

Numerisches Rechnen

Numerische Integration

M. Grepl

J. Berger & R. O'Connor

Institut für Geometrie und Praktische Mathematik
RWTH Aachen

Wintersemester 2015/16

Vorlesungsinhalt

7. Numerische Integration

berechne $I = \int_a^b f(x) dx$

8. Lineare Gleichungssysteme, iterative Verfahren

geg.: $A \in \mathbb{R}^{n \times n}$, $b \in \mathbb{R}^n$, $n \gg 1$, A dünnbesetzt;

ges.: $x \in \mathbb{R}^n$, so dass $Ax = b$

9. Optimierung

geg.: Kostenfunktion $J : \mathbb{R}^n \rightarrow \mathbb{R}$, Menge $X \subseteq \mathbb{R}^n$;

ges.: Optimum x^* , so dass $x^* = \arg \min_{x \in X} J(x)$

Übersicht

Themen: Dahmen & Reusken Kap. 10.1-10.3

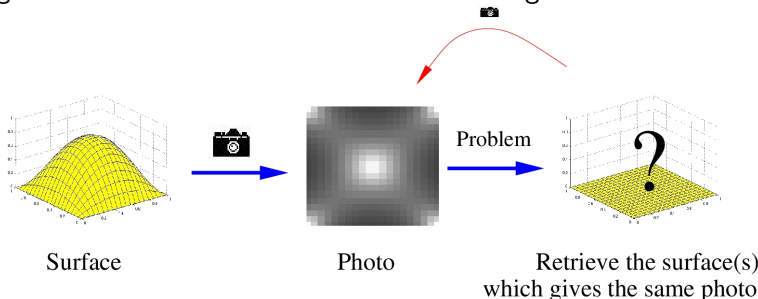
- ▶ Newton-Cotes-Formeln
- ▶ Gauß-Quadratur

Was Sie mitnehmen sollten:

- ▶ Was ist die Grundidee der numerischen Integration
- ▶ Wie kann man den Fehler abschätzen
- ▶ Was ist der Exaktheitsgrad einer Quadraturformel
- ▶ Welchen Exaktheitsgrad haben die Newton-Cotes-Formeln und die Gauß-Quadratur

Problem Definition

Goal: Compute three-dimensional shape of object/scene from brightness of one or more black and white images.



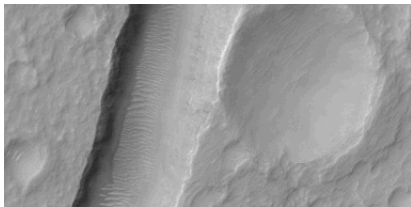
Quelle: Prados & Faugeras

Description of object in terms of, e.g.,

- ▶ depth $z(x, y)$
- ▶ surface normal $\vec{n} = (n_x, n_y, n_z)$
- ▶ surface gradient $(p(x, y), q(x, y))$

Examples

Mars Global Surveyor Orbiter (Quelle: NASA)



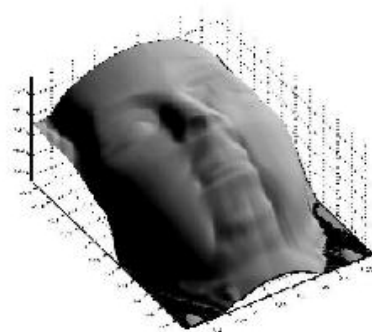
Other examples:

- ▶ Scanning electron microscope
- ▶ Lunar topography

Examples



a) Image



b) 3D surface reconstructed
from the single image a)

Quelle: wikiiii.de

Image Formation

Image Irradiance Equation

$$R(p(x, y), q(x, y)) = I(x, y)$$

where

- ▶ R is the reflectance map
- ▶ I is the brightness image

Difficulties:

- ▶ Loss of data through projection ($3D \rightarrow 2D$)
- ▶ Reconstruction of 3D information from 2D data is ill-posed, i.e., gradients cannot be determined uniquely
- ▶ Absolute depth information not available
- ▶ Regularization/Additional constraints necessary

Other approaches:

- ▶ Shape from Stereo, Shape from Motion, Shape from Texture,

Solution Techniques

Four Groups:

- ▶ Minimization
⇒ Minimization of energy function (brightness constraint, smoothness constraint).
- ▶ Propagation
⇒ Propagate information from a set of surface points (e.g. singular points).
- ▶ Local
⇒ Derive shape based on assumption of surface type (e.g. spherical) and intensity gradients.
- ▶ Linear
⇒ Based on linearization of reflectance map in terms of surface gradient.

