



KERNFORSCHUNGSANLAGE JÜLICH
GESELLSCHAFT MIT BESCHRÄNKTER HAFTUNG
Zentralinstitut für Angewandte Mathematik

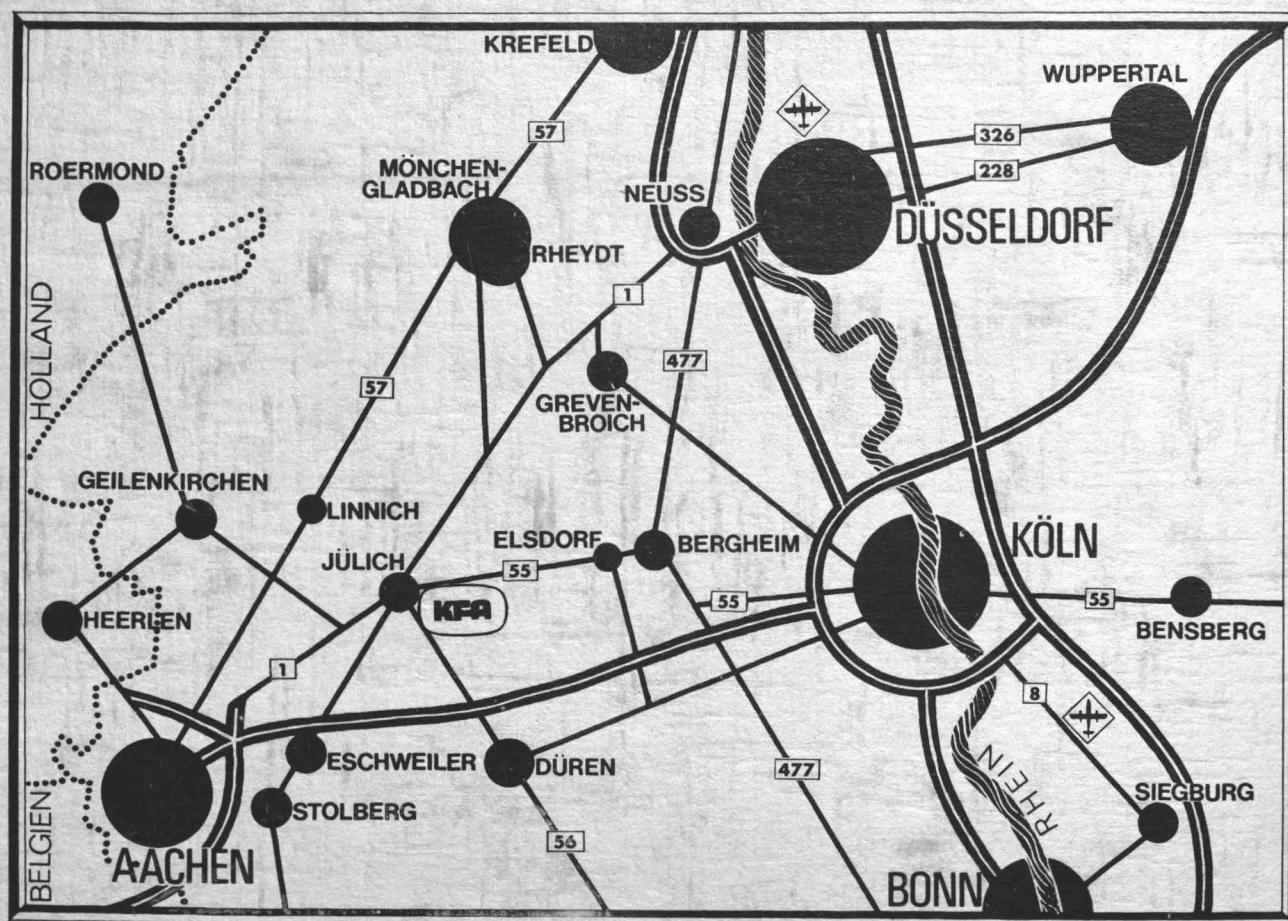
Symmetrische WILF'sche Quadraturformeln

von

H. Engels und U. Eckhardt

Jül - 1094 - MA
August 1974

Als Manuskript gedruckt



Berichte der Kernforschungsanlage Jülich - Nr. 1094

Zentralinstitut für Angewandte Mathematik Jül - 1094 - MA

Dok.: Numerical Integration

WILF Quadrature Formula - Computer Calculation

Im Tausch zu beziehen durch: ZENTRALBIBLIOTHEK der Kernforschungsanlage Jülich GmbH,
Jülich, Bundesrepublik Deutschland

Symmetrische WILF'sche Quadraturformeln

von

H. Engels* und U. Eckhardt

Zusammenfassung. Unter Verwendung einer neuen Charakterisierung WILF'scher Quadraturformeln von ENGELS gelang es, diese Formeln für das Intervall $[-1, 1]$ mit bis zu 32 Stützstellen zu berechnen mit 25 zuverlässigen dezimalen Ziffern.

In der vorliegenden Arbeit wird dargestellt, wie diese Formeln berechnet wurden. Weiterhin werden die berechneten Knoten und Gewichte sowie Programme zur WILF-Quadratur angegeben.

Summary. Using a new characterization for WILF quadrature formulae given by ENGELS, these formulae are computed for the interval $[-1, 1]$ up to 32 nodes with 25 significant digits.

In this paper it is described how to perform the calculation of nodes and weights of these formulae. In addition, nodes and weights as well as FORTRAN-subroutines for WILF-quadrature are listed, which can be implemented immediately.

Wir danken Frau E. Schöller für die sorgfältige Ausführung der Reinschrift dieser Arbeit.

1. EINLEITUNG UND ÜBERBLICK

In [6] leitete H. S. WILF ein Quadraturverfahren her aus der Forderung

$$Q_n := \sum_{v=0}^{\infty} \left[\int_0^1 x^v dx - \sum_{i=1}^n A_i x_i^v \right]^2 \rightarrow \text{Min.} \quad (1.1)$$

Dabei sind x_i , $i = 1(1)n$, die Stützstellen und A_i , $i = 1(1)n$, die Gewichte einer Mittelwertformel der Gestalt

$$\int_0^1 f(x) dx = \sum_{i=1}^n A_i f(x_i) + E_n f. \quad (1.2)$$

H.S. WILF versuchte, das aus den notwendigen Minimumsbedingungen für (1.1) folgende nichtlineare Gleichungssystem

$$\frac{\partial Q_n}{\partial A_i} = 0, \quad \frac{\partial Q_n}{\partial x_i} = 0, \quad i = 1(1)n, \quad n = 1, 2, \dots \quad (1.3)$$

zu lösen. Dieses System ist aber numerisch derart schlecht konditioniert, daß man nur für kleine Werte von n zu halbwegs zuverlässigen Lösungen kommen kann.

Der wesentliche Wert der von H.S. WILF vorgeschlagenen Quadraturmethode liegt in der Existenz von einfachen optimalen ableitungsfreien Fehlerschranken für Integranden $f(z)$, die im Inneren des Einheitskreises $|z| < 1$ analytisch und auf dem Rande $|z| = 1$ quadratisch integrierbar sind. H. S. WILF gab mit Majorisierungsüberlegungen einen ersten Beweis für die Konvergenz

$$\lim_{n \rightarrow \infty} Q_n = 0 \quad (1.4)$$

des Quadraturverfahrens.

