R. Butterfield. *Boundary Element Methods in Engineering Science*. McGraw-Hill Book Company, 1981.
- [7] R. E. Bank und A. Weiser. Some a posteriori error estimators for elliptic partial differential equations. *Mathematics of Computation*, 44(170):283–301, 1985.
- [8] A. Bernhardt. Untersuchung des Abkühlprozesses einer abgesetzten Welle unter besonderer Beachtung des Verhaltens in den Ecken. Jahresarbeit, Technische Universität Karl-Marx-Stadt, Sektion Mathematik, 1988.
- [9] D. Braess. *Finite Elemente – Theorie, schnelle Löser und Anwendungen in der Elastizitätstheorie*. Springer Lehrbuch, 1992.
- [10] J. H. Bramble, J. E. Pasciak und J. Xu. Parallel multilevel preconditioners. *Mathematics of Computation*, 55(191):1–22, 1990.
- [11] U. Breitschuh und R. Jurisch. *Die Finite-Element-Methoden*. Akademie Verlag, Berlin, 1993.
- [12] S. Brenner und L. R. Scott. *The Mathematical Theory of Finite Element Methods*. Springer Verlag, New York, 1994.
- [13] P. Ciarlet. *The finite element method for elliptic problems*. North-Holland, Amsterdam, 1978.
- [14] R. Courant. Variational methods for the solution of problems of equilibrium and vibrations. *Bulletin of American Mathematical Society*, 49:1–23, January 1943.

- [15] A. S. Cybenko, N. G. Vaščenko, N. G. Kriščuk und Ju. O. Lavendel. *Avtomatizirovannaja sistema obsluživanja konečno–elementnych rasčetov*. Golovnoe izdatel'stvo izdatel'skogo obědinenija "višča škola", Kiev, 1986.
- [16] Dankert. *Numerische Methoden der Mechanik*. Fachbuchverlag, Leipzig, 1977.
- [17] N. Elsner. *Grundlagen der technischen Thermodynamik*. Akademie Verlag, Berlin, 1985.
- [18] G. M. Fichtenholz. *Differential– und Integralrechnung*, Bd. 3. VEB Deutscher Verlag der Wissenschaften, Berlin, 1977.
- [19] G. M. Fichtenholz. *Differential– und Integralrechnung*, Bd. 2. VEB Deutscher Verlag der Wissenschaften, Berlin, 1978.
- [20] U. Fischer. *Finite–Elemente–Programme in der Festkörpermechanik*. Fachbuchverlag, Leipzig, 1987.
- [21] K. O. Friedrichs. Finite–difference schemes for the Neumann and Dirichlet problems. Technical report, N. Y. Univ., 1962.
- [22] R. H. Gallagher. *Finite Element Analysis: Fundamentals*. Prentice–Hall, Englewood Cliffs, New Jersey, 1975.
- [23] A. George und J. W. Lui. *Computer Solution of Large Sparse Positive Definite Systems*. Prentice Hall, Englewood Cliffs, New Jersey, 1981.
- [24] V. Girault und P. A. Raviart. *Finite Element approximation of the Navier–Stokes equations, Lecture Notes in Mathematics 749*. Springer–Verlag, Berlin, 1979.
- [25] H. Göldner. *Lehrbuch Höhere Festigkeitslehre. Band 1*. Fachbuchverlag, Leipzig, 1991.
- [26] H. Göldner. *Lehrbuch Höhere Festigkeitslehre. Band 2*. Fachbuchverlag, Leipzig–Köln, 1992.
- [27] H. Göring, H. G. Roos und L. Tobiska. *Finite–Element–Methode, Wissenschaftliche Taschenbücher, Bd. 285*. Akademie–Verlag, Berlin, 1985.
- [28] M. Griebel. *Multilevelmethoden als Iterationsverfahren über Erzeugendensystemen*. Teubner Skripten zur Numerik. B. G. Teubner Stuttgart, 1994.
- [29] D. F. Griffiths, editor. *The Mathematical Basis of Finite Element Methods*. The Clarendon Press, Oxford University Press, New York, 1984.
- [30] Ch. Großmann und H. G. Roos. *Numerik partieller Differentialgleichungen*. Teubner Studienbücher Mathematik. Teubner–Verlag, Stuttgart, 1992.
- [31] I. Gustafsson. A class of first order factorization methods. *BIT*, 18:142–156, 1978.
- [32] I. Gustafsson. On modified incomplete factorization methods. *Lecture Notes in Mathematics* 968, 334–351. Springer–Verlag, Berlin, 1982.
- [33] W. Hackbusch. *Multi–Grid Methods and Applications, Springer Series in Computational Mathematics 4*. Springer–Verlag, Berlin, 1985.