physik441 Computerphysik — Vorlesung 3

Marcus Petschlies[1]

[1] *Helmholtz-Institut für Strahlen- und Kernphysik*

(Dated: April 24, 2023)

Wir beginnen mit den Verfahren zur Numerischen Integration von reell-wertigen Funktionen einer Variablen. Aufbauend auf der Polynominterpolation schauen wir die Familie von Integrationsschemen aus den Newton-Cotes Formeln an, insbesondere die Trapez- und Simpson-Regel, und ihre iterative Verfeinerung. Die Betrachtung der Integrationsfehler führt uns schließlich zur Anwendung der Extrapolation in Form der Romberg-Integration.

Numerische Integration.—

Wir wollen *bestimmte Integrale* berechnen, also

$$\mathcal{I} = \int_a^b f(x)\,dx\,. \tag{1}$$

Wir kennen das aus der Analysis, wo wir im besten Fall eine Stammfunktion finden $\int f(x)\,dx = F(x)$ und $F'(x) = f(x)$, so dass einfach gilt $\mathcal{I} = F(b) - F(a)$.

Wird im Allgemeinen nicht klappen, selbst für analytisch bekannte Funktionen $f(x)$, geschweige denn, wenn wir die Funktion f nur auswerten können, ohne Formel.

In diesem Fall benutzen wir numerische Integration, nähern also das Integral durch eine endliche Summe an

$$\int_a^b f(x)\,dx = \sum_{k=0}^{n} \sigma_k + \quad \text{Fehler}\,, \tag{2}$$

und wir müssen uns um zwei Hauptpunkte kümmern

1. Integrationsmethoden für die Definition der Summe;
2. Konvergenz der Summe und den Approximationsfehler.

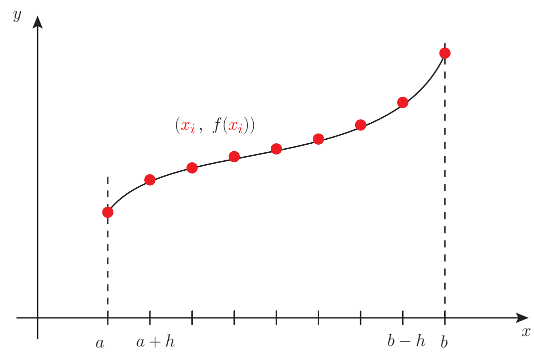
Newton & Cotes

Wir integrieren über das Intervall $[a, b]$ und benutzen dazu äquidistante Stützstellen $x_i = a + h\,i$ mit $i = i, \ldots, n$ und $h = (b-a)/n$. h ist die *Schrittweite*.

Idee Wir interpolieren die gegebene Funktion f auf $[a, b]$ durch ein Polynom vom Grad n mit den $n+1$ Stützstellen $(x_i, f_i = f(x_i))$. Dann integrieren wir das Polynom. Letztere Integration können wir immer algebraisch machen. Insgesamt also,

$$\begin{aligned}\int_a^b f(x)\,dx &\approx \int_a^b P_n(x)\,dx\,, \quad P_n \in \Pi_n\,,\\ P_n(x_i) &= f_i\,, \quad i = 0\ldots, n\,.\end{aligned} \tag{3}$$

wie in der folgenden Abbildung illustriert



Jetzt wollen wir aus dieser Idee praktische Verfahren ableiten, mit konkreten Interpolationsformeln, die wir als Algorithmen programmieren können.

Wir kennen die exakte Lösung des Interpolationsproblems: die Lagrangesche Interpolationsformel für P_n

$$P_n(x) = \sum_{i=0}^{n} f_i \prod_{j\neq i} \frac{x - x_j}{x_i - x_j}\,. \tag{4}$$

Um eine skalen-invariante Integrationsvorschrift zu erhalten, führen wir die dimensionslose Variable t ein mit $x = a + h\cdot t$ ein, wobei dann $0 \le t \le n$ gilt. Damit schreiben wir die Lagrange-Polynome um

$$\begin{aligned}\phi_i(t) &= \prod_{j\neq i} \frac{x - x_j}{x_i - x_j} = \prod_{j\neq i} \frac{t - j}{i - j}\,,\\ dx &= h\,dt\,.\end{aligned} \tag{5}$$

Das Integral des Polynoms P_n wird dann

$$\begin{aligned}\int_a^b P_n(x)\,dx &= \sum_{i=0}^{n} f_i \int_a^b \prod_{j\neq i} \frac{x - x_j}{x_i - x_j}\\ &= h \sum_{i=0}^{n} f_i \int_0^n \phi_i(t)\,dt = h \sum_{i=0}^{n} f_i\,\alpha_i\,.\end{aligned} \tag{6}$$

Die *Gewichte* α_i hängen nur von n ab, nicht mehr von a, b oder f, und sind also universell einsetzbar.