N. RICHTER zeigte in [4] und [5], daß die Stützstellen der WILF-Quadratur stets reell, paarweise verschieden und in $(0,1)$ gelegen sind (der in [1] angegebene Beweis enthält einen Fehler). Weiterhin zeigte U. ECKHARDT in [1], daß die Gewichte A_i für $i = 1(1)n$, $n = 1, 2, \dots$ stets positiv sind und daß Q_n mit n monoton gegen Null strebt.

Interessant ist, daß bei einer WILF'schen Quadraturformel i.a.

$$\sum_{i=1}^n A_i < 1 \quad (1.5)$$

gilt (vgl. [1]), d.h. daß die Funktion $f(x) = 1$ nicht exakt integriert

wird im Gegensatz zu den auf Polynomgenauigkeit fußenden Quadraturverfahren wie die Verfahren von NEWTON-COTES, GAUSS, TSCHEBYSCHEFF, ROMBERG etc. .

In [2] erbrachte H. ENGELS den Nachweis, daß das WILF'sche Quadraturverfahren durch Integration eines geeignet gewählten rationalen HERMITE'schen Interpolationsoperators erzeugt werden kann. Diese überraschende Eigenschaft eröffnet zugleich neue Wege zur Berechnung der Stützstellen, Gewichte und Fehlerkoeffizienten der WILF'schen Quadraturformeln. Es gelang damit, für $n = 2(1)32$ Stützstellen und Gewichte mit 25 zuverlässigen Dezimalstellen zu berechnen. Bei den Fehlerkonstanten reicht eine geringere Genauigkeit für praktische Zwecke vollkommen aus.

Anders als bei der WILF'schen Arbeit [6] wurden die Formeln für das Intervall $[-1, 1]$ anstelle von $[0, 1]$ bestimmt. Dabei ergaben sich durchweg symmetrische Quadraturformeln. Der allgemeine Nachweis der Symmetrie ist bisher noch nicht erbracht worden.

Die nunmehr bekannten Quadraturformeln wurden in einer Reihe von Beispielen mit den GAUSS'schen Quadraturformeln hinsichtlich ihrer Güte verglichen. Sie erwiesen sich dabei den in Unterprogrammen sehr häufig verwendeten GAUSS'schen Quadraturformeln zumindest als ebenbürtig. Liegt außerhalb des Integrationsintervalles im Endlichen eine Singularität oder auf dem Rande eine wesentliche Singularität des Integranden, so sind sie i.a. besser als die GAUSS'schen Formeln.

2. ERZEUGUNG DER WILF-QUADRATUR DURCH INTERPOLATION

Wie in [2] gezeigt wurde, ist der Interpolationsoperator

$$H_n(f; x) = \sum_{i=1}^n \left\{ \left[1 - 2\lambda_i'(x_i) \frac{h_i(x)}{h_i'(x_i)} \right] f(x_i) + \frac{h_i(x)}{h_i'(x_i)} f'(x_i) \right\} \lambda_i^2(x) \quad (2.1)$$

ein HERMITE'scher Interpolationsoperator mit den Eigenschaften (vgl. [2] und [3]):

$$H^{(r)}(f; x_i) = f^{(r)}(x_i), \quad r = 0, 1, \quad i = 1(1)n, \quad (2.2)$$

und für

$$h_i(x) = x - x_i, \quad i = 1(1)n, \quad (2.3)$$

$$\lambda_i(x) = \frac{\omega_n(x) h_i'(x_i)}{\omega_n(x_i) h_i'(x)} \quad i = 1(1)n \quad (2.4)$$

$$\omega_n(x) = \prod_{i=1}^n \frac{x - x_i}{1 - xx_i} \quad (2.5)$$

gilt

$$H\left(\frac{1}{1 - xx_i}; x\right) = \frac{1}{1 - xx_i}, \quad i = 1(1)n, \quad (2.6)$$

$$H\left(\frac{x}{(1 - xx_i)^2}; x\right) = \frac{x}{(1 - xx_i)^2} \quad i = 1(1)n. \quad (2.7)$$

Die beiden letzten Eigenschaften ermöglichen die Darstellung des WILF'schen Quadraturverfahrens im Intervall $[-1, 1]$ als

$$\int_{-1}^1 f(x) dx = \int_{-1}^1 H_n(f; x) dx + E_n f \quad (2.8)$$

(vgl. (1.2)). Dazu muß natürlich notwendigerweise gelten

$$\int_{-1}^1 (x - x_i) \lambda_i^2(x) dx = 0, \quad i = 1(1)n. \quad (2.9)$$

Dies ist ein gekoppeltes nichtlineares Gleichungssystem für die Stützstellen x_i , $i = 1(1)n$. Ist dieses System gelöst, dann erhält man die Gewichte entweder aus

$$A_i = \int_{-1}^1 \lambda_i^2(x) dx \quad (2.10)$$

durch Integration oder als Lösungen des linearen Gleichungssystems

$$\sum_{i=1}^n \frac{A_i}{1 - x_i x_k} = \int_{-1}^1 \frac{dx}{1 - xx_k}, \quad k = 1(1)n. \quad (2.11)$$

Die Berechnung von Stützstellen und Gewichten ist also entkoppelt und somit einfacher geworden. erinnert man sich nun der Tatsache, daß die GAUSS'schen Quadraturen durch Integration formal desselben HERMITE-Operators $H_n(f; x)$, jedoch mit $\omega_n(x) = \prod_{i=1}^n (x - x_i)$ erzeugt werden können, wobei x_i , $i = 1(1)n$,

die n Nullstellen des n -ten LEGENDRE-Polynoms sind, so bietet sich für die numerische Berechnung der Stützstellen x_i , $i = 1(1)n$, der WILF-Quadratur ein Einbettungsprozeß folgender Art an:

Man setzt

$$\omega_n(x; \epsilon) = \prod_{i=1}^n \frac{x - x_i}{1 - \epsilon x x_i} . \quad (2.12)$$

Für $\epsilon = 0$ erhält man dann die GAUSS-Quadratur und für $\epsilon = 1$ die WILF-Quadratur. Die GAUSS'schen Stützstellen kann man mit hinreichend hoher Genauigkeit aus Tafeln entnehmen und so einen Einbettungsprozeß starten, indem man ϵ geeignete Werte von Null bis Eins durchlaufen läßt. Damit erzeugt man eine von ϵ abhängige Schar von Quadraturformeln, die nicht nur rechentechnische Bedeutung hat. Indem man ϵ von Null zu Eins ändert, läßt man Pole des Integranden außerhalb $[-\frac{1}{\epsilon}, \frac{1}{\epsilon}]$ zu. Man holt also die Pole des Integrationsoperators schrittweise an den Rand von $[-1, 1]$ heran.

