

Capitolo 4

La quadratura

4.1 Generalità

La misura delle aree è uno dei problemi scientifici più antichi. In epoca babilonese ed egiziana, ad esempio, venivano calcolate delle buone approssimazioni dell'area di un cerchio con il cosiddetto **metodo di esaustione**¹. Problema analogo era la costruzione di un quadrato con area uguale a quella di un cerchio assegnato mediante l'ausilio di riga e compasso in un numero finito di passi. Tale problema era noto come **quadratura del cerchio**² e, più in generale, il termine **quadratura** veniva utilizzato per riferirsi a qualunque problema di calcolo dell'area o del volume di un oggetto di forma nota. Il problema del calcolo delle aree e dei volumi è tuttora di interesse in molte applicazioni di varia natura come, ad esempio, nella determinazione del volume di terra da rimuovere per la costruzione di strade, o nel calcolo dell'area della superficie della carrozzeria di un'automobile. In genere i problemi sono relativi a figure geometriche di forma irregolare, per cui non è sempre possibile applicare formule elementari per il calcolo delle aree (o dei volumi).

♣ **Esempio 4.1.** Si supponga di voler calcolare l'area della regione piana chiusa mostrata in Figura 4.1. Tale regione può essere divisa in cinque sottoregioni tracciando delle linee orizzontali e verticali come mostrato in Figura 4.2.

La sottoregione A è un rettangolo la cui area può essere calcolata mediante il prodotto delle sue dimensioni. Le sottoregioni B, C, D, E hanno forma irregolare ed il calcolo della loro area può essere ricondotto al calcolo dell'area di una regione piana delimitata da una curva. ♣

¹Il metodo consisteva nel dimostrare che due grandezze sono uguali mostrando che la loro differenza è inferiore a qualsiasi quantità data, per quanto piccola. In particolare, per determinare l'area del cerchio, si inscrive al suo interno una successione di poligoni regolari con un numero di lati crescente. La superficie del poligono diventa sempre più grande ad ogni passo, ma rimane sempre inferiore a quella del cerchio. Inoltre la differenza tra le due aree diventa sempre più piccola. In tal modo l'area del cerchio veniva calcolata con la massima precisione possibile nonostante non la si conoscesse esattamente. Utilizzando tale metodo Archimede fornì anche un'approssimazione per il numero irrazionale π (il rapporto tra area del cerchio e quadrato del raggio), stimando che esso appartiene all'intervallo chiuso $[223/71, 22/7]$ commettendo un errore dell'ordine di 10^{-4} .

²Tale problema non ha soluzione. F. Lindemann, nel 1882 mostrò infatti che il numero π è un numero irrazionale.

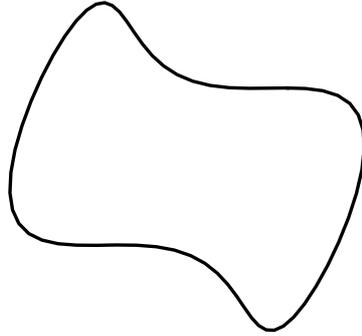


Figura 4.1: Regione piana chiusa

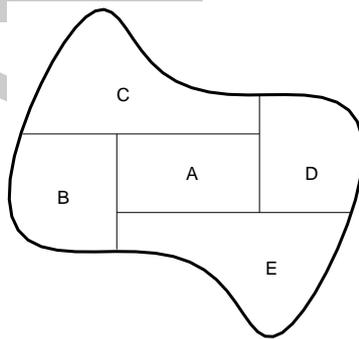


Figura 4.2: Suddivisione della regione in sottoregioni

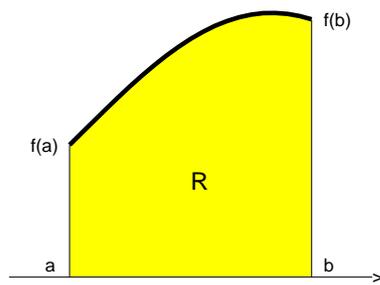


Figura 4.3: Area di un rettangoloide

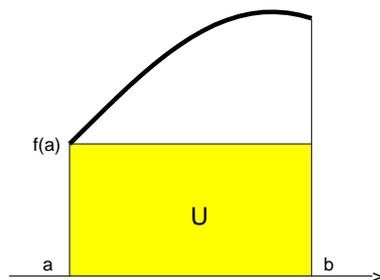


Figura 4.4: Approssimazione di $\mathcal{A}(R)$ mediante la formula rettangolare

Se si indica con $y = f(x)$ la funzione (non negativa) che rappresenta la curva, alla regione R in Figura 4.3 si dà il nome di **rettangoloide** di base l'intervallo $[a, b]$, relativo alla funzione $y = f(x)$, la cui area $\mathcal{A}(R)$ è detta **integrale definito di $f(x)$ tra a e b** ed è indicata con

$$\mathcal{A}(R) \equiv I[f] = \int_a^b f(x) dx \tag{4.1}$$

4.2 Formule di quadratura: primi esempi

Il metodo più naturale per approssimare l'area del rettangoloide R è quello di sovrapporre ad esso delle figure elementari e di calcolare l'area di ognuna di esse. Sia, ad esempio, R il rettangoloide di base l'intervallo $[a, b]$, relativo alla funzione $y = f(x)$ mostrato in Figura 4.3.

Un primo modo per approssimare l'area $\mathcal{A}(R)$ del rettangoloide R è quello di utilizzare un rettangolo U con lati di misura $(b - a)$ (la base) e $f(a)$ (l'altezza). Detta allora $\mathcal{A}(U)$ l'area del rettangolo U si ha che

$$I[f] \equiv \mathcal{A}(R) \simeq \mathcal{A}(U) = f(a)(b - a) \tag{4.2}$$

Tale formula è detta **formula rettangolare** (Figura 4.4). Inoltre è possibile scegliere il rettangolo V con lati di misura $(b - a)$ e $f(c)$ dove $c = (a + b)/2$ è il punto medio dell'intervallo $[a, b]$, ottenendo così la **formula del punto medio** (Figura 4.5):

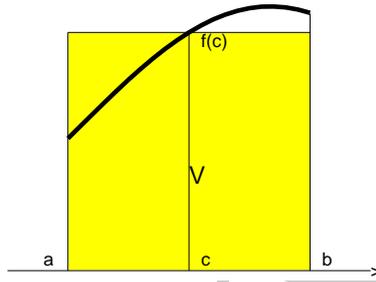


Figura 4.5: Approssimazione di $\mathcal{A}(R)$ mediante la formula del punto medio

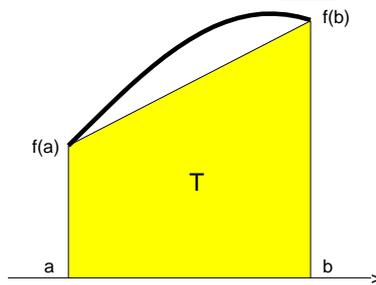


Figura 4.6: Approssimazione di $\mathcal{A}(R)$ mediante la formula trapezoidale

$$I[f] \equiv \mathcal{A}(R) \simeq \mathcal{A}(V) = f(c)(b - a) \quad (4.3)$$

Un terzo metodo per approssimare l'area di R è quello di utilizzare un trapezio T con basi di misura $f(a)$ e $f(b)$ ed altezza $(b - a)$, cioè

$$I[f] \equiv \mathcal{A}(R) \simeq \mathcal{A}(T) = \frac{(b - a)}{2} [f(a) + f(b)] \quad (4.4)$$

Quest'ultima formula è detta **formula trapezoidale** ed è indicata anche con il simbolo $T[f]$. Come mostrato in Figura 4.6, la formula trapezoidale equivale a calcolare l'area al di sotto del segmento di retta passante per la coppia di punti $(a, f(a))$, $(b, f(b))$.

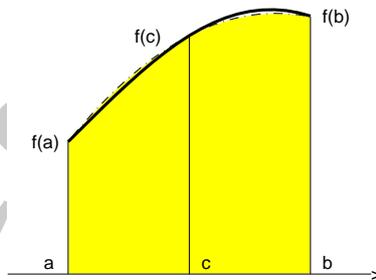


Figura 4.7: Approssimazione di $\mathcal{A}(R)$ mediante la formula di Simpson

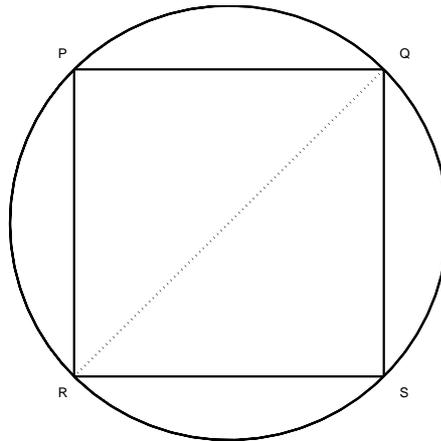


Figura 4.8: Decomposizione del cerchio in 5 sottoregioni

Analogamente è possibile ricavare altre formule. Infatti la parabola passante per i punti $(a, f(a))$, $(b, f(b))$ e $(c, f(c))$ (dove c è il punto medio di $[a, b]$) ha equazione:

$$y = f(a) \frac{(x - c)(x - b)}{(a - c)(a - b)} + f(c) \frac{(x - a)(x - b)}{(c - a)(c - b)} + f(b) \frac{(x - a)(x - c)}{(b - a)(b - c)}$$

per cui calcolando l'area al di sotto di tale parabola si ottiene:

$$I[f] \equiv \mathcal{A}(R) \simeq \frac{(b - a)}{2} \left(\frac{f(a)}{3} + \frac{4}{3}f(c) + \frac{f(b)}{3} \right) \tag{4.5}$$

Tale formula è nota come **formula di Simpson**³ (Figura 4.7) ed è indicata con $S[f]$.

♣ **Esempio 4.2.** Si voglia calcolare l'area del cerchio Ω di raggio 1 in un sistema aritmetico floating point con base $\beta = 10$ e precisione $t = 5$ (si ricorda che il valore di tale area è $\mathcal{A}(\Omega) = \pi = 3.1415926\dots$).

Il cerchio può essere diviso in 5 sottoregioni (Figura 4.8). Una di queste è un quadrato che ha come diagonale il segmento \overline{QR} lungo 2 e come lato un segmento la cui lunghezza è $\sqrt{2}$. L'area del quadrato è quindi 2. Le restanti sottoregioni sono calotte circolari uguali tra loro e la relativa area può essere calcolata con la formula trapezoidale. Una di tali sottoregioni è mostrata in Figura 4.9.

Se si considera il punto medio M di $[P, Q]$ come origine del riferimento cartesiano si ha che i vertici P e Q hanno rispettivamente coordinate $P \equiv (-\frac{\sqrt{2}}{2}, 0)$ e $Q \equiv (\frac{\sqrt{2}}{2}, 0)$ e la curva che sottende il segmento \overline{PQ} è il grafico della funzione:

$$f(x) = \sqrt{1 - x^2} - \frac{\sqrt{2}}{2}$$

³Nonostante il nome, tale formula fu proposta per primo da Cavalieri nel 1639 con il nome di *regola parabolica* e ripresa da James Gregory nel 1668. Solo nel 1743 Thomas Simpson rese famosa tale formula. Per tale motivo essa è nota anche con il nome di **formula di Cavalieri-Simpson**.

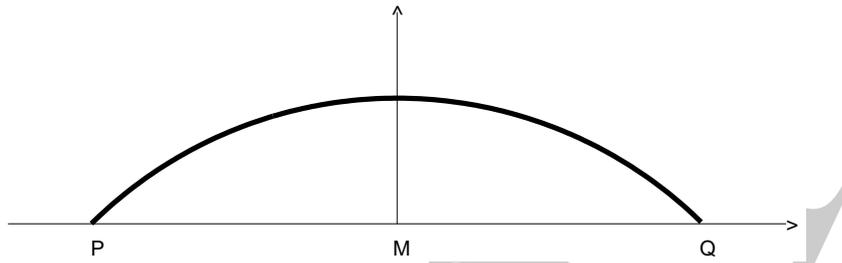


Figura 4.9: Area della calotta circolare

Tale regione è composta da due sottoregioni simmetriche relative ai due sottointervalli $[P, M]$ e $[M, Q]$. È sufficiente quindi calcolare solo l'area della sottoregione relativa al segmento $[M, Q]$. Cioè

$$I[f] = \int_0^{\sqrt{2}/2} \left(\sqrt{1-x^2} - \frac{\sqrt{2}}{2} \right) dx = \frac{\pi-2}{8} = 0.142699\dots$$

Utilizzando la formula trapezoidale (4.4) sull'intervallo $[M, Q]$ risulta $(b-a) = \sqrt{2}/2$ e

$$I[f] \simeq \mathcal{A}(T) = \frac{\sqrt{2}}{4} \left(1 - \frac{\sqrt{2}}{2} \right) = 0.10355 \times 10^0$$

mentre l'errore assoluto commesso nell'approssimazione di $I[f]$ con $\mathcal{A}(T)$ nella sottoregione relativa al segmento $[M, Q]$ è

$$|(\pi-2)/8 - 0.10355 \times 10^0| \simeq 0.39149 \times 10^{-1}$$

Un'approssimazione dell'area del cerchio è quindi:

$$\mathcal{A}(\Omega) \simeq 2 + 8(0.10355 \times 10^0) = .28284 \times 10^1$$

e poiché l'area del cerchio di raggio 1 è $\pi = 3.141592\dots$, l'errore assoluto commesso nell'approssimare $\mathcal{A}(\Omega)$ con il valore ottenuto, è

$$|\pi - .28284 \times 10^1| \simeq 0.31315 \times 10^0$$



Uno degli obiettivi che si pone la quadratura è quindi quello di determinare formule accurate per il calcolo di un integrale definito di una funzione $y = f(x)$ (non solo non negativa). Le formule finora proposte (4.2), (4.3), (4.4), (4.5) per il calcolo di (4.1) sono una combinazione lineare di valori della funzione integranda $f(x)$, con coefficienti dipendenti dall'ampiezza dell'intervallo di integrazione. Si può dare quindi la seguente definizione:

Definizione 4.1. (Formula di quadratura)

Fissata una funzione integrabile nel senso di Riemann ⁴ ed un intero n , dati n punti $x_i \in [a, b]$ detti **nod**i e n valori A_i detti **pesi**, la combinazione lineare:

$$Q[f] = A_1 f(x_1) + \dots + A_n f(x_n)$$

prende il nome di **formula di quadratura**.

Una formula di quadratura rappresenta un'approssimazione dell'integrale $I[f]$, per cui è possibile dare anche la seguente definizione:

Definizione 4.2. (Errore di discretizzazione)

La differenza:

$$E[f] = I[f] - Q[f] = \int_a^b f(x) dx - \sum_{i=1}^n A_i f(x_i)$$

è l'**errore di discretizzazione della formula di quadratura** $Q[f]$.

In Figura 4.10 è fornita la rappresentazione grafica dell'errore di discretizzazione della formula trapezoidale per il calcolo di (4.1).

♣ **Esempio 4.3.** Si calcoli l'errore della formula trapezoidale relativo all'integrale:

$$I[f] = \int_1^2 (3x + 1) dx = 5.5$$

Applicando la formula trapezoidale si ha:

$$T[f] = (4 + 7)/2 = 5.5$$

⁴Si ricorda che una funzione è detta *integrabile nel senso di Riemann in $[a, b]$* se, detta $P_1, P_2, \dots, P_m, \dots$ una generica successione di partizioni dell'intervallo $[a, b]$ in sottointervalli $[t_{j-1}, t_j]$, tale che la successione dei sottointervalli di massima ampiezza $\Delta_1, \Delta_2, \dots, \Delta_m, \dots$ converga a zero al crescere di m , le corrispondenti successioni di somme $S_1[f], S_2[f], \dots, S_m[f], \dots$:

$$S_m[f] = \sum_{j=1}^m f(\xi_j)(t_j - t_{j-1}) \quad \xi_j \in [t_{j-1}, t_j]$$

(chiamate somme di Riemann), hanno un limite comune S (che è proprio l'integrale definito della funzione, detto anche integrale di Riemann) che è indipendente dalla scelta della partizione e dei punti ξ_j , cioè:

$$\lim_{m \rightarrow \infty} S_m[f] = S = \int_a^b f(x) dx.$$

In particolare sono funzioni integrabili nel senso di Riemann le funzioni continue e le funzioni limitate con un numero finito di discontinuità. Nel seguito diremo semplicemente funzioni integrabili per indicare le funzioni integrabili nel senso di Riemann.

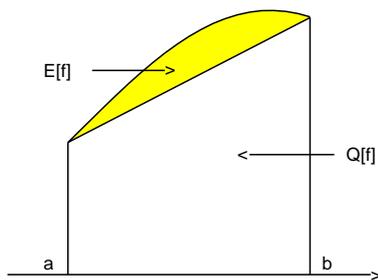


Figura 4.10: Rappresentazione grafica dell'errore di discretizzazione

In tal caso l'errore di discretizzazione della formula trapezoidale è nullo. Si può facilmente osservare che tale proprietà è vera per ogni polinomio $p(x) \in \Pi_1$, cioè $E[p] = 0 \forall p \in \Pi_1$. Per tale motivo si dice che la formula trapezoidale è **esatta per polinomi di primo grado**. ♣

4.2.1 Le formule di quadratura composite

Dalla Figura 4.10 è possibile notare che in generale l'errore di una formula trapezoidale dipende anche dall'ampiezza dell'intervallo $[a, b]$. Si può pensare allora di dividere l'intervallo $[a, b]$ in m sottointervalli (talvolta chiamati *pannelli*) di uguale ampiezza $\tau = (b - a)/m$ mediante i punti equidistanti:

$$a = t_0 < t_1 < \dots < t_m = b \quad \text{con} \quad t_j = a + j\tau, \quad j = 0, \dots, m \quad (4.6)$$

e, poiché:

$$\int_a^b f(x) dx = \sum_{j=1}^m \int_{t_{j-1}}^{t_j} f(x) dx$$

si può utilizzare la formula trapezoidale in ognuno dei sottointervalli $[t_{j-1}, t_j]$ ⁵.

♣ **Esempio 4.4.** Si utilizzi la formula trapezoidale con 4 sottointervalli per calcolare l'integrale:

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

In questo caso l'ampiezza dei sottointervalli coincide con la distanza tra i nodi in cui viene valutata la funzione integranda, ed è $h = \tau = t_j - t_{j-1} = \frac{b-a}{4}$ e per la proprietà additiva degli integrali risulta:

$$\int_a^b f(x) dx = \sum_{j=1}^4 \int_{t_{j-1}}^{t_j} f(x) dx \simeq \sum_{j=1}^4 \frac{(t_j - t_{j-1})}{2} [f(t_{j-1}) + f(t_j)] =$$

⁵L'idea di ridurre l'errore di una formula di quadratura applicando la stessa formula su più sottointervalli dell'intervallo di integrazione è basata sull'osservazione che, scelti m nodi t_j nell'intervallo $[a, b]$ la somma di Riemann è una formula rettangolare composta in cui l'altezza del rettangolo non coincide con il valore della funzione nell'estremo inferiore dell'intervallo.

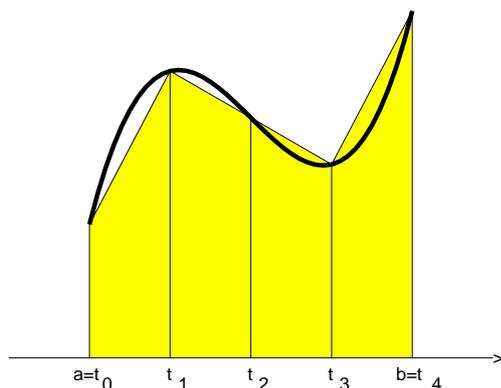


Figura 4.11: Formula trapezoidale composta su 4 sottointervalli

$$h \left[\frac{f(t_0)}{2} + f(t_1) + f(t_2) + f(t_3) + \frac{f(t_4)}{2} \right]$$

In Figura 4.11 è fornita la rappresentazione grafica della formula di quadratura ottenuta utilizzando la formula trapezoidale nei 4 sottointervalli $[t_0, t_1]$, $[t_1, t_2]$, $[t_2, t_3]$ e $[t_3, t_4]$, dell'intervallo $[a, b]$. ♣

♣ **Esempio 4.5.** Calcolare l'integrale (vedi esempio 4.2):

$$I[f] = \int_0^{\sqrt{2}/2} \left(\sqrt{1-x^2} - \frac{\sqrt{2}}{2} \right) dx$$

dividendo l'intervallo $[0, \sqrt{2}/2]$ rispettivamente in 2 e 4 sottointervalli in cui si applica la formula trapezoidale. Facendo riferimento alla Figura 4.12, mediante la formula trapezoidale in un solo sottointervallo, $T_1[f]$, si è avuta un'approssimazione dell'area del rettangoloide relativo all'intervallo $[M, Q]$ ed un errore $E_1[f]$ rispettivamente di:

$$T_1[f] = 0.10355 \times 10^0,$$

$$|E_1[f]| = |I[f] - T_1[f]| \simeq 0.39149 \times 10^{-1}.$$

Se si utilizza invece la formula trapezoidale sui due sottointervalli $[M, R]$ e $[R, Q]$ con R punto medio di $[M, Q]$ si ha un'approssimazione

$$T_2[f] = \frac{\sqrt{2}}{4} \left[\frac{1}{2} \left(1 - \frac{\sqrt{2}}{2} \right) + \left(\sqrt{\frac{7}{8}} - \frac{\sqrt{2}}{2} \right) \right] = 0.13250 \times 10^0$$

con un errore

$$|E_2[f]| = |I[f] - T_2[f]| = |(\pi - 2)/8 - 0.13250 \times 10^0| \simeq 0.10199 \times 10^{-1}.$$

Dimezzando ancora l'ampiezza dei sottointervalli $[M, R]$ e $[R, Q]$, applicando la formula trapezoidale sui quattro sottointervalli, $[M, S]$, $[S, R]$, $[R, T]$ e $[T, Q]$, con S punto medio di $[M, R]$ e T punto medio di $[R, Q]$, si ha:

$$\begin{aligned} T_4[f] &= \frac{\sqrt{2}}{8} \left[\frac{1}{2} \left(1 - \frac{\sqrt{2}}{2} \right) + \left(\sqrt{\frac{31}{32}} - \frac{\sqrt{2}}{2} \right) + \left(\sqrt{\frac{7}{8}} - \frac{\sqrt{2}}{2} \right) + \left(\sqrt{\frac{23}{32}} - \frac{\sqrt{2}}{2} \right) \right] \\ &= 0.14014 \times 10^0, \end{aligned}$$

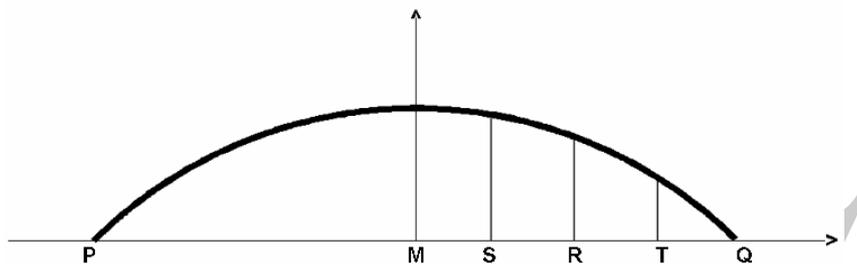


Figura 4.12: Area della calotta circolare

$$|E_4[f]| = |I[f] - T_4[f]| = |(\pi - 2)/8 - 0.140140 \times 10^0| \simeq 0.25591 \times 10^{-2}.$$

Dall'esempio si può notare che dimezzando l'ampiezza dei sottointervalli l'errore si riduce di circa 4 volte. ♣

Definizione 4.3. (Formula trapezoidale composta)

Suddiviso l'intervallo $[a, b]$ in m sottointervalli di ampiezza

$$\tau = \frac{(b - a)}{m},$$

mediante i punti equidistanziati:

$$a = x_0 < x_1 < \dots < x_m = b \quad \text{con} \quad x_i = a + i\tau, \quad i = 0, \dots, m \quad (4.7)$$

la formula di quadratura $T_m[f]$ che si ottiene applicando, in ognuno degli intervalli $[x_{j-1}, x_j]$, con $j = 1, \dots, m$, la formula trapezoidale $T[f]$, è detta **formula trapezoidale composta**. In questo caso la distanza tra i nodi h coincide con τ , per cui, posto

$$x_i = a + ih, \quad i = 0, \dots, m - 1, \quad x_m = b,$$

si ha:

$$\int_a^b f(x) dx \simeq T_m[f] = h \left(\frac{f(x_0)}{2} + f(x_1) + \dots + f(x_{m-1}) + \frac{f(x_m)}{2} \right) \quad (4.8)$$

È possibile definire altre formule composte, come, ad esempio, la formula di Simpson composta $S_m[f]$ su m sottointervalli $[x_{j-1}, x_j]$ di ampiezza $\tau = (b - a)/m$. In tal caso, dalla (4.5) si ha che la distanza tra i nodi è $h = \tau/2 = (b - a)/2m$, per cui posto

$$x_0 = a \quad x_i = a + ih \quad i = 1, \dots, 2m$$

la formula di Simpson composta risulta:

$$S_m[f] = h \left(\frac{f(x_0)}{3} + \frac{4}{3}f(x_1) + \frac{2}{3}f(x_2) + \dots + \frac{2}{3}f(x_{2m-2}) + \frac{4}{3}f(x_{2m-1}) + \frac{f(x_{2m})}{3} \right)$$

In generale si ha:

Definizione 4.4. (Formula di quadratura composta)

Fissata una formula di quadratura

$$Q[f] = \sum_{i=1}^n A_i f(x_i)$$

ed un intervallo $[a, b]$, si divida tale intervallo in m sottointervalli

$$[t_{j-1}, t_j] \quad (j = 1, \dots, m).$$

Si definisce $Q_m[f]$ **formula di quadratura composta** la formula che si ottiene applicando $Q[f]$ in ogni sottointervallo $[t_{j-1}, t_j]$ di ampiezza $\tau_j = (t_j - t_{j-1})$, cioè:

$$Q_m[f] = \sum_{j=1}^m Q^{(j)}[f] = \sum_{j=1}^m \sum_{i=1}^n A_i^{(j)} f(x_i^{(j)}). \quad (4.9)$$

dove con $Q^{(j)}[f]$ si è indicata la formula $Q[f]$ nel j -mo intervallo $[t_{j-1}, t_j]$, con $x_i^{(j)}$ si è indicato l' i -mo nodo di $Q^{(j)}[f]$ e con $A_i^{(j)}$ l' i -mo peso di $Q^{(j)}[f]$.

Si noti il ruolo differente che giocano gli n nodi x_i della Definizione 4.1 e gli $m+1$ punti t_j della Definizione 4.4. I primi rappresentano gli n nodi di una formula di quadratura $Q[f]$ mentre i secondi sono utilizzati per suddividere l'intervallo $[a, b]$ in m sottointervalli in ognuno dei quali viene applicata la formula di quadratura fissata. In generale $Q_m[f]$ ha mn nodi, tranne nel caso in cui il primo e l'ultimo nodo di $Q[f]$ coincidano con gli estremi di ogni intervallo $[t_{j-1}, t_j]$ (come ad esempio accade per la formula trapezoidale e per la formula di Simpson). In tal caso $Q_m[f]$ possiede $m(n - 1) + 1$ nodi.

Definizione 4.5. (Formule convergenti)

Data una funzione integrabile $y = f(x)$, la successione di formule composte $\{Q_m[f]\}$ è detta **convergente** se

$$\lim_{m \rightarrow \infty} E_m[f] = 0 \quad \Leftrightarrow \quad \lim_{m \rightarrow \infty} Q_m[f] = \int_a^b f(x) dx$$

dove $E_m[f]$ è l'errore dell' m -ma formula di quadratura $Q_m[f]$.

L'esempio 4.5 ha messo in luce che l'errore di discretizzazione della formula trapezoidale composta si riduce quando si dimezza l'ampiezza dei sottointervalli h .

Teorema 4.1. Sia $T[f]$ la formula trapezoidale e $f(x)$ una funzione integrabile nell'intervallo $[a, b]$. Detta $T_m[f]$ la formula composta su m sottointervalli di ampiezza $\tau = (b - a)/m$ definita dalla (4.8) si ha:

$$\lim_{m \rightarrow \infty} E_m[f] = 0$$

dove $E_m[f] = I[f] - T_m[f]$ è l'errore della formula trapezoidale composta $T_m[f]$.

Dimostrazione La formula trapezoidale $T[f]$ applicata all'intervallo $[t_{j-1}, t_j] \subseteq [a, b]$ di ampiezza τ è:

$$\int_{t_{j-1}}^{t_j} f(x) dx \simeq \frac{(t_j - t_{j-1})}{2} [f(t_{j-1}) + f(t_j)]$$

La formula trapezoidale composta $T_m[f]$ è allora:

$$\begin{aligned} T_m[f] &= \frac{1}{2} \sum_{j=1}^m (t_j - t_{j-1}) (f(t_{j-1}) + f(t_j)) \\ &= \frac{1}{2} \left[\sum_{j=1}^m (t_j - t_{j-1}) f(t_{j-1}) + \sum_{j=1}^m (t_j - t_{j-1}) f(t_j) \right] \end{aligned}$$

Ma per l'integrabilità della funzione $f(x)$ si ha

$$\lim_{m \rightarrow \infty} \sum_{j=1}^m (t_j - t_{j-1}) f(t_{j-1}) = \lim_{m \rightarrow \infty} \sum_{j=1}^m (t_j - t_{j-1}) f(t_j) = \int_a^b f(x) dx = I[f]$$

da cui:

$$\lim_{m \rightarrow \infty} E_m[f] = I[f] - \lim_{m \rightarrow \infty} T_m[f] = I[f] - \frac{1}{2} [2I[f]] = 0$$

cioè la tesi. ■

Questo tipo di comportamento di riduzione dell'errore dimostrato per la formula trapezoidale composta al crescere del numero di sottointervalli è comune ad altre formule composite⁶.

Un aspetto centrale della quadratura è fornire, oltre al valore dell'integrale $I[f]$, anche una stima dell'errore $E[f]$ basata su quantità calcolabili.