Wir schauen uns die Fälle $n = 1, 2$ genauer an.

$n = 1$

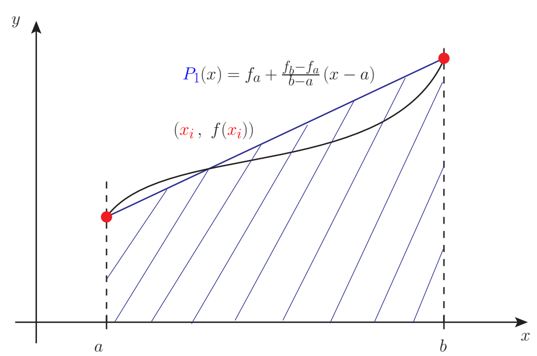
Wir benutzen also $x_0 = a$, $x_1 = b$, $h = b - a$ und die folgenden Gewichte

$$\phi_0(t) = \frac{t-1}{0-1}\,, \quad \alpha_0 = \int_0^1 (1-t)\,dt = 1 - \frac{1}{2} = \frac{1}{2}\,,$$
$$\phi_1(t) = \frac{t-0}{1-0}\,, \quad \alpha_1 = \int_0^1 t\,dt = \frac{1}{2}\,, \tag{7}$$

Diese Gewichte α_0, α_1 ergeben die **Trapez-Regel**

$$\int_a^b P_1(x)\,dx = (b-a)\left(\frac{f(a)}{2} + \frac{f(b)}{2}\right)\,, \tag{8}$$

deren geometrische Interpretation im folgenden Plot dargestellt ist.



$n = 2$

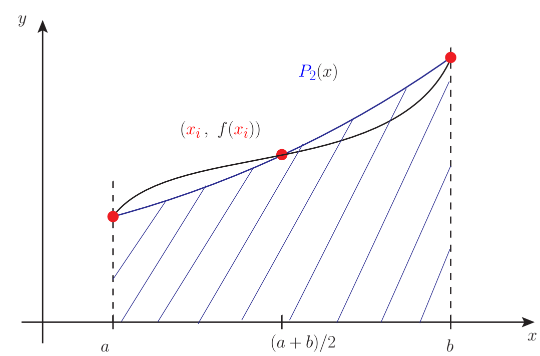
Die zweite wichtige Integrationsregel folgt aus den Newton-Cotes-Formeln für $n = 2$. In diesem Fall haben wir die Stützpunkte $x_0 = a$, $x_1 = \frac{a+b}{2}$, $x_2 = b$, mit Schrittweite $h = \frac{b-a}{2}$. Wir berechnen wieder die Gewichte

$$\phi_0(t) = \frac{(t-1)\,(t-2)}{(0-1)\,(0-2)} = \frac{t^2 - 3t + 2}{2}\,,$$
$$\alpha_0 = \int_0^2 \frac{t^2 - 3t + 2}{2}\,dt = \frac{1}{3}\,,$$
$$\phi_1(t) = \frac{(t-0)\,(t-2)}{(1-0)\,(1-2)} = -t^2 + 2t\,,$$
$$\alpha_1 = \int_0^2 (-t^2 + 2t)\,dt = \frac{4}{3}\,,$$
$$\phi_2(t) = \frac{(t-0)\,(t-1)}{(2-0)\,(2-1)} = \frac{t^2 - t}{2}\,,$$
$$\alpha_2 = \int_0^2 \frac{t^2 - t}{2}\,dt = \frac{1}{3}\,, \tag{9}$$

und wir erhalten die Integrationsregel

$$\int_a^b P_2(x)\,dx = \frac{b-a}{2}\,\frac{1}{3}\left(f(a) + 4\,f(\frac{a+b}{2}) + f(b)\right)\,. \tag{10}$$

Das ist die **Simpson-Regel**, anschaulich dargestellt im folgenden Plot.