Main Idea

Approach:

- ▶ Determine surface normals based on image irradiance equation: reflectance map = image intensities
- ▶ Integrate gradients to obtain height information

Note that depth, surface normal, and surface gradient satisfy

$$p(x, y) = \frac{\partial z(x, y)}{\partial x}, \quad q(x, y) = \frac{\partial z(x, y)}{\partial y}$$

and

$$\vec{n} = \begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix} = \frac{1}{\sqrt{1 + p^2 + q^2}} \begin{bmatrix} -p \\ -q \\ 1 \end{bmatrix}$$

Lambert Reflector

Reflectance map

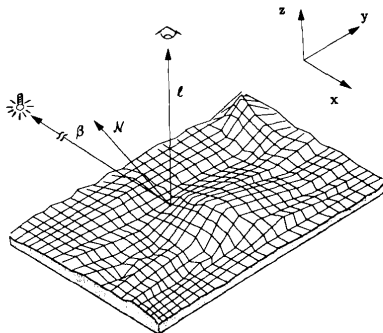
$$\begin{aligned} R(p, q) &= I_0 \rho \cos(\vec{s}, \vec{n}) \\ &= I_0 \rho \vec{s} \cdot \vec{n} \end{aligned}$$

where

- ▶ I_0 : incident light intensity
- ▶ ρ : albedo
- ▶ \vec{s} : light source direction
- ▶ \vec{n} : surface normal

Lambert (diffusive reflexion):

- ▶ Reflection of incident light is the same in all directions.
- ▶ Point of observation is irrelevant, only angle between surface normal and light source is important.



Quelle: Frankot & Chellappa

Determination of surface normal

Assume we are given three images with different lighting but same camera position. We then have

$$I_i(x, y) = I_{0,i} \rho (\vec{s}_i \cdot \vec{n}), \quad i = 1, 2, 3,$$

which we can rewrite as

$$\underline{I} = \begin{bmatrix} I_1 \\ I_2 \\ I_3 \end{bmatrix} = \begin{bmatrix} I_{0,1} \vec{s}_1^T \\ I_{0,2} \vec{s}_2^T \\ I_{0,3} \vec{s}_3^T \end{bmatrix} \rho \vec{n} = S \vec{n}$$

where

$$S = \rho \begin{bmatrix} I_{0,1} s_{1,x} & I_{0,1} s_{1,y} & I_{0,1} s_{1,z} \\ I_{0,2} s_{2,x} & I_{0,2} s_{2,y} & I_{0,2} s_{2,z} \\ I_{0,3} s_{3,x} & I_{0,3} s_{3,y} & I_{0,3} s_{3,z} \end{bmatrix}$$

\Rightarrow Solve 3×3 linear system for each pixel.

Determination of depth

Given the “needle diagram” (i.e. \vec{n} at each pixel) we obtain

$$\vec{n} = \begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix} = \frac{1}{\sqrt{1 + p^2 + q^2}} \begin{bmatrix} -p \\ -q \\ 1 \end{bmatrix}$$

or

$$\frac{\partial z(x, y)}{\partial x} = p(x, y) = -\frac{n_x}{n_z}, \quad \frac{\partial z(x, y)}{\partial y} = q(x, y) = -\frac{n_y}{n_z}.$$

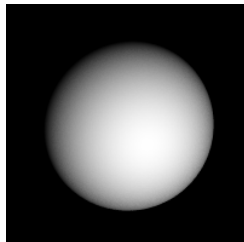
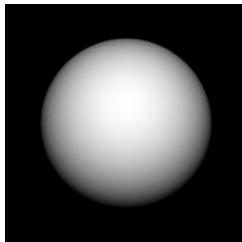
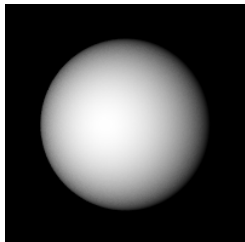
Integration along an (arbitrary) curve yields

$$z(x, y) = z_0(x, y) + \int_{(x_0, y_0)}^{(x, y)} (p \, dx + q \, dy).$$

\Rightarrow Numerical integration required (and possibly least-squares).