♣ **Esempio 4.6.** Si determini una stima calcolabile dell'errore della formula trapezoidale composta. Preliminarmente si dimostra il:

Teorema 4.2. Sia $f \in C^2[a, b]$ e

$$T[f] = \frac{(b-a)}{2} [f(a) + f(b)]$$

la formula trapezoidale. Detto $E[f] = I[f] - T[f]$ l'errore della formula trapezoidale, si ha:

$$|E[f]| \leq \frac{5M(b-a)^3}{12} \quad M = \max_{x \in [a,b]} |f''(x)|.$$

Dimostrazione Siano

$$\begin{aligned} f(x) &= f(a) + f'(a)(x-a) + f''(\xi_1)(x-a)^2/2 & \xi_1 &\in [a, x] \\ f(x) &= f(b) + f'(b)(x-b) + f''(\xi_2)(x-b)^2/2 & \xi_2 &\in [b, x] \end{aligned}$$

⁶La dimostrazione di ciò verrà data nel Teorema 4.9.

le formule di Taylor per la funzione $f(x)$ rispettivamente di punti iniziali a e b . Sommando membro a membro, dividendo per 2 e integrando termine a termine si ha:

$$I[f] = \int_a^b f(x) dx = \int_a^b \frac{[f(a) + f(b)]}{2} dx + \int_a^b \frac{[f'(a)(x-a) + f'(b)(x-b)]}{2} dx + \int_a^b \frac{[f''(\xi_1)(x-a)^2 + f''(\xi_2)(x-b)^2]}{4} dx \quad (4.10)$$

da cui, applicando la formula trapezoidale $T[f]$ ai primi due integrali al secondo membro (le cui funzioni integrande sono polinomi di grado zero e di primo grado per cui la formula trapezoidale è esatta), si ha:

$$I[f] = \int_a^b f(x) dx = \frac{(b-a)}{2}[f(a) + f(b)] + \frac{[f'(a)(b-a) + f'(b)(a-b)](b-a)}{4} + \int_a^b \frac{[f''(\xi_1)(x-a)^2 + f''(\xi_2)(x-b)^2]}{4} dx$$

Da cui si ha:

$$E[f] = I[f] - T[f] = \frac{[f'(a) - f'(b)](b-a)^2}{4} + \int_a^b \frac{[f''(\xi_1)(x-a)^2 + f''(\xi_2)(x-b)^2]}{4} dx \quad (4.11)$$

Per il Teorema di Lagrange applicato alla funzione $f'(x)$ è possibile trovare un punto $\xi \in [a, b]$ tale che:

$$f'(b) - f'(a) = f''(\xi)(b-a)$$

da cui, sostituendo nella (4.11), si ha

$$E[f] = -\frac{f''(\xi)(b-a)^3}{4} + \int_a^b \frac{[f''(\xi_1)(x-a)^2 + f''(\xi_2)(x-b)^2]}{4} dx$$

Posto $M = \max_{x \in [a, b]} |f''(x)|$ è allora possibile scrivere:

$$|E[f]| \leq \left| \frac{M(b-a)^3}{4} \right| + \frac{M}{4} \left| \int_a^b [(x-a)^2 + (x-b)^2] dx \right| \quad (4.12)$$

Per l'integrale a secondo membro della (4.12) con facili calcoli si ha:

$$\int_a^b [(x-a)^2 + (x-b)^2] dx = \frac{2(b-a)^3}{3}$$

e quindi:

$$|E[f]| \leq \frac{M(b-a)^3}{4} + \frac{M(b-a)^3}{6} = \frac{5M(b-a)^3}{12}$$

cioè la tesi. ■

Corollario 4.1. *Nelle stesse ipotesi del teorema precedente, detta $T_m[f]$ la formula trapezoidale composta su m sottointervalli $[t_{j-1}, t_j]$ di ampiezza $h = \tau = (b-a)/m$ si ha che l'errore $E_m[f] = I[f] - T_m[f]$ è:*

$$|E_m[f]| \leq \frac{5M(b-a)h^2}{12} \quad \text{con} \quad M = \max_{x \in [a,b]} |f''(x)|$$

Dimostrazione Per il Teorema 4.2, in ogni sottointervallo $[t_{j-1}, t_j]$ di ampiezza $h = \tau = (b-a)/m$ si ha:

$$|E[f]| \leq \frac{5Mh^3}{12}$$

e per ottenere una maggiorazione di $E_m[f]$ basta sommare il contributo dell'errore in ogni sottointervallo $[t_{j-1}, t_j]$, cioè:

$$|E_m[f]| \leq \sum_{j=1}^m \frac{5Mh^3}{12} = m \frac{5Mh^3}{12} = \frac{5M(b-a)h^2}{12} \quad (4.13)$$

da cui la tesi. ■

Osserviamo che, dalla (4.13), se si dimezza la distanza tra i nodi h , l'errore si riduce di circa quattro volte. Detti allora $E_m[f]$ ed $E_{2m}[f]$ rispettivamente gli errori delle due formule trapezoidali composte $T_m[f]$ e $T_{2m}[f]$ assumiamo per il corollario del Teorema 4.2 che $|E_m[f]| \simeq 4|E_{2m}[f]|$. Approssimando $I[f]$ con le due formule considerate si ha:

$$\begin{aligned} I[f] - T_m[f] &= E_m[f] \\ I[f] - T_{2m}[f] &= E_{2m}[f], \end{aligned}$$

da cui, sottraendo membro a membro e passando ai valori assoluti, supponendo che E_m ed E_{2m} abbiano lo stesso segno, si ha

$$|T_{2m}[f] - T_m[f]| = |E_{2m}[f] - E_m[f]| \simeq 3|E_{2m}[f]|$$

e quindi

$$|E_{2m}[f]| \simeq |T_{2m}[f] - T_m[f]|/3.$$

L'errore $E_{2m}[f]$ può essere, quindi, stimato utilizzando la differenza tra le due formule $T_m[f]$ e $T_{2m}[f]$. ♣

È utile osservare che, sfruttando la proprietà di convergenza delle formule composte, se ad esempio è stata già calcolata la formula trapezoidale composta $T_m[f]$ per avere una stima più accurata dell'integrale (mantenendo l'equispaziatura tra i nodi) è possibile dimezzare la distanza tra i nodi, h , ottenendo così la formula $T_{2m}[f]$ i cui nodi hanno distanza $h/2$. Tale scelta comporta da un lato un miglioramento dell'accuratezza del risultato e dall'altro l'utilizzo del valore di $T_m[f]$ nel calcolo di $T_{2m}[f]$, richiedendo la valutazione della funzione integranda solo nei punti medi degli m sottointervalli $[x_{j-1}, x_j]$ di $[a, b]$:

$$T_{2m}[f] = \frac{1}{2}T_m[f] + \frac{h}{2} \left[f\left(a + \frac{h}{2}\right) + f\left(a + \frac{3h}{2}\right) + \dots + f\left(b - \frac{h}{2}\right) \right]. \quad (4.14)$$

In questo caso le due formule trapezoidali composite $T_m[f]$ e $T_{2m}[f]$ si dicono innestate.

Definizione 4.6. (Formule innestate)

Due formule $Q'[f]$ e $Q''[f]$, relative ad uno stesso intervallo, tali che l'insieme dei nodi di $Q'[f]$ è contenuto nell'insieme dei nodi di $Q''[f]$, costituiscono una **coppia di formule innestate**.

Data quindi una coppia di formule innestate $Q'[f]$ e $Q''[f]$ l'errore $E''[f] = I[f] - Q''[f]$ può essere stimato mediante la differenza:

$$|E''[f]| \simeq \alpha |Q'[f] - Q''[f]| \quad \alpha > 0 \quad (4.15)$$

dove α è una costante che dipende dalle formule di quadratura utilizzate.

La disponibilità di formule innestate è particolarmente utile per ridurre la complessità computazionale delle formule di quadratura (e più in generale di un algoritmo per la quadratura). Infatti essa è determinata attraverso **il numero di valutazioni della funzione integranda** nei nodi x_i , perchè il numero di operazioni relative a tale calcolo è predominante (la funzione integranda può infatti essere combinazione di polinomi algebrici, funzioni razionali, irrazionali, trigonometriche e così via).

4.3 Esempio di sviluppo di software matematico per la quadratura

4.3.1 L'algoritmo

I dati di un problema di quadratura sono la funzione integranda f , l'intervallo di integrazione $[a, b]$ ed una tolleranza To1 che rappresenta l'accuratezza richiesta. Il modo più naturale per progettare un algoritmo di quadratura è quindi quello di prevedere come dati di input tali valori⁷ e di produrre come output un valore Res (che sia un'approssimazione di $I[f]$) con un errore Err (che rappresenti una stima attendibile dell'errore $E[f]$) minore di To1 . Si noti quindi che un buon algoritmo di quadratura deve essere in grado di fornire, oltre ad una stima del valore dell'integrale, anche una stima dell'errore commesso.

Caratteristiche importanti di un tale algoritmo devono essere l'**affidabilità**, cioè la garanzia che il risultato soddisfi i vincoli della tolleranza richiesta, l'**efficienza**, cioè la capacità di calcolare il risultato con il minor numero possibile di valutazioni della funzione integranda e la **robustezza**, cioè la capacità dell'algoritmo di far fronte a situazioni anomale come ad esempio il caso di funzioni non integrabili.

⁷La funzione integranda f va fornita all'algoritmo sotto forma di procedura.

Strategia implementativa. Al fine di soddisfare i vincoli richiesti sull'errore, un algoritmo per la quadratura può essere basato su uno schema iterativo che calcoli una successione convergente ad $I[f]$. Tale successione può essere costruita in base al Teorema 4.1, sfruttando cioè la proprietà di convergenza delle formule composite al crescere del numero di sottointervalli in cui si divide l'intervallo di integrazione $[a, b]$.

Scelta della famiglia di formule. Poiché l'efficienza maggiore si raggiunge quando l'algoritmo calcola un risultato soddisfacente i requisiti di accuratezza con il minor numero di valutazioni di $f(x)$, le formule composite utilizzate in algoritmi di quadratura sono tali che, ad ogni passo dell'algoritmo, è possibile riutilizzare tutte, o quasi, le informazioni ottenute nei passi precedenti, in modo da ridurre il numero di valutazioni della funzione integranda. In particolare le formule più utilizzate sono quelle di tipo innestato come, ad esempio, le formule trapezoidali composite

$$\{T_m[f]\}_{m=2^k}, \quad k = 0, 1, 2, \dots \quad (T_1[f] \equiv T[f])$$

ottenute raddoppiando ad ogni passo il numero di intervalli.

Criterio di arresto. La scelta del criterio di arresto di una successione convergente al valore dell'integrale è legata alla disponibilità di stime ragionevoli ed affidabili dell'errore associato alle formule di quadratura utilizzate. Si è detto che una stima per la formula trapezoidale composta è fornita dalla (4.15) con $\alpha = 1/3$. In questo caso un criterio di arresto può essere basato sulla relazione⁸:

$$|T_m[f] - T_{2m}[f]|/3 < \text{To1} \quad (4.16)$$

Oltre a verificare se il risultato soddisfa la tolleranza (assoluta) richiesta To1 , il criterio di arresto dell'algoritmo deve prevedere un controllo sul numero di valutazioni della funzione integranda, il quale deve essere minore di una costante Maxval fornita in input, in quanto, come in tutti gli algoritmi iterativi, è possibile che un criterio di arresto basato solo sulla stima dell'errore sia soddisfatto solo dopo un numero eccessivo di passi. Una variabile di output Iflag segnala tale eventualità.

Tali considerazioni sono alla base della seguente Procedura 4.1.

⁸In generale i criteri di arresto delle procedure implementate nelle librerie di software matematico sono più sofisticati della (4.16) e spesso richiedono che l'errore relativo, o una combinazione dell'errore relativo e dell'errore assoluto sia minore di una tolleranza assegnata, cioè, ad esempio

$$\alpha|Q_m[f] - Q_{2m}[f]| < \max(\text{tol}_{abs}, \text{tol}_{rel}|Q_{2m}[f]|)$$

dove tol_{abs} e tol_{rel} sono rispettivamente la tolleranza richiesta per l'errore assoluto e per l'errore relativo. Controlli ulteriori possono essere effettuati anche sull'ampiezza degli intervalli. Infatti nel caso in cui la funzione integranda presenti delle singolarità, l'algoritmo può suddividere indefinitamente il dominio di integrazione senza generare un risultato soddisfacente la tolleranza richiesta.

```
procedure Trapez1(in: a, b, Tol, f, Maxval ; out: Int, Err, Iflag)  
  
  /# SCOPO: calcolo di un integrale definito mediante la formula  
    trapezoidale composta con stima dell' errore di  
    discretizzazione  
  
  /# SPECIFICHE PARAMETRI:  
  /# PARAMETRI DI INPUT:  
    var: a      : reale   {primo estremo dell' intervallo di integrazione}  
    var: b      : reale   {secondo estremo dell' intervallo di integrazione}  
    var: Tol    : reale   {tolleranza da soddisfare}  
    var: f      : reale   {funzione integranda}  
    var: Maxval : intero  {massimo numero di valutazioni}  
  
  /# PARAMETRI DI OUTPUT:  
    var: Int    : reale   {stima dell' integrale}  
    var: Err    : reale   {stima dell' errore di discretizzazione}  
    var: Iflag  : intero  {indicatore d' errore}
```

Procedura 4.1: algoritmo per la formula trapezoidale composta - continua

```

/# VARIABILI LOCALI:
var: h      : reale  {distanza tra i nodi}
var: x      : reale  {nodo attuale per il calcolo della formula}
                          {di quadratura}
var: oldint : reale  {formula trapezoidale composta}
                          {al passo precedente, necessaria}
                          {per il calcolo dell' errore}
var: sum    : reale  {somma parziale per la costruzione della}
                          {formula trapezoidale composta}
var: m      : intero {numero dei nodi per la formula}
                          {trapezoidale composta}
var: fval   : intero {numero di valutazioni della}
                          {funzione integranda}

/# INIZIO ISTRUZIONI:
  m := 2;
  h := (b - a)/m;
  x := (a + b)/m;
  oldint := h * (f(a) + f(b));
  Int := oldint/2 + h * f(x);
  fval := 3;
  Err := |Int - oldint|/3;
  while (Err > Tol and fval < Maxval) do
    {ciclo principale di}
    {iterazione con test sull'errore}
    m := 2 * m;
    h := h/2;
    sum := 0.;
    for k = 1, m, 2 do
      {calcolo della formula trapezoidale}
      {composita su m punti medi}
      x := a + k * h;
      sum := sum + f(x);

```

Procedura 4.1: algoritmo per la formula trapezoidale composta - continua

```

        fval := fval + 1;
    endfor
    oldint := Int;
    Int := oldint/2 + h * sum;
    Err := |Int - oldint|/3           { stima dell'errore }
endwhile
if (fval < Maxval) then           { definizione della variabile Iflag }
    Iflag := 0;
else
    Iflag := 1;
endif
end Trapez1

```

Procedura 4.1: algoritmo per la formula trapezoidale composta - fine

4.3.2 Il software

Lo scopo di una routine per il calcolo di un integrale definito su un intervallo $[a, b]$ è quello di rendere trasparente all'utente i vari dettagli relativi all'algoritmo implementato e di fornire il risultato a partire da un numero minimo di informazioni indispensabili da precisare in input. In particolare:

Definizione 4.7. (Integratore automatico)

Si dice **integratore automatico** (Figura 4.13) una routine basata su un algoritmo che richieda almeno i parametri di input:

- una routine per il calcolo di $f(x)$;
- gli estremi dell'intervallo a e b ;
- una tolleranza Tol da soddisfare;
- un limite di valutazioni massime, $Maxval$ per la funzione integranda;

e fornisca almeno i parametri di output:

- una stima Int di $I[f]$ ed Err di $E[f]$
- un indicatore $Iflag$ che avvisi l'utente se è stata raggiunta la tolleranza Tol oppure il motivo per cui non è stata raggiunta.

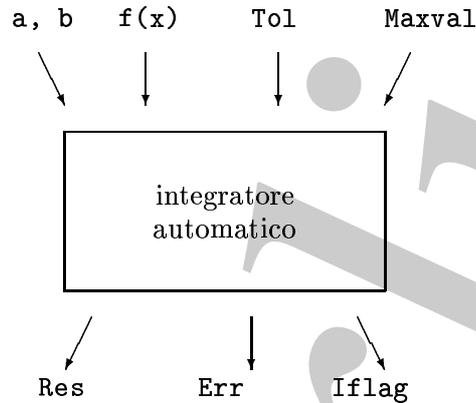


Figura 4.13: Schema di integratore automatico

A partire dalla procedura Trapez1 è possibile costruire un elemento di software matematico per la quadratura. L'utente deve fornire un programma chiamante ed una routine $F(X)$ che restituisca il valore della funzione integranda in ogni nodo previsto dalla formula di quadratura. Se, ad esempio, si utilizza il linguaggio di programmazione Fortran, la testata di tale subroutine può essere:

```
SUBROUTINE TRAPEZ1(A,B,F,TOL,MAXVAL,INT,ERR,IFLAG,FVAL)
```

Il programma chiamante INTEG illustra l'uso della subroutine TRAPEZ1 nel caso si voglia integrare la funzione $f(x) = \sqrt{x(1-x)}$ nell'intervallo $[0, 1]$ con una tolleranza $TOL=0.00001$ e $MAXVAL=10000$. Il parametro, $FVAL$, fornisce in output il numero di valutazioni di funzioni effettuate. Separatamente, nella FUNCTION $F(X)$, è implementata la funzione integranda. Tale unità di programma deve essere fornita dall'utente insieme al programma chiamante e nel caso si voglia calcolare l'integrale di una diversa funzione è sufficiente modificare solo l'istruzione $F=\text{SQRT}(X*(1.-X))$ che definisce la funzione.

```

        PROGRAM INTEG
C
C  dichiarazione delle variabili
C
        REAL A, B, TOL, INT, ERR, F
        INTEGER FVAL, IFLAG, MAXVAL
        EXTERNAL F
C
C  inizializzazione delle variabili di input
C
        A=0.
        B=1.
        TOL=.00001
        MAXVAL=10000
C
C  chiamata di TRAPEZ1
C
        CALL TRAPEZ1(A,B,F,TOL,MAXVAL,INT,ERR,IFLAG,FVAL)
C
C  stampa dei risultati
C
        IF (IFLAG.eq.0) THEN
            PRINT*, 'Integrale   =',INT
            PRINT*, 'Errore     =',ERR
            PRINT*, 'Val. fun.   =',FVAL
            PRINT*, 'Iflag      =',IFLAG
        ELSE
            PRINT*, 'tolleranza non raggiunta'
        ENDIF
        STOP
        END
C
C  FUNCTION F(X)
C
        REAL FUNCTION F(X)
        REAL X
        F=SQRT(X*(1.-X))
        RETURN
        END

```

Esempio di programma chiamante FORTRAN per la subroutine TRAPEZ1

Un programma FORTRAN, che implementa la versione in singola precisione della subroutine TRAPEZ1 e del programma chiamante precedente, è stato compilato ed ese-

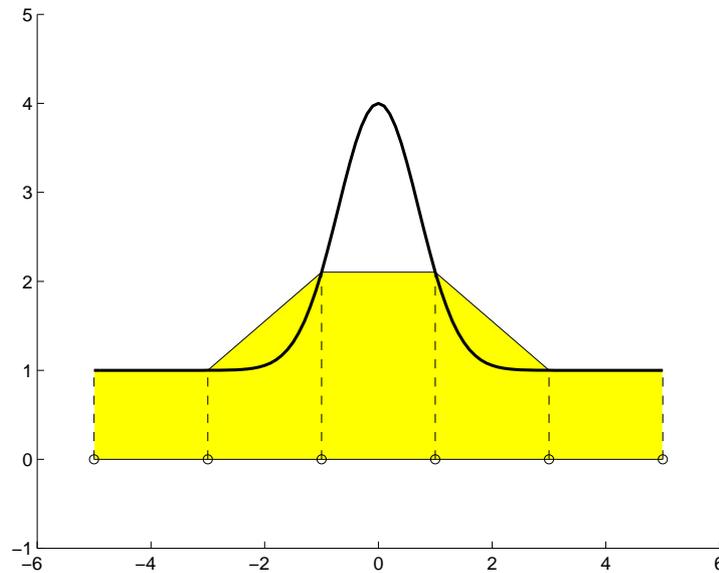


Figura 4.14: Approssimazione di $I[f]$ mediante la formula trapezoidale composta su 5 sottointervalli dell'intervallo $[-5, 5]$

guito su un calcolatore che utilizza il sistema aritmetico standard IEEE. L'esecuzione ha dato come risultati:

```

Integrale = 0.3926973641
Errore    = 0.3695487976E-05
Val. Fun. = 4097
Iflag     = 0

```

4.3.3 Gli algoritmi adattativi per la quadratura

Nel paragrafo 4.3.1 è stato sviluppato un algoritmo per la quadratura basato sulla famiglia di formule trapezoidali composte. Tale algoritmo dimezza successivamente l'ampiezza dei sottointervalli finché una stima dell'errore commesso non risulti minore di una tolleranza assegnata.

Qualunque sia la funzione integranda, l'algoritmo utilizza sempre la stessa distribuzione dei nodi nell'intervallo $[a, b]$ e ciò lo rende, in alcuni casi, poco efficiente.

♣ **Esempio 4.7.** Si calcoli

$$I[f] = \int_{-5}^5 (3e^{-x^2} + 1) dx$$

mediante la formula trapezoidale composta con 5 e 10 intervalli.

Nella Figura 4.14 si nota come la prima scelta dei nodi non tiene ben conto dell'andamento della funzione nell'intervallo $[-1, 1]$, mentre in Figura 4.15 si vede come raddoppiando il numero di intervalli non si migliora significativamente l'approssimazione di $I[f]$ negli intervalli $[-5, -2]$ e $[2, 5]$. ♣

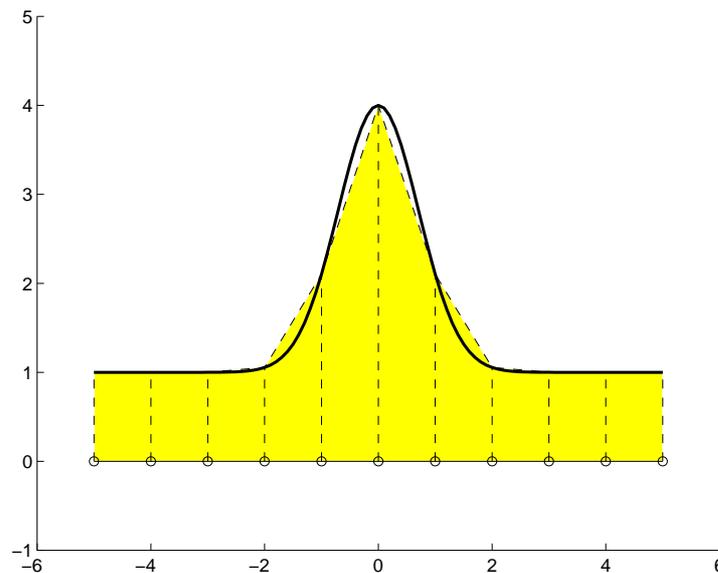


Figura 4.15: Approssimazione di $I[f]$ mediante la formula trapezoidale composta su 10 sottointervalli dell'intervallo $[-5, 5]$

L'esempio precedente mostra che la strategia di raddoppiare il numero degli intervalli della partizione di $[a, b]$ è in genere poco efficiente, in quanto non tiene conto dell'andamento della funzione integranda.

Una valida alternativa consiste nel cercare di adattare la distribuzione dei nodi alle proprietà della funzione integranda.

Definizione 4.8. (Algoritmo adattativo)

Un **algoritmo adattativo** per la quadratura è un algoritmo che sceglie dinamicamente (cioè durante l'esecuzione) la distribuzione dei nodi, in maniera da adattare il partizionamento dell'intervallo di integrazione al particolare andamento della funzione integranda. Un algoritmo in cui l'insieme dei nodi è scelto secondo uno schema fissato, indipendentemente dalla funzione integranda, è detto **algoritmo non adattativo**.

Gli algoritmi non adattativi possono essere utilizzati efficacemente nel caso in cui sia noto l'andamento della funzione integranda, scegliendo preventivamente la distribuzione più adatta alla funzione da integrare. Viceversa gli algoritmi adattativi sono più utili nel caso in cui non si hanno informazioni sufficienti sull'andamento della funzione, poiché, in questo caso, è l'algoritmo stesso a scegliere in maniera opportuna i nodi. All'interno della classe degli algoritmi adattativi sono possibili due strategie: la strategia **adattativa globale** e la strategia **adattativa locale**.

La **strategia adattativa globale** opera come segue. Sia ε la tolleranza richiesta sull'intervallo iniziale $\Delta = [a, b]$; vengono calcolati, in tale intervallo, l'integrale $Q(\Delta)$ (ad

esempio mediante la formula trapezoidale e la trapezoidale composta $T_2[f]$) e l'errore $E(\Delta)$ (mediante la (4.16) con $m = 1$). Se $E(\Delta) < \varepsilon$ allora l'algoritmo termina e $Q(\Delta)$ è il risultato. In caso contrario si divide Δ in due sottointervalli Δ_1 e Δ_2 nei quali vengono calcolati $Q(\Delta_i)$ ed $E(\Delta_i)$, $i = 1, 2$. Se $E = E(\Delta_1) + E(\Delta_2) < \varepsilon$ allora si termina e $Q = Q(\Delta_1) + Q(\Delta_2)$ è il risultato. In caso contrario si divide a metà il sottointervallo in cui $E(\Delta_i)$ è maggiore. In generale al k -mo passo dell'algoritmo, l'intervallo $[a, b]$ è diviso in $k+1$ intervalli disgiunti di differente ampiezza. Se $\sum E(\Delta_i) < \varepsilon$ l'algoritmo termina e $\sum Q(\Delta_i)$ è il risultato. Altrimenti, viene diviso a metà il sottointervallo $\bar{\Delta}$ con massima stima dell'errore, cioè:

$$|E(\bar{\Delta})| = \max_{i=1, \dots, k+1} |E(\Delta_i)|$$

La strategia adattativa globale può essere così riassunta:

```

Posto l' intervallo corrente  $\Delta = [a, b]$ , inizializza  $\varepsilon$ 
calcola un' approssimazione dell' integrale,  $Q(\Delta)$ ,
mediante una formula di quadratura opportuna e la formula di
quadratura composta
calcola  $E(\Delta)$  (mediante la (4.15), con  $\alpha$  opportuno)
while ( $E(\Delta) > \varepsilon$ )
    cerca il sottodominio  $\bar{\Delta}$  con stima dell'errore massima
    dividi  $\bar{\Delta}$  in 2 parti uguali
    calcola in tali parti una stima dell'integrale e dell'errore
    aggiorna  $Q(\Delta)$  e  $E(\Delta)$ 
endwhile

```

Algoritmo adattativo globale

Nella **strategia adattativa locale** al primo passo vengono calcolati $E(\Delta)$ e $Q(\Delta)$ analogamente alla strategia globale. Se $E(\Delta) < \varepsilon$ allora l'algoritmo termina e $Q(\Delta)$ è il risultato. In caso contrario si divide Δ in due sottointervalli Δ_1 e Δ_2 nei quali vengono calcolati $Q(\Delta_i)$ ed $E(\Delta_i)$, $i = 1, 2$. Si esamina dapprima l'intervallo Δ_1 . Fissata una *tolleranza locale* ε_1 (generalmente $\varepsilon_1 = \text{amp}(\Delta_1)\varepsilon/(b-a)$ con $\text{amp}(\Delta_1)$ ampiezza dell'intervallo Δ_1), se $E(\Delta_1) < \varepsilon_1$ si passa ad esaminare l'intervallo Δ_2 , altrimenti si divide ulteriormente Δ_1 . In generale, al passo k -mo l'intervallo $\Delta = [a, b]$ è diviso in $k+1$ sottointervalli. Detto Δ_k l'intervallo in esame, se $E(\Delta_k) < \varepsilon_k$ si passa ad esaminare Δ_{k+1} , altrimenti si suddivide ulteriormente Δ_k . Ogni volta che viene soddisfatto il criterio di tolleranza locale relativo al sottointervallo Δ_k viene aggiunto $Q(\Delta_k)$ ad una somma parziale che, al termine della procedura, fornirà il valore di $I[f]$. L'algoritmo termina quando l'ultimo intervallo soddisfa il criterio di tolleranza locale.

La strategia adattativa locale può essere così riassunta:

```

Posto l'intervallo corrente  $\Delta = [a, b]$ , inizializza  $\varepsilon$ 
calcola  $\overline{Q}(\Delta)$  mediante una formula di quadratura opportuna
repeat
  dividi l'intervallo corrente in 2 parti
  calcola  $Q(\Delta)$  mediante la formula di quadratura composita
  calcola  $E(\Delta)$  (mediante la (4.15), con  $\alpha$  opportuno)
  if ( $E(\Delta) > \varepsilon_{loc}$ ) then
    poni l'intervallo corrente  $\Delta =$  intervallo piu' a sinistra
  else
    somma  $Q(\Delta)$  al valore della somma parziale dell'integrale
    poni l'intervallo corrente  $\Delta =$  successivo intervallo
  endif
until ( intervalli tutti esaminati )

```

Algoritmo adattativo locale

La Figura 4.16 mostra come la routine di MATLAB `quad` per la quadratura, basata su un algoritmo adattativo, scelga i nodi per la funzione

$$f(x) = 3e^{-x^2} + 1 \quad \text{nell'intervallo} \quad [-2, 8].$$

Si nota che dove la funzione integranda presenta un picco, l'algoritmo utilizza dei sottointervalli di ampiezza inferiore rispetto ai sottointervalli in cui la funzione è pressoché costante.

Dal punto di vista implementativo la strategia globale può essere realizzata mediante l'uso di una lista ordinata⁹ secondo le stime crescenti degli errori $E(\Delta_i)$, in cui vengono conservate le informazioni relative a tutti gli intervalli esaminati (estremi degli intervalli e valori di $E(\Delta_i)$ e $Q(\Delta_i)$). In questo modo l'intervallo da suddividere (cioè quello con massima stima dell'errore) è sempre in testa alla lista. Viceversa la strategia locale può essere realizzata mediante l'uso di una pila¹⁰, costruita inserendo durante ogni suddivisione i due intervalli così ottenuti nella testa della pila, prima quello di destra poi quello di sinistra; in tal modo l'intervallo da esaminare (quello più a sinistra) si trova sempre in testa alla pila. Entrambe le strategie presentano vantaggi e svantaggi, anche se le routine più efficienti attualmente esistenti sono basate sulla strategia globale. Quest'ultima, infatti, a parità di tolleranza richiesta, utilizza generalmente un numero minore di valutazioni della funzione integranda rispetto alla strategia locale. Inoltre, se

⁹Una lista ordinata è una struttura dati dinamica (nel senso che la sua dimensione può variare durante l'esecuzione del programma). Tale struttura è composta da un insieme di nodi, ognuno dei quali formato da due campi: l'informazione e un puntatore al successivo nodo nella lista. Mediante il campo puntatore è possibile accedere ad ogni nodo della lista ed è possibile modificare l'ordine degli elementi solo intervenendo su tale campo.

¹⁰Una pila è una struttura dati dinamica in cui è possibile inserire ed estrarre gli elementi solo da un estremo della struttura dati. Tale estremo è detto *testa della pila*. Una struttura dati con tali caratteristiche è detta di tipo *Last In First Out* (LIFO).

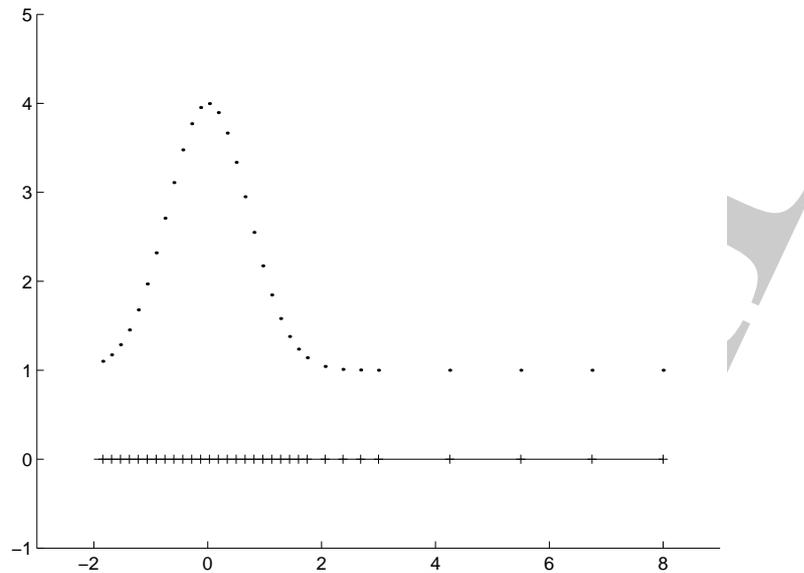


Figura 4.16: Scelta dei nodi da parte di una routine adattativa

l'algoritmo termina per aver raggiunto il massimo numero di valutazioni della funzione integranda, è sempre disponibile una stima di $I[f]$. Diversamente, la strategia locale produce una stima dell'integrale della funzione solo fino all'intervallo che ha esaminato per ultimo procedendo dall'estremo inferiore dell'intervallo $[a, b]$ verso quello superiore.

Tuttavia la strategia locale richiede un'occupazione minore di memoria per la pila, in quanto, se essa contiene ad esempio 50 sottointervalli, l'intervallo in testa alla pila ha ampiezza al più $2^{-50}(b-a) \simeq 10^{-15}(b-a)$ che, per usuali valori di $(b-a)$, è un numero minore della precisione relativa dei sistemi floating point utilizzati dalla maggior parte dei calcolatori. In tal caso si ha che la pila non conterrà mai più di 50 intervalli. Per tale motivo la strategia locale ha avuto un maggior successo rispetto alla strategia globale negli anni '70, quando la capacità limitata della memoria dei calcolatori costituiva un serio vincolo alla dimensione dei programmi eseguibili. Con la tecnologia odierna tale problema è superato e la maggior parte degli algoritmi adattativi sono basati sulla strategia globale.