3. NUMERISCHE BERECHNUNG DER WILF'SCHEN KNOTEN

Für das Intervall $[-1, 1]$ erhält man aus (2.9) das nichtlineare Gleichungssystem

$$\phi_k(x_1, \dots, x_n) := \int_{-1}^1 \frac{1}{x - x_k} \prod_{j=1}^n \frac{(x - x_j)^2}{(1 - \epsilon x x_j)^2} dx = 0, \quad k = 1(1)n \quad (3.1)$$

zur Bestimmung der Knoten x_i . Das Integral in (3.1) kann geschlossen ausgewertet werden. Durch partielle Integration erhält man die Darstellung

$$\phi_k = \sum_{i=1}^n \sum_{j=1}^n A_i^{(k)} A_j^{(k)} \psi_{ij}^{(k)}, \quad (3.2)$$

wobei

$$A_i^{(k)} = \epsilon^{-n+1} \frac{\prod_{v \neq k} (1 - \epsilon x_i x_v)}{\prod_{v \neq i} (x_i - x_v)}, \quad (3.3)$$

$$\psi_{ij}^{(k)} = \int_{-1}^1 \frac{x - x_k}{(1 - \epsilon x x_i)(1 - \epsilon x x_j)} dx = -a_{ij} + \epsilon x_k b_{ij}, \quad (3.4)$$

mit

$$a_{ij} = - \int_{-1}^1 \frac{x dx}{(1 - \epsilon x x_i)(1 - \epsilon x x_j)} \quad (3.5)$$

$$b_{ij} = - \frac{1}{\epsilon} \int_{-1}^1 \frac{dx}{(1 - \epsilon x x_i)(1 - \epsilon x x_j)}. \quad (3.6)$$

Die letzten beiden Integrale sind geschlossen auswertbar, wobei man gegebenenfalls die Fälle $x_i = x_j$ und $x_i = 0$ gesondert behandeln muß.

Aus der Darstellung (3.1-6) kann man ebenfalls leicht die Größen $\partial\phi_k/\partial x_\alpha$ ausrechnen. Auch hierbei ist eine Reihe von Fallunterscheidungen zu treffen, die Fälle $\alpha = k$, $x_i = x_j$ und $x_i = 0$ sind wieder gesondert zu behandeln. Durch diese vielen Fallunterscheidungen ergibt sich, daß ein Programm, das die Größen ϕ_k und $\partial\phi_k/\partial x_\alpha$ berechnet, ziemlich komplex sein muß.

Eine beträchtliche Ersparnis an Rechenzeit und Programmieraufwand ergibt sich, wenn man sich auf symmetrische Knoten beschränkt, d.h. man setzt voraus

$$x_j = -x_{n-j+1}, \quad j = 1(1)n. \quad (3.7)$$

Zunächst wurde jedoch ohne diese zusätzliche Annahme gerechnet, um gegebenenfalls auch asymmetrisch liegende Knoten nicht von vornherein auszuschließen.

Setzt man auf das nichtlineare Gleichungssystem (3.1) das NEWTON'sche Verfahren an, dann stellt man fest, daß der Einzugsbereich dieses Verfahrens sehr klein ist. Zunächst ist also zu überlegen, wie man sich Startwerte für das NEWTON-Verfahren beschafft. Für kleine Werte von n (d.h. $n \leq 10$) wurde dazu das bereits erwähnte Einbettungsverfahren verwendet. Ausgehend von den bekannten GAUSS'schen Knoten als Lösungen des Gleichungssystems (3.1) für $\epsilon = 0$ wurde für eine Folge $\{\epsilon_k\}$ mit $\lim \epsilon_k = 1$ eine Folge von Lösungen $\{x_j(\epsilon_k)\}$ berechnet. Aus diesen $x_j(\epsilon_k)$ wurde durch Extrapolation ein Startwert für die Iteration mit $\epsilon = 1$ ermittelt. Es ergeben sich dabei zwei Schwierigkeiten. Einmal wird das Gleichungssystem (3.1) beim Übergang von $\epsilon = 0$ zu $\epsilon \neq 0$ singulär, was man am deutlichsten an den beiden Integralen (3.5) und (3.6) sieht, die bei $\epsilon = 0$ völlig harmlose algebraische Integrale sind, für $\epsilon > 0$ jedoch logarithmische Lösungen aufweisen. Zum anderen kann es bei $\epsilon = 1$ zu Schwierigkeiten kommen, wenn ein x_j gegen ± 1 rückt, wie das auch tatsächlich bei den WILF'schen Quadraturformeln der Fall ist. Bei der Auswertung der Funktionen ϕ_k ergeben sich demgemäß Anteile sehr verschiedener Größenordnung, die zu erheblichen Genauigkeitsverlusten führen.

Für hinreichend große n ergab sich experimentell, daß das folgende einfache Verfahren sehr gute Startwerte für die NEWTON-Iteration liefert: Es seien

die $x_j^{(n)}$ bereits bekannt, $-1 < x_1^{(n)} < x_2^{(n)} < x_3^{(n)} < \dots < x_n^{(n)} < 1$. Damit setzt man

$$y_j = \frac{1}{2} (x_{j-1}^{(n)} + x_j^{(n)}), \quad j = 2, 3, \dots, n,$$

$$y_1 = \frac{1}{2} (-1 + x_1^{(n)}),$$

$$y_{n+1} = \frac{1}{2} (1 + x_n^{(n)}) .$$

Die y_j , $j = 1(1)n+1$, unterscheiden sich von den $x_j^{(n+1)}$ mit wachsendem n immer weniger und liefern vorzügliche Startwerte.

4. GENAUIGKEITSBETRACHTUNGEN

Die bisher publizierten Zahlenwerte für WILF'sche Quadraturformeln in [6] (bis $n = 3$) und [1] (bis $n = 5$) für das Intervall $[0, 1]$ wurden mit sehr schlecht konvergierenden Verfahren direkt aus den Gleichungen (1.3) gewonnen. Dabei wirkte sich verhängnisvoll aus, daß beide Sätze von Gleichungen sehr schlecht konditioniert sind und daß man daher erhebliche Genauigkeitsverluste in Kauf nehmen mußte. Bei dem hier gewählten NEWTON'schen Verfahren hat man quadratische Konvergenz, infolgedessen geben die Defekte des Gleichungssystems einen wesentlich besseren Anhalt über den Fehler in den Knoten. Hinzu kommt, daß zur endgültigen Berechnung der Knoten nur wenige Iterationen benötigt wurden. Um aus 16-stelligen Näherungswerten die definitiven 33-stelligen Knoten zu berechnen, wurden weniger als 10 Iterationen gebraucht. Für $n = 30$ kann man bei einer Darstellungsgenauigkeit von 33 Dezimalstellen die ϕ_k (vgl. (3.1)) auf die Größenordnung 10^{-25} reduzieren. Die gleiche Größenordnung haben die Korrekturen, die an den x_j noch anzubringen sind. Aus diesen - und anderen - Überlegungen ist es gerechtfertigt anzunehmen, daß die Knoten auf etwa 25 Dezimalstellen bekannt sind. Der erhebliche Genauigkeitsverlust von 33 auf 25 Stellen ist auf Auslöschung zurückzuführen, wenn bei der Berechnung der ϕ_k und der $\partial\phi_k/\partial x_\alpha$ annähernd gleiche Größen voneinander subtrahiert werden.

Die Gewichte A_j wurden aus (2.10) errechnet. Die dort auftretenden Integrale sind geschlossen auswertbar. Der Fehler der Gewichte dürfte in der gleichen Größenordnung liegen, wie der der Knoten.

Sämtliche Rechnungen wurden auf der IBM 370/165 des Zentralinstituts für Angewandte Mathematik der Kernforschungsanlage Jülich durchgeführt. Alle Programme wurden in PL1 geschrieben, was sich insbesondere bei den zahlreichen Fallunterscheidungen bei der Berechnung der ϕ_k und der $\partial\phi_k/\partial x_\alpha$ als sehr vorteilhaft erwies. Für die mühsame Arbeit der Erstellung der Programme möchten wir Frau M. LISKEN herzlich danken.

