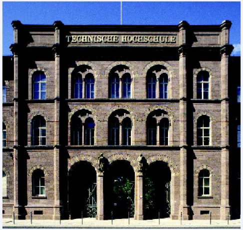
Übung zum Fach Rechnerunterstütze Mechanik I

J. Ruck, H. Erdle, T.-A. Langhoff, T. Böhlke

Chair for Continuum Mechanics
Institute of Engineering Mechanics
Department of Mechanical Engineering
Karlsruhe Institute of Technology (KIT)

WS 2017/2018



Karlsruher Institut für Technologie

Ü9: Numerische Integration (1d)

Themen der 9. Übung

- Numerische Integration (eindimensional)
- Lagrange'sche Interpolationspolynome
- Newton-Cotes-Quadratur
- Ergebnisverifikation
- Berechnung der Legendre-Polynome
- Gauss-Christoffel-Quadratur (**Testat**)

Wichtige Vorarbeit für die Finite Elemente Methode

Ausgabe der 2. Programmieraufgabe mit Abgabe
Abgabetermin: 11.01.2018

Lagrange-Polynome

Gegeben: $N+1$ Stützstellen $\tau_i \in [a,b]$ $(i=0,\dots,N)$
Definition: Die $N+1$ Polynome p_i $(i=0,\dots,N)$ vom Grad N mit $p_i(\tau_j)=\delta_{ij}$ heißen Lagrange-Polynome

Motivation

- Gesucht ist ein Interpolationspolynom $P(t)$ mit $P(\tau_i)=\hat{f}_i=f(\tau_i)$
- Ansatz:

$$P(t)=\sum_{i=0}^{N}\alpha_i p_i(t)$$

- Dann gilt:

$$P(\tau_i)=\sum_{j=0}^{N}\alpha_j p_j(\tau_i)=\sum_{j=0}^{N}\alpha_j\delta_{ij}=\alpha_i \overset{!}{=} \hat{f}_i$$

Interpolationspolynom kann effizient berechnet werden

Konstruktion

Da $p_i(\tau_j) = 0$ für $i \neq j$ gilt

$$p_i(t) = c_i \prod_{j \neq i} (t - \tau_j)\,.$$

Damit gilt:

$$p_i(\tau_i) = c_i \prod_{j \neq i} (\tau_i - \tau_j) \overset{!}{=} 1$$

$$\Rightarrow c_i = \frac{1}{\prod\limits_{j \neq i} (\tau_i - \tau_j)}$$

Folgerungen:

- Die Polynome p_i existieren, falls $\tau_i \neq \tau_j$ $(i \neq j)$
- Falls $|\tau_i - \tau_j| \ll 1$ gilt: $|c_i| \gg 1$
  $\rightarrow$ Achtung: Rundungsfehler!

Quadraturformeln

Gegeben: Funktion $f(x)$ auf einem Intervall $[a,b]$

Gesucht: Approximation $\hat{I}(f)$ von $I(f) = \int_a^b f(x)\mathrm{d}x$

Definition

Eine Quadraturformel $\hat{I}$ zur Approximation von I ist die gewichtete Summe

$$\hat{I}(f) = (b-a)\sum_{i=0}^{N} \lambda_i f(\tau_i) \qquad (\tau_i \in [a,b]\ \forall i = 0,\dots,N)$$

an den Knoten τ_i und mit den Gewichten λ_i.

Anforderungen an die Gewichte

- Für konstante Funktionen $f = c$ gilt $I(f) = (b-a)c$ und

$$\hat{I}(f) = c(b-a)\left(\sum_{i=0}^{N} \lambda_i\right) \overset{!}{=} c(b-a) \quad \rightarrow \sum_{i=0}^{N} \lambda_i = 1$$

- Sei $f(x) \geq 0 \rightarrow I(f) \geq 0$. Dann gilt für spezielles $f(t)$:

$$f(\tau_i) > 0,\ f(\tau_j) = 0\ (j \neq i) \underset{\lambda_i < 0}{\rightarrow} \hat{I}(f) = \lambda_i \hat{f}_i < 0$$

$\rightarrow$ Forderung: $\lambda_i > 0$. Für $\lambda_i > 0$ ist $\hat{I}$ positiv.

Newton-Cotes-Formeln

Für eine äquidistante Unterteilung gilt $\tau_i = \frac{b-a}{N} i + a$. Sei $P(t)$ das Interpolationspolynom von $f(t)$ mit den Stützstellen τ_i. Dann gilt

$$\int_a^b P(t) \mathrm{d}t = \sum_{i=0}^{N} \underbrace{f(\tau_i)}_{\hat{f}_i} \int_a^b p_i(t) \mathrm{d}t,$$

mit den Lagrange-Polynomen $p_i(t)$. Es gilt

$$\int_a^b p_i(t) \mathrm{d}t = (b-a) \underbrace{\frac{1}{N} \int_0^N \prod_{j \neq i} \frac{s-j}{i-j} \, \mathrm{d}s}_{=:\lambda_i} = (b-a)\lambda_i.$$