Images of sphere

Sphere under three different lighting conditions

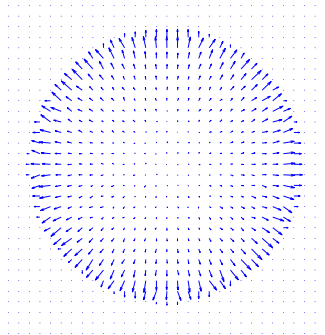
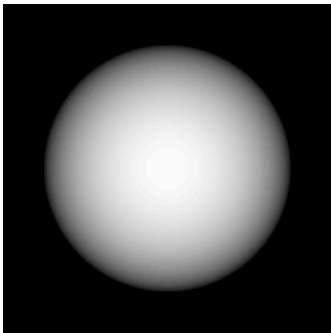


Quelle: S. Sarkar

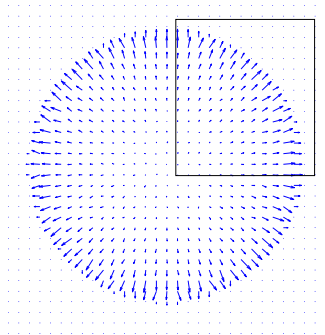
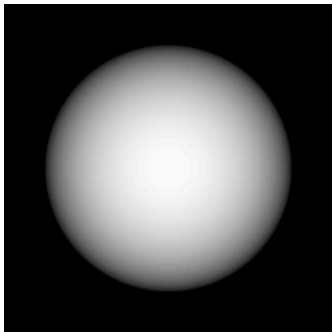
Approach:

- ▶ Determine “needle diagram”
- ▶ Recover depth information

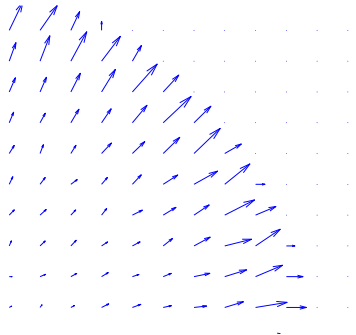
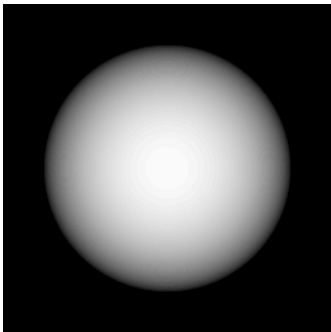
Needle Diagram



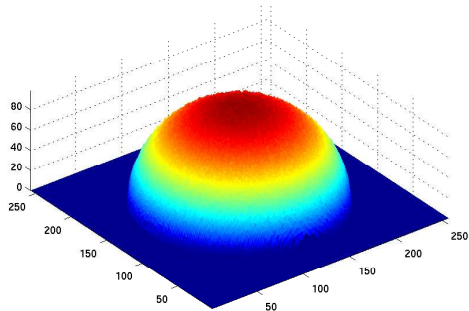
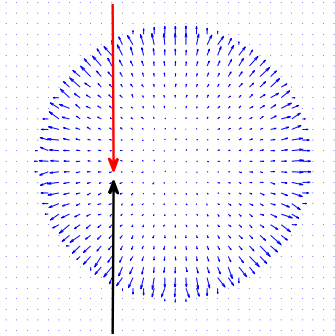
Needle Diagram