Diese Verfahren kann man für größere n weiter treiben und erhält die Integrationsformeln mittels Interpolation mit Polynomen von höherem Grad. Die Wahl von n ist entscheidend für den Approximationsfehler und die *Ordnung* des Verfahrens.

Approximationsfehler

Für den Fehler gilt folgends

$$\int_a^b P_n(x)\,dx - \int_a^b f(x)\,dx = h^{p+1}\,K\,f^{(p)}(\xi)\,, \quad \xi \in (a, b)\,. \tag{11}$$

Da wir hier die p-te Ableitung von f benutzen, muss f dann auf $[a,\, b]$ hinreichend of differenzierbar sein.

Wichtig ist, p und die Konstante K hängen von n, aber *nicht* von f ab.

Der Wert von p gibt die

Ordnung des Verfahrens: Das ist die *größte* ganze Zahl, sodass alle Polynome vom Grad $< p$ mit dem Verfahren exakt integriert werden.

In Gl. (11) sehen wir das an $f^{(p)}$. Wenn wir für f ein Polynom vom Grad $p-1$ einsetzen, dann ist die p-te Ableitung identische Null (Grad $p-1$ heißt höchste Potenz ist x^{p-1}).

Die Ordnung der beiden Regeln Trapez und Simpson sind dann

Regel	n	Fehler	Ordnung
Trapez	$n=1$	$\frac{h^3}{12}\, f^{(2)}(\xi)$	$p=2$
Simpson	$n=2$	$\frac{h^5}{90}\, f^{(4)}(\xi)$	$p=4$

Wir beweisen die Fehlerformel nicht im Detail, aber wir können die Ordnung der beiden für uns relevanten Verfahren nachweisen.

Die Trapez-Regel ist angegeben mit Ordnung $p = 2$, d.h. Polynome vom Grad 1 müssen exakt integriert werden. Das ist offensichtlich richtig, weil eine lineare Funktion natürlich identisch zum Interpolationspolynom 1. Ordnung ist.

Für ein Polynom vom Grad 2 können wir schreiben

$$f(x) = f_{\frac{a+b}{2}} + \frac{f_b - f_a}{b-a}\left(x - \frac{a+b}{2}\right) + 2\frac{f_b + f_a - 2f_{\frac{a+b}{2}}}{(b-a)^2}\left(x - \frac{a+b}{2}\right)^2 \tag{12}$$

Das Integral ergibt wie gewohnt

$$\begin{aligned}\int_a^b f(x)\,dx &= f_{\frac{a+b}{2}}\,(b-a) + \frac{4}{3}\,\frac{f_b + f_a - 2f_{\frac{a+b}{2}}}{(b-a)^2}\left(\frac{b-a}{2}\right)^3 \\ &= \frac{b-a}{2}\,\frac{1}{3}\left\{\frac{1}{6}\,f_a + \frac{4}{6}\,f_{\frac{a+b}{2}} + \frac{1}{6}\,f_b\right\},\end{aligned} \tag{13}$$

was natürlich identisch mit der Simpson-Regel und ungleich $(b-a)\left(\frac{f_a}{2} + \frac{f_b}{2}\right)$ ist. Also werden Polynome vom Grad 2 nicht mehr exakt integriert.

Für die Simpson-Regel schreiben wir das nicht mehr ganz so explizit. Stattdessen folgendermaßen. Dass Polynome vom Grad 4 im Allgemeinen nicht exakt integriert werden, sehen wir am Gegenbeispiel $f(x) = x^4$ auf dem Intervall $[0,\, 1]$. Exakter Wert ist

$$\int_0^1 f(x)\,dx = \frac{1}{5}\,, \tag{14}$$

während die Simpson-Regel $(0 + 4/2^4 + 1)/6 = 5/24$ ergibt. Stimmt also nicht.