Esaminiamo ora due algoritmi adattativi per la quadratura. Il primo algoritmo, chiamato `Qlocal` (Procedura 4.2), fa uso della strategia locale. In particolare esso utilizza una procedura `Trap(c, d, Q)` che calcola l'approssimazione Q di $I[f]$ in un generico intervallo $[c, d]$ mediante la formula trapezoidale. L'algoritmo utilizza tre pile per conservare rispettivamente i valori di Q e dei relativi estremi degli intervalli. Nei linguaggi di programmazione dove non sono previste¹¹, le tre pile possono essere realizzate con array monodimensionali, dove il primo elemento dell'array rappresenta il fondo della

¹¹Ad esempio nel linguaggio FORTRAN 77 non è prevista l'allocazione dinamica della memoria e quindi tutte le strutture dati dinamiche come la pila, la lista e la coda non possono essere definite. Tale problema è stato superato nel linguaggio FORTRAN 90 che permette l'allocazione dinamica della memoria come accade nel linguaggio C.

pila e l' n -mo ne rappresenta la testa. L'algoritmo effettua inoltre un controllo sul massimo numero di valutazioni della funzione integranda e sulla minima ampiezza di ogni intervallo¹².

```

procedure Qlocal(in: a, b, Tol, f, MaxvalMinlen; out: Int, Err, Iflag)

  /# SCOPO: calcolo di un integrale definito mediante un algoritmo
    adattativo locale

  /# SPECIFICHE PARAMETRI:
  /# PARAMETRI DI INPUT:
    var: a      : reale   {primo estremo dell' intervallo di integrazione}
    var: b      : reale   {secondo estremo dell' intervallo di integrazione}
    var: Tol    : reale   {tolleranza da soddisfare}
    var: f      : reale   {funzione integranda}
    var: Maxval : intero  {massimo numero di valutazioni}
    var: Minlen : reale   {minima ampiezza dell' intervallo}

  /# PARAMETRI DI OUTPUT:
    var: Int    : reale   {stima dell' integrale}
    var: Err    : reale   {stima dell' errore di discretizzazione}
    var: Iflag  : intero  {indicatore d' errore}

  /# VARIABILI LOCALI:
    var: Q1     : reale   {stima dell'integrale mediante la formula}
    {trapezoidale}
    var: Q2     : reale   {stima dell'integrale mediante la formula}
    {trapezoidale}
    var: Locerr : reale   {errore di discretizzazione della formula}
    {trapezoidale composta}
    var: p1(100) : reale   {array monodimensionale per la memorizzazione}
    {degli estremi sinistri degli intervalli}

```

Procedura 4.2: algoritmo adattativo locale - continua

¹²Si tenga presente che i controlli descritti non sono gli unici possibili. Numerose routine prevedono ad esempio controlli sul tempo massimo di esecuzione del programma, sull'accumulo degli errori di round-off o sulla dimensione massima delle aree di lavoro utilizzate per le strutture dati.

```

var: p2(100) : reale {array monodimensionale per la memorizzazione}
                        {degli estremi destri degli intervalli}
var: p3(100) : reale {array monodimensionale per la memorizzazione}
                        {dei risultati parziali}
var: inf      : reale {estremo sinistro dell' intervallo corrente}
var: sup      : reale {estremo destro dell' intervallo corrente}
var: res      : reale {stima corrente dell' integrale definito}
var: newres   : reale {stima dell' integrale mediante la formula}
                        {trapezoidale composta}

var: n        : intero {dimensione delle liste}
var: nval     : intero {numero di valutazioni di funzione}

/# INIZIO ISTRUZIONI:
    Int := 0.;
    Err := 0.;
    call Trap(a, b, Q1) {chiamata della routine}
    n := 1; {per la formula trapezoidale}
    nval := 2;
    p1(n) := a; {inizializzazione della pila}
    p2(n) := b;
    p3(n) := Q1;
    repeat {ciclo principale di iterazione}
        inf := p1(n); {il sottointervallo in testa}
        sup := p2(n); {alla pila viene suddiviso}
        res := p3(n); {in due parti}
        n := n - 1;
        call Trap(inf, (inf + sup)/2, Q1);
        call Trap((inf + sup)/2, sup, Q2);
        newres := Q1 + Q2;
        Locerr := |res - newres|/3;
        if (Locerr < Tol(sup - inf)/(b - a) {test sull'}

```

Procedura 4.2: algoritmo adattativo locale - continua

```

    or( $sup - inf \leq Minlen$ )then                                {accuratezza locale}
       $Int := Int + newres$ ;
       $Err := Err + Locerr$ ;
else
       $p1(n + 1) := (inf + sup)/2$ ;                                {test sull'accuratezza}
       $p2(n + 1) := sup$ ;                                         {fallito: i due nuovi}
       $p3(n + 1) := Q2$ ;                                         {sottodomini vengono}
       $p1(n + 2) := inf$ ;                                         {inseriti nella pila}
       $p2(n + 2) := (inf + sup)/2$ ;
       $p3(n + 2) := Q1$ ;
       $n := n + 2$ ;
endif
       $nval := nval + 4$ ;
until( $n = 0$  or  $nval \geq Maxval$ )
if( $n = 0$ )then                                               {definizione della variabile}
       $Iflag := 0$ ;                                             {Iflag}
else
       $Iflag := 1$ ;
endif
return  $Int, Err, Iflag$ 
end Qlocal

```

Procedura 4.2: algoritmo adattativo locale - fine

Il secondo algoritmo, chiamato `Qglobal` (Procedura 4.3), è invece basato su una strategia globale. Esso utilizza una procedura `quadra(a,b,Q,E)` per la stima del valore di $I[f]$ e dell'errore nell'intervallo $[a, b]$ che può essere realizzata mediante una qualunque coppia di formule di quadratura. Per conservare i sottodomini ancora da esaminare, sono state utilizzate 4 liste contenenti gli estremi degli intervalli (`alist` e `blist`), le stime dell'integrale nei vari sottodomini (`qlist`) e le stime dell'errore (`elist`). Tali liste sono ordinate secondo i valori dell'errore presenti in `elist`, in maniera che le informazioni relative all'intervallo con massima stima dell'errore si trovino sempre nella testa della lista. Anche in questo caso, nei linguaggi dove non sono previste, le liste ordinate possono essere realizzate mediante array monodimensionali dove la testa è rappresentata dall' n -mo elemento dell'array. Una procedura `sort(alist,blist,qlist,elist,n)`

è dedicata all'ordinamento. In realtà poichè si è interessati solo all'errore di massimo modulo, la procedura `sort` esegue solo la ricerca del massimo e lo posiziona nell' n -mo elemento delle liste. L'algoritmo termina quando si è raggiunto il massimo numero di valutazioni della funzione integranda o quando la stima dell'errore globale è minore di `Tol`.

```

procedure Qglobal(in: a, b, Tol, f, Maxval ; out: Int, Err, Iflag)

  /* SCOPO: calcolo di un integrale definito mediante un algoritmo
    adattativo globale */

  /* SPECIFICHE PARAMETRI: */
  /* PARAMETRI DI INPUT: */
  var: a      : reale  {primo estremo dell' intervallo di integrazione}
  var: b      : reale  {secondo estremo dell'intervallo di integrazione}
  var: Tol    : reale  {tolleranza da soddisfare}
  var: f      : reale  {funzione integranda}
  var: Maxval : intero {massimo numero di valutazioni}

  /* PARAMETRI DI OUTPUT: */
  var: Int    : reale  {stima dell' integrale}
  var: Err    : reale  {stima dell' errore di discretizzazione}
  var: Iflag  : intero {indicatore d' errore}

  /* VARIABILI LOCALI: */
  var: Q1     : reale  {prima approssimazione dell'integrale}
                          {mediante la formula trapezoidale}
  var: Q2     : reale  {stima dell'integrale mediante}
                          {la formula trapezoidale}
  var: c      : reale  {estremo sinistro dell' intervallo corrente}
  var: d      : reale  {estremo destro dell' intervallo corrente}
  var: Iold   : reale  {stima dell' integrale definito nel sottointervallo}
                          {in cui è massima la stima dell' errore}
  var: Eold   : reale  {massima stima dell' errore}

```

Procedura 4.3: algoritmo adattativo globale - continua

```

var: E           : reale   {errore di discretizzazione della formula}
                                {trapezoidale, sull' intervallo di integrazione}
var: E1        : reale   {errore di discretizzazione della formula}
                                {trapezoidale, sul sottintervallo sinistro}
var: E2        : reale   {errore di discretizzazione della formula}
                                {trapezoidale, sul sottintervallo destro}
var: alist(100) : reale   {array monodimensionale per la memorizza-}
                                {zione degli estremi sinistri degli intervalli}
var: blist(100) : reale   {array monodimensionale per la memorizza-}
                                {zione degli estremi destri degli intervalli}
var: qlist(100) : reale   {array monodimensionale per la memorizza-}
                                {zione dei risultati parziali}
var: elist(100) : reale   {array monodimensionale per la memorizza-}
                                {zione degli errori parziali}
var: n           : intero  {dimensione delle liste}
var: nval       : intero  {numero di valutazioni di funzione}

/# INIZIO ISTRUZIONI:

    call quadra(a, b, Q1, E) {prima approssimazione dell'integrale con la}
                                {formula trapezoidale mediante la}
                                {procedura quadra}
    nval := 3;                    {Si suppone che la complessita'}
    n := 1;                        {computazionale di quadra sia}
                                {di tre valutazioni di funzione}
    alist(n) := a;                {inizializzazione delle liste}

```

Procedura 4.3: algoritmo adattativo globale - continua

```

blist( $n$ ) :=  $b$ ;
qlist( $n$ ) :=  $Q1$ ;
elist( $n$ ) :=  $E$ ;
Int :=  $Q1$ ;
Err :=  $E$ ;
while( $Err > Toland$                                 {ciclo principale di}
  nval <  $Maxval$ )do                                {iterazione con test sull'errore}
   $c$  := alist( $n$ );                                  {il sottointervallo con massimo}
   $d$  := blist( $n$ );                                  {errore viene diviso}
   $Iold$  := qlist( $n$ );                               {in due sottointervalli}
   $Eold$  := elist( $n$ );
   $n$  :=  $n - 1$ ;
  call quadra( $c, (c + d)/2, Q1, E1$ )
  call quadra(( $c + d$ )/2,  $d, Q2, E2$ )
   $Int$  :=  $Int - Iold + Q1 + Q2$ ;                    {aggiornam. dei valori dell'integrale}
   $Err$  :=  $Err - Eold + E1 + E2$ ;                    {e dell'errore}
   $nval$  :=  $nval + 6$ ;
  alist( $n + 1$ ) :=  $c$ ;                               {i due nuovi sottointervalli}
  blist( $n + 1$ ) := ( $c + d$ )/2;                       {vengono aggiunti alla lista}
  qlist( $n + 1$ ) :=  $Q1$ ;
  elist( $n + 1$ ) :=  $E1$ ;
  alist( $n + 2$ ) := ( $c + d$ )/2;
  blist( $n + 2$ ) :=  $d$ ;
  qlist( $n + 2$ ) :=  $Q2$ ;
  elist( $n + 2$ ) :=  $E2$ ;
   $n$  :=  $n + 2$ ;

```

Procedura 4.3: algoritmo adattativo globale - continua

```
    call sort(alist, blist, qlist, elist, n)           {le liste vengono ordinate}
                                                         {in base agli errori; il}
                                                         {sottointervallo con massimo}
                                                         {errore si trova nell'n-mo nodo}

endwhile
if(nval < Maxval)then                                {definizione della variabile Iflag}
    Iflag := 0;
else
    Iflag := 1;
endif
return Int, Err, Iflag
end Qglobal
```

Procedura 4.3: algoritmo adattativo globale - fine

4.3.4 MATLAB e la quadratura

MATLAB è un *Problem Solving Environment* (PSE), per il calcolo numerico e la visualizzazione di dati.

Originariamente scritto per la risoluzione di problemi di algebra lineare¹³, si è evoluto negli anni ed attualmente copre tutti i principali campi della matematica computazionale.

Per l'integrazione numerica, MATLAB mette a disposizione due function: `quad` e `quad8`. Entrambe le function sono basate su un algoritmo automatico di tipo adattativo con strategia locale. Come nucleo computazionale `quad` utilizza la formula di Simpson, mentre `quad8` utilizza la formula di Newton-Cotes con 9 nodi¹⁴.

Per far uso della funzione `quad` è possibile utilizzare una delle funzioni seguenti:

```
>> res = quad('fun',a,b)
>> res = quad('fun',a,b,tol)
>> res = quad('fun',a,b,tol,graf)
```

Tale funzione restituisce nella variabile `res` il valore calcolato di un integrale definito tra i limiti `a` e `b` della funzione `fun`. Quest'ultima deve essere resa disponibile a `quad` mediante un file di nome `fun.m`.

♣ **Esempio 4.8.** Si calcoli:

$$I[f] = \int_0^{2\pi} |\sin(x)| dx = 4$$

mediante la funzione `quad` e si valuti il numero di operazioni floating point effettuate. È necessario creare il file `fun.m` contenente le operazioni che definiscono la funzione integranda:

```
function y=fun(x)
y = abs(sin(x));
end
```

e quindi eseguire i comandi:

```
>> a=0.;
>> b=2*3.1415926;
>> res = quad('fun',a,b)
res =
    4.00000207024612
>> flops
ans =
    840
```

¹³Il nome sta infatti per MATrix LABoratory.

¹⁴Le formule di Newton-Cotes sono trattate nel paragrafo 4.4.3



In generale `quad` calcola il valore dell'integrale con una tolleranza pari a 10^{-3} , a meno che non venga specificato il parametro opzionale `tol`.

♣ **Esempio 4.9.** Si calcoli l'integrale precedente, a meno di una tolleranza 10^{-4} . In tal caso è necessario eseguire i comandi:

```
>> a=0.;
>> b=2*3.1415926;
>> tol=10^(-4);
>> res = quad('fun',a,b,tol)
res =
    4.00000013081371
>> flops
ans =
    1688
```

Si noti come l'accuratezza maggiore richieda un maggior numero di operazioni floating point. ♣

Infine, se si utilizza il parametro opzionale `graf`, assegnando ad esso un valore diverso da 0, `quad` visualizza un grafico con i valori della funzione nei nodi utilizzati dalla routine per il calcolo dell'integrale. L'utilizzo di `quad8` è analogo a quello di `quad`. Entrambe le funzioni hanno un limite di 10 suddivisioni per ogni sottointervallo onde evitare di raffinare indefinitamente un sottointervallo nel caso in cui esso contenga un'eventuale singolarità. In tal caso viene prodotto il messaggio:

```
Recursion level limit reached in quad. Singularity likely.
```

che indica la presenza di tale situazione. Conseguenza di ciò è che sia `quad` sia `quad8` non sono in grado di calcolare con efficienza, ad esempio, integrali del tipo:

$$I[f] = \int_0^1 \frac{1}{\sqrt{x}} dx$$

♣ **Esempio 4.10.** Si calcoli l'integrale dell'esempio 4.8 nell'intervallo $[0, 5]$ con le due funzioni `quad` e `quad8`. Si valutino il numero di operazioni floating point e si mostrino anche i nodi utilizzati dalla funzione `quad8`. Per tale esempio si ha:

```

>> a=0.;
>> b=5;
>> tol=10^(-4);
>> graf = 1;
>> flops(0);
>> res = quad('fun',a,b,tol)
Recursion level limit reached
in quad. Singularity likely.
Recursion level limit reached
in quad. Singularity likely.
res =
    3.28366224139591
>> flops
ans =
    1116
>> flops(0);
>> res = quad8('fun',a,b,tol,graf)
res =
    3.28366221766898
>> flops
ans =
    1903

```

La presenza del parametro `graf` nella chiamata di `quad8` fa sì che, oltre al risultato venga visualizzata anche la Figura 4.17.

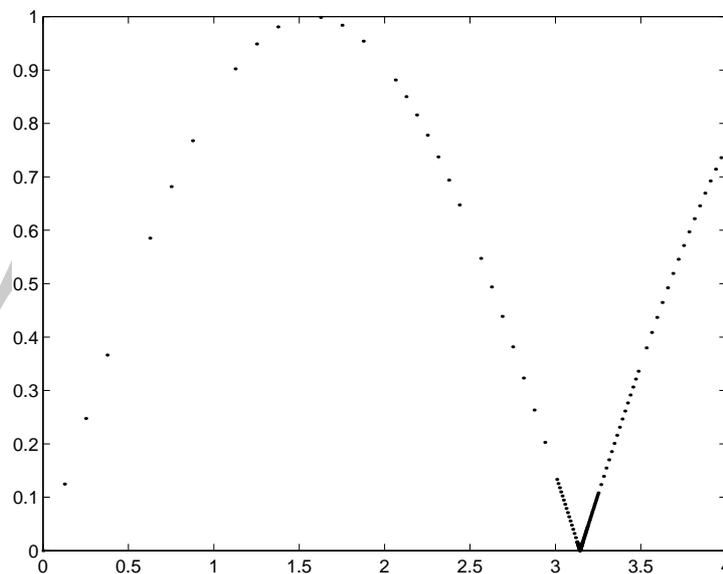


Figura 4.17: Generazione dei nodi da parte della funzione `quad8`

In questo caso la funzione `quad8` utilizza un maggior numero di nodi proprio in corrispondenza di π dove la derivata prima è singolare. Inoltre da un punto di vista computazionale la funzione `quad8` effettua un maggior numero di valutazioni di funzione rispetto a `quad`, ma è in grado di integrare con più accuratezza ed efficienza funzioni aventi discontinuità nella derivata prima. ♣

4.4 Principali formule di quadratura

4.4.1 Costruzione delle formule di quadratura

Dal teorema fondamentale del calcolo integrale è noto che

$$\int_a^b f(x) dx = F(b) - F(a)$$

con $F(x)$ funzione tale che $F'(x) = f(x)$ (funzione primitiva di $f(x)$).

♣ **Esempio 4.11.** Si calcoli:

$$I[f] = \int_1^3 \frac{dx}{x} = \ln 3 - \ln 1 = \ln 3 = 1.098612288668\dots \simeq 0.109861 \times 10^1$$

In questo caso si ottiene un risultato approssimato dell'integrale, dovuto all'approssimazione della funzione $\ln(x)$. ♣

♣ **Esempio 4.12.** Si calcoli:

$$I[f] = \int_0^1 e^{x^2} dx = \int_0^1 \sum_{i=0}^{\infty} \frac{x^{2i}}{i!}$$

Per calcolare tale integrale è necessario dapprima sviluppare in serie di Taylor la funzione integranda, quindi, troncare la serie ad un fissato termine L_{max} , cioè:

$$I[f] \simeq \sum_{i=0}^{L_{max}} \left[\frac{x^{2i+1}}{i!(2i+1)} \right]_0^1 \quad \text{ed} \quad E[f] = \sum_{i=L_{max}+1}^{\infty} \frac{x^{2i+1}}{i!(2i+1)}$$

♣

Il calcolo dell'integrale negli esempi precedenti richiede un'approssimazione numerica che riguarda la primitiva della funzione integranda, o la stessa funzione $f(x)$. L'idea alla base della quadratura è invece quella di sviluppare metodi che approssimino direttamente il valore di $I[f]$.

Al funzionale:

$$Q[f] = \sum_{i=1}^n A_i f(x_i) \tag{4.17}$$

si dà il nome di **formula di quadratura**, mentre al funzionale:

$$E[f] = I[f] - Q[f] = \int_a^b f(x) dx - \sum_{i=1}^n A_i f(x_i)$$

si dà il nome di **errore di discretizzazione**.

Si noti che la formula di quadratura (4.17) è un funzionale lineare così come $I[f]$ ed $E[f]$.

Il funzionale $Q[f]$ è determinato univocamente quando si fissa il valore di n ed i $2n$ parametri della formula di quadratura (n pesi e n nodi).

Uno degli obiettivi fondamentali della quadratura è far sì che l'errore $E[f]$ associato a $Q[f]$ sia "in qualche senso" il più piccolo possibile, in maniera da determinare una formula di quadratura "accurata".

A tal fine la ricerca dei pesi e dei nodi più opportuni può avvenire prevalentemente seguendo due strategie:

- si può richiedere di minimizzare il massimo errore su un generico spazio di funzioni \mathcal{G} , cioè:

$$\min_{A_i, x_i} \sup_{f \in \mathcal{G}} |E[f]| = \min_{A_i, x_i} \sup_{f \in \mathcal{G}} |I[f] - Q[f]| \quad (4.18)$$

dove il minimo è scelto al variare dei parametri della formula di quadratura (pesi e nodi). Tali formule sono note come **formule ottimali rispetto allo spazio di funzioni \mathcal{G}** .

- si può imporre che l'errore sia nullo per uno spazio di funzioni \mathcal{F} a dimensione finita, cioè

$$E[f] = I[f] - Q[f] = 0 \quad \forall f \in \mathcal{F}. \quad (4.19)$$

Tali formule sono note come **formule di annullamento dell'errore** oppure come **formule esatte per lo spazio di funzioni \mathcal{F}** .

Usualmente, per semplificare la costruzione delle formule di quadratura, in entrambi gli approcci è possibile fissare delle condizioni sui parametri; per esempio si può richiedere che i pesi siano uguali o, più comunemente, si possono fissare a priori i nodi.

Per la loro maggiore semplicità rispetto alle formule ottimali, le formule esatte per uno spazio di funzioni sono quelle più diffuse e sono perciò quelle che considereremo. D'altra parte esse sono anche quelle maggiormente utilizzate per la realizzazione di software matematico per la quadratura.

4.4.2 Formule esatte per uno spazio di funzioni

Le formule esatte per uno spazio di funzioni \mathcal{F} sono formule di quadratura che, applicate ad ogni funzione $f \in \mathcal{F}$, forniscono un risultato esatto, nel senso che annullano l'errore $E[f]$.

Fissato uno spazio di funzioni \mathcal{F} di dimensione finita q e scelta una sua base $\{\phi_r(x)\}$, $r = 1, \dots, q$, $\forall f \in \mathcal{F}$ sussiste una relazione del tipo:

$$f(x) = \sum_{r=1}^q b_r \phi_r(x), \quad \forall f \in \mathcal{F}$$

da cui, per la linearità del funzionale $E[f]$, si ha:

$$E[f] = E \left[\sum_{r=1}^q b_r \phi_r(x) \right] = b_1 E[\phi_1(x)] + \dots + b_q E[\phi_q(x)].$$

Pertanto, costruire una formula di quadratura esatta per \mathcal{F} equivale a determinare i nodi ed i pesi in maniera tale che risulti verificata la relazione:

$$E[\phi_r(x)] = 0, \quad r = 1, \dots, q. \tag{4.20}$$

La (4.20) rappresenta un sistema di equazioni non lineari nelle incognite x_i ed A_i . Se si fissano a priori gli n nodi distinti, il sistema (4.20) equivale ad un sistema lineare di q equazioni nelle n incognite A_i ($i = 1, \dots, n$). In tal caso il sistema è compatibile e determinato se e solo se $q = n$. Il calcolo dei pesi A_i mediante il sistema (4.20) è detto **metodo dei coefficienti indeterminati**.

Usualmente, lo spazio di funzioni \mathcal{F} che si sceglie è quello dei polinomi Π_n di un grado n fissato. La scelta di questo spazio è essenzialmente spiegata dal fatto che le sue proprietà sono conosciute meglio rispetto a quelle di altri spazi di funzione¹⁵.

♣ **Esempio 4.13.** Si determini una formula di quadratura nell'intervallo $[-1, 1]$ tale che $E[f] = 0 \forall f \in \Pi_2$. Per la linearità del funzionale $E[f]$, la relazione precedente si verifica se e solo se

$$E[f] = E[b_2 x^2 + b_1 x + b_0] = \sum_{r=0}^2 b_r E[x^r] = 0,$$

cioè se e solo se $E[x^r] = 0$ per $r = 0, 1, 2$. Si ha quindi:

$$\begin{cases} A_1 + A_2 + \dots + A_n = \int_{-1}^1 dx = 2 \\ A_1 x_1 + A_2 x_2 + \dots + A_n x_n = \int_{-1}^1 x dx = 0 \\ A_1 x_1^2 + A_2 x_2^2 + \dots + A_n x_n^2 = \int_{-1}^1 x^2 dx = 2/3 \end{cases}$$

Il sistema precedente è un sistema non lineare di 3 equazioni nelle $2n$ incognite costituite dai nodi e dei pesi. Se i nodi x_i vengono fissati, ad esempio equidistanziati, si ottiene un sistema di equazioni lineare nelle n incognite A_i . Per garantire la compatibilità del sistema si deve porre quindi $n = 3$ in modo da avere un sistema lineare di 3 equazioni in 3 incognite, la cui matrice dei coefficienti è

$$V = \begin{pmatrix} 1 & 1 & 1 \\ -1 & 0 & 1 \\ 1 & 0 & 1 \end{pmatrix}$$

Poichè V è non singolare il sistema ha l'unica soluzione:

$$A_1 = A_3 = \frac{1}{3} \quad A_2 = \frac{4}{3}$$

¹⁵Se $p(x)$ e $q(x)$ sono due polinomi, $p(x) \pm q(x)$, $p(x) \cdot q(x)$ e $p(q(x))$ sono anch'essi dei polinomi. Inoltre anche la derivata $p'(x)$ e l'integrale $\int p(x) dx$ sono dei polinomi. I polinomi sono anche strettamente collegati alle funzioni analitiche attraverso gli sviluppi in serie di potenze ed alle funzioni continue attraverso il teorema di Weierstrass. Un'ultima (ma non meno importante) proprietà dei polinomi è che essi sono rappresentati da espressioni per la cui valutazione sono noti algoritmi efficienti.

da cui:

$$Q[f] = \frac{1}{3}(f(-1) + 4f(0) + f(1)).$$

Tale formula è nota come formula di Simpson, già ottenuta per altra via nel paragrafo 4.2 nel generico intervallo $[a, b]$. ♣

Definizione 4.9. (Grado di precisione)

Una formula di quadratura ha **grado di precisione** q se è esatta per uno spazio di funzioni di dimensione al più q . In particolare si parla di **grado di precisione algebrico** nel caso in cui \mathcal{F} coincide con lo spazio dei polinomi Π_q o di **grado di precisione trigonometrico** se si considera lo spazio dei polinomi trigonometrici del tipo:

$$S_q(x) = a_0 + \sum_{k=1}^q (a_k \cos kx + b_k \sin kx)$$

In generale, se si sceglie come famiglia di funzioni \mathcal{F} quella dello spazio Π_q dei polinomi algebrici al più di grado q , si ottiene un sistema di $q+1$ equazioni in n incognite:

$$\begin{cases} A_1 + A_2 + \cdots + A_n & = \int_a^b dx \\ A_1 x_1 + A_2 x_2 + \cdots + A_n x_n & = \int_a^b x dx \\ \cdots & \cdots \\ A_1 x_1^q + A_2 x_2^q + \cdots + A_n x_n^q & = \int_a^b x^q dx \end{cases} \quad (4.21)$$

la cui matrice dei coefficienti è:

$$V = \begin{pmatrix} 1 & \cdots & 1 \\ x_1 & & x_n \\ \vdots & & \vdots \\ x_1^q & \cdots & x_n^q \end{pmatrix} \quad (4.22)$$

Si osservi che tale sistema ha un'unica soluzione se e solo se $q = n - 1$ e $x_i \neq x_j$, $\forall i \neq j$. Ciò vuol dire che le formule i cui pesi soddisfano il sistema precedente, hanno grado di precisione algebrico $n - 1$. Inoltre, dalla prima equazione si ha

$$A_1 + \cdots + A_n = \int_a^b dx = (b - a) \quad (4.23)$$

cioè la somma dei pesi di una formula di quadratura esatta per lo spazio dei polinomi è uguale all'ampiezza dell'intervallo di integrazione.

Si noti che la matrice (4.22) è una matrice di Vandermonde il cui indice di condizionamento, $\mu(V)$, cresce almeno esponenzialmente con la dimensione n (cfr. §B.1.4) e quindi il relativo sistema (4.21) è malcondizionato.

Un modo alternativo per il calcolo dei pesi si basa sull'osservazione che una formula esatta per lo spazio dei polinomi Π_{n-1} è ovviamente esatta anche per i polinomi fondamentali di Lagrange $l_j(x)$, nei nodi x_1, \dots, x_n ovvero

$$\int_a^b l_j(x) dx = \sum_{i=1}^n A_i l_j(x_i) = A_i \quad j = 1, \dots, n$$

Ricordando l'espressione dei polinomi fondamentali di Lagrange (equazione (3.17) del paragrafo 3.2.5), ciascun l_i si può scrivere come:

$$l_i(x) = \frac{\omega(x)}{(x - x_i)\omega'(x_i)}, \quad \text{con } \omega(x) = (x - x_1) \cdots (x - x_n)$$

da cui si deduce, facilmente, che i pesi hanno espressione:

$$A_i = \int_a^b \frac{\omega(x)}{(x - x_i)\omega'(x_i)} dx, \quad \text{con } \omega(x) = (x - x_1) \cdots (x - x_n) \quad (4.24)$$

Si noti però che anche questo metodo è poco efficace, per la necessità del calcolo dell'integrale (4.24).

Da quanto detto si evince che, in generale, è consigliabile determinare formule di quadratura con un grado di precisione algebrico basso ed utilizzarle opportunamente come descritto nel paragrafo 4.2.1 (formule composite). Altre ragioni saranno approfondite in seguito.

4.4.3 Formule di Newton-Cotes e formule di Gauss

Un modo per ottenere formule esatte per lo spazio dei polinomi Π_{n-1} , è quello di fissare gli n nodi x_i con $x_i \neq x_j$ per $i \neq j$. Benché la scelta di tali nodi possa essere del tutto arbitraria, una scelta naturale è quella di fissarli equidistanziati nell'intervallo $[a, b]$, cioè:

$$x_1 = a < x_2 = a + h < \dots < x_n = a + (n - 1)h = b \quad h = \frac{(b - a)}{(n - 1)}$$

In tal caso, ad esempio, si ha:

$n = 2$ Formula trapezoidale:

$$T[f] = h \left[\frac{f(a)}{2} + \frac{f(b)}{2} \right] \quad h = (b - a)$$

esatta per polinomi di grado 1.

$n = 3$ Formula di Simpson:

$$S[f] = h \left[\frac{f(a)}{3} + \frac{4}{3}f(a + h) + \frac{f(b)}{3} \right] \quad h = \frac{(b - a)}{2}$$

esatta per polinomi di grado 2.

$n = 4$ **Formula dei '3/8':**

$$Q[f] = h \left[\frac{3}{8}f(a) + \frac{9}{8}f(a+h) + \frac{9}{8}f(a+2h) + \frac{3}{8}f(b) \right] \quad h = \frac{(b-a)}{3}$$

esatta per polinomi di grado 3.

Dalle espressioni precedenti si deduce che le formule così ottenute si possono esprimere come:

$$Q[f] = h [B_1 f(x_1) + \cdots + B_n f(x_n)] \quad \text{con} \quad h = \frac{b-a}{n-1} \quad (4.25)$$

dove i valori B_i sono **indipendenti** dall'intervallo di integrazione $[a, b]$ e dalla funzione integranda, ma dipendono solo dal numero di nodi n . Inoltre i pesi relativi a nodi simmetrici rispetto al punto medio dell'intervallo $[a, b]$ sono uguali, cioè:

$$B_i = B_{n-i+1} \quad i = 1, 2, \dots, \left\lfloor \frac{n+1}{2} \right\rfloor$$

Per essi è quindi possibile una tabulazione opportuna (Tabella 4.1).

Definizione 4.10. (Formule di Newton-Cotes)

Le formule di quadratura con n nodi equidistanziati, esatte per lo spazio dei polinomi Π_{n-1} , sono dette **formule di Newton-Cotes**.

Dalla Tabella 4.1 si nota che per $n \leq 8$ e per $n = 10$ i pesi B_i sono tutti positivi¹⁶, mentre per $n = 9$ così come per $n \geq 11$, i pesi B_i sono di segno alterno e sono crescenti in modulo. La formula con $n = 10$ ha grado di precisione 9 e rappresenta la formula di Newton-Cotes con pesi tutti positivi, con grado di precisione algebrico più alto.

Un'espressione asintotica, al variare di n dei coefficienti B_i per le formule di Newton-Cotes è fornita dalle relazioni seguenti. Se $Q[f]$ è la formula di Newton-Cotes in $[0, 1]$, abbiamo una rappresentazione asintotica dei suoi pesi B_i [2]:

$$B_0 = B_n = \frac{1}{n \log n} \left[1 + \mathcal{O} \left(\frac{1}{\log n} \right) \right] \quad (4.26)$$

e, per ciascun $i = 1, 2, \dots, n-1$

$$B_i = \frac{(-1)^{i-1} \binom{n}{i}}{n \log^2 n} \left[\frac{1}{i} + \frac{(-1)^n}{n-i} \right] \left[1 + \mathcal{O} \left(\frac{1}{\log n} \right) \right] \quad (4.27)$$

dove $\mathcal{O} \left(\frac{1}{\log n} \right)$ è un infinitesimo di ordine uguale a $\frac{1}{\log n}$.

¹⁶Come si vedrà nel paragrafo 4.6, tale caratteristica influenza fortemente il condizionamento di una formula di quadratura.

formula	n	$B_1 = B_n$	$B_2 = B_{n-1}$	$B_3 = B_{n-2}$	$B_4 = B_{n-3}$	$B_5 = B_{n-4}$
trapezoidale	2	$\frac{1}{2}$				
Simpson	3	$\frac{1}{3}$	$\frac{4}{3}$			
dei '3/8'	4	$\frac{3}{8}$	$\frac{9}{8}$			
Boole	5	$\frac{14}{45}$	$\frac{64}{45}$	$\frac{24}{45}$		
N-C 6 nodi	6	$\frac{95}{288}$	$\frac{375}{288}$	$\frac{250}{288}$		
N-C 7 nodi	7	$\frac{41}{140}$	$\frac{216}{140}$	$\frac{27}{140}$	$\frac{272}{140}$	
N-C 8 nodi	8	$\frac{5257}{17280}$	$\frac{25039}{17280}$	$\frac{9261}{17280}$	$\frac{20923}{17280}$	
N-C 9 nodi	9	$\frac{3956}{14175}$	$\frac{23552}{14175}$	$\frac{-3712}{14175}$	$\frac{41984}{14175}$	$\frac{-18160}{14175}$
N-C 10 nodi	10	$\frac{25713}{89600}$	$\frac{141669}{89600}$	$\frac{9720}{89600}$	$\frac{174096}{89600}$	$\frac{52002}{89600}$

Tabella 4.1: Coefficienti delle formule di Newton-Cotes con $n \leq 10$

Si noti, infine, che se $Q^{(j)}[f]$ è una formula di Newton-Cotes, nel generico intervallo $[t_{j-1}, t_j]$ di ampiezza $\tau = (b - a)/m$, per la (4.25) essa avrà espressione

$$Q^{(j)}[f] = h \sum_{i=1}^n B_i f(x_i^{(j)}) \quad \text{con} \quad h = \frac{\tau}{n-1} = \frac{(b-a)}{m(n-1)}$$

con i pesi B_i opportunamente scelti nella Tabella 4.1. In tal caso la formula composta corrispondente è:

$$Q_m[f] = \sum_{j=1}^m Q^{(j)}[f] = h \sum_{j=1}^m \sum_{i=1}^n B_i f(x_i^{(j)})$$

con

$$h = \frac{(b-a)}{m(n-1)} \quad \text{e} \quad x_i^{(j)} = t_{j-1} + (i-1)h.$$

Una strategia alternativa per costruire formule di quadratura esatte per uno spazio di polinomi è quella di non fare alcuna ipotesi a priori sulla disposizione dei nodi.