Needle Diagram



Depth information



Numerical integration (“Trapezregel”):

$$z(x, y) = \int_0^y -\frac{n_y}{n_z} dy \quad \text{oder} \quad z(x, y) = - \int_{256}^y -\frac{n_y}{n_z} dy$$

Grundlegende Idee

Die gängige Strategie zur näherungsweise Berechnung des Integrals

$$\int_a^b f(x) dx$$

läßt sich folgendermaßen umreißen:

N10.1

1. Man unterteile $[a, b]$ in Teilintervalle $[t_{k-1}, t_k]$ z.B. mit $t_j = a + jh$, $j = 0, \dots, n$, $h = \frac{b-a}{n}$.
2. Approximiere f auf jedem Intervall $[t_{k-1}, t_k]$ durch eine *einfach* zu integrierende Funktion g_k , und verwende

$$\int_a^b f(x) dx = \sum_{k=1}^n \int_{t_{k-1}}^{t_k} f(x) dx \approx \sum_{k=1}^n \int_{t_{k-1}}^{t_k} g_k(x) dx$$

als Näherung für das exakte Integral.

Kondition

Sei

$$I = \int_a^b f(x) dx, \quad \tilde{I} = \int_a^b \tilde{f}(x) dx$$

Es gilt

$$\begin{aligned} |I - \tilde{I}| &= \left| \int_a^b f(x) - \tilde{f}(x) dx \right| \leq \int_a^b |f(x) - \tilde{f}(x)| dx \\ &\leq (b - a) \|f - \tilde{f}\|_{\infty}. \end{aligned}$$

und damit

$$\frac{|I - \tilde{I}|}{|I|} \leq (b - a) \frac{\|f - \tilde{f}\|_{\infty}}{\left| \int_a^b f(x) dx \right|} = \underbrace{\frac{\int_a^b \|f\|_{\infty} dx}{\left| \int_a^b f(x) dx \right|}}_{= \kappa_{\text{rel}}} \cdot \frac{\|f - \tilde{f}\|_{\infty}}{\|f\|_{\infty}}.$$

Grundlegende Idee

Die gängige Strategie zur näherungsweise Berechnung des Integrals

$$\int_a^b f(x) dx$$

läßt sich folgendermaßen umreißen:

1. Man unterteile $[a, b]$ in Teilintervalle $[t_{k-1}, t_k]$ z.B. mit $t_j = a + jh$, $j = 0, \dots, n$, $h = \frac{b-a}{n}$.
2. Approximiere f auf jedem Intervall $[t_{k-1}, t_k]$ durch eine *einfach* zu integrierende Funktion g_k , und verwende

$$\int_a^b f(x) dx = \sum_{k=1}^n \int_{t_{k-1}}^{t_k} f(x) dx \approx \sum_{k=1}^n \int_{t_{k-1}}^{t_k} g_k(x) dx$$

als Näherung für das exakte Integral.

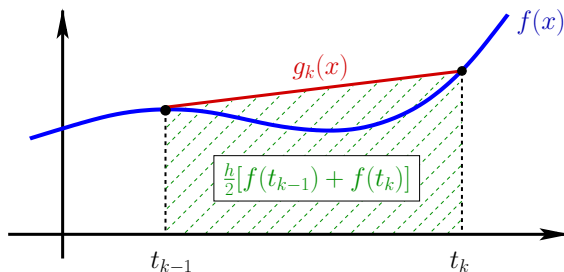
Trapezregel

- ▶ Lineare Interpolation an den Intervallenden von $[t_{k-1}, t_k]$, d.h.

$$g_k(x) = \frac{x - t_{k-1}}{h} f(t_k) + \frac{t_k - x}{h} f(t_{k-1}),$$

- ▶ Das Integral ist damit gegeben durch

$$\int_{t_{k-1}}^{t_k} g_k(x) dx = \frac{h}{2} [f(t_{k-1}) + f(t_k)].$$



Summierte Trapezregel

Wiederholtes Anwenden der Trapezregel auf jedem Teilintervall $[t_{k-1}, t_k]$ als Näherung für $\int_a^b f(x) dx$ liefert die

N10.2

Summierte Trapezregel

$$T(h) = h \left[\frac{1}{2} f(a) + f(t_1) + \dots + f(t_{n-1}) + \frac{1}{2} f(b) \right]$$

Für den Verfahrensfehler der Teilintegrale gilt folgende Darstellung:

Lemma 10.1

Sei $f \in C^2([t_{k-1}, t_k])$. Es gilt:

$$\frac{h}{2} [f(t_{k-1}) + f(t_k)] = \int_{t_{k-1}}^{t_k} f(x) dx + \frac{f''(\xi_k)}{12} h^3,$$

für $\xi_k \in [t_{k-1}, t_k]$.