Um zu sehen, dass Polynome vom Grad 3 immer noch exakt integriert werden, sei $f(x)$ also ein Polynom vom Grad 3, und $g(x)$ das eindeutig bestimmte Polynom vom Grad 2 durch die Punkte $(a,\, f(a))$, $(\frac{a+b}{2},\, f(\frac{a+b}{2}))$, und $(b,\, f(b))$. Die Differenz $q(x) = f(x) - g(x)$ ist dann also ein Polynom vom Grad 3 mit Nullstellen bei a, $(a+b)/2$ und b. Es hat also die Form

$$q(x) = A\,(x-a)\,(x - \frac{a+b}{2})\,(x-b)\,. \tag{15}$$

$q(x)$ ist *antisymmetrisch* bzgl. des Intervallmittelpunkts $(a+b)/2$. Das sehen wir an

$$\begin{aligned}q(b-(x-a)) &= A\,\left(b-(x-a)-a\right)\times \\ &\quad\times\left(b-(x-a)-\frac{a+b}{2}\right)\left(b-(x-a)-b\right) \\ &= (-1)^3\,q(x)\,.\end{aligned} \tag{16}$$

Das Integral von q über $[a,\, b]$ ist auf Grund der Antisymmetrie gleich Null.

Also werden beliebige Polynome vom Grad 3 auch exakt integriert und die Ordnung von Simpson ist $p = 4$.

Die Newton-Cotes Formeln kann man auch für höhere n auswerten und erhält die zugehörigen Integrationsregeln. Die Gewichte für z.B. $n = 8$ lauten (hier nur auf 3 Nachkommastellen, um den Punkt zu machen)

i	0	1	2	3	4	5	6	7	8
α_i	0.279	1.662	-0.262	2.962	-1.281	2.962	-0.262	1.662	0.279

Wir haben also positive und negative Gewichte. Dieses Verhalten setzt sich für $n > 8$ fort, für $n \geq 11$ haben auch die ungeraden n positive und negative Gewichte.

Dazu überspannen die Gewichte für große n viele Größenordnungen, so z.B. für $n = 50$ von 10^{-1} bis 10^{11} mit unterschiedlichen Vorzeichen, sodass die sehr genaue

Addition und Subtraktion einzelner Terme essentiell wird.

Das wird *numerische instabil*, und das wollen wir nicht.

Trapeze - viele sind gut, mehr sind besser

Der Fall einer wachsenden Anzahl immer feiner gewählter Stützstellen interessiert uns natürlich. Wir wollen also in der Lage sein, auch mit $n \gg 50$ numerisch stabil eine Approximation des Integrals auszuwerten.

Dafür gehen wir weg vom Versuch, die Funktion auf dem Integrationsintervall durch ein einziges Polynom zu beschreiben. Stattdessen benutzen wir eine *stückweise* Polynominterpolation.

Wir nehmen also wieder das Intervall $[a,\, b]$ und unterteilen es in n Teilintervalle $[x_i,\, x_{i+1}]$ mit $x_i = a + \frac{b-a}{n}\, i$ und $i = 0, 1, \ldots, n$. Dann integrieren wir auf den einzelnen Teilintervallen nach der Trapez-Regel und summieren

$$\begin{aligned}\int_a^b f(x)\, dx &= \sum_{i=0}^{n-1} \int_{x_i}^{x_{i+1}} f(x)\, dx \\ &\approx \sum_{i=0}^{n-1} \frac{h}{2}\, \left(f(x_i) + f(x_{i+1})\right) \\ &= \frac{b-a}{n}\, \left(\frac{1}{2}\, f(a) + f(a+h) + \cdots + f(b-h) + \frac{1}{2}\, f(b)\right) \\ &= T(h)\,. \end{aligned} \tag{17}$$

Trapez-Regel mit Restart Die Integrationsregel nach Gl. (17) eignet sich für ein iteratives Verfahren, bei dem wir die Integration schrittweise verfeinern.

Das teure bei der Auswertung von Gl. (17) ist im Allgemeinen die Auswertung der Funktion f. Von der wissen wir vielleicht gar nicht, was sie kostet. Die Idee der Verfeinerung ist dann, Funktionsauswertungen aus einem Iterationsschritt im folgenden wiederzuverwenden.