Eine ausführliche Beschreibung der Programme zur Erstellung von WILF'schen Quadraturformeln wird demnächst in der Reihe der Berichte der KFA Jülich erscheinen. Die dabei verwendeten Methoden eignen sich in hervorragender Weise zur Berechnung von neuen Quadraturformeln aus bereits bekannten, etwa LOBATTO- und RADAU-Formeln WILF'schen Typs aus GAUSS- und WILF-Formeln.

5. DIE PROGRAMME

Um einem möglichst großen Benutzerkreis entgegenzukommen, wurden die Programme zur numerischen Quadratur mit WILF-Formeln in FORTRAN geschrieben. Sie werden unter dem Namen DWQN (= Double-precision WILF Quadrature with N nodes) aufgerufen, wobei N eine der Zahlen 2, 4, 8, 16, 24 oder 32 ist. Das Programm führt dann eine WILF-Quadratur mit N Knoten in doppelter Genauigkeit (d.h. 16 Dezimalstellen) aus. Das allgemeine Format des Programmaufrufs von DWQN zur Berechnung von

$$Y = \int_{XA}^{XB} FCT(x)dx$$

sieht für N = 32 folgendermaßen aus:

```
CALL DWQ32(XA,XB,FCT,Y).
```

Dabei ist

- XA - die untere Integrationsgrenze,
- XB - die obere Integrationsgrenze,
- FCT - Name eines externen Unterprogrammes zur Berechnung des Integranden,

Y - Rückgabeparameter, der den Wert des Integrals enthält.

Die Programme DWQN für die oben erwähnten Werte von N findet man in der Tabelle III aufgelistet.

Programme für weitere Werte von N = 2(1)32 sind auf Anfrage beim Zentralinstitut für Angewandte Mathematik der KFA Jülich erhältlich.

6. NUMERISCHE BEISPIELE

Durch den Einsatz der modernen Großrechner wächst naturgemäß das Interesse an Quadraturformeln mit Optimalitätscharakter. Am deutlichsten zeigt sich dieser Trend bei den GAUSS'schen Quadraturformeln. Während diese Formeln noch vor einem Jahrzehnt als numerisch uninteressant galten wegen der irrationalen Stützstellen und Gewichte, bereitet es heute keine Schwierigkeiten, diese Formeln etwa auf einer Platte abzuspeichern und bei Bedarf abzurufen. So stehen dem Benutzer des Zentralinstituts für Angewandte Mathematik der KFA Jülich beispielsweise GAUSS-Formeln mit bis zu 80 Stützstellen zur Verfügung. In der Mehrzahl aller vorkommenden Fälle sind diese Quadraturformeln allen anderen Formeln weit überlegen, insbesondere den häufig benutzten ROMBERG-Verfahren und den adaptiven Quadraturverfahren.

Die WILF'schen Quadraturformeln sind nicht als Konkurrenz sondern als Ergänzung der GAUSS'schen Formeln zu verstehen. Generell läßt sich sagen, daß sie gerade dort, wo die GAUSS'schen Formeln versagen, gute Resultate liefern und umgekehrt. So quadrieren sie beispielsweise ein Polynom nicht exakt, andererseits liefern sie gute Näherungswerte und bieten die Möglichkeit einer Fehlerabschätzung bei Funktionen mit wesentlichen Singularitäten auf dem Rande des Integrationsintervalls bzw. mit Polen in der Nähe desselben.

Aus der Fülle von Beispielen, die mit den vorliegenden Formeln gerechnet wurden, seien hier nur einige typische herausgegriffen. Es wird jeweils nur der tatsächliche Fehler angegeben, eine Fehlerabschätzung ist in den meisten Fällen praktisch nicht möglich. Der angegebene Fehler bezieht sich auf den Fall $n = 16$. ϵ_G ist der Fehler der GAUSS-Quadratur, ϵ_W der der WILF-Quadratur (jeweils wahrer Wert-Näherung). Die ersten vier Funktionen sind typische Anwendungsbeispiele für WILF'sche Quadraturformeln. Bei der Funktion 1 liegt in der Nähe des Integrationsintervalles ein Pol des Integranden. Die zweite Funktion steht als Beispiel für eine Funktion, bei der es nicht möglich ist, eine klassische Fehlerabschätzung für die GAUSS'schen Quadraturformeln zu finden, da sämtliche Ableitungen dieser Funktion im Integrationsintervall unbeschränkt sind. Wie man sieht, ist auch der tatsächliche Fehler der GAUSS-Quadratur sehr groß. Ähnliches gilt für die Funktionen 3 und 4. Bei der fünften Funktion versagen nahezu alle

gewöhnlich benutzten Quadraturmethoden; nur Quadraturformeln sehr hoher Ordnung liefern noch brauchbare Ergebnisse. Die siebte Funktion hat keinerlei Komplikationen in der Nähe des Integrationsintervalles, demgemäß ist hier der Fehler der GAUSS-Quadratur sehr klein.

	Funktion	XA	XB	ϵ_G	ϵ_W	$\int_{XA}^{XB} f(x)dx$
1	$\frac{1}{1.01 - x}$	- 1	1	5.3 E-2	-5.3 E-5	$\ln 201 = 5.303\ 305$
2	$\frac{1}{x} \log \frac{1}{1 - x}$	0	1	-2.4 E-3	-3.2 E-6	$\frac{\pi^2}{6} = 1.644\ 934$
3	$\sqrt{x}$	0	1	-2.3 E-5	-7.6 E-7	$\frac{2}{3} = 0.666\ 667$
4	$\frac{1}{\sqrt{3 - x^2 - 2x}}$	- 1	1	3.8 E-2	-8.7 E-4	$\frac{\pi}{2} = 1.570\ 796$
5	e^{x^2}	0	12	2.2 E+60	-1.3 E+59	$1.444\ 493 \cdot 10^{61}$
6	e^{-10x^2}	- 1	1	$ \epsilon_G < 10^{-6}$	7.0 E-2	$\frac{2}{\sqrt{10}} \operatorname{erf}(\sqrt{10}) = 0.560495$

Tabelle 1

Vergleich von WILF- und GAUSS-Quadratur an Hand repräsentativer Beispiele unter Verwendung von je 16 Stützstellen.

ϵ_G = tatsächlicher Fehler der GAUSS-Quadratur

ϵ_W = tatsächlicher Fehler der WILF-Quadratur.

7. LITERATUR

- [1] ECKHARDT, U.: Einige Eigenschaften WILF'scher Quadraturformeln.
Numer. Math. 12 (1968), 1 - 7
- [2] ENGELS, H.: Über allgemeine GAUSS'sche Quadraturformeln.
Computing 10 (1972), 83 - 95
- [3] ENGELS, H.: Über ein numerisches Differentiationsverfahren mit
ableitungsfreien Fehlerschranken.
ZAMM 53 (1973), T184 - T185
- [4] RICHTER, N.: Properties of minimal integration rules.
SIAM J.Numer.Anal. 7 (1970), 67 - 79
- [5] RICHTER-DYN, N.: Properties of minimal integration rules. II.
SIAM J.Numer.Anal. 8 (1971), 497 - 508
- [6] WILF, H.S.: Exactness conditions in numerical quadrature.
Numer.Math. 6 (1964), 315 - 319

TABELLENANHANG

Tabelle I: Knoten und Gewichte der WILF'schen Quadraturformeln in $[-1, 1]$ für $N = 2(1)32$.