Summierte Trapezregel

Für den Verfahrensfehler des gesamten Integrals $T(h)$ ergibt sich damit die Abschätzung

$$\begin{aligned} \left| T(h) - \int_a^b f(x) dx \right| &= \left| \sum_{k=1}^n \frac{f''(\xi_k)}{12} h^3 \right| \\ &\leq \frac{h^3}{12} \sum_{k=1}^n |f''(\xi_k)| \leq \frac{h^3}{12} n \max_{x \in [a,b]} |f''(x)|. \end{aligned}$$

Mit $nh = b - a$ ergibt sich insgesamt die

Fehlerschranke

$$\left| T(h) - \int_a^b f(x) dx \right| \leq \frac{h^2}{12} (b - a) \max_{x \in [a,b]} |f''(x)|.$$

Summierte Trapezregel

Ebenfalls erhält man wegen

$$E(h) := T(h) - \int_a^b f(x) dx = \sum_{k=1}^n \frac{f''(\xi_k)}{12} h^3 = \frac{h^2}{12} \sum_{k=1}^n h f''(\xi_k)$$

und

$$\lim_{h \rightarrow 0} \frac{E(h)}{h^2} = \frac{1}{12} \int_a^b f''(x) dx = \frac{1}{12} (f'(b) - f'(a))$$

die

Fehlerschätzung

$$E(h) \approx \hat{E}(h) := \frac{h^2}{12} (f'(b) - f'(a)).$$

Beispiel 10.2

Zur näherungsweise Berechnung von

N10.3

$$I = \int_0^{\pi/2} (x \cos x + e^x) dx = \frac{\pi}{2} + e^{\frac{1}{2}\pi} - 2$$

mit der Trapezregel ergeben sich die in folgender Tabelle angegebenen Näherungswerte, Verfahrensfehler und Fehlerschätzungen.

n	$T(h)$	$ E(h) = T(h) - I $	$ \hat{E}(h) = \frac{h^2}{12} f'(\frac{\pi}{2}) - f'(0) $
4	4.396928	1.57e-02	1.59e-02
8	4.385239	3.97e-03	3.98e-03
16	4.382268	9.95e-04	9.96e-04
32	4.381523	2.49e-04	2.49e-04

Newton-Cotes-Formeln

- ▶ Für ein typisches Teilintervall $[t_{k-1}, t_k]$ stehe der Einfachheit halber im Folgenden $[c, d]$.
- ▶ Seien nun $x_0, \dots, x_m \in [c, d]$ paarweise verschiedene Punkte.
- ▶ Integration des Interpolationspolynoms liefert die Quadraturformel

$$I_m(f) = \int_c^d P(f|x_0, \dots, x_m)(x) dx.$$

Satz 10.3.

Sei $I_m(f)$ wie oben. Für jedes Polynom $Q \in \Pi_m$ gilt

$$I_m(Q) = \int_c^d Q(x) dx.$$

Man sagt, die Quadraturformel ist *exakt* vom Grade m .