Gewichte λ_i müssen **nur einmal** berechnet werden

Damit folgt: $\hat{I}(f) = (b-a) \sum_{i=0}^{N} \hat{f}_i \lambda_i$

Definition der Newton-Cotes-Formeln

Vor-/Nachteile

- Einfach zu berechnen
- Häufig äquidistante Daten

- Für $N > 7$ ist $\hat{I}$ **nicht positiv**
- Konsistenzordnung $\leq N$

Ergebnisverifikation

Idee: Wähle eine Familie von Funktionen $f_a(t)$ und teste $\hat{I}$

Häufig praktisch: $\sin$ und $\cos$-Funktionen, da die Oszillation von f leicht steuerbar ist

$$f_\omega(t) = \sin(\omega t) \qquad (\omega > 0)$$

Dann gilt mit $a = 0, b = \frac{\pi}{2}$:

$$I(f) = \frac{1 - \cos(\omega\pi/2)}{\omega}$$

Aufgabe: Testen Sie die Newton-Cotes-Formeln für verschiedene ω

Algorithmische Details

- Darstellung eines Polynoms $P(t)$ vom Grad n:

$$P(t) = \sum_{i=0}^{n} q_i t^i = \underbrace{(q_0, \dots, q_n)}_{=q} \begin{pmatrix} 1 \\ t \\ \vdots \\ t^n \end{pmatrix} .$$

- Dann gilt für Polynome $P_1(t)$ $\left(\to\ q^{(1)}\right)$ und $P_2(t)$ $\left(\to\ q^{(2)}\right)$:

$$P_1(t) + \alpha P_2(t) = \left(q^{(1)} + \alpha q^{(2)}\right) (1,\, t, \dots, t^n)^{\mathsf{T}} ,$$

$$P_1(t) t^j = \left(\underbrace{0, \dots, 0}_{j-\text{mal}}, q_0^{(1)}, \dots, q_n^{(1)} \right) \left(1,\, t, \dots, t^{n+j}\right)^{\mathsf{T}} ,$$

$$\int\limits_a^b P_1(t)\, \mathrm{d}t = q^{(1)} \cdot \begin{pmatrix} \int_a^b 1\, \mathrm{d}t \\ \int_a^b t\, \mathrm{d}t \\ \vdots \end{pmatrix} = q^{(1)} \cdot \tilde{I}$$

Legendre Polynome

Definition (L_2-Skalarprodukt)

$$\langle f, g \rangle = \int_a^b f(t) g(t) \, \mathrm{d}t$$

ist ein Skalarprodukt auf dem Raum der Lebesgue integrablen Funktionen.

Gesucht: Orthonormale Polynome $p_i(t)$ auf dem Intervall $[0, 1]$, das heißt:

$$\langle p_i, p_j \rangle = \delta_{ij}.$$

Die Polynome p_i werden als Legendre-Polynome bezeichnet.

Konstruktion der Legendre-Polynome

(Intervall: $t \in [0,1]$)

[A1] Setze $i = 0$

[A2] Initialisiere $p_i^0(t) = t^i$; setze $p_0(t) = 1$

[A3] Berechne $p_i^*(t) = p_i^0(t) - \sum_{j=0}^{i-1} \langle p_i^0, p_j \rangle p_j(t)$

[A4] $p_i(t) = \dfrac{p_i^*(T)}{\sqrt{\langle p_i^*, p_i^* \rangle}}$; $i = i + 1$

[A5] $i < N$? $\rightarrow$ [A2]

Gauss Christoffel Quadratur

Gesucht: Stützstellen τ_i und (positive) Gewichte λ_i, so dass das zugehörige Quadraturverfahren möglichst hohe Konsistenz besitzt

Ansatz

Sei $P(t)$ ein Polynom vom Grad $2N+1$. Dann gilt für spezielle c_{ij}:

$$P(t) = \sum_{i=0}^{N+1} \sum_{j=0}^{i-1} c_{ij} p_i(t) p_j(t) + \sum_{i=0}^{N} c_{ii} p_i^2(t).$$

Sind die p_i die Legendre Polynome ($a = 0, b = 1$ zur Vereinfachung), so gilt:

$$I(P) = \sum_{i=0}^{N+1} \sum_{j=0}^{i-1} c_{ij} \underbrace{\langle p_i, p_j \rangle}_{=0} + \sum_{i=0}^{N} c_{ii} \underbrace{\langle p_i, p_i \rangle}_{=1} = \sum_{i=0}^{N} c_{ii}.$$

Definition

$$p_k(t) = \prod_{0 \le i < k} (t - \tau_i) \qquad (k = 0, \dots, N).$$

Dann gilt für $k \le N$ und mit dem Verfahren $\hat{I} = (\boldsymbol{\lambda}^N, \boldsymbol{\tau}^N)$

$$\int_0^1 p_k(t) p_{N+1}(t) \mathrm{d}t = \hat{I}(p_k p_{N+1}) = \sum_{i=0}^{N} \lambda_i \underbrace{p_{N+1}(\tau_i)}_{=0} p_k(\tau_i) = 0$$

Die Stützstellen des Verfahrens $(\boldsymbol{\lambda}^N, \boldsymbol{\tau}^N)$ sind die Nullstellen des Legendre-Polynoms p_{N+1}.

Die Gewichte können leicht berechnet werden:
Da das Integral auch für Polynome vom Grad N exakt sein muss, entsprechen diese den Lagrange'schen Gewichten.

Integralberechnung

Gegeben:

$N+1$ Stützstellen $\boldsymbol{\tau}$, Gewichte $\boldsymbol{\lambda}$
Funktion $f(t)$, Intervall $[a,b]$

Integralberechnung:

$$\hat{I} = (b-a)\left(\sum_{i=0}^{N} \lambda_i f(a+\tau_i(b-a))\right)$$

Berechnung über M Teilintervalle durch Summation der Integrale auf den einzelnen Intervallen: (mit $h=(b-a)/M$)

$$\hat{I} = \sum_{j=0}^{M-1}\left(h\sum_{i=0}^{N} \lambda_i f(a+jh+\tau_i h)\right)$$