Die angegebenen Zahlenwerte sind jeweils symmetrisch zu ergänzen gemäß

$$\begin{aligned} x_j &= -x_{n-j+1} \\ A_j &= A_{n-j+1} \end{aligned} \quad j = 1(1)n$$

N	KNOTEN	GEWICHTE
=====		
2	0.6657426959532128630253806Q+00	0.9693032530017216145137782Q+00
3	0.0 0.8906556100572918953941645Q+00	0.1292986156109711031790707Q+01 0.3535069219451444841046465Q+00
4	0.5196924295935199844885557Q+00 0.9613969894636536457941923Q+00	0.8677961778941937087229099Q+00 0.1291489671916942736601938Q+00
5	0.0 0.7887055153023355306662715Q+00 0.9851676175475876436825624Q+00	0.1019923692924197284027916Q+01 0.4398335767103668170695296Q+00 0.5020457682753454091651266Q-01
6	0.4393148495835771415841974Q+00 0.9075474419210293401953670Q+00 0.9938673727789520981913882Q+00	0.7744695919512921426312676Q+00 0.2042056378792233242366825Q+00 0.2084723076612103689872333Q-01
7	0.0 0.7120745296896912731424911Q+00 0.9583973770896035725352129Q+00 0.9973054296661901519944826Q+00	0.8666436172887705955367483Q+00 0.4632737263452569341396498Q+00 0.9422888040766576723345534Q-01 0.9175584602692000858520676Q-02
8	0.3869563898972155989351321Q+00 0.8564573816129342964880581Q+00 0.9805731994455208112091703Q+00 0.9987547139516974592424840Q+00	0.7023586023192392565063217Q+00 0.2488142756043624934444051Q+00 0.4448686109231948984854121Q-01 0.4243528406812166059278885Q-02
9	0.0 0.6530926108715721072293613Q+00 0.9282180961131661795210790Q+00 0.9905882050694107499270129Q+00 0.9993995067921927817118970Q+00	0.7656449416486000019602281Q+00 0.4644844709029082638915230Q+00 0.1289859113086632580047249Q+00 0.2166020334911756547174353Q-01 0.2046943615010911651894526Q-02
10	0.3494719456049637084753347Q+00 0.8109562407396082228170254Q+00 0.9634686073716511698480446Q+00 0.9952834326072481689629024Q+00 0.9996997323955610916373859Q+00	0.6459899935977323761645272Q+00 0.2752771114701466022620744Q+00 0.6680582907626818103694054Q-01 0.1088005546392234534707868Q-01 0.1023696915353835508394606Q-02
11	0.0 0.6062046023550560563003508Q+00 0.8979452947425687260370460Q+00 0.9809852732969171503950318Q+00 0.9975636957215444468892321Q+00 0.9998450645530750604920967Q+00	0.6928207913952051059032775Q+00 0.4570068918451044135697977Q+00 0.1553511202828300820377880Q+00 0.3507693830217044789627147Q-01 0.5626396888291760094430171Q-02 0.5282569840007434500739329Q-03
12	0.3209693628253772934850892Q+00 0.7709614126041423344365902Q+00 0.9445735459167672447865464Q+00 0.9898702960108257646685493Q+00 0.9987071262784312936899058Q+00 0.9999178243740173157722109Q+00	0.6007497022196626811955779Q+00 0.2908366676332039888471787Q+00 0.8637024437445624874621248Q-01 0.1876935085160555948900692Q-01 0.2987464894561994723152238Q-02 0.2801904149637049987636289Q-03
13	0.0 0.5678864400080786163026054Q+00 0.8689038606755875221811676Q+00 0.9694976847112796305756214Q+00 0.9944822830975739406687180Q+00 0.9992971776460274428393318Q+00 0.9999553406346841009500769Q+00	0.6371788346164610111132589Q+00 0.4461116766280128365585143Q+00 0.1751483724947207629340447Q+00 0.4812643811344222516774317Q-01 0.1024731759526147080701148Q-01 0.1624502040870176323207766Q-02 0.1522758194620226528490939Q-03

14	0.2983709622306316859942054Q+00 0.7357785294874503064444873Q+00 0.9250341246129367087361047Q+00 0.9829441106953378294656992Q+00 0.9969316603722098397386398Q+00 0.9996095857045078471457787Q+00 0.9999751955739828222653171Q+00	0.5635410851747098177168816Q+00 0.2996516877285851934325335Q+00 0.1030210696910099579538889Q+00 0.2709172678467524244182392Q-01 0.5705384466910356500529013Q-02 0.9025436208342158990191120Q-03 0.8457692502078130497838774Q-04
15	0.0 0.5358600383975181570230233Q+00 0.841573317337192+348533752Q+00 0.9568609505760809136744838Q+00 0.9903039949687174550618287Q+00 0.9982609569036616383893844Q+00 0.9997788531195108633936372Q+00 0.9999859508556993348193752Q+00	0.5929095722051494173904626Q+00 0.4340582743597685490053740Q+00 0.1899807983759764751748584Q+00 0.6025332369271969191359401Q-01 0.1545784368118765432431847Q-01 0.3235787977395653488610256Q-02 0.5112814640786685345147739Q-03 0.4790434629859886349886132Q-04
16	0.2798945461029496332338835Q+00 0.7046662121588820526298901Q+00 0.9055729097579846859984814Q+00 0.9749018274890479836470871Q+00 0.9943984346521004615858563Q+00 0.9989970083917202437760505Q+00 0.9998724946776038177380757Q+00 0.9999919001247658954773624Q+00	0.5323003879744999245997492Q+00 0.3041768950494972666254233Q+00 0.1170114437536124065566814Q+00 0.3537275244553073098160388Q-01 0.8948548711238051097624603Q-02 0.1866924205669518128044345Q-02 0.2948002674546472926850877Q-03 0.2761879115644921579114269Q-04
17	0.0 0.5085981547900684522774643Q+00 0.8160859184705346492707751Q+00 0.9435366385042969562247740Q+00 0.9852175834771434902399218Q+00 0.996713994853646768840822Q+00 0.9994121812378502438265299Q+00 0.9999252870672143797838967Q+00 0.9999952539198792903872432Q+00	0.5566204492180594427989671Q+00 0.4218685502426800144890111Q+00 0.2010802992555656722913703Q+00 0.7126059513732600334549132Q-01 0.2094154274353155132254672Q-01 0.5255492528273435435060715Q-02 0.1094366843118136297373737Q-02 0.1727455250962756415503374Q-03 0.1618311537918977811226057Q-04
18	0.2644283589587128192277094Q+00 0.6769733495159981087232465Q+00 0.8865008408368036475878644Q+00 0.9660673865089973893344284Q+00 0.9911825708046157308799406Q+00 0.9980445061261830888312633Q+00 0.9996503814277631025003331Q+00 0.9999555674034689445511708Q+00 0.9999971774943713832919407Q+00	0.5056172406747329182053235Q+00 0.3059116920130372859458339Q+00 0.1287001745303020123295959Q+00 0.4334925217700991640658163Q-01 0.1252847511365927769756740Q-01 0.3129609972356877735702404Q-02 0.6509770530794967385669600Q-03 0.1027352043490994075218910Q-03 0.9624148142366766278775741Q-05
19	0.0 0.4850398464244690+41306312Q+00 0.7923828710412464+36225472Q+00 0.9300241527609628435717408Q+00 0.9794048683671869734224944Q+00 0.9946748115956882498734921Q+00 0.9988206053631765304251026Q+00 0.9997892057211032707007659Q+00 0.9999732120359943403133946Q+00 0.9999982983539015375930281Q+00	0.5261817622877983548447491Q+00 0.4100110046601272932323805Q+00 0.2093606695203749977500683Q+00 0.8112272050348481069588474Q-01 0.2648657065955440234246863Q-01 0.7579636357476125037768768Q-02 0.1888257823773491757294528Q-02 0.3925185099772092254924194Q-03 0.6193856423529956697173025Q-04 0.5802257097192969295832758Q-05