Newton-Cotes-Formeln

Wir erhalten damit als Näherung für das exakte Integral

$$\begin{aligned}\int_c^d P(f|x_0, \dots, x_m)(x) dx &= \int_c^d \sum_{j=0}^m f(x_j) \ell_{jm}(x) dx \\&= \sum_{j=0}^m f(x_j) \int_c^d \ell_{jm}(x) dx \\&= \sum_{j=0}^m f(x_j) \int_c^d \prod_{\substack{k=0 \\ k \neq j}}^m \frac{x - x_k}{x_j - x_k} dx \\&= h \sum_{j=0}^m f(x_j) \underbrace{\frac{1}{h} \int_c^d \prod_{\substack{k=0 \\ k \neq j}}^m \frac{x - x_k}{x_j - x_k} dx}_{c_j} \\&= h \sum_{j=0}^m f(x_j) c_j\end{aligned}$$

Newton-Cotes-Formeln

Lemma 10.4

Es gibt Gewichte c_0, \dots, c_m , so dass $I_m(f)$ die Form

$$I_m(f) = h \sum_{j=0}^m c_j f(x_j)$$

hat, wobei wieder $h = d - c$. Die c_j sind durch

$$c_j = \frac{1}{h} \int_c^d \prod_{\substack{k=0 \\ k \neq j}}^m \frac{x - x_k}{x_j - x_k} dx = \frac{1}{h} \int_c^d \ell_{jm}(x) dx$$

gegeben, wobei ℓ_{jm} ($0 \leq j \leq m$) die Lagrange-Fundamentalpolynome zu den Stützstellen x_0, \dots, x_m sind.

Newton-Cotes-Formeln

Wählt man speziell die Stützstellen x_j äquidistant

$$x_0 = c + \frac{1}{2}h =: c + \xi_0 h, \quad \text{wenn } m = 0$$

$$x_j = c + \frac{j}{m}h =: c + \xi_j h, \quad j = 0, \dots, m, \quad \text{wenn } m > 0,$$

erhält man die **Newton-Cotes-Formeln**.

Beispiel

Für $m = 1$ erhalten wir $x_0 = c$, $x_1 = d$ und

$$\begin{aligned} c_0 &= \frac{1}{h} \int_c^d \frac{x - x_1}{x_0 - x_1} dx = \frac{1}{2} \\ c_1 &= \frac{1}{h} \int_c^d \frac{x - x_0}{x_1 - x_0} dx = \frac{1}{2}. \end{aligned}$$

Newton-Cotes-Formeln

Mann kann die Quadraturformel in der Form

N10.4

$$I_m(f) = h \sum_{j=0}^m c_j f(c + \xi_j h)$$

mit *normierten* Stützstellen ξ_j und Gewichten c_j schreiben, die jetzt unabhängig vom speziellen Intervall $[c, d]$ sind, z.B.

m		ξ_j	c_j	$I_m(f) - \int_c^d f(x)dx$
0	Mittelpunktsregel	$\frac{1}{2}$	1	$-\frac{1}{24}h^3 f^{(2)}(\xi)$
1	Trapezregel	0, 1	$\frac{1}{2}, \frac{1}{2}$	$\frac{1}{12}h^3 f^{(2)}(\xi)$
2	Simpson-Regel	$0, \frac{1}{2}, 1$	$\frac{1}{6}, \frac{4}{6}, \frac{1}{6}$	$\frac{1}{90}(\frac{1}{2}h)^5 f^{(4)}(\xi)$
3	$\frac{3}{8}$ -Regel	$0, \frac{1}{3}, \frac{2}{3}, 1$	$\frac{1}{8}, \frac{3}{8}, \frac{3}{8}, \frac{1}{8}$	$\frac{3}{80}(\frac{1}{3}h)^5 f^{(4)}(\xi)$
4	Milne-Regel	$0, \frac{1}{4}, \frac{1}{2}, \frac{3}{4}, 1$	$\frac{7}{90}, \frac{32}{90}, \frac{12}{90}, \frac{32}{90}, \frac{7}{90}$	$\frac{8}{945}(\frac{1}{4}h)^7 f^{(6)}(\xi)$

Summierte Newton-Cotes-Formeln

Beispiel: Summierte Simpson-Regel.

$$S(h) = \int_a^b f(x) dx + E(h)$$

mit

$$S(h) = \frac{h}{6} \left[f(t_0) + 4f\left(\frac{t_0 + t_1}{2}\right) + 2f(t_1) + 4f\left(\frac{t_1 + t_2}{2}\right) + 2f(t_2) + \dots + 2f(t_{n-1}) + 4f\left(\frac{t_{n-1} + t_n}{2}\right) + f(t_n) \right]$$

und Fehlerschranke

$$E(h) = \sum_{k=1}^n \frac{1}{90} \left(\frac{1}{2}h\right)^5 f^{(4)}(\xi_k) = \frac{h^4}{2880} \sum_{k=1}^n h f^{(4)}(\xi_k),$$

für $\xi_k \in [t_{k-1}, t_k]$.