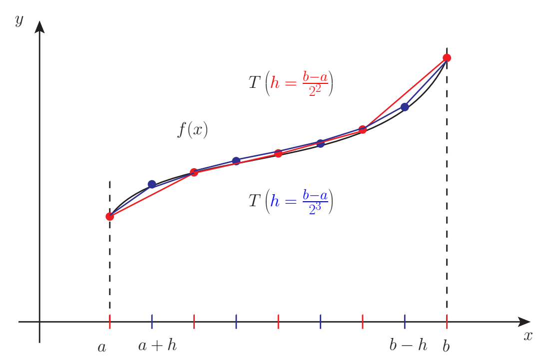
Dazu betrachten wir die Folge der Schrittweiten $h_m = (b-a)/2^m$, $m = 0,\, 1,\, \ldots$. In diesem Fall brauchen wir im Schritt von m nach $m+1$ lediglich die neuen Zwischenpunkte hinzufügen

$$\begin{aligned} T(h_{m+1}) &= h_{m+1} \Big(\frac{1}{2}\, f(a) + f(a+h_{m+1}) + f(a+2h_{m+1}) + \ldots \\ &\qquad \cdots + f(b-h_{m+1}) + \frac{1}{2}\, f(b)\Big) \qquad (18) \\ &= \frac{1}{2}\, T(h_m) + h_{m+1} \Big(f(a+h_{m+1}) + f(a+3h_{m+1}) + \ldots \\ &\qquad \cdots + f(b-3h_{m+1}) + f(b-h_{m+1})\Big)\,. \qquad (19) \end{aligned}$$

Wegen $h_m = 2\, h_{m+1}$ ist nämlich

$$T(h_m) = 2\, h_{m+1} \left(f(a)/2 + \sum_{k=1}^{2^{m+1}/2-1} f(a + k\, 2h_{m+1}) + f(b)/2\right). \tag{20}$$

Die Zwischenpunkte bei $a + (2k+1)\, h_{m+1}$ werden dann zusätzlich eingefügt. Dieses Vorgehen wird im folgenden Plot illustriert.



Auch andere Folgen von Schrittweiten $\{h_m\}$ können verwendet werden. Die hier betrachtete Folge ist die *Romberg-Folge*. Wir kommen auf sie bei der Anwendung der Extrapolation zurück.

Wir schauen uns den Integrationsfehler an, zunächst pro Teilintervall $[x_i,\, x_{i+1}]$. Dazu benutzen wir die Entwicklungen

$$\begin{aligned} f(x) &= f_i + \frac{f_{i+1} - f_i}{h}\, (x - x_i) + \mathcal{O}\left(h^2\right)\,, \\ &= f_{i+1} + \frac{f_{i+1} - f_i}{h}\, (x - x_{i+1}) + \mathcal{O}\left(h^2\right)\,, \qquad (21) \\ &= \frac{1}{2}\, (f_i + f_{i+1}) + \frac{f_{i+1} - f_i}{h}\, \left(x - (x_i + x_{i+1})/2\right) + \mathcal{O}\left(h^2\right)\,. \qquad (22) \end{aligned}$$

Durch Integration über das Teilintervall erhalten wir

$$\int_{x_i}^{x_{i+1}} f(x)\, dx \approx h\, \frac{1}{2}\, (f_i + f_{i+1}) + \mathcal{O}\left(h^3\right)\,, \tag{23}$$

und bei der Integration über alle Teilintervalle heißt das

$$T(h) = \sum_{i=0}^{n} \int_{x_i}^{x_{i+1}} f(x)\,dx + n\,\mathcal{O}\left(h^3\right) = \int_a^b f(x)\,dx + \mathcal{O}\left(h^2\right) \,, \tag{24}$$

wobei wir ausnutzen, dass $n\cdot h = (b-a)$, was die Ordnung h^0 ist.

The Simpsons

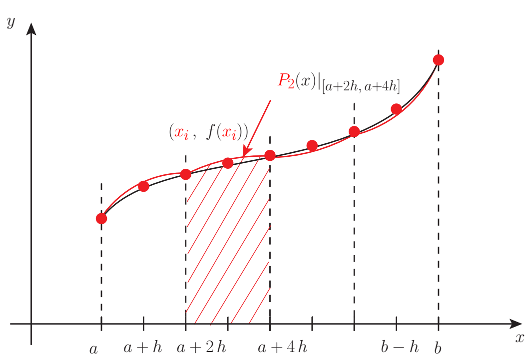
Ganz analog zur Trapez-Regel können wir auch eine *Simpson-Regel mit Restart* herleiten. Wir gehen dazu von einer **geraden Anzahl** von Teilintervallen aus. In jedem Teilintervall $[x_{2i},\, x_{2i+2}]$, das auch noch x_{2i+1} enthält, interpolieren wir f mit einem Polynom vom Grad 2 und setzen zusammen. Im i-ten der $n/2$ Teilintervalle heißt das