♣ **Esempio 4.14.** Determinare una formula di quadratura nell'intervallo $[-1,1]$ con il grado di precisione algebrico più alto possibile, utilizzando 2 nodi. Tale formula ha espressione:

$$Q[f] = A_1 f(x_1) + A_2 f(x_2).$$

Essendo disponibili 4 parametri (2 pesi e 2 nodi) è ragionevole integrare esattamente un qualunque polinomio di terzo grado. A tal fine è sufficiente imporre che $Q[f]$ sia esatta solo per le funzioni che costituiscono una base per lo spazio Π_3 . Cioè:

$$\mathcal{F} = \{1, x, x^2, x^3\}.$$

Si ha così il sistema:

$$\begin{cases} A_1 & + A_2 & = 2 \\ A_1 x_1 & + A_2 x_2 & = 0 \\ A_1 x_1^2 & + A_2 x_2^2 & = 2/3 \\ A_1 x_1^3 & + A_2 x_2^3 & = 0 \end{cases} \quad (4.28)$$

Il sistema precedente è non lineare. Se si osserva che l'intervallo $[-1, 1]$ è simmetrico rispetto allo 0, è plausibile che la formula cercata (e quindi i nodi) conservi tale caratteristica, cioè $x_2 = -x_1$. Dalla seconda equazione di (4.28) segue allora che $A_1 = A_2$ e dalla prima equazione si ricava che $A_1 = A_2 = 1$. Dalla terza equazione di (4.28) segue infine che

$$x_1^2 = \frac{1}{3} \Leftrightarrow x_1 = \frac{-1}{\sqrt{3}}, \quad x_2 = \frac{1}{\sqrt{3}}$$

Si ha quindi la formula:

$$Q[f] = f\left(\frac{-1}{\sqrt{3}}\right) + f\left(\frac{1}{\sqrt{3}}\right)$$

La formula così ottenuta ha grado di precisione algebrico 3 ed è detta **formula di Gauss** con 2 nodi. ♣

Teorema 4.3. *Sia $G[f]$ una formula di quadratura del tipo*

$$G[f] = \sum_{i=1}^n A_i f(x_i) \quad (4.29)$$

Condizione necessaria e sufficiente affinché $G[f]$ abbia grado di precisione algebrico $2n - 1$ è che sia esatta per polinomi $r(x)$ di grado al più $n - 1$, cioè tale che:

$$\int_a^b r(x) dx = \sum_{i=1}^n A_i r(x_i) \quad \forall r \in \Pi_{n-1} \quad (4.30)$$

e che abbia come nodi x_i $i = 1, \dots, n$ gli n zeri di un polinomio $\omega(x) \in \Pi_n$ ortogonale¹⁷ a tutti i polinomi $q(x)$ di grado al più $n - 1$, cioè tale che:

$$\int_a^b \omega(x) q(x) dx = 0 \quad \forall q \in \Pi_{n-1} \quad (4.31)$$

¹⁷Si ricorda che due polinomi p e q sono ortogonali nell'intervallo $[a, b]$ se e solo se

$$\int_a^b p(x)q(x) dx = 0.$$

Dimostrazione Se $G[f]$ ha grado di precisione $2n - 1$ la (4.30) è banalmente soddisfatta. Siano x_1, \dots, x_n i nodi di $G[f]$ e sia $q(x)$ un generico polinomio di grado al più $n - 1$. Posto allora $\omega(x) = (x - x_1) \cdots (x - x_n)$ e $p(x) = \omega(x)q(x)$ si ha che $p \in \Pi_{2n-1}$ e per tale polinomio, per ipotesi, si deve avere $E[p] = 0$, cioè:

$$0 = E[p] = \int_a^b p(x) dx - \sum_{i=1}^n A_i p(x_i) = \int_a^b \omega(x)q(x) dx - \sum_{i=1}^n A_i \omega(x_i)q(x_i)$$

ma poiché $\omega(x_i) = 0, i = 1, \dots, n$, si deve avere

$$\int_a^b \omega(x)q(x) dx = 0$$

cioè la (4.31).

Viceversa, siano verificate le condizioni (4.30) e (4.31), e sia $p(x)$ un generico polinomio di grado al più $2n - 1$. Detti x_1, \dots, x_n i nodi di $G[f]$ e posto $\omega(x) = (x - x_1) \cdots (x - x_n)$, dividendo $p(x)$ per $\omega(x)$ si ottiene:

$$p(x) = \omega(x)q(x) + r(x)$$

con $q(x)$ e $r(x)$ polinomi di grado al più $n - 1$, rispettivamente quoziente e resto della divisione. Si ha quindi che

$$\begin{aligned} E[p] &= \int_a^b p(x) dx - \sum_{i=1}^n A_i p(x_i) \\ &= \int_a^b \omega(x)q(x) dx + \int_a^b r(x) dx - \sum_{i=1}^n A_i \omega(x_i)q(x_i) - \sum_{i=1}^n A_i r(x_i) \end{aligned}$$

da cui, per le condizioni (4.30) e (4.31) ed osservando che $\omega(x_i) = 0, i = 1, \dots, n$, si ha $E[p] = 0 \quad \forall p \in \Pi_{2n-1}$.

Si ha poi che per il polinomio di grado $2n$:

$$p(x) = (x - x_1)^2 \cdots (x - x_n)^2$$

si avrà sempre:

$$\sum_{i=1}^n A_i p(x_i) = 0 \quad e \quad \int_a^b p(x) dx > 0$$

per ogni scelta dei pesi e dei nodi, per cui $2n - 1$ è effettivamente il massimo grado di precisione possibile per una formula di quadratura di annullamento dell'errore con n nodi. ■

Definizione 4.11. (Formula di Gauss)

Una formula di quadratura su n nodi

$$G[f] = \sum_{i=1}^n A_i f(x_i) \tag{4.32}$$

avente grado di precisione algebrico $2n - 1$ è detta **formula di Gauss**.

Ma, come già detto per le formule di Newton-Cotes, anche per le formule di Gauss la matrice del sistema lineare proveniente dal metodo dei coefficienti indeterminati ha indice di condizionamento elevato al crescere di n . Pertanto per il calcolo dei pesi A_i delle formule di Gauss si utilizza un'altra rappresentazione.

A tal fine, fissato un sistema di polinomi ortogonali $\{p_n(x)\}$ nell'intervallo $[a, b]$, si consideri il sistema di polinomi ortonormali $\{p_n^*(x)\}$ corrispondente, ottenuto ponendo:

$$p_n^*(x) = \frac{p_n(x)}{\|p_n(x)\|_2}$$

con:

$$\|p_n(x)\|_2^2 = \int_a^b p_n^2(x) dx$$

Teorema 4.4. *Fissati i nodi secondo la (4.31), i pesi A_i delle formule di Gauss sono forniti dalla relazione seguente:*

$$A_i = -\frac{d_{n+1}}{d_n p_{n+1}^*(x_i) p_n^{*'}(x_i)} \quad i = 1, \dots, n \quad (4.33)$$

dove d_n è il coefficiente direttore del polinomio $p_n^*(x)$ e x_i è l' i -mo nodo della formula di Gauss con n nodi.

Dimostrazione La dimostrazione parte dall'uguaglianza nota come uguaglianza di Christoffel-Darboux:

$$(x-t) \sum_{k=0}^n p_k^*(x) p_k^*(t) = \frac{d_n}{d_{n+1}} [p_{n+1}^*(x) p_n^*(t) - p_n^*(x) p_{n+1}^*(t)]$$

Valutando tale uguaglianza per $t = x_i$ e dividendo ambo i membri per $x - x_i$, poiché $p_n^*(x_i) = 0$, si ottiene:

$$\sum_{k=0}^{n-1} p_k^*(x) p_k^*(x_i) = -\frac{d_n}{d_{n+1}} \frac{p_n^*(x) p_{n+1}^*(x_i)}{x - x_i}$$

Integrando entrambi i membri di tale uguaglianza nell'intervallo $[a, b]$ si ha:

$$\sum_{k=0}^{n-1} \int_a^b p_k^*(x) p_k^*(x_i) dx = -\frac{d_n}{d_{n+1}} \int_a^b \frac{p_n^*(x) p_{n+1}^*(x_i)}{x - x_i} dx \quad (4.34)$$

per l'ortonormalità del sistema di polinomi $p_0^*(x), \dots, p_n^*(x)$ si ha:

$$p_k^*(x_i) \int_a^b p_k^*(x) dx = \begin{cases} 0 & \text{se } k \geq 1 \text{ per l'ortogonalità} \\ 1 & \text{se } k = 0 \text{ per la normalità} \end{cases}$$

per cui la (4.34) diventa:

$$1 = -\frac{d_n}{d_{n+1}} p_{n+1}^*(x_i) \int_a^b \frac{p_n^*(x)}{x - x_i} dx \quad (4.35)$$

Poichè le formule di Gauss sono formule esatte per lo spazio dei polinomi, per i pesi di tali formule vale la (4.24), o equivalentemente:

$$A_i = \int_a^b \frac{p_n^*(x)}{(x - x_i)p_n^{*\prime}(x_i)} dx,$$

per cui, moltiplicando e dividendo per $p_n^{*\prime}(x_i)$ il secondo membro della (4.35) si ha la tesi. ■

Utilizzando la (4.33) è possibile calcolare i pesi delle formule di Gauss per un numero qualunque di nodi. Su tali espressioni sono basate ad esempio le routine D01BBF e D01BCF della libreria NAG per il calcolo dei pesi di varie formule di Gauss.

Teorema 4.5. *Data una formula di Gauss*

$$G[f] = \sum_{i=1}^n A_i f(x_i)$$

si ha che $A_i > 0 \quad \forall i = 1, \dots, n$.

Dimostrazione Sia $\omega(x)$ il polinomio ortogonale di grado n nell'intervallo $[a, b]$. Detti x_1, \dots, x_n gli n nodi di $G[f]$ si ha che $\omega(x_i) = 0$. Per ogni fissato indice j , sia quindi $q(x) = \omega(x)/(x - x_j)$. Poichè gli zeri di $\omega(x)$ sono tutti semplici si ha:

$$q(x_i) \begin{cases} = 0 & \text{se } i \neq j \\ \neq 0 & \text{se } i = j \end{cases}$$

Poichè inoltre $q^2(x)$ è un polinomio di grado $2n - 2$, per tale polinomio la formula $G[f]$ è esatta. Di conseguenza si ha:

$$0 < \int_a^b q^2(x) dx = \sum_{i=1}^n A_i q^2(x_i) = A_j q^2(x_j)$$

da cui la tesi. ■

È immediata l'estensione dei teoremi precedenti, al caso in cui si voglia calcolare l'integrale:

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

con $\psi(x)$ funzione peso ammissibile, cioè tale che

- $\psi(x) \geq 0$ e non quasi ovunque nulla;
- $\psi(x)f(x)$ integrabile in $[a, b]$;

$\psi(x)$	intervallo	tipo
1	$[-1,1]$	Gauss-Legendre
$(1-x^2)^{-1/2}$	$[-1,1]$	Gauss-Chebyshev (I)
$(1-x^2)^{1/2}$	$[-1,1]$	Gauss-Chebyshev (II)
$(1-x)^\alpha(1+x)^\beta$ $\alpha, \beta > -1$	$[-1,1]$	Gauss-Jacobi
e^{-x}	$[0, \infty[$	Gauss-Laguerre
$e^{-x^2/2}$	$]-\infty, \infty[$	Gauss-Hermite

Tabella 4.2: Funzioni peso ed intervalli relativi alle principali formule di Gauss

- $|\int_a^b \psi(x)x^k dx| < \infty \quad \forall k = 0, 1, \dots$

In generale, solo se la funzione peso $\psi(x)$ è ammissibile è assicurata l'esistenza di un polinomio $\omega(x) \in \Pi_n$ ortogonale a tutti i polinomi $q(x)$ al più di grado $n-1$ rispetto a tale funzione peso e tale che le sue radici siano reali, distinte e tutte interne ad $[a, b]$.

A seconda della funzione peso $\psi(x)$ e dell'intervallo $[a, b]$, si hanno formule di Gauss che prendono il nome dalla famiglia di polinomi ortogonali relativi alla funzione $\psi(x)$.

Poichè i risultati esposti valgono per vari tipi di formule di Gauss, per semplicità in seguito, dove non dichiarato esplicitamente, si farà riferimento sempre alle formule di Gauss-Legendre.

Dalle proprietà degli zeri dei polinomi ortogonali si ricavano alcune caratteristiche delle formule di Gauss:

- le formule di Gauss sono di tipo *aperto*, cioè gli estremi dell'intervallo $[a, b]$ non appartengono all'insieme dei nodi;
- gli insiemi dei nodi di due distinte formule di Gauss sono disgiunti, cioè i nodi di una formula con n_1 punti sono diversi da quelli di una formula con n_2 punti ($n_1 \neq n_2$), ad eccezione del punto medio dell'intervallo $[a, b]$ presente in tutte le formule con un numero di nodi dispari.

Nella Tabella 4.3 sono riportati, ad esempio, i valori dei pesi e dei nodi delle formule di Gauss-Legendre nell'intervallo $[-1, 1]$, con 8 cifre significative (a nodi di modulo uguale e di segno opposto corrispondono pesi uguali).

È da notare, infine, che i polinomi ortogonali sono definiti generalmente nell'intervallo $[-1, 1]$. Per tale motivo i nodi ed i pesi delle formule di Gauss vengono calcolati in $[-1, 1]$ e poi trasformati per il generico intervallo $[a, b]$.

♣ **Esempio 4.15.** Si calcoli l'integrale

$$I[f] = \int_1^2 f(t) dt$$

n	nodì	pesi
1	0.0	2.
2	$\pm .57735027$	1.
3	$\pm .77459667$ 0.0	.55555556 .88888889
4	$\pm .86113631$ $\pm .33998104$.34785485 .65214515
5	$\pm .90617984$ $\pm .53846931$ 0.0	.23692689 .47862867 .56888889
6	$\pm .93246951$ $\pm .66120939$ $\pm .23861919$.17132449 .36076157 .46791393
7	$\pm .96028985$ $\pm .74153118$ $\pm .40584515$ 0.0	.10122853 .27970539 .38183005 .41795918

Tabella 4.3: Nodi e pesi delle formule di Gauss-Legendre

mediante la formula di Gauss-Legendre con tre nodi. In tal caso utilizzando la trasformazione lineare

$$t = (x + 3)/2 \quad x \in [-1, 1]$$

che trasforma l'intervallo $[-1, 1]$ nell'intervallo $[1, 2]$ si ha:

$$I[f] = \frac{1}{2} \int_{-1}^1 f\left(\frac{x+3}{2}\right) dx \simeq \frac{1}{2} \sum_{i=1}^3 A_i f\left(\frac{x_i+3}{2}\right)$$

con i pesi A_i ed i nodi $x_i \in [-1, 1]$ riportati nella Tabella 4.3. ♣

La trasformazione dell'intervallo $[-1, 1]$ nel generico intervallo $[a, b]$, è la seguente:

$$t = \left[\frac{(b-a)x + a + b}{2} \right], \quad x \in [-1, 1] \tag{4.36}$$

Con tale trasformazione si ha:

$$I[f] = \int_a^b f(t) dt = \frac{b-a}{2} \int_{-1}^1 f\left[\frac{(b-a)x + a + b}{2} \right] dx$$

e quindi

$$I[f] \simeq Q[f] = \frac{(b-a)}{2} \sum_{i=1}^n A_i f\left[\frac{(b-a)x_i + a + b}{2} \right]$$

4.4.4 Formule di Gauss con nodi preassegnati

Come mostrato nel paragrafo 4.4.3, le formule di Gauss, per un numero fissato di nodi, hanno il grado di precisione algebrico più alto. Pertanto tali formule sono preferite nella costruzione di software matematico. In questo paragrafo si vuole descrivere un modo per stimare l'errore delle formule di Gauss con quantità calcolabili. Nel paragrafo 4.2 si è osservato che una strategia per stimare l'errore di una formula composta è basata sulla (4.15). Tale tipo di stima ha carattere generale e può essere applicata a qualunque coppia di formule di quadratura¹⁸:

$$(Q^{(n_1)}[f]; Q^{(n_2)}[f])$$

rispettivamente con n_1 e n_2 nodi, in cui $n_1 > n_2$. In generale, il costo computazionale per valutare $E[f]$ mediante la relazione

$$|E[f]| \simeq \alpha |Q^{(n_1)}[f] - Q^{(n_2)}[f]|$$

è $n_1 + n_2$ valutazioni di funzione.

Come è noto però, le coppie di formule più adatte per calcolare una stima dell'errore sono le formule innestate (vedi Definizione 4.6), in cui, cioè, l'insieme dei nodi di una formula contiene l'insieme dei nodi dell'altra. In tal caso infatti la complessità computazionale per la stima di $E[f]$ si riduce a n_1 valutazioni di funzione.

Purtroppo però le formule di Gauss, benché convenienti in quanto hanno grado di precisione algebrico massimo, non godono della proprietà di essere innestate e la complessità computazionale per valutare l'errore $E[f]$ è di $n_1 + n_2$ valutazioni della funzione integranda.

Tale inconveniente viene superato da formule di Gauss particolari, in cui parte dei nodi sono preassegnati.

♣ **Esempio 4.16.** Sia :

$$G[f] = f\left(\frac{-1}{\sqrt{3}}\right) + f\left(\frac{1}{\sqrt{3}}\right)$$

la formula di Gauss-Legendre con due nodi. Si vuole costruire una formula con cinque nodi del tipo:

$$K[f] = B_1 f\left(\frac{-1}{\sqrt{3}}\right) + B_2 f\left(\frac{1}{\sqrt{3}}\right) + \sum_{j=1}^3 C_j f(\xi_j)$$

determinando i nodi ξ_j , $j = 1, \dots, 3$ ed i pesi B_i , $i = 1, 2$ e C_j , $j = 1, \dots, 3$ in maniera che $K[f]$ abbia grado di precisione algebrico massimo. Essendo disponibili 8 parametri si può ragionevolmente pensare di poter integrare esattamente un qualunque polinomio di grado $d = 7$. Analogamente all'esempio 4.14,

¹⁸In questo paragrafo indicheremo esplicitamente il numero dei nodi n nell'espressione di $Q^{(n)}[f]$ per evidenziare la dipendenza della formula di quadratura da n .

è necessario quindi risolvere il sistema non lineare:

$$\begin{cases} B_1 + B_2 + C_1 + C_2 + C_3 & = 2 \\ \frac{1}{\sqrt{3}}(-B_1 + B_2) + C_1\xi_1 + C_2\xi_2 + C_3\xi_3 & = 0 \\ \frac{1}{3}(B_1 + B_2) + C_1\xi_1^2 + C_2\xi_2^2 + C_3\xi_3^2 & = \frac{2}{3} \\ \frac{1}{3\sqrt{3}}(-B_1 + B_2) + C_1\xi_1^3 + C_2\xi_2^3 + C_3\xi_3^3 & = 0 \\ \frac{1}{9}(B_1 + B_2) + C_1\xi_1^4 + C_2\xi_2^4 + C_3\xi_3^4 & = \frac{2}{9} \\ \frac{1}{9\sqrt{3}}(-B_1 + B_2) + C_1\xi_1^5 + C_2\xi_2^5 + C_3\xi_3^5 & = 0 \\ \frac{1}{27}(B_1 + B_2) + C_1\xi_1^6 + C_2\xi_2^6 + C_3\xi_3^6 & = \frac{2}{27} \\ \frac{1}{27\sqrt{3}}(-B_1 + B_2) + C_1\xi_1^7 + C_2\xi_2^7 + C_3\xi_3^7 & = 0 \end{cases}$$

ottenuto imponendo che $K[f]$ sia esatta per i monomi x^r per $r = 0, 1, \dots, 7$.

Per risolvere tale sistema, si osservi che, poiché l'intervallo $[-1, 1]$ è simmetrico rispetto allo 0, è ragionevole porre $\xi_1 = -\xi_3$ e $\xi_2 = 0$. Pertanto dalla seconda e quarta equazione si ha $C_1 = C_3$ e quindi anche $B_1 = B_2$. In seguito, dalla terza, quinta e settima equazione si ricava prima $\xi_1 = \xi_3 = 0.92582009977255\dots$ e quindi $B_1 = B_2 = 0.490$ e $C_1 = C_3 = 0.1979$. Infine, dalla prima equazione si ricava $C_2 = 0.62$. ♣

L'idea è quella di partire da una formula di Gauss-Legendre con n nodi e di costruire una formula a $2n + 1$ nodi che utilizzi gli n nodi della formula di Gauss-Legendre. In generale, assegnata una formula di Gauss-Legendre con n nodi:

$$G^{(n)}[f] = \sum_{i=1}^n A_i f(x_i),$$

si vuole costruire una formula di quadratura con $2n + 1$ nodi del tipo

$$K^{(2n+1)}[f] = \sum_{i=1}^n B_i f(x_i) + \sum_{j=1}^{n+1} C_j f(\xi_j). \tag{4.37}$$

Tale formula viene detta *formula di Gauss con nodi preassegnati*¹⁹. Nella (4.37) i nodi x_i sono assegnati e bisogna calcolare i pesi B_i e C_j ed i nodi ξ_j in modo da ottenere una formula con il grado di precisione algebrico massimo possibile. La (4.37) ha $n + 2(n + 1) = 3n + 2$ parametri, ed è quindi possibile aspettarsi una formula di quadratura con grado di precisione algebrico $3n + 1$. La determinazione di tali parametri avviene in base al teorema seguente:

¹⁹Nonostante in letteratura il caso più trattato sia quello esposto, una forma più generale della precedente formula è:

$$K^{(n_1+n_2)}[f] = \sum_{i=1}^{n_1} B_i f(x_i) + \sum_{j=1}^{n_2} C_j f(\xi_j)$$

Teorema 4.6. Sia $K^{(2n+1)}[f]$ del tipo (4.37). Condizione necessaria e sufficiente affinché $K^{(2n+1)}[f]$ abbia grado di precisione algebrico $3n+1$ è che sia esatta per polinomi di grado al più $2n$, cioè tale che:

$$K^{(2n+1)}[v] = \int_a^b v(x) dx \quad \forall v \in \Pi_{2n} \quad (4.38)$$

e che abbia come $n+1$ nodi gli zeri ξ_j , $j = 1, \dots, n+1$ di un polinomio $\omega(x) \in \Pi_{n+1}$ ortogonale a tutti i polinomi $q(x)$ di grado al più n rispetto alla funzione peso $r(x) = (x-x_1) \cdots (x-x_n)$, cioè tale che:

$$\int_a^b r(x)q(x)\omega(x) dx = 0 \quad \forall q \in \Pi_n \quad (4.39)$$

Dimostrazione Se la formula $K^{(2n+1)}[f]$ ha grado di precisione $3n+1$ la condizione (4.38) è banalmente soddisfatta. Siano quindi x_1, \dots, x_n e ξ_1, \dots, ξ_{n+1} i nodi di $K^{(2n+1)}[f]$ e sia $q(x)$ un polinomio di grado al più n . Posto allora $r(x) = (x-x_1) \cdots (x-x_n)$, $\omega(x) = (x-\xi_1) \cdots (x-\xi_{n+1})$ e $p(x) = r(x)q(x)\omega(x)$ si ha che $p \in \Pi_{3n+1}$ e per tale polinomio la formula $K^{(2n+1)}[f]$ è esatta, cioè:

$$\begin{aligned} 0 = E[p] &= \int_a^b p(x) dx - \sum_{i=1}^n B_i p(x_i) - \sum_{j=1}^{n+1} C_j p(\xi_j) = \\ &= \int_a^b r(x)q(x)\omega(x) dx - \sum_{i=1}^n B_i r(x_i)q(x_i)\omega(x_i) - \sum_{j=1}^{n+1} C_j r(\xi_j)q(\xi_j)\omega(\xi_j) \end{aligned}$$

Poiché $r(x_i) = 0$ e $\omega(\xi_j) = 0 \forall i$ e j la (4.39) è provata.

Viceversa si supponga che le (4.38) e (4.39) siano verificate e sia $p(x)$ un polinomio di grado al più $3n+1$. Dividendo $p(x)$ per il polinomio $r(x)\omega(x)$ di grado $2n+1$ si ha che $p(x) = r(x)\omega(x)q(x) + v(x)$ con $v(x)$ polinomio di grado al più $2n$. Per la (4.39) risulta allora:

$$\int_a^b p(x) dx = \int_a^b r(x)\omega(x)q(x) dx + \int_a^b v(x) dx = \int_a^b v(x) dx$$

Si noti inoltre che $p(x_i) = v(x_i)$ e $p(\xi_j) = v(\xi_j)$, poiché in tali punti il polinomio $r(x)\omega(x)$ si annulla. Per la (4.38) allora si ha:

$$\int_a^b v(x) dx = \sum_{i=1}^n B_i v(x_i) + \sum_{j=1}^{n+1} C_j v(\xi_j) = \sum_{i=1}^n B_i p(x_i) + \sum_{j=1}^{n+1} C_j p(\xi_j)$$

e per l'arbitrarietà di $p(x)$ il teorema è dimostrato. ■

Definizione 4.12. (Formula di Gauss-Kronrod)

Una formula del tipo

$$K^{(2n+1)}[f] = \sum_{i=1}^n B_i f(x_i) + \sum_{j=1}^{n+1} C_j f(\xi_j)$$

con grado di precisione algebrico $3n+1$, costruita in base al teorema precedente è detta **formula di Gauss-Kronrod**.

Il teorema precedente determina le condizioni per cui una formula del tipo (4.37) abbia grado di precisione algebrico $3n + 1$, ma nulla dice circa l'esistenza del polinomio $\omega(x)$ e quindi dei nodi ξ_j . A tal proposito si osservi che i polinomi di Stieltjes $s_n(x)$ godono della proprietà (4.39) del Teorema 4.6. Infatti si dimostra che

$$\int_{-1}^1 l_n(x) s_{n+1}(x) x^k dx = 0 \quad k = 0, \dots, n.$$

con $l_n(x)$ polinomio di Legendre di grado n e $s_{n+1}(x)$ polinomio di Stieljes di grado $n + 1$. Gli zeri del polinomio di Stieltjes $s_{n+1}(x)$ di grado $n + 1$ sono quindi i nodi ξ_j cercati e tali zeri separano gli zeri del polinomio di Legendre $l_n(x)$ di grado n che rappresentano invece i nodi della formula di Gauss $G^{(n)}[f]$. Infine è possibile dimostrare che i pesi delle formule di Gauss-Kronrod sono tutti positivi.

Il costo computazionale della coppia di formule:

$$(G^{(n)}[f]; K^{(2n+1)}[f]) \tag{4.40}$$

è di $2n + 1$ valutazioni di funzione. Tale numero di valutazioni è lo stesso della formula di Gauss con $2n + 1$ nodi che non permette di ottenere una stima calcolabile dell'errore. Le coppie del tipo (4.40) hanno anche lo stesso costo computazionale delle coppie di formule di Gauss $G^{(n)}[f]$ e $G^{(n+1)}[f]$ ma la formula $K^{(2n+1)}[f]$ ha precisione molto maggiore di $G^{(n+1)}[f]$. Per tale motivo le coppie (4.40) sono attualmente le più utilizzate nello sviluppo di routine per il calcolo degli integrali. Numerose routine della libreria NAG e del package QUADPACK sono infatti basate su tali formule.

I risultati fin qui riportati valgono quando la formula $G[f]$ è una formula di Gauss-Legendre. Essi possono comunque essere estesi anche al caso in cui la $G[f]$ sia una formula di Gauss-Jacobi. In generale comunque l'esistenza dei nodi ξ_j non è assicurata per le altre formule di Gauss riportate nella Tabella 4.2.

Il metodo descritto, sviluppato inizialmente da Kronrod, è stato successivamente generalizzato da Patterson, che, a partire da una formula di Gauss-Legendre con n nodi, ha sviluppato una famiglia di formule innestate

$$\mathcal{K} = \{P^{(i)}[f]\}_{i=2^k(n+1)-1} \quad k = 1, 2, \dots$$

con grado di precisione algebrico $3 \cdot 2^{k-1}(n + 1) - 2$, ciascuna delle quali costruita in base al Teorema 4.6. Tale famiglia di formule è alla base della routine QNG del package QUADPACK e della routine D01AHF della libreria NAG.

4.5 Errori e criteri di convergenza per le formule di quadratura

4.5.1 Errore di discretizzazione

Ad ogni formula di quadratura $Q[f]$ è associato un errore $E[f]$, chiamato errore di discretizzazione, tale che

$$E[f] = I[f] - Q[f].$$

In questo paragrafo si vuole determinare un'espressione di $E[f]$. Uno strumento fondamentale in tale senso è il seguente:

Teorema 4.7. [Teorema di Peano]

Sia $E[f]$ il funzionale lineare del tipo ²⁰

$$E[f] = \int_a^b f(x) dx - \sum_{i=1}^n A_i f(x_i)$$

Se risulta $E[p] = 0 \quad \forall p(x) \in \Pi_q$ allora si ha che

$$E[f] = \int_a^b f^{(q+1)}(t)K(t) dt \quad \forall f \in C^{q+1}[a, b] \quad (4.41)$$

dove la funzione

$$K(t) = \frac{1}{q!} E_x[(x-t)_+^q] \quad \text{con} \quad (x-t)_+^q = \begin{cases} (x-t)^q & x \geq t \\ 0 & x < t \end{cases},$$

è detta kernel di Peano e con il simbolo E_x si indica il funzionale E applicato alla funzione $(x-t)_+^q$ considerata come funzione della sola variabile x .

Dimostrazione Sia $f \in C^{q+1}[a, b]$. Si consideri la formula di Taylor della funzione $f(x)$ di punto iniziale a :

$$f(x) = f(a) + f'(a)(x-a) + \dots + \frac{f^{(q)}(a)(x-a)^q}{q!} + \frac{1}{q!} \int_a^x f^{(q+1)}(t)(x-t)^q dt. \quad (4.42)$$

L'ultimo addendo di tale formula è uguale a:

$$\frac{1}{q!} \int_a^b f^{(q+1)}(t)(x-t)_+^q dt.$$

Applicando il funzionale $E[f]$ ad entrambi i membri della (4.42) e ricordando che esso è nullo per tutti i polinomi di grado al più q si ha:

$$E[f] = \frac{1}{q!} E_x \left[\int_a^b f^{(q+1)}(t)(x-t)_+^q dt \right]$$

Per la continuità della $f^{(q+1)}(t)$ e per la linearità del funzionale $E[f]$ è possibile invertire l'ordine dell'integrale e del funzionale, cioè:

$$E[f] = \frac{1}{q!} \int_a^b f^{(q+1)}(t) E_x[(x-t)_+^q] dt,$$

²⁰Si osservi che tale teorema è valido in generale per funzionali lineari del tipo

$$\Phi[f] = \int_a^b \left(\sum_{j=0}^m \alpha_j(x) f^{(j)}(x) \right) dx + \sum_{j_0=1}^{n_0} A_{j_0} f(a_{j_0}) + \dots + \sum_{j_m=1}^{n_m} A_{j_m} f^{(m)}(a_{j_m})$$

con $\alpha_j(x)$ funzioni continue a tratti su $[a, b]$, $f \in C^{m+1}[a, b]$, $a_{j_i} \in [a, b]$, $A_{j_k} \in \mathfrak{R}$ di cui il funzionale $E[f]$ è un caso particolare ottenuto ponendo $m = 0$ e $\alpha_0(x) \equiv 1$.

da cui la tesi. ■

Conseguenze del Teorema di Peano sono:

Corollario 4.2. *Nelle ipotesi del teorema di Peano risulta:*

$$|E[f]| \leq \max_{[a,b]} |f^{(q+1)}(t)| \int_a^b |K(t)| dt.$$

Dimostrazione Dalla (4.41) si ha:

$$|E[f]| = \left| \int_a^b f^{(q+1)}(t)K(t) dt \right| \leq \int_a^b |f^{(q+1)}(t)| |K(t)| dt$$

da cui segue la tesi. ■

Corollario 4.3. *Nelle ipotesi del Teorema di Peano, se $K(t)$ non cambia di segno in $[a, b]$ risulta:*

$$E[f] = \frac{f^{(q+1)}(\xi)}{(q+1)!} E[t^{q+1}] \quad \xi \in [a, b].$$

Dimostrazione Per il teorema del valor medio degli integrali deve esistere un punto $\xi \in [a, b]$ tale che

$$E[f] = \int_a^b f^{(q+1)}(t)K(t) dt = f^{(q+1)}(\xi) \int_a^b K(t) dt \tag{4.43}$$

Ponendo $f(t) = t^{q+1}$ nella (4.41) si ha che

$$E[t^{q+1}] = (q+1)! \int_a^b K(t) dt \quad \text{cioè} \quad \int_a^b K(t) dt = \frac{E[t^{q+1}]}{(q+1)!}$$

da cui, sostituendo nella (4.43) segue la tesi. ■

♣ **Esempio 4.17.** Si calcoli l'errore di discretizzazione della formula trapezoidale

$$T[f] = \frac{b-a}{2} (f(a) + f(b)).$$

con $f \in C^2[a, b]$.

Cominciamo col verificare che il kernel $K(t)$ non cambia segno nell'intervallo $[a, b]$. Si ha quindi:

$$\begin{aligned} K(t) &= \int_a^b (x-t)_+ dx - \frac{(b-a)}{2} [(a-t)_+ + (b-t)_+] = \\ &= \int_t^b (x-t) dx - \frac{(b-a)}{2} [(b-t)] = \frac{1}{2}(a-t)(b-t) \leq 0 \quad t \in [a, b]. \end{aligned}$$

È allora possibile applicare il Corollario 4.3 del Teorema di Peano:

$$E[f] = \frac{f''(\xi)}{2} E[x^2] \quad \xi \in [a, b]. \quad (4.44)$$

Ma

$$E[x^2] = \int_a^b x^2 dx - \frac{b-a}{2}(a^2 + b^2) = -\frac{1}{6}(b-a)^3,$$

da cui, sostituendo nella (4.44), si ricava

$$E[f] = -\frac{(b-a)^3}{12} f''(\xi) \quad \xi \in [a, b].$$

♣

Si osservi che, per una formula di quadratura generica, non c'è garanzia che il kernel $K(t)$ sia di segno costante in $[a, b]$. Per le formule di Newton-Cotes si può dimostrare che risulta $K(t) \leq 0 \quad \forall t \in [a, b]$. Per tali formule è quindi possibile applicare il Corollario 4.3 del teorema di Peano.