Summierte Newton-Cotes-Formeln

Es gilt, wegen $nh = b - a$,

$$|E(h)| \leq \frac{h^4}{2880} (b - a) \|f^{(4)}\|_{\infty},$$

$$E(h) \approx \frac{h^4}{2880} \int_a^b f^{(4)}(x) dx = \frac{h^4}{2880} (f^{(3)}(b) - f^{(3)}(a)).$$

Beachte: Beim Aufsummieren der einzelnen Teilintegrale,

$$\int_a^b f(x) dx = \sum_{k=1}^n \int_{t_{k-1}}^{t_k} f(x) dx,$$

geht im Fehler eine h -Potenz verloren.

Beispiel 10.5

Zur näherungsweise Berechnung von (siehe Beispiel 10.2)

$$I = \int_0^{\pi/2} (x \cos x + e^x) dx = \frac{\pi}{2} + e^{\frac{1}{2}\pi} - 2$$

ergeben sich die Resultate wie in folgender Tabelle.

n	$S(h)$	$ E(h) $	$\frac{h^4}{2880} f^{(3)}(\frac{\pi}{2}) - f^{(3)}(0) $
4	4.381343022	6.93e - 05	6.92e - 05
8	4.381278035	4.33e - 06	4.33e - 06
16	4.381273978	2.70e - 07	2.70e - 07
32	4.381273725	1.69e - 08	1.69e - 08

Gauß-Quadratur

Zielvorgabe

Entwickle für $m \in \mathbb{N}$ eine Formel

$$\sum_{i=0}^m \omega_i f(x_i) = \int_c^d P(f|x_0, \dots, x_m)(x) dx$$

mit:

- ▶ positiven Gewichten $\omega_i, i = 0, \dots, m$;
- ▶ mit möglichst hohem Exaktheitsgrad $n \geq m$, d.h.,

$$\int_c^d Q(x) dx = \sum_{i=0}^m \omega_i Q(x_i), \quad \forall Q \in \Pi_n.$$

Zur Erinnerung: Der Exaktheitsgrad bei Newton-Cotes-Formeln $I_m(f)$ ist entweder m oder $m + 1$.

Gauß-Quadratur

- ▶ Exaktheitsgrad kann *höchstens* $2m + 1$ sein.
⇒ Gaußsche Quadraturformeln

Satz 10.6

Sei $m \geq 0$. Es existieren Stützstellen $x_0, \dots, x_m \in (c, d)$ und positive Gewichte $\omega_0, \dots, \omega_m$, so dass mit $h = d - c$

$$h \sum_{i=0}^m \omega_i f(x_i) = \int_c^d f(x) dx + E_f(x)$$

und

$$E_Q = 0 \quad \text{für alle } Q \in \Pi_{2m+1}.$$

Ferner gilt für passendes $\xi \in [c, d]$

$$|E_f(h)| = \frac{((m+1)!)^4}{((2m+2)!)^3 (2m+3)} h^{2m+3} |f^{(2m+2)}(\xi)|.$$

Numerisches Test

- ▶ Betrachte Quadraturformel

$$I_{k,n} \approx \int_a^b f(x) dx = I(f),$$

wobei $[a, b]$ in n Teilintervalle mit Länge $\frac{b-a}{n} = h$ unterteilt wird und auf jedem Teilintervall eine Gauß-Quadratur mit $k = m + 1$ Stützstellen angewandt wird.