$$\begin{aligned} \mathcal{I}_i &= \frac{h}{3}\,\left(f(x_{2i}) + 4\,f(x_{2i+1}) + f(x_{2i+2})\right) \,, \\ i &= 0, \dots, n/2-1 \,. \end{aligned} \tag{25}$$

Anschließend summieren wir über alle Teilintervalle und erhalten

$$\begin{aligned} \int_a^b f(x)\,dx &= \sum_{i=0}^{n/2-1} \int_{x_{2i}}^{x_{2i+2}} f(x)\,dx \\ &\approx \sum_{i=0}^{n/2-1} \frac{h}{3}\,\left(f(x_{2i}) + 4\,f(x_{2i+1}) + f(x_{2i+2})\right) \\ &= \frac{b-a}{6}\Big(f(a) + 4\,f(a+h) + 2\,f(a+2h) + \dots \\ &\qquad \dots + 2\,f(b-2h) + 4\,f(b-h) + f(b)\Big) \\ &= S(h)\,. \end{aligned} \tag{26}$$

Die Regel funktioniert also analog zur Trapez-Regel für $T(h)$ mit den Koeffizienten aus dem Simpson-Verfahren.



Integrationsfehler bei Simpson Wir schauen uns wieder die Skalierung des Fehlers bei der Integration an, zunächst pro Teilintervall, dann für das gesamt Integrationsintervall (wir sammeln Fallbeispiele für die Skalierung mit h für später).

Im Teilintervall $[x_{2i},\, x_{2i+2}]$ haben wir mit $h = x_{2i+1} - x_{2i} = x_{2i+2} - x_{2i+1}$

$$\begin{aligned} P_2(x) &= f_{2i+1} + \frac{f_{2i+2} - f_{2i}}{2\,h}\,(x - x_{2i+1}) \\ &\quad + \frac{f_{2i+2} + f_{2i} - 2\,f_{2i+1}}{2\,h^2}\,(x - x_{2i+1})^2 \\ &= f(x) + \mathcal{O}\left(h^4\right) \end{aligned} \tag{27}$$

Achtung: Der Fehlerterm, der mit h^3 skaliert, $\propto (x - x_{2i+1})^3$ ist *antisymmetrisch* bezüglich x_{2i+1}. Wir integrieren aber von $x_{2i+1} - h$ bis $x_{2i+1} + h$. D.h. dieser Fehlerterm spielt keine Rolle und der führende relevante Fehlerterm ist der mit h^4. Das sehen wir einfach aus dem Vergleich mit der Taylor-Entwicklung um x_{2i+1}:

$$f(x) = f_{2i+1} + f_{2i+1}^{(1)}\,(x - x_{2i+1}) + \frac{1}{2}\,f_{2i+1}^{(2)}\,(x - x_{2i+1})^2 + \dots \,, \tag{28}$$

zusammen mit

$$\begin{aligned} \frac{f_{2i+2} - f_{2i}}{2\,h} &= f_{2i+1}^{(1)} + \mathcal{O}\left(h^2\right) \,, \\ \frac{f_{2i+2} + f_{2i} - 2\,f_{2i+1}}{h^2} &= f_{2i+1}^{(2)} + \mathcal{O}\left(h^2\right) \,. \end{aligned} \tag{29}$$

Durch Integration über $[x_{2i+1} - h,\, x_{2i+1} + h] = [x_{2i},\, x_{2i+2}]$ erhalten wir dann

$$\begin{aligned} \int_{x_{2i+1}-h}^{x_{2i+1}+h} P_2(x) &= \frac{h}{3}\,\left(f_{2i} + 4\,f_{2i+1} + f_{2i+2}\right) \\ &= \int_{x_{2i+1}-h}^{x_{2i+1}+h} f(x)\,dx + \mathcal{O}\left(h^5\right) \,. \end{aligned} \tag{30}$$