20	0.2512386698053110680245893Q+00 0.6521594686929911644738034Q+00 0.8630387015269199813387868Q+00 0.9566932715160179386480505Q+00 0.9873751959115050644932793Q+00 0.9957453162411685884006372Q+00 0.9992797505285404695238369Q+00 0.9998712934193178949778437Q+00 0.9999836444150495744165122Q+00 0.9999989610527562306655306Q+00	0.4824988857990555270989005Q+00 0.3058016639024842375878484Q+00 0.1384423934339318940655502Q+00 0.5088413521930350899887093Q-01 0.1630103042140104333129847Q-01 0.4637364586982534323609701Q-02 0.1153412720761432564928137Q-02 0.2396734328872703806565277Q-03 0.3781723511562193655927798Q-04 0.3542593857594843869755070Q-05
21	0.0 0.4644245356949531613236952Q+00 0.7703582314624033503377197Q+00 0.9164102276533092812700223Q+00 0.9730253857518002228882883Q+00 0.9921820636849305702679853Q+00 0.9979881260945864249625917Q+00 0.9995549951079258234161885Q+00 0.9999204878478412139139828Q+00 0.9999898960889864283911738Q+00 0.9999993581764392883533741Q+00	0.5001810630896220275409655Q+00 0.3986919848590628932464307Q+00 0.2154954911716481127404845Q+00 0.8989364746250696045382833Q-01 0.3195492885095292241488160Q-01 0.1011871006442698280405657Q-01 0.2868354493646874587738844Q-02 0.7127320994560821185112573Q-03 0.1480687239094312936543876Q-03 0.2336224438956507721103207Q-04 0.2188485189161492720015305Q-05
22	0.2398193454984493139888112Q+00 0.6297836563851550275372373Q+00 0.8502869364451138144902889Q+00 0.9469724529676304990168465Q+00 0.9830675748396805637379674Q+00 0.9951100880752825292609457Q+00 0.9987429941979656835920282Q+00 0.9997220455265265833381405Q+00 0.9999503393269241639045206Q+00 0.9999936895119709686492593Q+00 0.9999995991440643970197845Q+00	0.4622267651796915855162009Q+00 0.3044599273926089806011403Q+00 0.1465538518629106253091886Q+00 0.5791504810415210835378617Q-01 0.2015956862020346410778978Q-01 0.6338247399800267103172520Q-02 0.1792805408099690192981597Q-02 0.4452167201751898879093532Q-03 0.9248023144736050899332985Q-04 0.1459112644359527401441093Q-04 0.1366835743414904175131546Q-05
23	0.0 0.4461919300148010764592254Q+00 0.7498777571161409031912966Q+00 0.9029039665617016693430513Q+00 0.9662134339496446216157792Q+00 0.9892828878130779742207304Q+00 0.9969117965558368629867174Q+00 0.9992066744531571084170896Q+00 0.9998246081481390541752835Q+00 0.9999686649742264146159967Q+00 0.9999960182230481790898221Q+00 0.9999997470691367058698650Q+00	0.4776402195248796777072790Q+00 0.3879890690937980392839322Q+00 0.2199834088363696296746118Q+00 0.9766084993628387834107407Q-01 0.3725926791260015700882120Q-01 0.1279967984320174560844855Q-01 0.4006497899200740802921864Q-02 0.1131743341488678371127686Q-02 0.2809502859420912392024673Q-03 0.5835396114896601423697658Q-04 0.9206685497799202303615917Q-05 0.8624419384355996801056130Q-06

24	0.2298085841040433588535110Q+00 0.6094872063753156973204157Q+00 0.8332864264477571379601808Q+00 0.9370510514585020613601209Q+00 0.9783442276165602566760961Q+00 0.9931592168861257407039884Q+00 0.9980315427215103989804632Q+00 0.9994945352633550771943388Q+00 0.9998882621504475227842400Q+00 0.9999800377521306132157817Q+00 0.9999974533873128041283351Q+00 0.999998388691254107713756Q+00	0.4442676728273933128664476Q+00 0.3022934902029223451734375Q+00 0.1533032118978526644782416Q+00 0.6442360913903910906352613Q-01 0.2402723044102733518618656Q-01 0.8185782603165586983362769Q-02 0.2555218239546755985350781Q-02 0.7211902671145960086392875Q-03 0.1789921723034259016496016Q-03 0.3717510458579347381376753Q-04 0.5865173363106986069600382Q-05 0.5494229861521091965583437Q-06
25	0.0 0.4299194753765594880004763Q+00 0.7308044413132134381299790Q+00 0.8896110960079164817659421Q+00 0.9590797675713717923369130Q+00 0.9860254082216479696860586Q+00 0.9955972267019984102030846Q+00 0.9987341878715413455191562Q+00 0.9996750468642387717780593Q+00 0.9999281710541282210277927Q+00 0.9999871677838670547435097Q+00 0.9999983694090382491972029Q+00 0.9999998964215397849541328Q+00	0.4578577937151621332831973Q+00 0.3779156199588325785159664Q+00 0.2231966560352564489870735Q+00 0.1045223071132925592777848Q+00 0.4234660394318568374577140Q-01 0.1556460055686419439048416Q-01 0.5274845150210979223464308Q-02 0.1643704525789996012775588Q-02 0.4636805876143134255183658Q-03 0.1150646825591733709355097Q-03 0.2389714227292826624068647Q-04 0.3770265384107437088925875Q-05 0.3531811559707052977467790Q-06
26	0.2209400515237258574995127Q+00 0.5909772955864953501071161Q+00 0.8170427298191262755625282Q+00 0.9270391265314232166508931Q+00 0.9732804166175428661622539Q+00 0.9909179727985949568649656Q+00 0.9971434555712910490381939Q+00 0.9991791862424361355855862Q+00 0.9997893195247914104732641Q+00 0.9999534324143172120384445Q+00 0.9999916308033592664317491Q+00 0.9999989428809497354829681Q+00 0.9999999328496620813262376Q+00	0.4282171846720751746288992Q+00 0.2995778911433934402299516Q+00 0.1589146123431075482669344Q+00 0.7041688558012043165223407Q-01 0.2784878930988497852619103Q-01 0.1014013752040388495589357Q-01 0.3424994609587128802140490Q-02 0.1066094531293154017421044Q-02 0.3006403585917894158477110Q-03 0.7459878568638738313136423Q-04 0.1549268437024787595530877Q-04 0.2444279772426840482927064Q-05 0.2289687870689587369787863Q-06
27	0.0 0.4152819583624965103369057Q+00 0.7130078311976761148697168Q+00 0.8766004171642746119918218Q+00 0.9517146689518614054358404Q+00 0.9824573259808460817973321Q+00 0.9940553400250693431772393Q+00 0.9981322542876620547689489Q+00 0.9994635016581502259002977Q+00 0.9998623100766057058946646Q+00 0.999969566850923800788191Q+00 0.9999945631915078373835196Q+00 0.9999993091464419673640969Q+00 0.9999999561155936915939746Q+00	0.4403160626042210938765217Q+00 0.3684533163894872383631786Q+00 0.2254158234677868907235804Q+00 0.1105751706671054284951536Q+00 0.4718714558447021759268475Q-01 0.1836828439782448714698939Q-01 0.6647663274017231914062871Q-02 0.2240532603252275555846456Q-02 0.6969172143346194050422738Q-03 0.1964902464990713554046305Q-03 0.4875293710526748798898764Q-04 0.1012488140137254449667376Q-04 0.1597397839373185491412651Q-05 0.1496367659792918200865525Q-06