♣ **Esempio 4.18.** Si calcoli l'errore di discretizzazione della formula di Simpson

$$S[f] = \frac{(b-a)}{2} \left[\frac{f(a)}{3} + \frac{4}{3} f\left(\frac{a+b}{2}\right) + \frac{f(b)}{3} \right]$$

con $f \in C^4[a, b]$.

Tale formula è esatta per tutti i polinomi $p(x) \in \Pi_2$, per cui dal Corollario 4.3 del Teorema di Peano (con $q = 2$) si ha:

$$E[f] = \frac{f'''(\xi)}{6} E[x^3] \quad \xi \in [a, b].$$

Si osservi che

$$E[x^3] = \int_a^b x^3 dx - \frac{(b-a)}{2} \left[\frac{a^3}{3} + \frac{4}{3} \left(\frac{a+b}{2}\right)^3 + \frac{b^3}{3} \right]$$

e con facili calcoli risulta:

$$E[x^3] = \frac{b-a}{4} [(b^2 + a^2)(a+b) - (a^3 + b^3 + a^2b + ab^2)] = 0$$

Da ciò si ricava che la formula di Simpson è esatta anche per tutti i polinomi $p(x) \in \Pi_3$. Riapplicando quindi il Corollario 4.3 del Teorema di Peano (questa volta con $q = 3$), se la funzione è dotata anche di derivata quarta, si ha:

$$E[f] = \frac{f^{(4)}(\xi)}{24} E[x^4] \quad \xi \in [a, b] \tag{4.45}$$

e poiché risulta:

$$E[x^4] = -\frac{4}{15} \left(\frac{b-a}{2}\right)^5$$

sostituendo nella (4.45) si ha che l'errore di discretizzazione associato alla formula di Simpson è:

$$E[f] = -\frac{f^{(4)}(\xi)}{90} \left(\frac{b-a}{2}\right)^5 \quad \xi \in [a, b].$$



Risultati analoghi a quelli dell'esempio precedente si hanno per altre formule di Newton-Cotes con un numero di nodi n dispari e con $f \in C^{n+1}[a, b]$, cioè vale il:

Teorema 4.8. *Se $f(x) \in C^{(n+1)}[a, b]$ e se il numero dei nodi $n = 2k + 1$ ($k \geq 1$) è dispari le formule di Newton-Cotes hanno grado di precisione algebrico n e l'espressione dell'errore associato risulta essere:*

$$E[f] = \frac{f^{(n+1)}(\xi)}{(n+1)!} E[x^{n+1}] \quad \xi \in [a, b].$$

Dimostrazione Dal teorema di Peano e per la linearità del funzionale $E[f]$, integrando per parti si ha:

$$\begin{aligned} E[f] &= \frac{1}{(n-1)!} E_x \left[\int_a^b f^{(n)}(t) (x-t)_+^{n-1} dt \right] \\ &= \frac{1}{(n-1)!} E_x \left[-f^{(n)}(a) \frac{(x-a)^n}{n} + \int_a^b f^{(n+1)}(t) \frac{(x-t)_+^n}{n} dt \right] \\ &= -\frac{f^{(n)}(a)}{n!} E_x[(x-a)^n] + \left[\int_a^b f^{(n+1)}(t) K(t) dt \right]. \end{aligned}$$

Ma per la (4.24) si ha:

$$\begin{aligned} E_x[(x-a)^n] &= \int_a^b (x-a)^n dx - \sum_{i=1}^n (x_i - a)^n \int_a^b \frac{\omega(x)}{(x-x_i)\omega'(x_i)} dx \\ &= \int_a^b (x-a)^n dx - \int_a^b L_{n-1}(x) dx \\ &= \int_a^b [(x-a)^n - L_{n-1}(x)] dx \end{aligned}$$

dove, in questo caso, $L_{n-1}(x)$ è il polinomio di grado $n-1$ interpolante nel senso di Lagrange la funzione $(x-a)^n$ nei nodi x_i con $i = 1, \dots, n$. Poiché il resto di ordine n del polinomio interpolante di Lagrange, della funzione $(x-a)^n$ è²¹:

$$(x-a)^n - L_{n-1}(x) = \frac{n!}{n!} \omega(x)$$

risulta:

$$E_x[(x-a)^n] = \int_a^b \omega(x) dx$$

Se il numero di nodi $n = 2k + 1$ è dispari si può facilmente verificare che $\omega(x_k + t) = -\omega(x_k - t)$ e quindi la funzione $\omega(x)$ è una funzione dispari rispetto al nodo x_k che coincide con il punto medio dell'intervallo $[a, b]$. Si ha quindi che

$$\int_a^b \omega(x) dx = 0,$$

da cui la tesi. ■

Si può, quindi, affermare che una formula di quadratura di Newton-Cotes con un numero di nodi $n = 2k + 1$ dispari ha lo stesso grado di precisione algebrico della successiva formula con un numero di nodi $n = 2k + 2$ pari. Nella Tabella 4.4 sono riportate le espressioni degli errori di alcune formule di Newton-Cotes. Si noti, ad esempio, come l'errore della formula di Simpson abbia le stesse caratteristiche dell'errore della formula dei '3/8'.

Nella stessa Tabella, dove nell'espressione dell'errore compare il termine $f^{(k)}(\xi)$, è tacitamente assunto che $f(x) \in C^k[a, b]$. Ovviamente è possibile applicare il Teorema di Peano ed i suoi Corollari anche alle formule di Gauss.

♣ **Esempio 4.19.** Si calcoli l'errore della formula di Gauss-Legendre con $n = 2$ nodi. In tal caso la formula ha grado di precisione algebrico $q = 3$ ed il kernel di Peano è (vedi esempio 4.14):

$$K(t) = \frac{1}{3!} E_x [(x-t)_+^3] = \frac{1}{6} \left(\int_{-1}^1 (x-t)_+^3 dx - \left[\left(-\frac{1}{\sqrt{3}} - t \right)^3 + \left(\frac{1}{\sqrt{3}} - t \right)^3 \right] \right)$$

da cui

$$K(t) = \frac{(1+t)^4}{24} \geq 0 \quad \forall t \in [-1, 1].$$

²¹Sussiste infatti il:

Teorema: Sia $f \in C^n[a, b]$ e $p \in \Pi_{n-1}$ il polinomio di Lagrange interpolante la funzione $f(x)$ in n nodi distinti x_i per $i = 1, \dots, n$. Per ogni $x \in [a, b]$ esiste un punto $\xi \in [a, b]$ tale che

$$f(x) - p(x) = \frac{1}{n!} f^{(n)}(\xi) \omega(x)$$

Formula	Errore
Trapezoidale	$-\frac{f''(\xi)}{12}(b-a)^3$
Simpson	$-\frac{f^{(4)}(\xi)}{90}\left(\frac{b-a}{2}\right)^5$
dei '3/8'	$-\frac{3f^{(4)}(\xi)}{80}\left(\frac{b-a}{3}\right)^5$
Boole	$-\frac{8f^{(6)}(\xi)}{945}\left(\frac{b-a}{4}\right)^7$
N-C 6 nodi	$-\frac{275f^{(6)}(\xi)}{12096}\left(\frac{b-a}{5}\right)^7$
N-C 7 nodi	$-\frac{9f^{(8)}(\xi)}{1400}\left(\frac{b-a}{6}\right)^9$
N-C 8 nodi	$-\frac{8183f^{(8)}(\xi)}{518400}\left(\frac{b-a}{7}\right)^9$
N-C 9 nodi	$-\frac{2368f^{(10)}(\xi)}{467775}\left(\frac{b-a}{8}\right)^{11}$
N-C 10 nodi	$-\frac{673175f^{(10)}(\xi)}{163459296}\left(\frac{b-a}{9}\right)^{11}$

Tabella 4.4: Errori delle formule di Newton-Cotes con $n \leq 10$

Formula	Errore
$n = 2$	$f^{(4)}(\xi)/135$
$n = 3$	$f^{(6)}(\xi)/91854$
$n = 4$	$f^{(8)}(\xi)/3472875$

Tabella 4.5: Errori delle formule di Gauss-Legendre in $[-1, 1]$

Formula	Errore
$n = 2$	$(b-a)^5 f^{(4)}(\xi)/4320$
$n = 3$	$(b-a)^7 f^{(6)}(\xi)/11757312$
$n = 4$	$(b-a)^9 f^{(8)}(\xi)/1778112000$

Tabella 4.6: Errori delle formule di Gauss-Legendre in $[a, b]$

Applicando il Corollario 4.3 del Teorema di Peano si ha allora

$$E[f] = \frac{f^{(4)}(\xi)}{24} E[x^4] \quad (4.46)$$

La quantità $E[x^4]$ è data da:

$$E[x^4] = \int_{-1}^1 x^4 dx - \left[\left(-\frac{1}{\sqrt{3}}\right)^4 + \left(\frac{1}{\sqrt{3}}\right)^4 \right] = \frac{2}{5} - \frac{2}{9} = \frac{8}{45}$$

da cui, sostituendo nella (4.46), si ricava

$$E[f] = \frac{8}{45} \frac{f^{(4)}(\xi)}{24} = \frac{f^{(4)}(\xi)}{135} \quad \forall f \in C^4[-1, 1]$$

♣

Nella Tabella 4.5 sono riportati gli errori di discretizzazione di alcune formule di Gauss-Legendre nell'intervallo $[-1, 1]$.

Più in generale, utilizzando la trasformazione (4.36), è possibile determinare l'errore delle formule di Gauss in un intervallo $[a, b]$ (Tabella 4.6).

4.5.2 Errore delle formule composite e stima calcolabile dell'errore

Dalla Tabella 4.6 si può notare che l'errore dipende dall'ampiezza dell'intervallo di integrazione $[a, b]$; un modo per ridurre l'errore consiste quindi, come mostrato nel paragrafo 4.2.1, nel dividere l'intervallo di integrazione $[a, b]$ in m sottointervalli di uguale ampiezza $\tau = (b-a)/m$ in ognuno dei quali si applica una fissata formula di quadratura (formule composite). Per valutare l'errore di una formula composta è sufficiente applicare il teorema di Peano in ogni sottointervallo $[t_{j-1}, t_j]$ dell'intervallo $[a, b]$.

♣ **Esempio 4.20.** Si calcoli l'errore della formula trapezoidale composta $T_m[f]$ per $f \in C^2[a, b]$. In tal caso applicando il Corollario 4.3 del Teorema di Peano al j -mo sottointervallo $[t_{j-1}, t_j]$ di ampiezza $\tau = t_j - t_{j-1} = (b - a)/m$ si ha che l'errore in tale sottointervallo è:

$$E^{(j)}[f] = -\frac{(b-a)^3}{12m^3} f''(\xi_j) \quad \xi_j \in [t_{j-1}, t_j],$$

da cui, sommando sull'indice j dei sottointervalli, l'errore su tutto l'intervallo $[a, b]$ è:

$$E[f] = \sum_{j=1}^m E^{(j)}[f] = -\frac{(b-a)^3}{12m^3} \sum_{j=1}^m f''(\xi_j)$$

Poiché si ha

$$\min_{\xi \in [a, b]} f''(\xi) \leq \min_j f''(\xi_j) \leq \frac{1}{m} \sum_{j=1}^m f''(\xi_j) \leq \max_j f''(\xi_j) \leq \max_{\xi \in [a, b]} f''(\xi)$$

per la continuità di $f''(x)$ nell'intervallo $[a, b]$ esiste un punto $\xi \in [a, b]$ tale che

$$f''(\xi) = \frac{1}{m} \sum_{j=1}^m f''(\xi_j),$$

da cui si ha:

$$E_m[f] = -\frac{(b-a)^3}{12m^2} f''(\xi).$$

♣

Per ricavare gli errori delle formule di Newton-Cotes composite $Q_m[f]$, si noti che $Q[f]$ nell'intervallo $[t_{j-1}, t_j]$ di ampiezza $\tau = (b - a)/m$ ha errore

$$E^{(j)}[f] = \int_{t_{j-1}}^{t_j} f(x) dx - h \sum_{i=1}^n B_i f(x_i) \quad h = \frac{(b-a)}{m(n-1)}$$

con $E^{(j)}[f]$ opportunamente scelto nella Tabella 4.4. Per ottenere l'errore di $E_m[f]$ su tutto $[a, b]$ è sufficiente sommare il contributo degli errori $E^{(j)}[f]$, cioè:

$$E_m[f] = \sum_{j=1}^m E^{(j)}[f]$$

Procedendo in maniera analoga al precedente esempio 4.20 si hanno le espressioni degli errori delle formule di Newton-Cotes composite riportate nella Tabella 4.7 che segue.

Nello sviluppo di software matematico per la quadratura è rilevante la determinazione di una stima calcolabile dell'errore di discretizzazione, basata cioè solo su quantità calcolabili. Il Corollario 4.3 del Teorema di Peano fornisce uno strumento generale in tal senso.

Formula	Errore
trapezoidale	$-\frac{f''(\xi)(b-a)^3}{12m^2}$
Simpson	$-\frac{f^{(4)}(\xi)(b-a)^5}{180m^4}$
dei '3/8'	$-\frac{f^{(4)}(\xi)(b-a)^5}{80m^4}$
Boole	$-\frac{2f^{(6)}(\xi)(b-a)^7}{945m^6}$
N-C 6 nodi	$-\frac{55f^{(6)}(\xi)(b-a)^7}{12096m^6}$
N-C 7 nodi	$-\frac{3f^{(8)}(\xi)(b-a)^9}{2800m^8}$
N-C 8 nodi	$-\frac{1169f^{(8)}(\xi)(b-a)^9}{518400m^8}$
N-C 9 nodi	$-\frac{296f^{(10)}(\xi)(b-a)^{11}}{46775m^{10}}$
N-C 10 nodi	$-\frac{673175f^{(10)}(\xi)(b-a)^{11}}{1471133664m^{10}}$

Tabella 4.7: Errori delle formule di Newton-Cotes composite per $n \leq 10$

♣ **Esempio 4.21.** Si determini una stima calcolabile dell'errore di discretizzazione per la formula di Simpson composta per una funzione $f \in C^4[a, b]$. A tale scopo, posto $h = (b - a)/2$, siano:

$$Q[f] = \frac{(b-a)}{6} [f(a) + 4f(c) + f(b)]$$

$$Q_2[f] = \frac{(b-a)}{12} \left[f(a) + 4f\left(a + \frac{h}{2}\right) + 2f(c) + 4f\left(a + 3\frac{h}{2}\right) + f(b) \right]$$

rispettivamente la formula di Simpson sull'intervallo $[a, b]$ e la sua composta negli intervalli $[a, c]$ e $[c, b]$ con c punto medio di $[a, b]$. Al fine di determinare una stima per

$$|E_2[f]| = |I[f] - Q_2[f]|$$

si noti che, dal Corollario 4.3 del Teorema di Peano si ricava:

$$E[f] = -\frac{f^{(4)}(\xi)}{90} h^5, \quad E_2[f] = -\frac{f^{(4)}(\xi')}{90} \left(\frac{h}{2}\right)^5 - \frac{f^{(4)}(\xi'')}{90} \left(\frac{h}{2}\right)^5 \quad (4.47)$$

e dalle formule di Taylor di punto iniziale ξ della funzione $f^{(4)}(x)$:

$$f^{(4)}(\xi') = f^{(4)}(\xi) + \mathcal{O}(h), \quad f^{(4)}(\xi'') = f^{(4)}(\xi) + \mathcal{O}(h); \quad (4.48)$$

sostituendo, quindi, le (4.48) nella (4.47), si ha:

$$\begin{aligned} E_2[f] &= -2 \frac{[f^{(4)}(\xi) + \mathcal{O}(h)]}{90} \left(\frac{h}{2}\right)^5 = -2 \frac{f^{(4)}(\xi)}{90 \cdot 32} h^5 + \mathcal{O}(h^6) = \\ &= \frac{E[f]}{16} + \mathcal{O}(h^6). \end{aligned}$$

Procedendo poi come mostrato nell'esempio 4.6, si perviene alla stima calcolabile dell'errore $E_2[f]$:

$$|E_2[f]| \simeq \frac{|Q[f] - Q_2[f]|}{15} \quad (4.49)$$

♣

Per una coppia di formule di quadratura ($Q'[f], Q''[f]$) dove $Q''[f]$ è "in qualche senso" più accurata di $Q'[f]$, è possibile utilizzare come stima dell'errore un'opportuna funzione Ψ della differenza tra $Q'[f]$ e $Q''[f]$, cioè:

$$E[f] \simeq \Psi [|Q'[f] - Q''[f]|]$$

con Ψ funzione crescente che dipende dalle particolari formule di quadratura scelte e da considerazioni di carattere sperimentale²².

²²Ad esempio alcune routine di QUADPACK, relativamente alla coppia di formule di Gauss-Kronrod ($G^{(7)}[f], K^{(15)}[f]$) rispettivamente con 7 e 15 nodi, utilizzano come stima calcolabile dell'errore la quantità:

$$E[f] = \left(200|G^{(7)}[f] - K^{(15)}[f]\right)^{1.5}.$$

4.5.3 Criteri di convergenza delle formule di quadratura

Abbiamo già dimostrato, mediante il Teorema 4.1, la convergenza della formula trapezoidale composta costruita su m sottointervalli di uguale ampiezza $\tau = (b - a)/m$. Più in generale, sussiste il seguente:

Teorema 4.9. *Sia $f(x)$ una funzione integrabile nell'intervallo $[a, b]$ e sia*

$$Q[f] = \sum_{i=1}^n A_i f(x_i) \quad x_i \in [-1, 1] \quad (4.50)$$

una formula di quadratura tale che:

$$Q[1] = \sum_{i=1}^n A_i = \int_{-1}^1 dx = 2 \quad (4.51)$$

Partizionato l'intervallo $[a, b]$ in m sottointervalli $[t_{j-1}, t_j]$ $j = 1, \dots, m$, sia $Q_m[f]$ la formula composta ottenuta applicando la (4.50) negli m sottointervalli. Per ogni fissato m , si indichi con τ_m la massima ampiezza dei sottointervalli, cioè

$$\tau_m = \max_{j=1, \dots, m} (t_j - t_{j-1}).$$

Se

$$\lim_{m \rightarrow \infty} \tau_m = 0 \quad (4.52)$$

allora si ha che²³

$$\lim_{m \rightarrow \infty} Q_m[f] = I[f] \iff \lim_{m \rightarrow \infty} E_m[f] = 0$$

Dimostrazione Data $Q[f]$ del tipo (4.50), dividendo $[a, b]$ in m sottointervalli $[t_{j-1}, t_j]$ per la proprietà additiva degli integrali, si ha:

$$I[f] = \int_a^b f(x) dx = \sum_{j=1}^m \int_{t_{j-1}}^{t_j} f(x) dx \quad (4.53)$$

Mediante la trasformazione

$$x = \left[\frac{(t_j - t_{j-1})s + t_j + t_{j-1}}{2} \right]$$

dell'intervallo $[-1, 1]$ nell'intervallo $[t_{j-1}, t_j]$ si ha

$$\int_{t_{j-1}}^{t_j} f(x) dx = \frac{(t_j - t_{j-1})}{2} \int_{-1}^1 f \left[\frac{(t_j - t_{j-1})s + t_j + t_{j-1}}{2} \right] ds \quad (4.54)$$

²³In particolare l'ipotesi (4.52) è banalmente soddisfatta nel caso di intervalli di uguale ampiezza $\tau = (b - a)/m$.

Posto allora

$$x_i^{(j)} = \frac{(t_j - t_{j-1})s_i + t_j + t_{j-1}}{2}$$

applicando la (4.50) all'integrale al secondo membro della (4.54) si ha

$$\int_{-1}^1 f \left[\frac{(t_j - t_{j-1})s + t_j + t_{j-1}}{2} \right] ds \simeq \sum_{i=1}^n A_i f(x_i^{(j)}) \quad \text{con } x_i^{(j)} \in [t_{j-1}, t_j]$$

Sostituendo tale espressione nella (4.54), si ha

$$\int_{t_{j-1}}^{t_j} f(x) dx \simeq \frac{(t_j - t_{j-1})}{2} \sum_{i=1}^n A_i f(x_i^{(j)})$$

e quindi la formula di quadratura composita $Q_m[f]$ sull'intero intervallo $[a, b]$ è

$$I[f] \simeq Q_m[f] = \sum_{j=1}^m \frac{(t_j - t_{j-1})}{2} \sum_{i=1}^n A_i f(x_i^{(j)}) = \frac{1}{2} \sum_{i=1}^n A_i \sum_{j=1}^m (t_j - t_{j-1}) f(x_i^{(j)})$$

Nell'ultima espressione, per ogni fissato indice i , la somma più interna è una somma di Riemann per cui, per l'ipotesi (4.52), si ha

$$\lim_{m \rightarrow \infty} E_m[f] = \lim_{m \rightarrow \infty} [I[f] - Q_m[f]] = I[f] - \frac{1}{2} I[f] \sum_{i=1}^n A_i$$

Per l'ipotesi (4.51), si ha quindi la tesi. ■

♣ **Esempio 4.22.** La Figura 4.18 mostra l'andamento degli errori associati alle successioni $T_m[f]$ e $S_m[f]$, rispettivamente formule trapezoidali e di Simpson composite, al crescere del numero di sottointervalli m , per l'integrale:

$$\int_0^\pi e^x \cos(x) dx = -\frac{(e^\pi + 1)}{2} = -12.0703463... \tag{4.55}$$

ottenuti utilizzando la doppia precisione di un calcolatore con aritmetica standard IEEE.

Dalla figura si può notare come l'errore si riduca quando il numero dei sottointervalli m aumenta. Nella Tabella 4.8 è invece riportato il rapporto degli errori delle formule trapezoidali e di Simpson composite per il problema (4.55) quando il numero dei sottointervalli m raddoppia. Si può osservare che per la formula trapezoidale composita l'errore $E_m[f]$ viene diviso approssimativamente per 4 quando m raddoppia (cioè quando si dimezza h), mentre per la formula di Simpson composita l'errore $E_m[f]$ viene diviso approssimativamente per 16 raddoppiando il numero di nodi. Tale fenomeno trova giustificazione dalle espressioni degli errori riportati nella Tabella 4.7, dalla quale si ricava il fattore di riduzione dell'errore delle varie formule composite, quando si raddoppia m ($1/m^2$ per la formula trapezoidale composita, $1/m^4$ per la formula di Simpson composita.) ♣

Diamo ora la seguente:

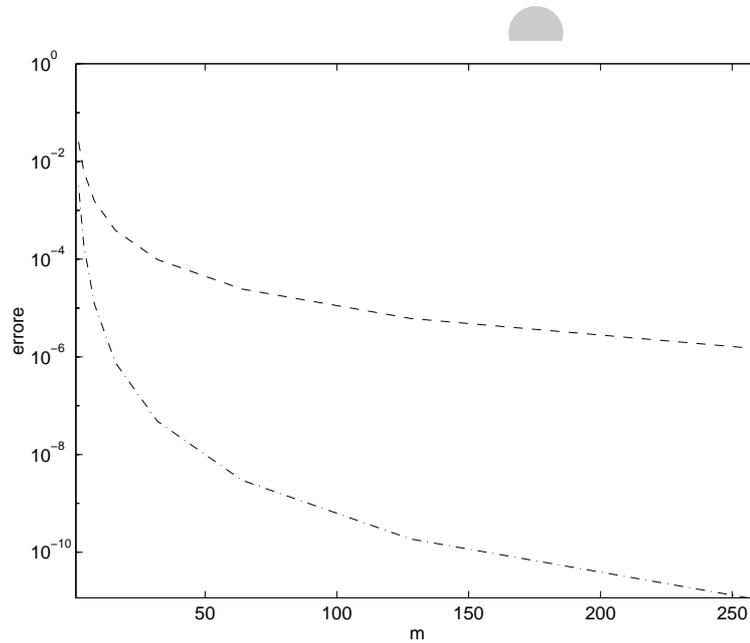


Figura 4.18: Errori delle formule composite di Simpson e trapezoidale

Trapezoidale comp.		Simpson comp.	
m	$ E_m[f]/E_{m/2}[f] $	m	$ E_m[f]/E_{m/2}[f] $
2	4.2693258D+0	2	13.915154D+0
4	4.2024268D+0	4	15.537889D+0
8	4.0590479D+0	8	15.888486D+0
16	4.0152590D+0	16	15.972376D+0
32	4.0038452D+0	32	15.993104D+0
64	4.0009632D+0	64	15.998199D+0
128	4.0002409D+0	128	15.998005D+0
256	4.0000602D+0	256	15.986632D+0

Tabella 4.8: Errori delle formule trapezoidali e di Simpson composite

Definizione 4.13. (Ordine di convergenza)

Una famiglia di formule di Newton-Cotes composite $\{Q_m[f]\}$ ha **ordine di convergenza** q se e solo se

$$E_m[f] = \mathcal{O}\left(\frac{1}{m^q}\right) \Leftrightarrow E_m[f] = \mathcal{O}(h^q) \quad h = \frac{b-a}{m(n-1)}$$

Da tale definizione e dalla Tabella 4.7 è possibile dire quindi che la formula trapezoidale composta ha ordine di convergenza 2 mentre la formula di Simpson composta ha ordine di convergenza 4.

Il Teorema 4.9 garantisce la convergenza delle formule composite al crescere del numero di sottointervalli m , facendo tendere a zero la distanza fra i nodi fissato l'intervallo di integrazione $[a, b]$. Il teorema che segue fornisce un risultato relativo alle proprietà di convergenza al crescere del numero di nodi n , delle formule di quadratura esatte per lo spazio dei polinomi.

Teorema 4.10. *Siano assegnate una funzione $f(x) \in C[a, b]$ ed una famiglia di formule del tipo:*

$$Q_n[f] = \sum_{i=1}^n A_i^{(n)} f(x_i^{(n)}) \quad n = 1, 2, \dots$$

ognuna delle quali esatta per lo spazio dei polinomi Π_{n-1} . Se la somma dei moduli dei pesi è limitata da una costante M indipendente dal numero di nodi n , cioè:

$$\sum_{i=1}^n |A_i^{(n)}| \leq M \quad \forall n = 1, 2, \dots \tag{4.56}$$

si ha:

$$\lim_{n \rightarrow \infty} E[f] = \lim_{n \rightarrow \infty} \left[\int_a^b f(x) dx - \sum_{i=1}^n A_i^{(n)} f(x_i^{(n)}) \right] = 0$$

Dimostrazione Dimostriamo che per ogni ε esiste un indice ν tale che $\forall n \geq \nu$ l'errore $E[f] \leq K\varepsilon$ con K costante indipendente da m . Fissato quindi un $\varepsilon > 0$, poichè la funzione $f(x)$ è continua, per il teorema di Weierstrass è possibile determinare un ν ed un polinomio $p(x) \in \Pi_\nu$ tale che

$$|f(x) - p(x)| < \varepsilon \quad \forall x \in [a, b] \tag{4.57}$$

L'errore $E[f]$ è quindi

$$\begin{aligned} E[f] &= \int_a^b f(x) dx - \sum_{i=1}^n A_i^{(n)} f(x_i^{(n)}) \\ &= \int_a^b f(x) dx - \int_a^b p(x) dx + \int_a^b p(x) dx - \sum_{i=1}^n A_i^{(n)} f(x_i^{(n)}) \end{aligned} \tag{4.58}$$

Per ipotesi si ha che $\forall n \geq \nu$

$$\int_a^b p(x) dx = \sum_{i=1}^n A_i^{(n)} p(x_i^{(n)}) \quad p \in \Pi_n$$

da cui, sostituendo nella (4.58), si ha:

$$E[f] = \int_a^b f(x) dx - \sum_{i=1}^n A_i^{(n)} f(x_i^{(n)}) = \int_a^b [f(x) - p(x)] dx + \sum_{i=1}^n A_i^{(n)} [p(x_i^{(n)}) - f(x_i^{(n)})]$$

e quindi per la (4.57):

$$|E[f]| = \left| \int_a^b f(x) dx - \sum_{i=1}^n A_i^{(n)} f(x_i^{(n)}) \right| \leq \varepsilon(b-a) + \varepsilon \sum_{i=1}^n |A_i^{(n)}| = \varepsilon \left((b-a) + \sum_{i=1}^n |A_i^{(n)}| \right)$$

Per l'ipotesi (4.56), posto $K = [(b-a) + M]$, si ha quindi la tesi. ■

Si può affermare quindi che, se la funzione integranda è continua, per il Teorema 4.10 e per la (4.23), le famiglie delle formule di Gauss sono convergenti. Nulla si può dire invece circa la convergenza delle formule di Newton-Cotes nell'ipotesi di sola continuità della $f(x)$, in quanto i pesi di tali formule sono di segno alterno per $n \geq 11$ e non verificano l'ipotesi (4.56) del Teorema 4.10.

♣ **Esempio 4.23.** Se si calcola il valore dell'integrale:

$$\int_{-\sqrt{3}}^{\sqrt{3}} \frac{4}{1+x^2} dx = \frac{8}{3}\pi$$

mediante la successione di formule di Newton-Cotes ottenute al variare di $n = 2, 3, 4, \dots$, tale successione è divergente, mentre nel caso dell'integrale:

$$\int_0^1 \frac{4}{1+x^2} dx = \pi$$

si ottengono i valori in Tabella 4.9.

In tal caso infatti, al crescere del numero di nodi n , l'errore delle formule di Newton-Cotes tende a zero, mentre ciò non accade nel primo caso.

Si noti che i due problemi hanno la stessa funzione integranda che è di classe C^∞ e differiscono solo per l'intervallo di integrazione. Per comprendere il perché di tale fenomeno si osservi che, essendo formule esatte per lo spazio dei polinomi, le formule di Newton-Cotes sono in particolare esatte anche per il polinomio di Lagrange $L_{n-1}(f, x)$ interpolante la funzione $f(x)$ nei nodi equispaziati $x_i, i = 1, \dots, n$. È possibile verificare che tali formule sono convergenti per tutte le funzioni per cui valgono i criteri di convergenza uniforme delle famiglie dei polinomi interpolanti di Lagrange costruiti su nodi

n	$Q[f]$
2	$.300000 \times 10^1$
3	$.313333 \times 10^1$
4	$.313846 \times 10^1$
5	$.314211 \times 10^1$
6	$.314187 \times 10^1$
7	$.314157 \times 10^1$
8	$.314157 \times 10^1$
9	$.314159 \times 10^1$
10	$.314159 \times 10^1$

Tabella 4.9: Successione dei valori ottenuti dalle formule di Newton-Cotes per $n = 2, 3, \dots, 10$

equispaziati²⁴. In tal caso è infatti possibile passare al limite sotto il segno di integrale, cioè:

$$\begin{aligned} \lim_{n \rightarrow \infty} \sum_{i=1}^n A_i f(x_i) &= \lim_{n \rightarrow \infty} \sum_{i=1}^n A_i L_n(f, x_i) = \\ \lim_{n \rightarrow \infty} \int_a^b L_n(f, x) dx &= \int_a^b \lim_{n \rightarrow \infty} L_n(f, x) dx = \int_a^b f(x) dx \end{aligned}$$

Nell'esempio in esame, la funzione definita nel campo complesso $f(z) = 4/(1 + z^2)$ (di cui la $f(x)$ è la restrizione al campo reale), ha singolarità in corrispondenza dell'unità immaginaria e del suo opposto $\pm i$. Tali singolarità, nel secondo caso cadono all'esterno dell'ellisse di centro $1/2$ e semiassi $5/8$ e $3/8$ (Figura 4.19), mentre nel primo caso cadono all'interno dell'ellisse di centro 0 e con semiassi $5\sqrt{3}/4$ e $3\sqrt{3}/4 = 1.299\dots$



²⁴Sussiste infatti il seguente

Teorema 4.11. *Condizione necessaria e sufficiente affinché la successione $\{L_{n-1}(x)\}$ converga uniformemente in $[-1, 1]$ alla funzione $f(x)$ definita in $[-1, 1]$ è che la funzione di variabile complessa $f(z)$ con $z = x + iy$ sia olomorfa in tutti i punti della regione del piano limitata dalla curva di equazione:*

$$\begin{aligned} &\frac{1}{2}(1+x)\log[(1+x)^2 + y^2] + \frac{1}{2}(1-x)\log[(1-x)^2 + y^2] \\ &+ \pi|y| - y \left(\operatorname{arctg} \frac{y}{1+x} + \operatorname{arctg} \frac{y}{1-x} \right) = 2\log 2 \end{aligned} \tag{4.59}$$

Tale regione è approssimabile ad un'ellisse il cui centro si trova nell'origine degli assi ed i cui semiassi hanno lunghezza $5/4$ e $3/4$, con il semiasse maggiore orizzontale che interseca l'asse delle ascisse. Allo stesso risultato si può pervenire anche facendo ricorso al *teorema della lemniscata*, che determina le condizioni di convergenza uniforme di una successione di polinomi interpolanti una funzione data. A causa dell'espressione della curva, in generale è abbastanza difficile verificare se la funzione $f(z)$ sia olomorfa nella regione limitata dalla (4.59). In generale, per un intervallo $[a, b]$, è sufficiente verificare che la $f(z)$ sia olomorfa all'interno dell'ellisse \mathcal{E} di centro $(a + b)/2$ e semiassi $5(b - a)/8$ e $3(b - a)/8$ che racchiude la figura in esame.