Über das gesamte Integrationsintervall hinweg reduziert sich bei Integration die Fehlerordung um 1

$$\begin{aligned} S(h) &= \sum_{i=0}^{n/2-1} \int_{x_{2i+1}-h}^{x_{2i+1}-h} f(x)\,dx + n\,\mathcal{O}\left(h^5\right) \\ &= \int_a^b f(x)\,dx + \mathcal{O}\left(h^4\right) \,. \end{aligned} \tag{31}$$

Beispiel-Code

Als Implementationsbeispiel nehmen wir das Programm `trapez.c`. Als kleine Neuerung führen wir hier *Zeiger auf eine Funktion* ein Dazu definieren wir uns einen Datentyp Zeiger-auf-Funktion `func_type`

```
typedef double (*func_type)(double const x,
                            void * p);
```

und später eine Funktion

```
double Gamma_integrand ( double x, void * p  )
{
  ...
}
```

die genau diesem Typ entspricht.

Der wirklich große Vorteil ist, dass wir die Integration dann ganz allgemein programmieren können, unabhängig von den Details der Funktion, die als Integrand dient.

Wir können also die Integration für jede Funktion dieses Typs aufrufen.

Wir benutzen hier die Trapez-Regel mit Restart, um die Integration für die Schrittweiten $h_m = h_0/2^m$ zu verfeinern. Der Kern ist dabei das Hinzufügen der neuen Zwischenpunkte in

```
int_val = 0.5 * ( int_val +
  trapez_integration_restarted(f, xa, xe, h, p));
```

innerhalb der Iteration über das Abbruch-Kriterium.

Integration mit Extrapolation.—

Wir haben uns den Fehler bei der Trapez- und Simpson-Regel angeschaut in Gl. (24) und (31) und gesehen, dass

$$\begin{aligned} \text{Trapez} \quad T(h) &= \int_a^b f(x)\,dx + \mathcal{O}\left(h^2\right)\,, \\ \text{Simpson} \quad S(h) &= \int_a^b f(x)\,dx + \mathcal{O}\left(h^4\right)\,. \end{aligned} \tag{32}$$

Tatsächlich gilt allgemein die *asymptotische Entwicklung*

$$T(h) = \tau_0 + \tau_1\,h^2 + \tau_2\,h^4 + \cdots + \tau_m\,h^{2m} + \alpha_{m+1}(h)\,h^{2m+2}\,. \tag{33}$$

τ_0 ist das gesuchte Integral.

Der Hintergrund hier ist eine wunderschöne Verbindung zur *Euler-Maclaurinschen Summenformel*: Für $g \in \mathcal{C}^{2m+2}([0,\,1])$, also g mindestens $(2m+2)$-mal stetig differenzierbar, gilt

$$\begin{aligned} \int_0^1 g(t)\,dt &= \frac{g(0)}{2} + \frac{g(1)}{2} \\ &+ \sum_{k=1}^{m} \frac{B_{2k}}{(2k)!}\left(g^{2k-1}(0) - g^{(2k-1)}(1)\right) \\ &+ \frac{B_{2m+2}}{(2m+2)!}\,g^{(2m+2)}(\xi)\,, \end{aligned} \tag{34}$$

mit $\xi \in (0,\,1)$ und den Bernoulli-Zahlen B_{2k}.

Die Formel (34) ist auch für unseren Fall anwendbar, weil wir immer die Variablentransformation $t \to x = x_i + t\,h$ durchführen können und $f(x) = f(x_i + t\cdot h) = g(t)$. Dann gilt

$$\int_{x_i}^{x_{i+1}} f(x)\,dx = h\int_0^1 f(x_i + th)\,dt \tag{35}$$

und $d^k\,g(t)/dt^k = h^k\,d^k f(x)/dx^k$.