28	0.2130128056116792315153774Q+00 0.5740134326266000256086614Q+00 0.8015407227320085286759571Q+00 0.9170192348052427718417470Q+00 0.9679417169326601987581953Q+00 0.9884134972127595275490709Q+00 0.9950814435718489177562374Q+00 0.9987696388505192228482602Q+00 0.9996466810421870276676915Q+00 0.9999093284272857446207438Q+00 0.9999799594201542499223389Q+00 0.9999964198335415921767689Q+00 0.9999995450697343256642323Q+00 0.9999999711019185132843045Q+00	0.4137626069974171773450458Q+00 0.2965026917921393791916115Q+00 0.1635734719057168308794920Q+00 0.7591617472482078723793741Q-01 0.3158598953191849415726179Q-01 0.1216800026790251445333469Q-01 0.4386395897520698498347015Q-02 0.1476341526408762737574383Q-02 0.4590072198029213879573177Q-03 0.1293958050273375319107450Q-03 0.3210436460008856852582205Q-04 0.6667292859410188185325285Q-05 0.1051893995083788248477483Q-05 0.9853649286470863570649871Q-07
29	0.0 0.4020245971080615402384534Q+00 0.6953678484373842468465907Q+00 0.8639150874189899353182663Q+00 0.9441913039595017296805669Q+00 0.9786222520537108155062466Q+00 0.9923001337697693714330056Q+00 0.9973992566846342301186417Q+00 0.9991838124050338171873515Q+00 0.9997656430307390138402427Q+00 0.9999398600754258759157259Q+00 0.9999867078005554719695889Q+00 0.9999975254102880370273794Q+00 0.9999996982620298018392830Q+00 0.9999999803330012688157196Q+00	0.4246235412324889257550172Q+00 0.3595690084961603836523986Q+00 0.2268545411965786201903424Q+00 0.1159104757521892703399179Q+00 0.5176667370883965388226164Q-01 0.2117593955737880365896263Q-01 0.8102031824595391032810875Q-02 0.2913165975085689255038982Q-02 0.9796067316890357339786448Q-03 0.3044782515276930847012077Q-03 0.8582593762670978542154487Q-04 0.2129374800805972898944029Q-04 0.4422167169688178440381422Q-05 0.6976813969082896939560302Q-06 0.6535550963030953318081366Q-07
30	0.2058720334470114785143896Q+00 0.5583966791787688373667782Q+00 0.7867535500102566786368753Q+00 0.9070529839546005784308524Q+00 0.9623845339678766752094585Q+00 0.9856729996479551734506059Q+00 0.9948513540044132808484886Q+00 0.9982624386314019176216604Q+00 0.9994548650495515521572039Q+00 0.9998434871652521786752185Q+00 0.9999598373467804578967948Q+00 0.999991123270492229262330Q+00 0.9999984142163657112779183Q+00 0.9999997984953072796212102Q+00 0.9999999872000199182894461Q+00	0.4006580531564503705514831Q+00 0.2931999918726044557797748Q+00 0.1674327625980917166111058Q+00 0.8095018667855256601643972Q-01 0.3521273522203805052224907Q-01 0.1424194195848058482155091Q-01 0.5424473842643703956297745Q-02 0.1947131341885498069469405Q-02 0.6543718993534512314908631Q-03 0.2033505073404951162155539Q-03 0.5731586269922185815586820Q-04 0.1422031465605693771153908Q-04 0.2953185157095609271289508Q-05 0.4659210852850203413381692Q-06 0.4364529031872472386090689Q-07
31	0.0 0.3999446732035971598013657Q+00 0.6807759092927535276484786Q+00 0.8515804233086545384018822Q+00 0.9365637135743046538827530Q+00 0.9745600161906506011989100Q+00 0.9903471673431158209100738Q+00 0.9965362958363836088005298Q+00 0.9988317241559785914384215Q+00 0.9996335418068497900016260Q+00 0.9998947934678663753703456Q+00 0.9999730035372204267803304Q+00 0.9999940332859093720696990Q+00 0.9999989340774392817965170Q+00 0.9999998645538113262951848Q+00 0.9999999913961881476019787Q+00	0.4104780368348212348678434Q+00 0.3512235820491207528461159Q+00 0.2276770368753094168153496Q+00 0.1206110121102164947256800Q+00 0.5608134277018535632688466Q-01 0.2396108755523904443724555Q-01 0.9617987209797056416398947Q-02 0.3652339335884041965837931Q-02 0.1309555978437655544323058Q-02 0.4399302664447741764550549Q-03 0.1366939052258313809892938Q-03 0.3852740325980075732255474Q-04 0.9558551402434447520068439Q-05 0.1985054855584466562403774Q-05 0.3131799883019484396205906Q-06 0.2933722275630895509604708Q-07

32

0.19939629135452099316375040+00
 0.54396124820327925928980210+00
 0.77264813990284304358535020+00
 0.89719604856448087690742920+00
 0.95665700374034482844684800+00
 0.98272271333794285177978460+00
 0.99346105377716070278729690+00
 0.99765598897611325743009730+00
 0.99920957978874664063224910+00
 0.99975212895334920300807070+00
 0.99992884167562228482303580+00
 0.99998174056942722386086210+00
 0.99999596437074667475554770+00
 0.99999927905631289896347450+00
 0.99999990839009927956172090+00
 0.99999999418075664472227920+00

0.38870726467417230077892000+00
 0.28976275517801969028719700+00
 0.17061873840139172336541760+00
 0.85550909521045530327066060-01
 0.38711961514669685478962970-01
 0.16339624844091289178535840-01
 0.65254936530158996203106870-02
 0.24730507373804783244095050-02
 0.88606138081700407455188390-03
 0.29758494313950589252028230-03
 0.92456947626699656449943340-04
 0.26058484053541799717572820-04
 0.64650000517029240518793320-05
 0.13426050505676747069573500-05
 0.21182130369076404796773510-06
 0.19842418889839165081053760-07

Tabelle II: Die Fehlerkonstanten Q_n der WILF'schen Quadraturformeln in $[-1, 1]$ für $N = 2(1)32$.

Zur Definition von Q_n vgl. (1.1).