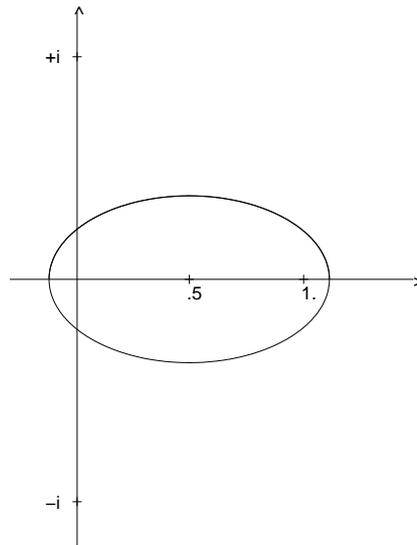


Figura 4.19: Regione di convergenza per le formule di Newton-Cotes nell'intervallo $[0, 1]$

4.6 Condizionamento di una formula di quadratura

Per tutti i problemi matematici che si vogliono risolvere a partire da dati rappresentati in un sistema aritmetico a precisione finita o comunque affetti da errore, la prima analisi da effettuare per stimare l'attendibilità del risultato è quella del **condizionamento** del problema, misurare cioè quanto varia la soluzione del problema introducendo una perturbazione sui dati. Poiché i dati di una formula di quadratura sono i pesi A_i ed i valori della funzione integranda $f(x)$ nei nodi x_i , in questo paragrafo e nel successivo useremo il simbolo $Q[A, f]$ al posto di $Q[f]$ in modo da evidenziare la dipendenza del problema Q dai dati A e f . Siano ΔA_i e $\Delta f(x_i)$ rispettivamente le perturbazioni effettuate sui pesi A_i e sulle valutazioni $f(x_i)$, cioè:

$$\tilde{A}_i = A_i + \Delta A_i \quad \widetilde{f(x_i)} = f(x_i) + \Delta f(x_i)$$

Sussiste il seguente:

Teorema 4.12. *Se $Q[A, f] \neq 0$, l'errore relativo della soluzione del problema con dati perturbati $Q[\tilde{A}, \tilde{f}]$ rispetto alla soluzione del problema non perturbato $Q[A, f]$ è:*

$$\frac{|Q[\tilde{A}, \tilde{f}] - Q[A, f]|}{|Q[A, f]|} \leq \frac{\sum_{i=1}^n |A_i| |f(x_i)|}{|\sum_{i=1}^n A_i f(x_i)|} \left(\max_i \left| \frac{\Delta A_i}{A_i} \right| + \max_i \left| \frac{\Delta f(x_i)}{f(x_i)} \right| \right) \quad (4.60)$$

Dimostrazione Dalle posizioni fatte si ha:

$$\begin{aligned} Q[\tilde{A}, \tilde{f}] &= \sum_{i=1}^n (A_i + \Delta A_i)(f(x_i) + \Delta f(x_i)) \\ &= \sum_{i=1}^n A_i f(x_i) + \sum_{i=1}^n A_i \Delta f(x_i) + \sum_{i=1}^n \Delta A_i f(x_i) + \sum_{i=1}^n \Delta A_i \Delta f(x_i) \end{aligned}$$

da cui, trascurando i termini $\Delta A_i \Delta f(x_i)$, si ottiene:

$$\begin{aligned} |Q[\tilde{A}, \tilde{f}] - Q[A, f]| &\leq \sum_{i=1}^n (|A_i| |\Delta f(x_i)| + |\Delta A_i| |f(x_i)|) \\ &= \sum_{i=1}^n |A_i| |f(x_i)| \frac{|\Delta f(x_i)|}{|f(x_i)|} + \sum_{i=1}^n |A_i| |f(x_i)| \frac{|\Delta A_i|}{|A_i|} \end{aligned}$$

da cui:

$$\frac{|Q[\tilde{A}, \tilde{f}] - Q[A, f]|}{|Q[A, f]|} \leq \frac{\sum_{i=1}^n |A_i| |f(x_i)|}{|\sum_{i=1}^n A_i f(x_i)|} \left(\max_i \left| \frac{\Delta A_i}{A_i} \right| + \max_i \left| \frac{\Delta f(x_i)}{f(x_i)} \right| \right)$$

e quindi la tesi. ■

Dalla (4.60) si nota che la quantità:

$$\mu(Q[A, f]) = \frac{\sum_{i=1}^n |A_i| |f(x_i)|}{|\sum_{i=1}^n A_i f(x_i)|} \quad (4.61)$$

rappresenta il fattore di amplificazione delle perturbazioni

$$\frac{|\Delta A_i|}{|A_i|} \quad \text{e} \quad \frac{|\Delta f(x_i)|}{|f(x_i)|},$$

ed è quindi l'**indice di condizionamento relativo** del problema in esame²⁵. Si noti infine che $\mu(Q[A, f]) \geq 1, \forall Q[A, f]$.

²⁵Si osservi che, considerata la formula di quadratura $Q[A, f]$ in funzione delle sue $2n$ variabili, $A_i, i = 1, \dots, n$ e $f(x_i), i = 1, \dots, n$, dalla definizione di indice di condizionamento relativo di una funzione in n variabili,

$$C(f, x_1, \dots, x_n) = \frac{|f_{x_1 x_1}| + |f_{x_2 x_2}| + \dots + |f_{x_n x_n}|}{|f|}$$

segue che l'indice di condizionamento relativo per la formula di quadratura in esame può esprimersi come:

$$\mu(Q[A, f]) = 2 \cdot \frac{\sum_{i=1}^n |A_i| |f(x_i)|}{|\sum_{i=1}^n A_i f(x_i)|} \quad (4.62)$$

Tuttavia tale risultato non è in disaccordo con la (4.61), se si considera che la quantità (4.62) rappresenta il fattore di amplificazione del *massimo* errore relativo presente sulla totalità dei dati, siano essi pesi o valutazioni della funzione integranda nei nodi.

♣ **Esempio 4.24.** Se $Q[A, f]$ è una formula di quadratura con pesi non negativi, cioè $A_i \geq 0$, l'indice di condizionamento è:

$$\mu(Q[A, f]) = \frac{\sum_{i=1}^n A_i |f(x_i)|}{|\sum_{i=1}^n A_i f(x_i)|}$$

Se inoltre la funzione $f(x)$ non cambia di segno nell'intervallo di integrazione, i prodotti $A_i f(x_i)$ hanno tutti lo stesso segno e quindi $\sum_{i=1}^n A_i |f(x_i)| = |\sum_{i=1}^n A_i f(x_i)|$. In tal caso l'indice di condizionamento assume il valore ottimale $\mu(Q[A, f]) = 1$.

Esempi di formule con le caratteristiche citate sono le formule di Gauss e le formule composite basate su formule con pesi non negativi (ad esempio le formule composite trapezoidale e di Simpson). ♣

Se la funzione $f \in C[a, b]$ posto $M = \max_{[a, b]} |f(x)|$ per la (4.61) si ha:

$$\mu(Q[A, f]) \leq \frac{M \sum_{i=1}^n |A_i|}{|Q[A, f]|} \quad (4.63)$$

Pertanto $\mu(Q[A, f])$ dipende strettamente dalla somma dei valori assoluti dei pesi $|A_i|$ della formula di quadratura. Le formule di Newton-Cotes, per $n \geq 9$ hanno pesi di segno alterno e non limitati in modulo²⁶, per cui il condizionamento di tali formule peggiora al crescere di n . Per tale motivo le formule di Newton-Cotes vengono utilizzate con un numero di nodi molto basso, in genere $n \leq 8$. I pesi delle formule di Gauss, invece, sono tutti positivi, e per la (4.23) si ha

$$\sum_{i=1}^n |A_i| = \sum_{i=1}^n A_i = (b - a)$$

e quindi $\mu(Q[A, f])$ è limitato per ogni valore di n . Le formule di Gauss, quindi, possono essere considerate ben condizionate.

Osservazione 4.1. Si osserva che, per una funzione integrabile secondo Riemann in un intervallo $[a, b]$, assegnati $n + 1$ nodi $x_i^{(n)}$ in $[a, b]$ ($i = 0, \dots, n$) e $n + 1$ pesi $A_i^{(n)}$ ($i = 0, \dots, n$), una formula di quadratura può essere rappresentata come

$$Q_n[A, f] = A_0^{(n)} f(x_0^{(n)}) + \dots + A_n^{(n)} f(x_n^{(n)})$$

oppure, se ciò non genera ambiguità, semplicemente, come

$$Q_n[A, f] = A_0 f(x_0) + \dots + A_n f(x_n)$$

o, direttamente, come $Q[A, f]$.

Con questa notazione si può riformulare la definizione di formula di quadratura composta:

²⁶Ad eccezione, come si è detto, del caso $n = 10$ in quanto i pesi sono tutti positivi e la formula risulta ben condizionate.

Definizione 4.14. [11]

Sia $Q[A, f]$ una formula di quadratura in $[a, b]$. Partizionando questo intervallo in m ($m \geq 2$) sottointervalli $[t_{j-1}, t_j]$, ($j = 1, \dots, m$) ciascuno di ampiezza $\tau_j = t_j - t_{j-1}$, la **formula composta** di $Q[A, f]$ è:

$$Q_{m,n}(f) = \sum_{j=1}^m Q^{(j)}[f] = \sum_{j=1}^m \sum_{i=0}^n A_{i,j} f(x_{i,j})$$

dove $Q^{(j)}[f] = \sum_{i=0}^n A_{i,j} f(x_{i,j})$ si ottiene applicando $Q[A, f]$ ai sottointervalli $[t_{j-1}, t_j]$, cioè

$$A_{i,j} = A_i \frac{t_j - t_{j-1}}{b - a} \quad x_{i,j} = t_{j-1} + i \frac{t_j - t_{j-1}}{n} \quad i = 1, \dots, n .$$

Si può, dunque, dare la seguente

Definizione 4.15.

Sia f una funzione integrabile secondo Riemann su un intervallo reale $[a, b]$ e $Q_n[A, f]$ una formula di quadratura in $[a, b]$. Se $Q_n[A, f]$ è diversa da zero, l'**indice di condizionamento relativo** $\mu(Q)$ di $Q_n[A, f] = \sum_{i=0}^n A_i f(x_i)$, è definito come segue:

$$\mu(Q) = \mu(Q_n[A, f]) \stackrel{\text{def}}{=} \frac{\sum_{i=0}^n |A_i| |f(x_i)|}{|\sum_{i=0}^n A_i f(x_i)|} \quad (4.64)$$

L'indice di condizionamento relativo è molto utile nell'analisi degli errori di una formula di quadratura, quando è utilizzata come nucleo computazionale di un software di quadratura numerica. In effetti essa dipende dai pesi e dalle valutazioni della funzione.

Analogamente è possibile dimostrare che, se $\Delta A_i = 0, \forall i = 1, \dots, n$ allora si ha:

$$\sum_{i=1}^n \widetilde{A}_i \widetilde{\Delta f}(x_i) = \sum_{i=1}^n A_i f(x_i) + \sum_{i=1}^n A_i \Delta f(x_i)$$

da cui:

$$|Q[\widetilde{A}, \widetilde{f}] - Q[A, f]| \leq \max_i |\Delta f(x_i)| \sum_{i=1}^n |A_i|$$

Si definisce quindi **indice di condizionamento assoluto** la quantità:

$$\nu(Q[A, f]) = \sum_{i=1}^n |A_i|$$

Tale definizione mette nuovamente in risalto la dipendenza del condizionamento di una formula di quadratura dalla somma dei valori assoluti dei pesi.

4.6.1 Indice di condizionamento assoluto

In questo paragrafo si analizza l'indice di condizionamento assoluto per le seguenti famiglie di formule di quadratura: le formule di Gauss e di Gauss-Kronrod, le formule di Newton-Cotes e le formule interpolatorie composite.

Enunciamo alcune definizioni e risultati preliminari:

Definizione 4.16. [2]

Sia $Q_n[A, f]$ una formula di quadratura in $[a, b]$ con $x_0 < x_1 < \dots < x_n$. Se $x_0 = a$ e $x_n = b$ allora si dice che $Q_n[A, f]$ è **chiusa**, altrimenti, se $x_0, x_n \in (a, b)$, si dice che $Q_n[A, f]$ è **aperta**.

Proposizione 4.1. [11]

Sia s un intero non negativo, $[c, d]$ e $[a, b]$ due intervalli e

$$t = \phi(x) = x \cdot \frac{b-a}{d-c} + \frac{ad-bc}{d-c} \quad (4.65)$$

la trasformazione lineare da $[c, d]$ in $[a, b]$. Sia $w_1(x)$ una funzione peso su $[c, d]$, $w_2(t) = w_1(\phi^{-1}(t))$ la funzione peso su $[a, b]$ ottenuta da $w_1(x)$ e $Q[B, f] = \sum_{i=0}^n B_i f(x_i)$ una formula di quadratura in $[c, d]$, con grado di precisione algebrico s rispetto a $w_1(x)$, cioè:

$$\int_c^d w_1(x)q(x)dx = \sum_{i=0}^n B_i q(x_i) \quad \forall q \in \Pi_s \quad (4.66)$$

Consideriamo la formula di quadratura su $[a, b]$ $Q[A, f] = \sum_{i=0}^n A_i f(t_i)$, i cui pesi A_i sono proporzionali ai pesi B_i di $Q[B, f]$ per mezzo del fattore $\frac{b-a}{d-c}$ ed i cui nodi t_i sono ottenuti dai nodi x_i mediante la $t = \phi(x)$.

Allora $Q[A, f]$ ha grado di precisione algebrico s , su $[a, b]$, rispetto a $w_2(x)$, cioè:

$$\int_a^b w_2(t)q(t)dt = \sum_{i=0}^n A_i q(t_i) \quad \forall q \in \Pi_s .$$

Proposizione 4.2.

Sia s un intero non negativo, $[c, d]$ e $[a, b]$ due intervalli, w_1 e w_2 come nella **Proposizione 4.1** e $Q[B, f] = \sum_{i=0}^n B_i f(x_i)$ una formula di quadratura nell'intervallo $[c, d]$, con grado di precisione algebrico s rispetto a w_1 . Sia $Q[A, f]$ la formula di quadratura in $[a, b]$, con grado di precisione algebrico s rispetto a w_2 , ottenuta da $Q[B, f]$ attraverso la trasformazione lineare (4.65), allora:

$$\nu(Q[A, f]) = \frac{b-a}{d-c} \nu(Q[B, f]) .$$

Dimostrazione Per A_i e B_i sussiste $A_i = \frac{b-a}{d-c} \cdot B_i$ ($i = 0, \dots, n$), quindi:

$$\nu(Q[A, f]) = \sum_{i=0}^n |A_i| = \sum_{i=0}^n \frac{b-a}{d-c} \cdot |B_i| = \frac{b-a}{d-c} \nu(Q[B, f]) .$$



Le formule di Newton-Cotes con $n + 1$ nodi sono interpolatorie ed hanno grado di precisione algebrico almeno n , quindi dalla **Proposizione 4.1** e dalla **4.2** segue che:

Corollario 4.4.

Sia $Q_n[B, f]$ la formula di Newton-Cotes nell'intervallo $[0, 1]$ e $Q_n[A, f]$ la formula di Newton-Cotes in $[a, b]$, allora:

$$\nu(Q_n[A, f]) = (b - a) \cdot \nu(Q_n[B, f]) . \quad (4.67)$$

Formule di Gauss e Gauss-Kronrod

Per quanto riguarda le formule di Gauss e Gauss-Kronrod osserviamo, preliminarmente, che:

Osservazione 4.2. [25]

Se $Q[A, f] = \sum_{i=0}^n A_i f(x_i)$ ha pesi positivi ed è esatta per funzioni costanti, rispetto alla funzione peso w , allora:

$$\nu(Q) = \int_a^b w(x) dx .$$

Le formule di Gauss hanno pesi positivi [13] e sono esatte per funzioni costanti, allora:

Corollario 4.5. [2]

L'indice di condizionamento assoluto di una formula di Gauss nell'intervallo $[a, b]$, rispetto alla funzione peso w , è $\int_a^b w(x) dx$.

Le formule di Gauss-Kronrod hanno pesi positivi [17], sono esatte per funzioni costanti, così:

Corollario 4.6. [2]

L'indice di condizionamento assoluto di una formula di Gauss-Kronrod in $[a, b]$ è $b - a$.

Dal precedente corollario, segue che l'indice di condizionamento assoluto delle formule di Gauss e Gauss-Kronrod è indipendente dai nodi, da cui segue che esse sono formule "ben condizionate".

Formule di Newton-Cotes

Per l'indice di condizionamento assoluto di una formula di Newton-Cotes, segue:

Teorema 4.13. [11] (Kusmin)

Sia $Q_n[A, f]$ la formula di Newton-Cotes nell'intervallo $[a, b]$ con $n + 1$ nodi, allora

$$\lim_{n \rightarrow +\infty} \nu(Q_n[A, f]) = +\infty \quad (4.68)$$

Questo risultato prova che l'indice di condizionamento assoluto diverge al crescere di n . Ora, analizziamo il comportamento asintotico dell'indice di condizionamento assoluto delle formule di Newton-Cotes. Consideriamo il seguente:

Lemma 4.1.

Per ogni $n \in \mathbb{N} - \{1\}$ e per ciascun $i = 1, \dots, n - 1$, sussiste:

$$\frac{4}{n^2 - 1} \leq \left| \frac{1}{i} + \frac{(-1)^n}{(n - i)} \right| \leq \frac{n}{n - 1} \quad (4.69)$$

Dimostrazione Per un fissato $n \in \mathbb{N} - \{1\}$, consideriamo la funzione reale:

$$\varphi(x) = \left| \frac{1}{x} + \frac{(-1)^n}{n - x} \right| = \left| \frac{n - x + (-1)^n x}{(n - x)x} \right|.$$

Vogliamo determinare il massimo ed il minimo valore di φ assunto in corrispondenza di valori interi nell'intervallo $[1, n - 1]$. Osserviamo che $\varphi(x)$ è simmetrica rispetto a $x = n/2$. In effetti,

$$\begin{aligned} \varphi(n - x) &= \left| \frac{n - (n - x) + (-1)^n (n - x)}{x(n - x)} \right| = \left| \frac{x + (-1)^n (n - x)}{(n - x)x} \cdot (-1)^n \right| = \\ &= \left| \frac{(-1)^n x + (n - x)(-1)^{2n}}{(n - x)x} \right| = \left| \frac{n - x + (-1)^n x}{(n - x)x} \right| = \varphi(x). \end{aligned}$$

Quindi possiamo limitarci a trovare il massimo ed il minimo valore solo in $[1, n/2]$. Distinguiamo i due casi: n pari e n dispari.

Se n è pari, allora:

$$\varphi(x) = \left| \frac{n - x + x}{(n - x)x} \right| = \frac{n}{(n - x)x}.$$

Derivando

$$\varphi'(x) = \frac{n(2x - n)}{(n - x)^2 x^2}.$$

Se $x \in [1, n/2]$, φ' è negativa e si annulla se e solo se $x = n/2$. Quindi φ è una funzione decrescente ed assume il massimo ed il minimo valore nei punti estremi dell'intervallo: $x = 1$ e $x = n/2$ che sono interi (n è pari e $n/2$ è un intero), così:

$$\varphi\left(\frac{n}{2}\right) = \frac{4}{n} \leq \left| \frac{n - i + (-1)^n i}{(n - i)i} \right| \leq \frac{n}{n - 1} = \varphi(1) \quad (i = 1, \dots, n - 1) \quad (4.70)$$

Se n è dispari, allora:

$$\varphi(x) = \left| \frac{n - x - x}{(n - x)x} \right| = \frac{n - 2x}{(n - x)x}.$$

Derivando

$$\varphi'(x) = -\frac{[(n-2x)^2 + 2x(n-x)]}{x^2(n-x)^2}.$$

In $[1, n/2]$, φ' è negativa. Quindi φ è una funzione decrescente ed, in corrispondenza di valori interi, assume il massimo valore nell'estremo dell'intervallo $x = 1$ ed il minimo valore in corrispondenza dell'intero $x = (n-1)/2$ (n dispari), così:

$$\varphi\left(\frac{n-1}{2}\right) = \frac{4}{n^2-1} \leq \left| \frac{n-i+(-1)^ni}{(n-i)i} \right| \leq \frac{n-2}{n-1} = \varphi(1) \quad (i = 1, \dots, n-1) \quad (4.71)$$

Combinando la (4.70) e la (4.71) si prova la tesi. ■

Possiamo dimostrare, ora, il seguente:

Teorema 4.14.

Sia $Q_n[A, f] = \sum_{i=0}^n A_i f(x_i)$ la formula di Newton-Cotes nell'intervallo $[a, b]$, allora, asintoticamente, sussiste il risultato:

$$\psi_1(n) \frac{2^{n+2}}{(n^3-n) \log^2 n} (b-a) < \nu(Q_n[A, f]) < \psi_2(n) \frac{2^n}{(n-1) \log^2 n} (b-a) \quad (4.72)$$

dove

$$\psi_1(n) = \left[1 + \left(\frac{2(n^2-1) \log n - 8}{2^{n+2}} \right) \right] K_n, \quad \psi_2(n) = \left[1 + \left(\frac{2 \log n - 2}{2^n} \right) \right] K_n,$$

$$K_n = \left[1 + \mathcal{O}\left(\frac{1}{\log n}\right) \right] e \quad \lim_{n \rightarrow \infty} \psi_1(n) = \lim_{n \rightarrow \infty} \psi_2(n) = 1$$

Dimostrazione Consideriamo la formula di Newton-Cotes in $[0, 1]$, allora $\nu(Q_n[B, f]) = \sum_{i=0}^n |B_i|$. Dalla (4.27), per $i = 1, 2, \dots, n-1$:

$$|B_i| = \frac{1}{n \log^2 n} \binom{n}{i} \left| \frac{n-i+(-1)^ni}{(n-i)i} \right| K_n \quad (4.73)$$

Usando la (4.26), per B_0 e B_n :

$$\frac{1}{n \log n} K_n = |B_0| = |B_n| < \frac{1}{(n-1) \log n} K_n \quad (4.74)$$

Dalla (4.73) e dal **Lemma 4.1**, per $i = 1, 2, \dots, n-1$:

$$\frac{4}{(n^3-n) \log^2 n} \binom{n}{i} K_n \leq |B_i| \leq \frac{1}{(n-1) \log^2 n} \binom{n}{i} K_n \quad (4.75)$$

Infine, sommando le disuguaglianze nella (4.74) e nella (4.75), per $i = 0, \dots, n$:

$$|B_0| + |B_n| + \sum_{i=1}^{n-1} \frac{4 \binom{n}{i} K_n}{(n^3-n) \log^2 n} < |B_0| + |B_n| + \sum_{i=1}^{n-1} |B_i| < \frac{2K_n}{(n-1) \log n} + \sum_{i=1}^{n-1} \frac{\binom{n}{i} K_n}{(n-1) \log^2 n}$$

$$\begin{aligned}
\frac{2K_n}{n \log n} + \frac{4K_n}{(n^3 - n) \log^2 n} \sum_{i=1}^{n-1} \binom{n}{i} &< \nu(Q_n[B, f]) < \frac{2K_n}{(n-1) \log n} + \frac{K_n}{(n-1) \log^2 n} \sum_{i=1}^{n-1} \binom{n}{i} \\
\frac{[4(2^n - 2) + 2(n^2 - 1) \log n] K_n}{(n^3 - n) \log^2 n} &< \nu(Q_n[B, f]) < \frac{[(2^n - 2) + 2 \log n] K_n}{(n-1) \log^2 n} \\
\frac{2^{n+2} \left[1 + \left(\frac{2(n^2 - 1) \log n - 8}{2^{n+2}} \right) \right] K_n}{(n^3 - n) \log^2 n} &< \nu(Q_n[B, f]) < \frac{2^n \left[1 + \left(\frac{2 \log n - 2}{2^n} \right) \right] K_n}{(n-1) \log^2 n} \quad (4.76)
\end{aligned}$$

Sostituendo ψ_1 e ψ_2 nella (4.76) e moltiplicando per l'ampiezza dell'intervallo $[a, b]$, si ottiene la tesi. ■

Questo teorema è utile per stimare il comportamento asintotico dell'indice di condizionamento assoluto di una formula di Newton-Cotes. Questo risultato consente di dedurre il comportamento della convergenza per una classe di funzioni, come dimostrato nella prossima sezione.

Formule di quadratura interpolatorie composite

Analizziamo l'indice di condizionamento assoluto per la classe di formule interpolatorie composite. In accordo con la notazione nella **Definizione 4.14** dimostriamo il seguente:

Teorema 4.15.

Sia $Q[A, f] = \sum_{i=0}^n A_i f(x_i)$ una formula di quadratura interpolatoria. Sia $Q_{m,n}(f)$ una formula composta di $Q[A, f]$ su m ($m \geq 2$) sottointervalli. Allora:

$$\nu(Q_{m,n}(f)) \leq \nu(Q) .$$

dove la disuguaglianza stretta sussiste se e solo se $Q[A, f]$ è una formula chiusa e se ha A_0 ed A_n di segno discorde.

Per dimostrare il **Teorema 4.15** facciamo una distinzione tra formule base chiuse e formule base non chiuse. Se una formula base non è chiusa, allora:

Lemma 4.2.

Sia $Q[A, f] = \sum_{i=0}^n A_i f(x_i)$ una formula di quadratura non chiusa, $Q_{m,n}(f)$ una formula composta di $Q[A, f]$ su m ($m \geq 2$) sottointervalli e $Q^{(j)}[f] = \sum_{i=0}^n A_{i,j} f(x_{i,j})$ la formula ottenuta applicando $Q[A, f]$ in $[t_{j-1}, t_j]$. Allora

$$\nu(Q_{m,n}(f)) = \sum_{j=1}^m \nu(Q^{(j)}[f]) . \quad (4.77)$$

Dimostrazione $Q[A, f]$ è una formula non chiusa quindi, se denotiamo con W_k i pesi di $Q_{m,n}(f)$, segue che:

$$\nu(Q_{m,n}(f)) = \sum_{k=0}^{m(n+1)-1} |W_k| = \sum_{j=1}^m \sum_{i=0}^n |A_{i,j}| = \sum_{j=1}^m \nu(Q^{(j)}[f])$$

■

Lemma 4.3.

Sia $Q[A, f] = \sum_{i=0}^n A_i f(x_i)$ una formula di quadratura interpolatoria in $[a, b]$ e $Q^{(j)}[f] = \sum_{i=0}^n A_{i,j} f(x_{i,j})$ la formula ottenuta applicando $Q[A, f]$ in $[t_{j-1}, t_j]$. Allora:

$$\nu(Q^{(j)}[f]) = \frac{t_j - t_{j-1}}{b - a} \nu(Q) . \tag{4.78}$$

Dimostrazione Applicando $Q[A, f]$ in $[t_{j-1}, t_j]$, costruiamo $Q^{(j)}[f]$ mediante una trasformazione lineare. I pesi $A_{i,j}$ sono proporzionali ai pesi A_i , mediante il fattore $\frac{t_j - t_{j-1}}{b - a}$. La **Proposizione 4.2** completa la dimostrazione. ■

Da qui, sfruttando il **Lemma 4.2** ed il **Lemma 4.3**, dimostriamo la seguente:

Proposizione 4.3.

Sia $Q[A, f] = \sum_{i=0}^n A_i f(x_i)$ una formula di quadratura interpolatoria non chiusa. Sia $Q_{m,n}(f)$ ($m \geq 2$) una formula composta di $Q[A, f]$ su m sottointervalli. Allora:

$$\nu(Q_{m,n}(f)) = \nu(Q) .$$

Dimostrazione Dal **Lemma 4.2**:

$$\nu(Q_{m,n}(f)) = \sum_{j=1}^m \nu(Q^{(j)}[f]) .$$

$Q[A, f]$ è interpolatoria e dal **Lemma 4.3**:

$$\nu(Q^{(j)}[f]) = \frac{t_j - t_{j-1}}{b - a} \nu(Q)$$

allora

$$\nu(Q_{m,n}(f)) = \sum_{j=1}^m \nu(Q) \left(\frac{t_j - t_{j-1}}{b - a} \right) = \frac{\nu(Q)}{b - a} \sum_{j=1}^m (t_j - t_{j-1}) = \nu(Q)$$

■

Se la formula base è chiusa, allora:

Lemma 4.4.

Sia $Q[A, f] = \sum_{i=0}^n A_i f(x_i)$ una formula di quadratura chiusa e $Q_{m,n}(f)$ una formula composta di $Q[A, f]$ su m ($m \geq 2$) sottointervalli. Allora:

$$\nu(Q_{m,n}(f)) \leq \sum_{j=1}^m \nu(Q^{(j)}[f]) \quad (4.79)$$

dove l'uguaglianza sussiste se e solo se A_0 e A_n hanno lo stesso segno.

Dimostrazione $Q[A, f]$ è una formula chiusa, allora se denotiamo con W_k i pesi di $Q_{m,n}(f)$, segue che:

$$W_{tn} = A_{n,t} + A_{0,t+1} \quad t = 1, \dots, m-1$$

e quindi

$$|W_{tn}| \leq |A_{n,t}| + |A_{0,t+1}|$$

dove il segno dell'uguaglianza sussiste se e solo se $A_{n,t}$ e $A_{0,t+1}$ hanno lo stesso segno o, equivalentemente, se A_0 e A_n di $Q[A, f]$ hanno lo stesso segno. Questo dimostra la tesi:

$$\nu(Q_{m,n}(f)) = \sum_{k=0}^{mn} |W_k| \leq \sum_{j=1}^m \sum_{i=0}^n |A_{i,j}| = \sum_{j=1}^m \nu(Q^{(j)}[f]) .$$

■

Usando il **Lemma 4.3** ed il **Lemma 4.4**, segue:

Proposizione 4.4.

Sia $Q[A, f] = \sum_{i=0}^n A_i f(x_i)$ una formula di quadratura interpolatoria e chiusa. Sia $Q_{m,n}(f)$ ($m \geq 2$) una formula composta di $Q[A, f]$ su m sottointervalli. Allora:

$$\nu(Q_{m,n}(f)) \leq \nu(Q)$$

dove l'uguaglianza sussiste se e solo se A_0 e A_n hanno lo stesso segno.

Dimostrazione Dal **Lemma 4.4**:

$$\nu(Q_{m,n}(f)) \leq \sum_{j=1}^m \nu(Q^{(j)}[f]) .$$

$Q[A, f]$ è interpolatoria e, dal **Lemma 4.3**:

$$\nu(Q^{(j)}[f]) = \frac{t_j - t_{j-1}}{b-a} \nu(Q)$$

allora:

$$\nu(Q_{m,n}(f)) \leq \sum_{j=1}^m \nu(Q) \left(\frac{t_j - t_{j-1}}{b-a} \right) = \frac{\nu(Q)}{b-a} \sum_{j=1}^m (t_j - t_{j-1}) = \nu(Q)$$

Le **Proposizioni 4.3** e **4.4** dimostrano il **Teorema 4.15**. ■

Abbiamo due semplici corollari del **Teorema 4.15**. Le formule di Gauss e Gauss-Kronrod sono interpolatorie aperte, quindi:

Corollario 4.7.

Le formule composite di Gauss e di Gauss-Kronrod hanno lo stesso indice di condizionamento assoluto delle corrispondenti formule base.

Lo stesso risultato resta vero per una formula di Newton-Cotes. Infatti esse sono formule interpolatorie e chiuse ed hanno $A_0 = A_n$.

Corollario 4.8.

Le formule composite di Newton-Cotes hanno lo stesso indice di condizionamento assoluto delle corrispondenti formule base.

4.6.2 Errore di round-off nella valutazione di una formula di quadratura

L'approssimazione di un integrale $I[f]$ mediante una formula di quadratura è affetta, oltre che dall'errore di discretizzazione $E[f]$:

$$E[f] = I[f] - Q[A, f],$$

analizzato nei paragrafi precedenti, anche dall'errore di round-off R , generato dal calcolo della somma $Q[A, f]$ in un sistema aritmetico a precisione finita. Cioè:

$$R = Q^*[A, f] - Q[A, f]$$

dove $Q^*[A, f]$ è il risultato dell'algoritmo utilizzato per la somma che definisce la formula di quadratura e che diremo *risultato computazionale*. L'errore complessivo è quindi maggiorato dalla relazione:

$$|I[f] - Q^*[A, f]| \leq |E[f]| + |R|$$

♣ **Esempio 4.25.** Si calcoli:

$$I[f] = \int_0^{\pi/2} (3x + 4) dx = 9.9843,$$

mediante la formula trapezoidale composta $T_m[f]$ al variare del numero di sottointervalli $m = 2^k$, $k = 1, \dots, 16$ in un sistema aritmetico a precisione finita. Nella Tabella 4.10 sono riportati gli errori $|T_m^*[f] - I[f]|$ calcolati utilizzando un calcolatore con aritmetica standard IEEE in singola precisione. Si noti

m	$ T_m^*[f] - I[f] $	m	$ T_m^*[f] - I[f] $
2	.0000000000E+00	1024	.0000000000E+00
4	.0000000000E+00	2048	.5435943604E-04
8	.0000000000E+00	4096	.9346008301E-04
16	.0000000000E+00	8192	.5722045898E-04
32	.0000000000E+00	16384	.3242492676E-04
64	.0000000000E+00	32768	.1525878906E-04
128	.0000000000E+00	65536	.3433227539E-04
256	.0000000000E+00	131072	.1621246338E-04
512	.0000000000E+00	262144	.4596710205E-03

Tabella 4.10: Errore di round-off per la formula trapezoidale composta

che la formula trapezoidale composta ha grado di precisione algebrico 1, e quindi dovrebbe fornire, in un'aritmetica a precisione infinita, un errore nullo qualunque sia il valore del numero di sottointervalli m .