Probeweise können wir tatsächlich mal die Taylor-Entwicklung für f nehmen und auf $[x_i,\,x_{i+1}]$ schreiben

$$\begin{aligned} f(x) = \frac{1}{2}\,(f_i + f_{i+1}) &+ \frac{1}{2}\left(f_i^{(1)}\,(x - x_i) - f_{i+1}^{(1)}\,(x - x_{i+1})\right) \\ &+ \frac{1}{4}\left(f_i^{(2)}\,(x - x_i)^2 + f_{i+1}^{(2)}\,(x - x_{i+1})^2\right) + \ldots\,. \end{aligned} \tag{36}$$

Das Integral ergibt dann

$$\begin{aligned} \int_{x_i}^{x_{i+1}} f(x)\,dx &= \frac{h}{2}\,(f_i + f_{i+1}) + \frac{h^2}{4}\left(f_i^{(1)} - f_{i+1}^{(1)}\right) \\ &+ \frac{h^3}{12}\left(f_i^{(2)} + f_{i+1}^{(2)}\right) + \ldots \end{aligned} \tag{37}$$

Wir benutzen folgende Umformungen von $f_i^{(1)}$ und $f_{i+1}^{(1)}$

$$\begin{aligned} f_i^{(1)} - f_{i+1}^{(1)} &= -f_i^{(2)}\,h + \mathcal{O}\left(h^2\right) \\ &= -f_{i+1}^{(2)}\,h + \mathcal{O}\left(h^2\right)\,. \end{aligned} \tag{38}$$

D.h. die zweiten Ableitungen können wir schreiben als

$$f_i^{(2)} + f_{i+1}^{(2)} = -\frac{2}{h}\left(f_i^{(1)} - f_{i+1}^{(1)}\right) + \mathcal{O}(h)\,. \tag{39}$$

Insgesamt haben wir also

$$\int_{x_i}^{x_{i+1}} f(x)\,dx = \frac{h}{2}\,(f_i + f_{i+1}) + \frac{h^2}{2}\,\frac{1}{6}\left(f_i^{(1)} - f_{i+1}^{(1)}\right) + \mathcal{O}\left(h^4\right)\,. \tag{40}$$

Mit $B_2 = 1/6$ stimmt der Term der 1. Ableitungen also mit dem $m = 1$-Term der Euler-Maclaurin-Formel überein.

Gl. (34) gibt also genau die asymptotische Entwicklung in Gl. (33).

Warum heißt diese Entwicklung "asymptotisch"? Sie kann für jedes $h > 0$ divergieren. Sie ist aber trotzdem nützlich, weil wir $\alpha_{m+1}(h)$ abschätzen können

$$\alpha_{m+1}(h) \leq M_{m+1} = \left| \frac{B_{2m+2}}{(2m+2)!} (b-a) \right| \max_{x \in [a,\, b]} |f^{(2m+2)}(x)| \,. \tag{41}$$

Das endliche Maximum existiert, weil f (analog g) aus $\mathcal{C}^{2m+2}([a,\, b])$ sein soll.

Mit anderen Worten das Restglied in Gl. (33) wird schon klein und kann für hinreichend kleines h vernachlässigt werden. In diesem Fall folgt aus der asymptotischen Entwicklung (ohne das Restglied), dass $T(h)$ ein Polynom in h^2 ist.

Dieses Polynom können wir bei $h^2 = 0$ auswerten, so wie wir es mit dem Algorithmus von Neville-Aitken gesehen haben, d.h. wir

1. bestimmen $T_{0,i} = T(h_i)$ für eine Reihe von $n+1$ Schrittweiten (= Stützwerten);

2. verwenden die Neville-Rekursion speziell mit $x = 0$ als Auswertungspunkt zur Berechnung von $T_{n,n}$.

Die so erhaltene Auswertung des Interpolationspolynoms in h^2 bei $h^2 = 0$ ist der Schätzwert für das Integral. Das ist die **Romberg-Integration**. Romberg hat speziell die Folge der Schrittweiten $h_m = h_0/2^m$ vorgeschlagen.

Beispiel-Code

Ein Implementationsbeispiel ist in `romberg.c` zu sehen. Im Unterschied zu `trapez.c` checken wir hier das Konvergenzkriterium an den extrapolierten Werten aus Funktion `neville`.

Zum Abschluss.—

Wie immer Dank im Voraus für Hinweise und Kommentare.

Momentan geht nichts über die Lehrbücher [1] und [2].

[1] R. Freund and R. Hoppe, *Stoer/Bulirsch: Numerische Mathematik 1*, Springer-Lehrbuch (Springer Berlin Heidelberg, 2007).

[2] H. Schwarz and N. Köckler, *Numerische Mathematik* (Vieweg+Teubner Verlag, 2011).