N	Q(N)
=====	
2	0.2576818430+00
3	0.8632324400-01
4	0.3271243190-01
5	0.1354482000-01
6	0.5999892220-02
7	0.2804350440-02
8	0.1369818320-02
9	0.6943575440-03
10	0.3653125000-03
11	0.1954122740-03
12	0.1076840690-03
13	0.5063545690-04
14	0.3481035130-04
15	0.2033635970-04
16	0.1207095410-04
17	0.7269845390-05
18	0.4437263040-05
19	0.2742004030-05
20	0.1713935500-05
21	0.1082804690-05
22	0.6909220130-06
23	0.4449975790-06
24	0.2891280900-06
25	0.1894110510-06
26	0.1250552000-06
27	0.8317582150-07
28	0.5570890390-07
29	0.3756052680-07
30	0.2548461710-07
31	0.1737541630-07
32	0.1194215230-07

Tabelle III: Listen der Programme DWQN für $N = 2, 4, 8, 16, 24$
und 32.

```

SUBROUTINE DWQ2(YA,XB,FCT,Y)
REAL*8 A,B,C,Y,XA,XB,FCT
A=.500*(XB+XA)
B=.500*(XB-XA)
C= 0.665742695953212870+00*B
Y=B* 0.959303253001721615+00*(FCT(A+C)+FCT(A-C))
RETURN
END

```

```

SUBROUTINE DWQ4(XA,XB,FCT,Y)
REAL*8 A,B,C,Y,XA,XB,FCT
A=.500*(XB+XA)
B=.500*(XB-XA)
C= 0.519692429593519980+00*B
Y= 0.867796177894193710+00*(FCT(A+C)+FCT(A-C))
C= 0.961396989463653650+00*B
Y=B*(Y+ 0.129143967191694270+00*(FCT(A+C)+FCT(A-C)))
RETURN
END

```

```

SUBROUTINE DWQ6(XA,XB,FCT,Y)
REAL*8 A,B,C,Y,XA,XB,FCT
A=.500*(XB+XA)
B=.500*(XB-XA)
C= 0.386956389897215610+00*B
Y= 0.702358602319239260+00*(FCT(A+C)+FCT(A-C))
C= 0.956457381612934300+00*B
Y=Y+ 0.248814275604362490+00*(FCT(A+C)+FCT(A-C))
C= 0.980573199445520810+00*B
Y=Y+ 0.444888510923194900-01*(FCT(A+C)+FCT(A-C))
C= 0.998754713951697460+00*B
Y=B*(Y+ 0.424352840681216630-02*(FCT(A+C)+FCT(A-C)))
RETURN
END

```

```

SUBROUTINE DWQ16(XA,XB,FCT,Y)
REAL*8 A,B,C,Y,XA,XB,FCT
A=.500*(XB+XA)
B=.500*(XB-XA)
C= 0.279894546102949580+00*B
Y= 0.532300387974499930+00*(FCT(A+C)+FCT(A-C))
C= 0.704666212158882050+00*B
Y=Y+ 0.304176895049497270+00*(FCT(A+C)+FCT(A-C))
C= 0.905572909757984680+00*B
Y=Y+ 0.117011443753612420+00*(FCT(A+C)+FCT(A-C))
C= 0.974901827489047980+00*B
Y=Y+ 0.353727524455307310-01*(FCT(A+C)+FCT(A-C))
C= 0.994398434652100450+00*B
Y=Y+ 0.324954371123805080-02*(FCT(A+C)+FCT(A-C))
C= 0.998997008391720250+00*B
Y=Y+ 0.186692420566951810-02*(FCT(A+C)+FCT(A-C))
C= 0.999872494677603820+00*B
Y=Y+ 0.294900267454647310-03*(FCT(A+C)+FCT(A-C))
C= 0.999991900124765900+00*B
Y=B*(Y+ 0.276197911564492170-04*(FCT(A+C)+FCT(A-C)))
RETURN
END

```

```

SUBROUTINE DWQ24(XA,XB,FCT,Y)
REAL*8 A,B,C,Y,XA,XB,FCT
A=.500*(XB+XA)
B=.500*(XB-XA)
C= 0.229308584104042350+00*B
Y= 0.444267672927393310+00*(FCT(A+C)+FCT(A-C))
C= 0.609487206375315700+00*B
Y=Y+ 0.302293490202922340+00*(FCT(A+C)+FCT(A-C))
C= 0.833285426447757130+00*B
Y=Y+ 0.153303211897852660+00*(FCT(A+C)+FCT(A-C))
C= 0.937051051458502070+00*B
Y=Y+ 0.644236091390391060-01*(FCT(A+C)+FCT(A-C))
C= 0.978244227516560260+00*B
Y=Y+ 0.240272304410273350-01*(FCT(A+C)+FCT(A-C))
C= 0.995159216986125740+00*B
Y=Y+ 0.818578260316558690-02*(FCT(A+C)+FCT(A-C))
C= 0.998031542721510400+00*B
Y=Y+ 0.255521823954675600-02*(FCT(A+C)+FCT(A-C))
C= 0.999494535263355080+00*B
Y=Y+ 0.721190267114595980-03*(FCT(A+C)+FCT(A-C))
C= 0.999883262190447520+00*B
Y=Y+ 0.178992172303425900-03*(FCT(A+C)+FCT(A-C))
C= 0.999980037752130610+00*B
Y=Y+ 0.371751045857934750-04*(FCT(A+C)+FCT(A-C))
C= 0.999997453387312300+00*B
Y=Y+ 0.586517336310698600-05*(FCT(A+C)+FCT(A-C))
C= 0.999999838369125410+00*B
Y=B*(Y+ 0.549422986152109200-06*(FCT(A+C)+FCT(A-C)))
RETURN
END

```

```

SUBROUTINE DW032(XA,XB,FCT,Y)
REAL*8 A,B,C,Y,XA,XB,FCT
A=.500*(XB+XA)
B=.500*(XB-XA)
C= 0.19939629185452099D+00*B
Y= 0.33870726467417230D+00*(FCT(A+C)+FCT(A-C))
C= 0.54396124820327926D+00*B
Y=Y+ 0.28976275517801969D+00*(FCT(A+C)+FCT(A-C))
C= 0.77264813990284804D+00*B
Y=Y+ 0.17081873840139172D+00*(FCT(A+C)+FCT(A-C))
C= 0.89718604886448088D+00*B
Y=Y+ 0.85550909521045534D-01*(FCT(A+C)+FCT(A-C))
C= 0.95665700374034483D+00*B
Y=Y+ 0.38711961514669685D-01*(FCT(A+C)+FCT(A-C))
C= 0.98272271333794285D+00*B
Y=Y+ 0.16339624844091289D-01*(FCT(A+C)+FCT(A-C))
C= 0.99346105377716069D+00*B
Y=Y+ 0.65254936530158994D-02*(FCT(A+C)+FCT(A-C))
C= 0.99765598897611325D+00*B
Y=Y+ 0.24730507373804783D-02*(FCT(A+C)+FCT(A-C))
C= 0.99920967978874664D+00*B
Y=Y+ 0.88606138031700407D-03*(FCT(A+C)+FCT(A-C))
C= 0.99975212895334920D+00*B
Y=Y+ 0.29758494313950588D-03*(FCT(A+C)+FCT(A-C))
C= 0.99992884167562228D+00*B
Y=Y+ 0.92456947626599655D-04*(FCT(A+C)+FCT(A-C))
C= 0.99998174068942722D+00*B
Y=Y+ 0.26058484053541800D-04*(FCT(A+C)+FCT(A-C))
C= 0.99999596437074667D+00*B
Y=Y+ 0.64650000517029241D-05*(FCT(A+C)+FCT(A-C))
C= 0.99999927905631290D+00*B
Y=Y+ 0.13426050505676747D-05*(FCT(A+C)+FCT(A-C))
C= 0.9999999083900928D+00*B
Y=Y+ 0.21182130369076405D-06*(FCT(A+C)+FCT(A-C))
C= 0.9999999418075664D+00*B
Y=3*(Y+ 0.19842418889839165D-07*(FCT(A+C)+FCT(A-C)))
RETURN
END

```