Dalla Tabella si può notare che fino a $m = 1024$ l'errore $|T_m^*[f] - I[f]|$ è nullo, ma per $m > 1024$ l'errore è diverso da zero. Il degrado dell'accuratezza dopo un numero elevato di passi è dovuto ovviamente all'accumulo degli errori di round-off nella valutazione della somma della formula di quadratura. ♣

Allo scopo di fornire una stima per $|R|$ si considerino le somme:

$$S_k = \sum_{i=1}^k A_i f(x_i) = S_{k-1} + A_k f(x_k) \quad k = 1, \dots, n \quad \text{con } S_0 = 0 \quad (4.80)$$

Posto $|\sigma_i| \leq u$ e $|\tau_i| \leq u$ rispettivamente gli errori di round-off associati alle operazioni di prodotto e di somma e $u = \frac{1}{2} \times \beta^{1-t}$ la massima accuratezza relativa del sistema aritmetico floating point \mathcal{F} a precisione t , la sequenza calcolata da un algoritmo che valuta le (4.80) in \mathcal{F} , è la seguente:

$$\begin{aligned}
 S_1^* &= A_1 f(x_1)(1 + \sigma_1) \\
 S_2^* &= [S_1^* + A_2 f(x_2)(1 + \sigma_2)](1 + \tau_2) = \\
 &= [A_1 f(x_1)(1 + \sigma_1) + A_2 f(x_2)(1 + \sigma_2)](1 + \tau_2) = \\
 &= A_1 f(x_1)(1 + \sigma_1)(1 + \tau_2) + A_2 f(x_2)(1 + \sigma_2)(1 + \tau_2) \\
 &\vdots \\
 Q^*[A, f] &= S_n^* = \\
 &= A_1 f(x_1)(1 + \tau_2) \cdots (1 + \tau_n)(1 + \sigma_1) + \\
 &+ A_2 f(x_2)(1 + \tau_2) \cdots (1 + \tau_n)(1 + \sigma_2) + \\
 &+ A_n f(x_n)(1 + \tau_n)(1 + \sigma_n)
 \end{aligned} \quad (4.81)$$

Per semplificare le espressioni precedenti è possibile non fare distinzione tra gli errori di round-off nelle somme e nei prodotti (rispettivamente τ_i e σ_i), identificandoli tutti con

δ_i . In tal modo, ricordando che se $|\delta_i| \leq u$ e $nu < 1$ si ha²⁷:

$$\prod_{i=1}^n (1 + \delta_i) = 1 + \theta_n \quad |\theta_n| \leq \frac{nu}{1 - nu} \quad (4.82)$$

per cui, applicando la (4.82) all'ultima uguaglianza della (4.81), si ha:

$$\begin{aligned} Q^*[A, f] &= A_1 f(x_1)(1 + \theta_n) + A_2 f(x_2)(1 + \theta_n) \\ &+ A_3 f(x_3)(1 + \theta_{n-1}) + \dots + A_n f(x_n)(1 + \theta_2) \end{aligned} \quad (4.83)$$

Dalla (4.83) si ricava che $Q^*[A, f]$ può essere considerata, dal punto di vista della backward error analysis, come la formula di quadratura ottenuta in un sistema aritmetico

²⁷ Sussiste il seguente

Teorema 4.16. *Assegnato un insieme finito di valori*

$$\{\delta_i\}_{i=1, \dots, n}$$

ed una costante u , tali che

$$|\delta_i| \leq u = \frac{1}{2}\beta^{1-t}, \quad \rho_i = \pm 1, \quad i = 1, \dots, n, \quad e \quad nu < 1,$$

allora

$$\prod_{i=1}^n (1 + \delta_i)^{\rho_i} = 1 + \theta_n \quad \text{dove} \quad |\theta_n| \leq \frac{nu}{1 - nu} := \gamma_n$$

Dimostrazione Ragionando per induzione su n , per $\rho_n = 1$, se $n = 1$ la verifica del teorema è immediata. Si supponga che il teorema sia vero per $n - 1$, allora:

$$\prod_{i=1}^n (1 + \delta_i) = (1 + \delta_n)(1 + \theta_{n-1}) = 1 + \theta_n,$$

da cui, per l'ipotesi di induzione, si ha:

$$1 + \theta_n = (1 + \delta_n)(1 + \theta_{n-1}) = 1 + \delta_n + \theta_{n-1} + \delta_n \theta_{n-1}$$

Dalla precedente espressione si ricava $\theta_n = \delta_n + (1 + \delta_n)\theta_{n-1}$ e quindi:

$$|\theta_n| \leq u + (1 + u) \frac{(n-1)u}{1 - (n-1)u} = \frac{nu}{1 - (n-1)u} \leq \frac{nu}{1 - nu}$$

Per $\rho_n = -1$, si trova, analogamente, che

$$|\theta_n| \leq \frac{nu - (n-1)u^2}{1 - nu + (n-1)u^2} \leq \frac{nu}{1 - nu},$$

cioè la tesi. ■

a precisione infinita a partire dai pesi perturbati \widetilde{A}_i :

$$Q^*[A, f] = Q[\widetilde{A}, f] \quad \text{con} \quad \frac{|A_i - \widetilde{A}_i|}{|A_i|} \leq |\theta_i| \leq \frac{nu}{1 - nu} \quad i = 1, \dots, n$$

oppure come la formula di quadratura ottenuta in un sistema aritmetico a precisione infinita a partire dalle valutazioni di funzione perturbate $\widetilde{f}(x_i)$:

$$Q^*[A, f] = Q[A, \widetilde{f}] \quad \text{con} \quad \frac{|f(x_i) - \widetilde{f}(x_i)|}{|f(x_i)|} \leq |\theta_i| \leq \frac{nu}{1 - nu} \quad i = 1, \dots, n$$

Dal Teorema 4.12, facendo uso della (4.61), si può quindi affermare che l'errore relativo di round-off nel calcolo di una formula di quadratura è:

$$\frac{|Q^*[A, f] - Q[A, f]|}{|Q[A, f]|} \leq \mu(Q[A, f]) \left(\frac{nu}{1 - nu} \right)^2$$

Si noti infine che le formule di quadratura con pesi uguali del tipo:

$$Q[f] = A \sum_{i=1}^n f(x_i)$$

riducono ulteriormente l'errore di round-off, in quanto gli errori σ_i nella (4.81) risultano nulli.

4.7 Proprietà notevoli della formula trapezoidale composta

Si vuole approfondire un risultato relativo alla convergenza della formula trapezoidale composta per funzioni integrande sviluppabili in serie di Fourier nell'intervallo $[0, 1]$. Questo è il caso, ad esempio, di funzioni di classe almeno $C^1[0, 1]$ tali che $f(0) = f(1)$.

♣ **Esempio 4.26.** Si calcoli l'errore della formula trapezoidale composta relativamente all'integrale:

$$\int_0^1 (1 + .5 \sin(2\pi x))^{-1} dx = 1.154700534\dots$$

utilizzando un calcolatore con aritmetica standard IEEE. Si noti che la funzione integranda è una funzione periodica nell'intervallo $[0, 1]$. Nella Tabella 4.11 è mostrato l'andamento dell'errore $E_m[f]$ al crescere del numero di sottointervalli m .

L'esempio mostra che al crescere del numero di nodi la velocità di convergenza è molto maggiore di quella prevista, che come è noto, è dell'ordine $\mathcal{O}(h^2)$. Sette cifre decimali di accuratezza sono ottenute già con $m = 16$. Per valori più grandi il risultato non migliora solamente a causa della precisione finita del sistema aritmetico utilizzato. ♣

m	$E_m[f]$
2	0.1547005177×10^0
4	$0.1196610928 \times 10^{-01}$
8	$0.6139278412 \times 10^{-04}$
16	$0.1192092896 \times 10^{-06}$
32	$0.1192092896 \times 10^{-06}$
64	$0.1192092896 \times 10^{-06}$

Tabella 4.11: Errore della formula trapezoidale con una funzione periodica

In tal caso la formula trapezoidale composta su m intervalli di ampiezza $h = 1/m$:

$$[t_{j-1}, t_j], \quad t_j = jh, \quad j = 1, \dots, m, \quad t_0 = 0$$

risulta:

$$T_m[f] = \frac{1}{m} \sum_{j=1}^m f\left(\frac{j}{m}\right) \tag{4.84}$$

Si consideri ora lo sviluppo in serie di Fourier della funzione $f(t)$:

$$f(t) = \sum_{r=-\infty}^{\infty} a_r(f) e^{2\pi i r t}; \tag{4.85}$$

dove con $a_r(f)$ sono stati indicati i coefficienti di Fourier

$$a_r(f) = \int_0^1 f(t) e^{-2\pi i r t} dt. \tag{4.86}$$

Si nota che $a_0(f) = I[f]$. Sostituendo la (4.85) nella (4.84) si ottiene lo sviluppo:

$$T_m[f] = \frac{1}{m} \sum_{j=1}^m \sum_{r=-\infty}^{\infty} a_r(f) e^{2\pi i r j/m} = I[f] + \sum_{\substack{r=-\infty \\ r \neq 0}}^{\infty} a_r(f) T_m[e^{2\pi i r x}],$$

da cui si ricava:

$$E_m[f] = T_m[f] - I[f] = \sum_{\substack{r=-\infty \\ r \neq 0}}^{\infty} a_r(f) T_m[e^{2\pi i r x}]. \tag{4.87}$$

La formula precedente (detta formula della sommatoria di Poisson) esprime l'errore $E_m[f] = T_m[f] - I[f]$ come una serie coinvolgente i coefficienti di Fourier della funzione integranda. L'utilità di questo risultato sta nel fatto che esso consente di legare l'andamento asintotico dell'errore a quello dei coefficienti di Fourier della funzione f .

♣ **Esempio 4.27.** Calcolare l'errore della formula trapezoidale composta nel caso in cui $m = 3$. Dalla (4.87) si ha allora che

$$T_3[e^{2\pi i 1x}] = \frac{1}{3} \sum_{j=1}^3 e^{2\pi i j/3} = \frac{1}{3} \sum_{j=1}^3 \cos(2\pi \frac{j}{3}) + i \sin(2\pi \frac{j}{3}) = 0$$

$$T_3[e^{2\pi i 2x}] = \frac{1}{3} \sum_{j=1}^3 e^{4\pi i j/3} = \frac{1}{3} \sum_{j=1}^3 \cos(4\pi \frac{j}{3}) + i \sin(4\pi \frac{j}{3}) = 0$$

$$T_3[e^{2\pi i 3x}] = \frac{1}{3} \sum_{j=1}^3 e^{6\pi i j/3} = \frac{1}{3} \sum_{j=1}^3 \cos(6\pi \frac{j}{3}) + i \sin(6\pi \frac{j}{3}) = 1$$

Da cui si ha che:

$$E_3[f] = a_{\pm 3}(f) + a_{\pm 6}(f) + a_{\pm 9}(f) + \dots$$

♣

In generale, per un generico m risulta

$$T_m[e^{2\pi i r x}] = \begin{cases} 1 & \text{se } r/m \text{ intero} \\ 0 & \text{se } r/m \text{ non intero} \end{cases}$$

e quindi:

$$E_m[f] = T_m[f] - I[f] = \sum_{\substack{k=-\infty \\ k \neq 0}}^{\infty} a_{km}(f). \quad (4.88)$$

La sommatoria di Poisson si riduce quindi solo alla somma dei coefficienti di Fourier i cui indici sono multipli di m . La formula trapezoidale composta è quindi particolarmente adatta per le funzioni i cui coefficienti di Fourier decadono rapidamente con r . La velocità con cui gli $a_r(f)$ decadono a 0 è strettamente legata al grado di regolarità della funzione integranda. Infatti vale il seguente:

Teorema 4.17. Se $f(x) \in C^p[0, 1]$ si ha che

$$a_r(f) = - \sum_{q=1}^p \frac{f^{(q-1)}(1) - f^{(q-1)}(0)}{(2\pi i r)^q} + \frac{1}{(2\pi i r)^p} \int_0^1 f^{(p)}(t) e^{-2\pi i r t} dt.$$

Dimostrazione Integrando ripetutamente per parti il secondo membro della (4.86) si ha:

$$\begin{aligned} a_r(f) &= - \frac{f(1) - f(0)}{2\pi i r} + \frac{1}{2\pi i r} \int_0^1 f'(t) e^{-2\pi i r t} dt = \\ &= - \frac{f(1) - f(0)}{2\pi i r} - \frac{f'(1) - f'(0)}{(2\pi i r)^2} + \frac{1}{(2\pi i r)^2} \int_0^1 f''(t) e^{-2\pi i r t} dt = \dots \end{aligned}$$

per cui la tesi si ottiene dopo p integrazioni per parti. ■

Se la funzione integranda è di classe $C^p[0, 1]$ con $p > 1$ e periodica con le sue $p - 1$ derivate, i moduli dei coefficienti di Fourier soddisfano la maggiorazione:

$$|a_r(f)| \leq \frac{C}{r^p},$$

con C costante che dipende da $f^{(p)}(t)$ e, dalla formula (4.88), si ha

$$\lim_{m \rightarrow \infty} |T_m[f] - I[f]| = 0$$

e poiché il termine più significativo è $a_m(f)$ si può affermare che

$$E_m[f] \simeq \mathcal{O}(m^{-p}) = \mathcal{O}(h^p), \quad (4.89)$$

e che l'ordine di convergenza è p .

La funzione integranda dell'esempio precedente è periodica ed è di classe $C^\infty[0, 1]$, per cui l'ordine di convergenza della formula trapezoidale è, in questo caso, superiore a $\mathcal{O}(h^p)$ per tutti gli interi p .

In generale la funzione integranda non presenta le caratteristiche di regolarità e di periodicità necessarie per ottenere la rapida convergenza della formula trapezoidale composta. In tal caso si può cercare di modificare opportunamente tale funzione mediante una trasformazione del tipo:

$$x = \psi(t)$$

facendo in modo che abbia le caratteristiche richieste. Per scelte opportune di $\psi(t)$, si ha:

$$\int_0^1 f(x) dx = \int_0^1 f(\psi(t))\psi'(t) dt = \int_0^1 g(t) dt$$

con $g(t)$ soddisfacente le proprietà richieste. Una possibile scelta per $x = \psi(t)$ è, ad esempio, $\psi(t) = 3t^2 - 2t^3$. Tale funzione ha come derivata $\psi'(t) = 6t(1 - t)$ e quindi la funzione $g(t)$ risulta periodica nell'intervallo $[0, 1]$, poiché $\psi'(0) = \psi'(1) = 0$.

A questo punto è utile osservare che, se la funzione è sufficientemente regolare, dal Teorema 4.17 si ha che l'errore $E[f] = T_m[f] - I[f]$ è dato dalla somma delle quantità:

$$\begin{aligned} a_m(f) &= -\frac{f'(1) - f'(0)}{(2\pi im)^2} - \frac{f''(1) - f''(0)}{(2\pi im)^3} - \frac{f'''(1) - f'''(0)}{(2\pi im)^4} - \dots \\ a_{-m}(f) &= -\frac{f'(1) - f'(0)}{(2\pi im)^2} + \frac{f''(1) - f''(0)}{(2\pi im)^3} - \frac{f'''(1) - f'''(0)}{(2\pi im)^4} - \dots \\ a_{2m}(f) &= -\frac{f'(1) - f'(0)}{2^2(2\pi im)^2} - \frac{f''(1) - f''(0)}{2^3(2\pi im)^3} - \frac{f'''(1) - f'''(0)}{2^4(2\pi im)^4} - \dots \\ a_{-2m}(f) &= -\frac{f'(1) - f'(0)}{2^2(2\pi im)^2} + \frac{f''(1) - f''(0)}{2^3(2\pi im)^3} - \frac{f'''(1) - f'''(0)}{2^4(2\pi im)^4} - \dots \end{aligned}$$

da cui, sommando i termini relativi alle stesse derivate, si ha:

$$E[f] = \frac{f'(1) - f'(0)}{m^2(2\pi)^2} 2 \sum_{j=1}^{\infty} j^{-2} + 0 - \frac{f'''(1) - f'''(0)}{m^4(2\pi)^4} 2 \sum_{j=1}^{\infty} j^{-4} + 0 + \dots$$

Posto quindi:

$$B_2 = \frac{2!}{(2\pi)^2} 2 \sum_{j=1}^{\infty} j^{-2} \quad B_4 = \frac{4!}{(2\pi)^4} 2 \sum_{j=1}^{\infty} j^{-4} \quad B_{2k} = \frac{(2k)!}{(2\pi)^{2k}} 2 \sum_{j=1}^{\infty} j^{-2k} \quad (4.90)$$

si ottiene:

$$E[f] = T_m[f] - I[f] = \frac{B_2(f'(1) - f'(0))}{2!m^2} - \frac{B_4(f'''(1) - f'''(0))}{4!m^4} + \dots \quad (4.91)$$

I coefficienti B_{2k} della (4.90) sono detti coefficienti di Bernoulli e non dipendono ne' da m ne' dalla funzione integranda, mentre la (4.91) è nota come formula di Eulero-McLaurin. Ricordando che $h = 1/m$, dalla (4.91) si ottiene:

$$E[f] = T_m[f] - I[f] = \frac{B_2 h^2 (f'(1) - f'(0))}{2!} - \frac{B_4 h^4 (f'''(1) - f'''(0))}{4!} + \dots \quad (4.92)$$

Posto $C_{2j} = (-1)^{(j+1)} B_{2j} (f^{(2j-1)}(1) - f^{(2j-1)}(0)) / 2j!$ $j = 1, 2, \dots$, si ha, quindi:

$$E_m[f] = T_m[f] - I[f] = C_2 h^2 + C_4 h^4 + C_6 h^6 + \dots \quad (4.93)$$

da cui si ritrova la (4.89) nel caso in cui le derivate di f assumano lo stesso valore agli estremi dell'intervallo $[a, b]$. Si osservi che il primo termine della (4.93) è di ordine $\mathcal{O}(h^2)$ ovvero l'ordine di convergenza della formula trapezoidale composta è $p = 2$.

La formula di Eulero-McLaurin fornisce, dunque, un' espressione asintotica dell' errore di discretizzazione $E_m[f] = T_m[f] - I[f]$, in termini del passo di discretizzazione h . Tale formula rappresenta lo strumento matematico alla base di uno schema per la costruzione di una classe di formule di quadratura con ordine di convergenza che raddoppia ad ogni passo. A tal fine sussiste il risultato seguente:

Proposizione 4.5. *Sia $T_m[f]$ la formula trapezoidale composta su m sottointervalli. Posto $T_m^{(1)}[f] = (4T_{2m}[f] - T_m[f]) / 3$ si ha:*

$$T_m^{(1)}[f] - I[f] = -C_4 h^4 / 4 - 5C_6 h^6 / 16 + \dots$$

Dimostrazione Dalla (4.93) si ha:

$$\begin{aligned} T_m[f] - I[f] &= C_2 h^2 + C_4 h^4 + C_6 h^6 + \dots \\ T_{2m}[f] - I[f] &= C_2 (h/2)^2 + C_4 (h/2)^4 + C_6 (h/2)^6 + \dots \end{aligned}$$

da cui, moltiplicando ambo i membri della seconda espressione per 4 e sottraendo a quest'ultima la prima equazione, si ottiene:

$$4T_{2m}[f] - T_m[f] - 3I[f] = -3C_4 h^4 / 4 - 15C_6 h^6 / 16 + \dots$$

da cui, dividendo ambo i membri per 3 si ha:

$$\frac{4T_{2m}[f] - T_m[f]}{3} - I[f] = T_m^{(1)}[f] - I[f] = -C_4 h^4 / 4 - 5C_6 h^6 / 16 + \dots \quad (4.94)$$



La (4.94) definisce, in effetti, la formula $T_m^{(1)}[f]$ come una formula di quadratura il cui errore di discretizzazione è dato dall'ultimo membro della (4.94). Dalla (4.94) si osserva, inoltre, che $T_m^{(1)}[f]$ ha ordine di convergenza $p = 4$, cioè il doppio rispetto a $T_m[f]$. In altre parole, a partire da $T_m[f]$ abbiamo ricavato un'approssimazione di $I[f]$ che, utilizzando approssimazioni di $I[f]$ meno accurate, fornisce una formula di quadratura molto più accurata. Tale tecnica è nota, in generale, come *procedimento di estrapolazione di Romberg*. Il procedimento di estrapolazione di Romberg rientra tra gli schemi di "accelerazione della convergenza" di una successione²⁸. Applicando il procedimento di estrapolazione di Romberg alla coppia di formule $T_{2m}[f]$ e $T_{4m}[f]$, si ottiene:

$$T_{2m}^{(1)}[f] - I[f] = -C_4 h^4 / 2^6 - 5C_6 h^6 / 2^{10} + \dots = \mathcal{O}(h^4).$$

Così proseguendo si costruisce la successione $\{T_m^{(1)}[f]\}_{m=2^k}$ $k = 1, 2, \dots$ che converge ad $I[f]$ con ordine $p = 4$. Lo stesso procedimento può essere applicato alla sequenza

$$T_m^{(1)}[f], T_{2m}^{(1)}[f], T_{4m}^{(1)}[f], T_{8m}^{(1)}[f], \dots$$

In particolare:

Proposizione 4.6.

$$\begin{aligned} T_m^{(1)}[f] - I[f] &= D_4 h^4 + D_6 h^6 + \dots \\ T_{2m}^{(1)}[f] - I[f] &= D_4 (h/2)^4 + D_6 (h/2)^6 + \dots \end{aligned}$$

avendo definito in maniera opportuna i coefficienti D_i .

Dimostrazione Moltiplicando ambo i membri della seconda espressione per 16 e sottraendo a quest'ultima la prima equazione, si ottiene:

$$16T_{2m}^{(1)}[f] - T_m^{(1)}[f] - 15I[f] = 3D_6 h^6 / 4 + \dots$$

da cui, dividendo ambo i membri per 15 si ha:

$$\frac{16T_{2m}^{(1)}[f] - T_m^{(1)}[f]}{15} - I[f] = T_m^{(2)}[f] - I[f] = 1D_6 h^6 / 20 + \dots \tag{4.95}$$

ottenendo quindi una formula con ordine di convergenza $p = 6$. ■

In definitiva si ottiene la tabella seguente.

²⁸In generale, assegnata una successione $\{s_n\}_{n \in \mathbb{N}}$ convergente a s , uno schema di accelerazione della convergenza di $\{s_n\}_{n \in \mathbb{N}}$ è un qualsiasi procedimento che, a partire da $\{s_n\}_{n \in \mathbb{N}}$ costruisce una successione $\{s'_n\}_{n \in \mathbb{N}}$ tale che $\{s'_n\}_{n \in \mathbb{N}}$ converga a s e $\lim_{n \rightarrow \infty} \frac{s'_n}{s_n} = 0$.

$T_1[f]$			
$T_2[f]$	$T_1^{(1)}[f]$		
$T_4[f]$	$T_2^{(1)}[f]$	$T_1^{(2)}[f]$	
$T_8[f]$	$T_4^{(1)}[f]$	$T_2^{(2)}[f]$	$T_1^{(3)}[f]$
\vdots			\ddots

Tabella 4.12: Procedimento di estrapolazione di Romberg

Nella Tabella 4.12 la prima colonna ha ordine di convergenza $p = 2$, la seconda colonna ha ordine di convergenza $p = 4$, la terza $p = 6$ e così via, a patto, comunque, che la funzione sia sufficientemente regolare in quanto i coefficienti C_{2j} sono definiti a partire dalle derivate. Osserviamo, infine, che tali formule, al pari della formula trapezoidale composta da cui derivano, non sono di tipo interpolatorio, nel senso che, in generale, non sono esatte per la classe dei polinomi.

4.8 La quadratura multidimensionale

Il problema della quadratura multidimensionale è lo sviluppo e l'analisi di metodi numerici per il calcolo di:

$$I[f] = \int \dots \int_{\Omega} f(x_1, \dots, x_d) dx_1 \dots dx_d, \quad (4.96)$$

dove $\Omega \subseteq \mathbb{R}^d$ con $d > 1$ è una regione dello spazio euclideo d -dimensionale e $f(x_1, \dots, x_d)$ è una funzione integrabile in Ω . Analogamente al caso monodimensionale ($d = 1$) le formule di quadratura utilizzate (talvolta per $d > 1$ chiamate *formule di cubatura*) hanno espressione:

$$Q[f] = \sum_{i=1}^n A_i f(\underline{x}_i)$$

dove \underline{x}_i è un punto dello spazio euclideo d -dimensionale, mentre l'errore di discretizzazione ha espressione:

$$E[f] = \int \dots \int_{\Omega} f(x_1, \dots, x_d) dx_1 \dots dx_d - \sum_{i=1}^n A_i f(\underline{x}_i)$$

Benché molte di idee base della quadratura multidimensionale siano dirette estensioni di quelle in una dimensione, essa pone problemi sia computazionali sia teorici diversi, di non sempre risoluzione semplice.

Uno di questi è legato alle caratteristiche dei domini di integrazione Ω su cui è possibile definire $I[f]$. Infatti mentre per il caso monodimensionale è possibile definire solo tre tipi di domini (intervalli chiusi e limitati, intervalli limitati solo superiormente o

inferiormente ed intervalli illimitati), in più dimensioni è possibile integrare su sottospazi di \mathbb{R}^d come un ipercubo, una sfera, un simpleso o su una qualunque varietà algebrica come curve e superfici. La strategia che viene seguita consiste nello sviluppare formule per i domini più semplici e, quando possibile, effettuare per regioni diverse opportune trasformazioni per ricondurle a domini per cui le formule sono note.

Un altro problema caratteristico della quadratura multidimensionale è il numero di nodi necessario per ottenere l'accuratezza richiesta. Tale numero, infatti, in alcuni casi cresce esponenzialmente con il numero di dimensioni e ciò rende la scelta della formula di quadratura da utilizzare particolarmente delicata, poiché, per dimensioni elevate, l'algoritmo di quadratura può risultare impraticabile essendo, la sua complessità, non polinomiale. In questi casi, inoltre, l'errore di round-off può avere influenze notevoli sul risultato.

In questo paragrafo si presentano alcuni esempi significativi di formule di quadratura multidimensionale nel caso in cui $\Omega = [-1, 1]^d$.

4.8.1 Formule prodotto

Uno dei metodi più semplici ed immediati per la costruzione di formule per la quadratura multidimensionale è quello di comporre una o più formule di quadratura monodimensionale mediante prodotto cartesiano. Se ad esempio $d = 2$, siano $R[f]$ e $S[f]$ due formule di quadratura in una dimensione:

$$R[f] = \sum_{i=1}^{n_1} A_i f(x_i), \quad S[f] = \sum_{j=1}^{n_2} B_j f(y_j).$$

Definizione 4.17. (Prodotto cartesiano)

Se si indicano con $X = \{x_1, \dots, x_{n_1}\}$ e $Y = \{y_1, \dots, y_{n_2}\}$ rispettivamente gli insiemi dei nodi delle due formule di quadratura, si definisce **prodotto cartesiano** di X e Y l'insieme delle coppie:

$$X \times Y = \{(x_i, y_j), i = 1, \dots, n_1, j = 1, \dots, n_2\}.$$

Definizione 4.18. (Formula prodotto)

Si definisce **formula prodotto** la formula di quadratura definita sui nodi di $X \times Y$:

$$(R \times S)[f] = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} A_i B_j f(x_i, y_j).$$

Tale formula richiede $n_1 \cdot n_2$ valutazioni della funzione integranda.

Se le due formule di quadratura $R[f]$ e $S[f]$ hanno grado di precisione algebrico rispettivamente p_1 e p_2 allora si ha:

$$E[f] = \int_{-1}^1 \int_{-1}^1 x^r y^s dx dy - \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} A_i B_j x_i^r y_j^s =$$

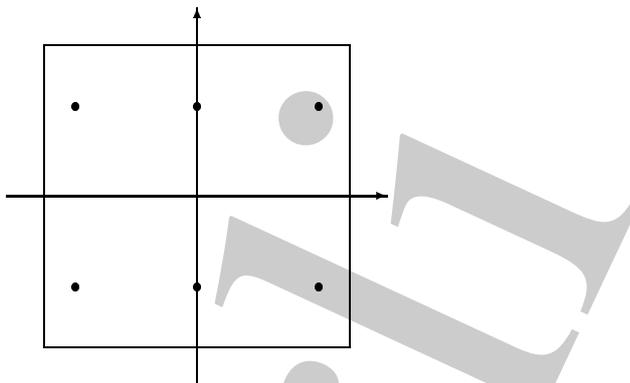


Figura 4.20: nodi della formula prodotto di Gauss con sei nodi nella regione $[-1, 1] \times [-1, 1]$

$$\int_{-1}^1 x^r dx \int_{-1}^1 y^s dy - \sum_{i=1}^{n_1} A_i x_i^r \sum_{j=1}^{n_2} B_j y_j^s = 0$$

$\forall r \leq p_1$ e $\forall s \leq p_2$. Si ha quindi che la formula prodotto è esatta per tutti i monomi $x^r y^s$ con $r \leq p_1$ e $s \leq p_2$ cioè essa ha grado di precisione algebrico $p = \min(p_1, p_2)$ anche se avrà errore nullo per qualche monomio di grado $p_1 + p_2$ ²⁹. In Figura 4.20 è mostrata la disposizione dei nodi nel quadrato unitario $[-1, 1] \times [-1, 1]$ nel caso in cui $R[f]$ è una formula di Gauss con tre nodi e $S[f]$ è una formula di Gauss con due nodi.

Il metodo esposto è facilmente generalizzabile ad un numero di dimensioni qualunque, per cui le formule prodotto sono molto utilizzate. Esse hanno però lo svantaggio che il numero di nodi cresce esponenzialmente con la dimensione d . Infatti, ad esempio, se si utilizza una formula in $d = 20$ dimensioni, in ognuna delle quali si utilizzano $n = 2$ nodi, si hanno complessivamente $2^{20} \simeq 10^6$ valutazioni di funzione della funzione integranda. Per tale motivo le formule prodotto sono generalmente basate su formule particolarmente precise (ad esempio le formule di Gauss) ed utilizzate per un numero di dimensioni molto limitato ($d \leq 5$). Ad esempio la routine automatica D01AJF della libreria NAG basata sulla formula trapezoidale prodotto è in grado di integrare funzioni fino ad un massimo di quattro variabili.

4.8.2 Formule monomiali

Un approccio diverso per lo sviluppo di formule per la quadratura multidimensionale è quello di costruire formule esatte per un insieme di monomi di grado fissato, cioè tale

²⁹La definizione di grado di precisione algebrico per una formula di quadratura multidimensionale è del tutto analoga al caso monodimensionale: si dice che $Q[f]$ ha grado di precisione p se è esatta per i monomi algebrici multidimensionali di grado al più p .

che

$$E[f] = 0 \quad \forall f(x_1, \dots, x_d) = x_1^{i_1} \cdots x_d^{i_d} \in \Pi_p, \quad i_1 + \dots + i_d = p$$

dove Π_p è l'insieme di polinomi di grado al più p in d variabili. Le formule che si ottengono in questo modo sono dette **formule monomiali**; tra le più note vi è la famiglia di formule di Genz e Malik definita nel seguito [8]. Sia $P^{(m,d)}$ l'insieme di tutte le distinte d -partizioni degli interi $0, 1, \dots, m$ con componenti non crescenti, cioè

$$P^{(m,d)} = \{(p_1, \dots, p_d) : m \geq p_1 \geq \dots \geq p_d \geq 0, |\mathbf{p}| \leq m\},$$

dove $|\mathbf{p}| = \sum_{i=1}^d p_i$. Ad esempio per $m=3$ e $d=4$ si ha:

$$P^{(3,4)} = \{(0, 0, 0, 0), (1, 0, 0, 0), (2, 0, 0, 0), (1, 1, 0, 0), (3, 0, 0, 0), (2, 1, 0, 0), (1, 1, 1, 0)\}$$

Fissati quindi $m+1$ valori reali distinti $\lambda_0 = 0, \lambda_1, \dots, \lambda_m$ (generatori) si pone $\boldsymbol{\lambda}_p = \{\lambda_{p_1}, \dots, \lambda_{p_d}\}$ e si definisce *formula base pienamente simmetrica* la somma:

$$f[\boldsymbol{\lambda}_p] = \sum_{q \in \Pi_p} \sum_{\mathbf{s}} f(s_1 \lambda_{q_1}, \dots, s_d \lambda_{q_d}),$$

dove Π_p è l'insieme di tutte le permutazioni del generico $\mathbf{p} \in P^{(m,d)}$ e la somma interna è eseguita su tutte le combinazioni di segno delle componenti dei nodi che si ottengono per $s_i = \pm 1$. Tale somma è quindi effettuata mediante tutte le permutazioni e cambio di segno per i valori $\lambda_{q_i} \neq 0$. Per esempio con $d = 4, m = 2$ e $\mathbf{p} = (2, 0, 0, 0)$ si ha:

$$f[\boldsymbol{\lambda}_p] = f(\lambda_2, 0, 0, 0) + f(-\lambda_2, 0, 0, 0) + f(0, \lambda_2, 0, 0) + f(0, -\lambda_2, 0, 0) + f(0, 0, \lambda_2, 0) + f(0, 0, -\lambda_2, 0) + f(0, 0, 0, \lambda_2) + f(0, 0, 0, -\lambda_2).$$

Infine posto $\underline{\delta} = \{\delta, \dots, \delta\}$, con δ positivo, la formula di quadratura $R^{(m,d)}[f]$ è definita mediante:

$$R^{(m,d)}[f] = \sum_{p \in P^{(m-1,d)}} w_p^{(m,d)} f[\boldsymbol{\lambda}_p] + W^{(m,d)} f[\underline{\delta}].$$

Si può notare che i generatori λ_i sono indipendenti da m , per cui la formula $R^{(m,d)}$ utilizza tutti i nodi di $R^{(m-1,d)}$, che utilizza tutti i nodi di $R^{(m-2,d)}$ e così via. Tali formule costituiscono quindi una famiglia di formule innestate e per tale motivo esse sono tra le più utilizzate nella realizzazione di routine di quadratura multidimensionale efficienti. Il calcolo dei parametri che caratterizzano la formula richiede la risoluzione di sistemi di equazioni non lineari. Mediante un'opportuna scelta di W, λ_i e δ è possibile determinare i pesi w_p in modo che la formula $R^{(m,d)}$ abbia grado di precisione $2m + 1$ [8].