- ▶ Für glatte Funktionen (d.h. $|f^{(2k)}|$ wird nicht allzu groß, wenn k größer wird) wird die Qualität der Gauß-Quadratur im Wesentlichen durch den Faktor $C_{k,h}$ bestimmt, wobei

$$C_{k,h} := \frac{(k!)^4}{((2k)!)^3(2k+1)} h^{2k+1}.$$

Wir erhalten folgende Werte ...

Numerisches Test

h	$k = 2$	$k = 4$	$k = 8$
4	$2.4\text{e} - 01$	$1.5\text{e} - 04$	$2.9\text{e} - 13$
2	$7.4\text{e} - 03$	$2.9\text{e} - 07$	$2.2\text{e} - 18$
1	$2.3\text{e} - 04$	$5.6\text{e} - 10$	$1.7\text{e} - 23$
0.5	$7.2\text{e} - 06$	$1.1\text{e} - 12$	$1.3\text{e} - 28$

- ▶ Für $I_{2k,n}$ und $I_{k,2n}$ wird die Anzahl der Funktionsauswertungen etwa verdoppelt im Vergleich zu $I_{k,n}$.

- ▶ In obiger Tabelle kann man sehen, dass man

$$|I - I_{2k,n}| \ll |I - I_{k,2n}|$$

erwarten darf.

- ▶ In der Praxis wird daher bei der Gauß-Quadratur n in der Regel klein gewählt, oft sogar $n = 1$.

Beispiel 10.7.

Gauß-Quadratur zur Berechnung von (siehe Beispiel 10.2)

$$I = \int_0^{\pi/2} (x \cos x + e^x) dx = \frac{\pi}{2} + e^{\frac{1}{2}\pi} - 2$$

mit $[c, d] = [0, \frac{\pi}{2}]$ (d.h. $n = 1$) ergibt die Resultate:

m	I_m	$ I_m - I $
1	4.3690643196	1.22e - 03
2	4.3813023502	2.86e - 05
3	4.3812734352	2.73e - 07
4	4.3812737083	5.18e - 10

Die Genauigkeit der Gauß-Quadratur mit 5 Funktionswerten ($m = 4; k = 5$) ist besser als die der Simpson-Regel angewandt auf $n = 32$ Teilintervalle (vgl. Beispiel 10.5), wobei insgesamt 65 Funktionwerte benötigt werden.

Beispiel 10.8.

- ▶ **Aufgabe:** Berechnung der Stützstellen und Gewichte der Gauß-Quadratur für $[c, d] = [-1, 1]$ und $m = 1$.
- ▶ Die Gauß-Quadraturformel

$$I_1(f) = 2(c_0 f(x_0) + c_1 f(x_1))$$

muß für $p \in \Pi_3$ exakt sein, d.h.

$$\int_{-1}^1 p(x) dx = 2(c_0 p(x_0) + c_1 p(x_1))$$

für $p(x) = x^k, k = 0, 1, 2, 3$.

- ▶ Aus

$$\int_{-1}^1 x^k dx = 2(c_0 x_0^k + c_1 x_1^k), \quad k = 0, 1, 2, 3,$$

erhält man die Gleichungen ...

Beispiel 10.8.

erhält man die Gleichungen ...

$$2 = 2(c_0 + c_1), \quad 0 = 2(c_0 x_0 + c_1 x_1),$$

$$\frac{2}{3} = 2(c_0 x_0^2 + c_1 x_1^2), \quad 0 = 2(c_0 x_0^3 + c_1 x_1^3).$$

- Dieses nichtlineare Gleichungssystem hat genau zwei Lösungen:

$$c_0 = c_1 = \frac{1}{2}, \quad x_0 = -\frac{1}{3}\sqrt{3}, \quad x_1 = \frac{1}{3}\sqrt{3}$$

$$c_0 = c_1 = \frac{1}{2}, \quad x_0 = \frac{1}{3}\sqrt{3}, \quad x_1 = -\frac{1}{3}\sqrt{3}$$

Allgemeiner Fall

- Gauß-Quadratur: Stützstellen und Gewichte **N10.5(1)**
- Integrationsgebiet **N10.5(2)**
- Gauß vs. Newton-Cotes **N10.6**