Come applicazione della teoria descritta consideriamo il problema dell'esistenza di valori per la sequenza di generatori $\{\lambda_i\}$ utilizzabili per il calcolo computazionale di integrali multipli, nel qual caso si desidera che siano $\delta, \lambda_i \in [0, 1]$. La scelta di δ determina

tutti i termini della sequenza $\{\lambda_i\}$ così che risulta necessario considerare l'effetto delle varie scelte di $\delta \in [0, 1]$. Significativi test computazionali indicano che non esiste scelta di $\delta^2 \in [0, 1]$ che restituisca tutti i $\lambda_1^2, \lambda_2^2, \dots, \lambda_6^2 \in [0, 1]$. Questo significa che per alcune applicazioni pratiche le formule $R^{(m,d)}$ risultano utili solo fino a $m = 6$ (avendo, in tal caso grado di precisione algebrico 13). Si riportano di seguito i valori dei parametri $\delta, \lambda_1, \lambda_2, \dots, \lambda_7$ con 15 cifre decimali, in corrispondenza di una scelta di δ^2 per la quale risultano: $\lambda_1, \lambda_2, \dots, \lambda_5 \in [0, 1]$.

$$\delta^2 = 0.4707000000000000,$$

$$\begin{aligned} \delta &= 0.686075797561756, & \lambda_1 &= 0.955907315804539, \\ \lambda_2 &= 0.406057174738240, & \lambda_3 &= 0.895254709252356, \\ \lambda_4 &= 0.847486391083961, & \lambda_5 &= 0.878694113728702, \\ \lambda_6 &= 1.529203497225463, & \lambda_7 &= 1.949470987454603. \end{aligned}$$

Si può osservare che $\lambda_6, \lambda_7 \in [1, 2]$ così che le formule con grado di precisione al più 15 possono essere utilizzate anche nelle applicazioni in cui la funzione integranda risulta definita fuori dalla regione di integrazione. Dal punto di vista pratico tali formule sono utilizzate per $6 \leq d \leq 20$ e riscontrano una naturale applicazione in algoritmi adattativi che al momento trovano implementazione nella classe più ampia di routine disponibili per la quadratura multidimensionale. Tra queste si ricordano la routine DCUHRE della collezione ACM *Transactions on Mathematical Software* e le routine D01EAF e D01FCF della libreria del NAg.

4.8.3 Metodi Monte Carlo

Quando la dimensione d aumenta (si può stimare $d \geq 20$) i metodi precedentemente descritti sono computazionalmente impraticabili e si fa ricorso a tecniche di tipo numerico-statistico. Ad esempio, nel dominio $B = [0, 1]^d$ l'integrale:

$$I[f] = \int_B f(t) dt = \int_{R^d} c_B(t) f(t) dt = \mu(f)$$

viene interpretato come il valore atteso $\mu(f)$ della funzione f , vista come variabile casuale, con densità di probabilità $c_B(t)$, dove $c_B(t)$ è la funzione caratteristica della regione B . Per simulare l'andamento della variabile casuale f , si scelgono gli n nodi in maniera casuale su $[0, 1]^d$. In tal modo si hanno i cosiddetti *metodi Monte Carlo* per la quadratura multidimensionale:

$$R[f] = \frac{1}{n} \sum_{i=1}^n f(x_i) \tag{4.97}$$

Per la legge forte dei grandi numeri, se le n variabili casuali sono uniformemente distribuite si ha che

$$\mathcal{P} \left(\lim_{n \rightarrow \infty} \frac{1}{n} \sum_{k=1}^n f(x_k) = I[f] \right) = 1.$$

dove con $\mathcal{P}(\mathcal{E})$ si è indicata la probabilità dell'evento \mathcal{E} . Si ha quindi che la probabilità che la (4.97) sia una buona approssimazione di $I[f]$ può essere resa arbitrariamente prossima a 1 se si sceglie n sufficientemente grande. Inoltre se esiste anche $I[f^2]$ si ha, per il teorema centrale del limite, che

$$\mathcal{P} \left(|R[f] - I[f]| \leq \frac{\lambda\sigma}{\sqrt{n}} \right) = \frac{1}{\sqrt{2\pi}} \int_{-\lambda}^{\lambda} e^{-x^2/2} dx + \mathcal{O} \left(\frac{1}{\sqrt{n}} \right).$$

dove $\sigma = [I[f^2] - (I[f])^2]^{1/2}$ è la deviazione standard.

In genere per ogni fissato λ tale espressione è presa come una misura dell'errore $E[f]$ prodotto da un metodo Monte Carlo. Per $\lambda \geq 2$ si ha che il primo addendo al secondo membro è circa uguale a 1, per cui è possibile porre

$$|E[f]| = |I[f] - R[f]| \simeq \mathcal{O}(n^{-1/2})$$

Questo errore è tipico dei metodi Monte Carlo ed ha lo svantaggio di decrescere lentamente al crescere di n . Un altro aspetto negativo dei metodi Monte Carlo è che sequenze realmente casuali non sono computazionalmente disponibili.

Osservazione 4.3. *Nello sviluppo di software basato sui metodi Monte Carlo, piuttosto che campioni reali, per la determinazione dei nodi si utilizzano sequenze di numeri, così dette pseudo-random, o pseudo casuali, calcolate, deterministicamente, mediante opportuni algoritmi chiamati generatori di numeri casuali. In tal caso si parla di formule pseudo-random o, anche, di metodi Quasi Monte Carlo.*

Si osserva, in particolare, che i generatori di numeri random sono basati su calcoli deterministici, ovvero completamente prevedibili, per cui le sequenze restituite in output non possono essere considerate di natura stocastica; per questa ragione i numeri generati in tal modo sono definiti pseudo-random. L'idea alla base di un metodo Quasi Monte Carlo è molto semplice: sostituire le successioni di numeri random del tradizionale metodo Monte Carlo, che mostrano una tendenza generale ad accumularsi in una certa regione dello spazio, con sequenze deterministiche di punti generati in maniera tale da assicurare un certo grado di uniformità o, meglio, un certo livello di discrepanza.

È da notare che l'errore prodotto da un metodo Monte Carlo, è indipendente dalla dimensione e dalla regolarità della funzione integranda. Per tale motivo i metodi Monte Carlo sono particolarmente utili nel caso di integrali in un gran numero di dimensioni o di funzioni integrande estremamente irregolari. Un esempio di routine basata su un metodo Monte Carlo è la routine D01GBF della libreria del NAG.

4.8.4 Le Lattice Rule

Le Lattice Rule possono essere considerate una particolare generalizzazione della formula trapezoidale composta nello spazio a più dimensioni. Esse hanno un' elevata velocità di convergenza nel caso in cui la funzione integranda sia sufficientemente regolare e periodica in ogni sua componente nel cubo unitario $\Omega = [0, 1]^d$ (vedi paragrafo 4.7). Inoltre, come la formula trapezoidale composta, utilizzano come nodi di integrazione un insieme estremamente regolare di punti, le cosiddette *Lattice* di integrazione:

Definizione 4.19. (Lattice)

Si definisce **Lattice** di integrazione nello spazio a d dimensioni un sottoinsieme discreto dello spazio euclideo \mathfrak{R}^d , che sia chiuso rispetto all'addizione (cioè tale che la somma di due elementi della lattice è ancora un elemento della lattice) e che contenga al suo interno l'insieme dei vettori d -dimensionali a componenti intere $Z^d = \{(z_1, \dots, z_d) : z_i \in Z\}$.

Definizione 4.20. (Lattice Rule)

Si definisce **Lattice Rule** una formula di quadratura del tipo (4.97) che utilizzi come nodi i punti di una Lattice che appartengono al cubo semiaperto $[0, 1]^d$.

♣ **Esempio 4.28.** Nello spazio a $d = 2$ dimensioni la formula:

$$R[f] = \frac{1}{5} \left[f(0, 0) + f\left(\frac{1}{5}, \frac{2}{5}\right) + f\left(\frac{2}{5}, \frac{4}{5}\right) + f\left(\frac{3}{5}, \frac{1}{5}\right) + f\left(\frac{4}{5}, \frac{3}{5}\right) \right]$$

è una Lattice Rule. Si noti infatti che l'insieme dei punti:

$$(x, y) = \alpha \left(\frac{1}{5}, \frac{2}{5}\right) + \beta \left(\frac{3}{5}, \frac{1}{5}\right) \quad \alpha, \beta \text{ interi}$$

è un insieme chiuso rispetto all'addizione ed inoltre contiene al suo interno Z^2 (Figura 4.21). I punti $(\frac{1}{5}, \frac{2}{5})$ e $(\frac{3}{5}, \frac{1}{5})$ sono chiamati **generatori** della Lattice Rule.

Nella figura sono rappresentati i nodi della Lattice ed è messo in evidenza il quadrato unitario $[0, 1]^2$. ♣

Un altro legame delle Lattice Rule con la formula trapezoidale composta consiste nel fatto che è possibile esprimere l'errore in termini dei coefficienti di Fourier della funzione integranda (per la trapezoidale composta si veda il paragrafo 4.7). Se si suppone infatti che la funzione $f(\underline{x}) = f(x_1, \dots, x_d)$ sia sviluppabile in serie di Fourier, cioè:

$$f(\underline{x}) = \sum_{\underline{h} \in Z^d} a_{\underline{h}}(f) e^{2\pi i \underline{h} \cdot \underline{x}} \quad (4.98)$$

dove $\underline{h} \cdot \underline{x}$ è l'usuale prodotto scalare in \mathfrak{R}^d e gli $a_{\underline{h}}(f)$ sono i coefficienti di Fourier della funzione f , applicando la formula (4.97) alla (4.98) si ha:

$$R[f] = \sum_{\underline{h} \in Z^d} a_{\underline{h}}(f) R[e^{2\pi i \underline{h} \cdot \underline{x}}]$$

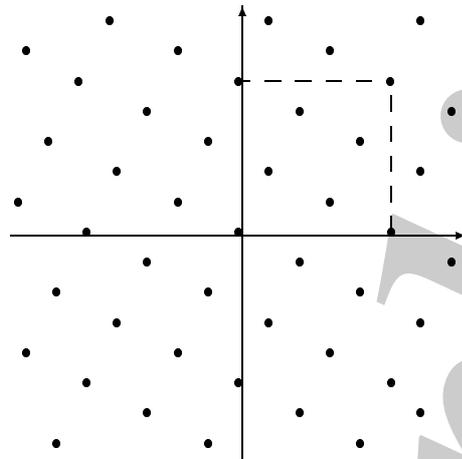


Figura 4.21: Esempio di Lattice in $d = 2$ dimensioni generata da $(\frac{1}{5}, \frac{2}{5})$ e $(\frac{3}{5}, \frac{1}{5})$

Poiché è possibile verificare, analogamente a come mostrato nel paragrafo 4.7, che

$$R[e^{2\pi i \underline{h} \cdot \underline{x}}] = \begin{cases} 1 & \text{se } \underline{h} \cdot \underline{x} \in \mathbb{Z} \\ 0 & \text{altrimenti} \end{cases}$$

si ha che l'errore $E[f]$ è:

$$E[f] = I[f] - R[f] = \sum_{\underline{h} \cdot \underline{x} \in \mathbb{Z}, \underline{h} \neq 0} a_{\underline{h}}(f)$$

Si noti infine che la formula trapezoidale composta prodotto è una particolare Lattice Rule, in quanto, ad esempio, la formula prodotto $(T_m \times T_m)[f]$ può essere generata dai nodi $(0, \frac{1}{m})$ e $(\frac{1}{m}, 0)$. Il recente sviluppo della teoria sulle Lattice Rule, giustifica la mancanza di routine basata su tali metodi. Esistono solo alcune implementazioni riportate sulla rivista specializzata di software matematico ACM *Transactions on Mathematical Software*.

4.9 Software matematico disponibile per la quadratura

Nella costruzione di software numerico per la quadratura sono molti gli obiettivi da realizzare. I principali sono l'abilità di rilevare e ridurre l'errore di round-off, fornire un risultato accurato, la facilità di uso e l'efficienza computazionale. Tali obiettivi sono spesso incompatibili tra loro, per cui è necessario sviluppare routine per classi di problemi al fine di ottenere un buon livello di efficienza ed accuratezza.

La disponibilità di routine di buona qualità per la quadratura è comunque molto ampia. Un gran numero di queste è raccolto nelle principali librerie di software matematico commerciale come le librerie del NAG e dell'IMSL, nonché nel package specifico per la quadratura denominato QUADPACK. Inoltre è disponibile una grande varietà di software di pubblico dominio reperibile per via elettronica. Quasi tutte le routine di queste librerie sono basate su schemi adattativi.

La libreria NAG (Mark 17 del 1996) del Numerical Algorithm Group di Oxford è sicuramente la fonte più ampia di software per il calcolo degli integrali. Sono presenti infatti circa 50 routine nel capitolo D01 dedicato alla quadratura per vari tipi di problemi, più varie routine ausiliarie che calcolano i pesi ed i nodi di particolari formule di Gauss. In particolare, oltre a numerose routine di tipo generale, sono disponibili routine per funzioni integrande particolari quali, ad esempio, quelle oscillanti o dotate di singolarità, etc. Esistono anche routine per l'integrazione di funzioni note su un insieme discreto di punti e per integrali definiti su intervalli non limitati. Sono inoltre disponibili delle routine per la quadratura multidimensionale fino ad un numero di dimensioni $d = 20$ su domini di tipo ipercubo, ipersfera e simpleso. Per buona parte delle routine esistono versioni in FORTRAN e C, in singola e doppia precisione. L'indirizzo World Wide Web (WWW) di Internet del NAG è : <http://www.nag.com>.

Nella libreria IMSL (versione 2 del 1993) della Visual Numeric, invece, sono disponibili circa 20 routine sviluppate in linguaggio FORTRAN e C per la quadratura, più alcune routine ausiliarie per il calcolo dei pesi e dei nodi delle formule di Gauss. In particolare sono presenti anche routine per problemi in cui la funzione integranda è dotata di singolarità e per funzioni oscillanti. Altre routine infine sono dedicate al calcolo di integrali multidimensionali su iperrettangoli. Informazioni sulla libreria IMSL sono reperibili all'indirizzo WWW di Internet <http://www.vni.com>.

Oltre alle librerie commerciali, esistono moltissime routine di dominio pubblico reperibili per via elettronica.

QUADPACK, ad esempio, è un package costituito da numerose routine specificamente progettate per il caso monodimensionale. Esso è stato sviluppato in linguaggio FORTRAN negli anni 80 da alcuni ricercatori dell'Università di Leuven, dell'Università di Vienna e del National Bureau of Standard americano. La maggior parte delle routine

sono basate su algoritmi adattativi che usano le formule di Gauss-Kronrod nonché particolari tecniche per l'accelerazione della convergenza. In particolare sono presenti routine per integrali con diverse funzioni peso, per funzioni con singolarità e per intervalli non limitati. Alcune routine di tale package sono presenti, con diverso nome, anche nella libreria del NAg. Le routine di QUADPACK sono reperibili all'indirizzo World Wide Web (WWW) di Internet: <http://www.netlib.org/quadpack>.

SLATEC è una collezione di routine FORTRAN portabili e di dominio pubblico sviluppata da ricercatori di diversi centri di ricerca americani tra cui il Lawrence Livermore National Laboratory, l'Oak Ridge National Laboratory ed il National Institute of Standard. La versione 4.1 del 1993 contiene oltre 1400 routine di cui circa 50 dedicate alla quadratura. Le routine di SLATEC sono reperibili all'indirizzo WWW <http://www.netlib.org/slatec>.

Alcune riviste, infine, sono specializzate nella pubblicazione di software portabile. La più autorevole in proposito è *Transactions on Mathematical Software (TOMS)* dell'Association of Computing Machinery (ACM). Nel corso degli anni sono state pubblicate oltre 800 routine scritte in FORTRAN, di cui circa 20 sono dedicate al calcolo degli integrali di vari tipi di problemi. L'indirizzo WWW di Internet per la reperibilità delle routine pubblicate dal TOMS è: <http://www.netlib.gov/toms>.

Analogamente il *Journal of Computational and Applied Mathematics* ha pubblicato numerose routine di software matematico di cui tre dedicate alla quadratura (due di tipo generale in una dimensione per integrali limitati e non limitati ed una per il calcolo di un integrale doppio definito su un triangolo).

4.9.1 Esempio d'uso

Di seguito si mostra un esempio di utilizzo di una routine per la quadratura. La routine in esame è D01AJF presente nella versione FORTRAN della libreria NAg. Tale routine calcola l'integrale di una funzione $f(x)$ nell'intervallo $[a, b]$. Le specifiche della routine sono le seguenti:

La routine D01AJF (peraltro presente anche in QUADPACK con il nome di QAGS) è una routine adattativa globale basata sulla coppia di formule di Gauss-Kronrod

$$(G^{(7)}, K^{(15)})$$

I parametri di input sono:

F funzione reale fornita dall'utente;

A, B estremi dell'intervallo di integrazione;

EPSABS, EPSREL tolleranze assoluta e relativa.

I parametri di output sono:

RESULT contiene l'approssimazione di $I[f]$;

ABSERR contiene una stima dell'errore $E[f]$;

IFAIL indicatore di errore.

La routine D01AJF utilizza inoltre le aree di lavoro

W array reale di dimensione LW;

LW intero;

IW array intero di dimensione LIW;

LIW intero.

Volendo utilizzare la routine D01AJF per calcolare il seguente integrale

$$I[f] = \int_0^{2\pi} \frac{x \sin(30x)}{\sqrt{1 - x^2/4\pi^2}} dx = -2.54326831\dots$$

con una tolleranza relativa $\varepsilon = 10^{-4}$, il programma chiamante può essere il seguente³⁰:

³⁰La routine X01AAF è disponibile nella libreria NAg per il calcolo di π . Si noti la variabile KOUNT utilizzata come contatore delle chiamate alla function FST

```
SUBROUTINE D01AJF(F, A, B, EPSABS, EPSREL, RESULT,
ABSERR, W, LW, IW, LIW, IFAIL)
INTEGER LW, IW(LIW), LIW, IFAIL
REAL F, A, B, EPSABS, EPSREL, RESULT, ABSERR, W(LW)
EXTERNAL F

C
INTEGER KOUNT
REAL PI
PARAMETER (LW=800, LIW=200)
EXTERNAL FST, X01AAF
COMMON /TELNUM/ PI, KOUNT

C
C inizializzazione variabili
C
PI=X01AAF(PI)
EPSABS=0.
EPSREL=1.0E-4

C
A=0.
B=2.*PI
KOUNT=0
IFAIL=-1

C
C chiamata di D01AJF
C
CALL D01AJF(FST,A,B,EPSABS,EPSREL,RESULT,ABSERR,W,LW,IW,LIW,IFAIL)

C
IF (IFAIL.EQ.0) THEN
WRITE(*,10) RESULT
WRITE(*,11) ABSERR
WRITE(*,12) KOUNT
ELSE
WRITE(*,20) ,IFAIL
ENDIF

C
STOP

C
```

Esempio di programma chiamante FORTRAN per la subroutine D01AJF - continua

```

C
10  FORMAT(' RISULTATO =',F9.5)
11  FORMAT(' ERRORE STIMATO =',E9.2)
12  FORMAT(' NO. VALUTAZ. FUNZIONE =',I4)
20  FORMAT(' ERRORE. IFAIL = ',I2)
C
      END
C
C
C
      REAL FUNCTION FST(X)
C
      INTEGER KOUNT
      REAL PI
      COMMON /TELNUM/ PI, KOUNT
C
      KOUNT=KOUNT+1
      FST=X*SIN(30.*X)/SQRT(1.-X**2/(4.*PI**2))
C
      RETURN
C
      END

```

Esempio di programma chiamante e function FORTRAN per la subroutine D01AJF - fine

Il risultato che si ottiene chiamando D01AJF con i dati forniti è:

RISULTATO	=-2.54326
ERRORE STIMATO	=0.13E-4
NO. VALUTAZ. FUNZIONE	=777

4.10 Computer Problems

Esercizio 1 Sviluppare una subroutine Fortran TRAP1 per il calcolo dell'area sottesa da una funzione data $f(x)$ nell'intervallo $[a, b]$, basata sulla formula trapezoidale composita su m sottointervalli, i cui parametri di input siano gli estremi dell'intervallo a e b , la funzione $f(x)$ ed il numero di sottointervalli m ed il cui parametro di output sia il valore dell'area T .

Dato un cerchio di raggio $r = 1$, si utilizzi la subroutine per il calcolo dell'area della calotta circolare delimitata dalla circonferenza e da un lato del quadrato inscritto del cerchio. Sapendo che l'area della calotta è:

$$Area = \frac{\pi - 2}{4}$$

si utilizzino di seguito $m = 1, 2, 4, 8, \dots, 1024$ sottointervalli e si osservi come si riduce l'errore ad ogni passo.

Esercizio 2 Data una parabola intersecante l'asse delle ascisse nei punti $P_1 = (0, 0)$ e $P_2 = (4, 0)$ ed avente come vertice il punto $P_3 = (2, 8)$, ripetere l'esercizio 1 calcolando l'area racchiusa tra la parabola e l'asse delle ascisse. Si ricordi, che Archimede (287 a.c.) ha dimostrato che l'area di un segmento parabolico è uguale a $4/3$ dell'area di un triangolo avente la stessa base e la stessa altezza del segmento parabolico.

Esercizio 3 Si ripetano gli esercizi 1 e 2 sostituendo alla formula trapezoidale la formula rettangolare e la formula del punto medio. Si commentino i risultati.

Esercizio 4 Si determini una stima calcolabile dell'errore relativo e si modifichi la subroutine TRAP1 dell'esercizio 1 in una subroutine TRAP2 in modo tale da sostituire il parametro di input relativo al numero di sottointervalli m con un parametro di input tol che rappresenti la tolleranza sull'errore relativo commesso ed aggiungendo i parametri di output E , che rappresenta una stima dell'errore relativo commesso, e V , che rappresenta il numero di valutazioni di funzioni effettuate.

Si calcoli quindi l'area della calotta circolare dell'esercizio 1 e l'area del segmento parabolico dell'esercizio 2 con tolleranze 0.001, 0.0001 e col la massima accuratezza del sistema aritmetico utilizzato, specificando il numero di valutazioni della funzione integranda.

Esercizio 5 Mediante la subroutine TRAP2 sviluppata nell'esercizio 4, si calcoli l'area

al di sotto delle funzioni di equazione

$$\begin{array}{lll}
 i) & y = e^x & x \in [0, 1] & (\text{val. esatto} = 1.718281\dots) \\
 ii) & y = x^2 e^{-x^2} & x \in [0, 4] & (\text{val. esatto} = 0.4431132\dots) \\
 iii) & y = \log(x) & x \in [1, 10] & (\text{val. esatto} = 14.02585\dots) \\
 iv) & y = \sin(x^2) & x \in [0, 5] & (\text{val. esatto} = 0.5279172\dots) \\
 v) & y = \sqrt{1-x^2} & x \in [0, 1] & (\text{val. esatto} = 1.570796\dots) \\
 vi) & e^{-x^2} & x \in [-1, 1] & (\text{val. esatto} = 1.493648\dots) \\
 vii) & 1/x^2 & x \in [0.1, 1] & (\text{val. esatto} = 9.000000\dots) \\
 viii) & \frac{\sin(e^x + \cos(x))}{(1+(1+x)\log(1+x))} & x \in [0.1, 1] & (\text{val. esatto} = 1.5782736\dots)
 \end{array}$$

con tolleranze 0.001, 0.0001 e con la massima accuratezza del sistema aritmetico floating-point utilizzato. Si fornisca anche una stima dell'errore relativo commesso e del numero di valutazioni di funzioni effettuate.

Esercizio 6 Si sviluppi un integratore automatico ADALOC basato su un algoritmo adattativo con strategia locale che faccia uso della formula trapezoidale e si ripeta l'esercizio 5, confrontando il numero di valutazioni di funzioni effettuate dalle subroutine TRAP2 e ADALOC.

Esercizio 7 Si sviluppi un integratore automatico ADAGLO basato su un algoritmo adattativo, con strategia globale, che faccia uso della formula trapezoidale e si ripeta l'esercizio 5, confrontando il numero di valutazioni di funzioni effettuate dalle subroutine TRAP2, ADALOC e ADAGLO.

Esercizio 8 Utilizzare le subroutine ADALOC e ADAGLO per calcolare l'area al di sotto delle funzioni nell'intervallo $[0, 1]$:

$$\begin{array}{ll}
 f_1(x) = x^2(1.2 - x)(1 - e^{2(x-1)}) & \text{val. esatto} = 0.00954996\dots \\
 f_2(x) = x^{0.1}(1.2 - x)(1 - e^{20(x-1)}) & \text{val. esatto} = 0.602297\dots
 \end{array}$$

utilizzando la tolleranza $tol = 0.001$ e la massima accuratezza del sistema aritmetico utilizzato. Si confronti il numero di valutazioni di funzioni effettuate.

Esercizio 9 Ripetere l'esercizio 8, fornendo in output, la dimensione massima della pila utilizzata dalla subroutine ADALOC e della lista utilizzata dalla subroutine ADAGLO.

Esercizio 10 Mediante la subroutine ADALOC si ripeta l'esercizio 8 fornendo, in output, gli estremi degli intervalli in cui la stima dell'errore soddisfa il criterio di tolleranza locale. Si raffronti il risultato con il grafico delle funzioni.

Esercizio 11 Si ripeta l'esercizio 5 utilizzando le funzioni matlab QUAD e QUAD8. Si valuti la complessità computazionale con la funzione FLOPS

Esercizio 12 Si scriva una funzione MATLAB ADAGLO basata sulla strategia adattativa globale che utilizzi la formula del punto medio. Tale funzione, oltre a fornire una

stima del valore dell'area al di sotto della funzione data e dell'errore relativo commesso, disegni anche il grafico della funzione con i nodi utilizzati dall'algoritmo. Si calcolino nuovamente le aree dell'esercizio 8.

Esercizio 13 Mostrare che i seguenti integrali sono calcolati esattamente dalle formule di fianco indicate:

$$\begin{array}{ll} \int_1^3 4 - x \, dx & \text{formula del punto medio} \\ \int_0^2 x^3 \, dx & \text{formula di Simpson} \\ \int_0^3 x^3 \, dx & \text{formula dei } 3/8 \end{array}$$

Esercizio 14 Mostrare che se si impone che la formula di quadratura con un nodo

$$\int_a^b f(x) \, dx \simeq A_1 f(x_1)$$

sia esatta per le funzioni $f(x) = 1$ e $f(x) = x$ (formula di Gauss Legendre con 1 nodo) si ottiene la formula del punto medio (ciò rappresenta una spiegazione al fatto che la formula del punto medio ha grado di precisione 1).

Esercizio 15 Qual è il più piccolo valore del numero di nodi n affinché la formula di Gauss abbia grado di precisione 7? Verificare la risposta con l'integrale $\int_0^2 x^7 \, dx$.

Esercizio 16 Usare le formule di Gauss Legendre con 3 e 5 nodi, per calcolare $\int_0^4 \sqrt{x} \, dx$. Perché il miglioramento è così modesto?

Esercizio 17 Utilizzare l'algoritmo adattativo locale per calcolare i seguenti integrali con una tolleranza $\pm\epsilon$. Mostrare tutti i passi intermedi.

$$\int_0^4 x^{3/2} \, dx, \epsilon = 0.002 \qquad \int_1^4 x^{1/2} \, dx, \epsilon = 0.001$$

Esercizio 18 Calcolare i seguenti integrali doppi utilizzando il prodotto cartesiano delle due regole indicate di fianco a ciascun integrale e mostrare con un disegno della regione di integrazione la posizione dei nodi:

$$\begin{array}{l} \int_{-1}^1 \int_0^1 (e^{3x} - y) \, dy \, dx \\ \int_{-1}^1 \int_0^{\pi/3} e^x \tan y \, dy \, dx \end{array} \quad \left\{ \begin{array}{l} \text{Gauss Legendre con } n = 4 \text{ per la variabile } x \\ \text{Gauss Legendre con } n = 2 \text{ per la variabile } y \\ \text{Gauss Legendre con } n = 4 \text{ per la variabile } x \\ \text{Gauss Legendre con } n = 2 \text{ per la variabile } y \end{array} \right.$$

Esercizio 19 Verificare che la formula di Newton Cotes con $n = 5$:

$$\int_0^1 df \simeq \frac{1}{90} \left(7f(0) + 32f\left(\frac{1}{4}\right) + 12f\left(\frac{1}{2}\right) + 32f\left(\frac{3}{4}\right) + 7f(1) \right)$$

ha grado di precisione algebrico 4.

Esercizio 20 Calcolare:

$$\int_0^1 \frac{4}{1+x^2} dx = \pi$$

mediante la formula dell'esercizio precedente. Confrontare il risultato ottenuto con quello esatto e confrontare l'errore.

Esercizio 21 Determinare una formula di quadratura del tipo:

$$\int_0^1 \sqrt{x} f(x) dx \simeq A_1 f(0) + A_2 f(1)$$

esatta per polinomi di primo grado.

Esercizio 22 Determinare i pesi della formula:

$$\int_0^1 f(x) dx \simeq A_1 f\left(-\frac{1}{2}\right) + A_2 f(0) + A_3 f\left(\frac{1}{2}\right)$$

esatta per polinomi di secondo grado col metodo dei coefficienti indeterminati (Formule di Newton Cotes aperte). Confrontare i risultati ottenuti da tale formula con la formula di Simpson sui seguenti integrali:

$$\int_0^1 \frac{dx}{1+x} = \ln 2 \quad \int_0^1 \frac{4}{1+x^2} = \pi \quad \int_0^1 e^{-x} dx = 1 - \frac{1}{e}$$

Bibliografia

- [1] Chawla M. - *Convergence of Newton-Cotes quadratures for analytic functions* - BIT, vol. 11 (1971), pp. 159–167
- [2] Davis P. e Rabinowitz P. - *Methods of numerical integration* - Academic Press Inc. (1984)
- [3] Delves L. and Mohamed J. - *Computational methods for integral equations* - Cambridge University Press (1985)
- [4] Engels P. - *Numerical quadrature and cubature* - Academic Press Inc. (1980)
- [5] Evans G. - *Practical Numerical Integration* - Wiley (1993)
- [6] Fox J. et al. - *Solving problems on concurrent processors* - vol. 1, Prentice Hall, 1988.
- [7] Gautschi W. e Inglese G. - *Lower bounds for the condition number of Vandermonde matrices* - Numer. Math. vol. 52 (1988), pp. 241-250
- [8] Genz A. e Malik A. - *An imbedded family of fully symmetric numerical integration rule* - SIAM J. Num. Anal., vol. 20, No.3, June 1983, pp. 580–588
- [9] Hammersley J. e Handscomb D. - *Monte Carlo methods* - Methuen, 1964
- [10] Krommer A. e Ueberhuber C. - *Numerical Integration on advanced computer systems* - Springer Verlag (1994)
- [11] Krommer A.R. e Ueberhuber Ch.W. - *Computational integration* - SIAM (1998)
- [12] Kronrod A. - *Nodes and weights of quadrature formulas* - Consultants bureau, 1965
- [13] Krylov V. - *Approximate calculation of integrals* - Mc Millan Comp. (1962)
- [14] Malcom L. e Simpson R. - *Local versus Global strategies for adaptive quadrature* - ACM Trans. Math. Soft., vol. 1 (1975), pp. 129–146
- [15] Math Works Inc. - *MATLAB - Reference guide* - 1992
- [16] Monegato G. - *Stieltjes polynomials and related quadrature rules* - SIAM Review, vol. 24 (1982), pp. 137–158

- [17] Monegato G. - *Positivity of weights of extended Gauss-Legendre quadrature rules* - Math. Comp (1978)
- [18] Numerical Algorithm Group - *NAG library, Mark 17* - NAG - Oxford (1996)
- [19] Numerical Algorithm Group - *NAG Numerical Parallel library, release 2* - NAG - Oxford (1997)
- [20] Piessens R. et al - *QUADPACK, a subroutine package for automatic integration* - Springer Verlag (1983)
- [21] Sloan I. e Joe S. - *Lattice Methods for Multiple Integration* - Clarendon Press 1994
- [22] Szegő G. - *Orthogonal polynomials* - American Math. Soc., 1939
- [23] Stroud A.H. - *Approximate calculation of multiple integrals* - Prentice Hall (1971)
- [24] Visual Numerics - *IMSL Math Library version 2* - 1993
- [25] Ueberhuber Ch.W. - *Numerical Computation 2, Methods Software and Analysis* - Springer (1997)