Appunti di Calcolo Numerico

a.a. 2019-2020

Indice

Introduzione

Lo studio di molti fenomeni naturali si basa sulla definizione di opportuni *modelli matematici* che permettono di prevedere lo sviluppo del fenomeno stesso senza dover effettuare fisicamente esperimenti complessi, costosi ed in taluni casi pericolosi (si pensi, ad esempio, al propagarsi di una sostanza tossica nelle acque di un fiume). Tuttavia, molto spesso le equazioni che costituiscono il modello non sono risolubili analiticamente, ovvero non esistono tecniche che permettono di esprimere la soluzione come combinazione finita di funzioni elementari note. Inoltre, molto spesso, il problema matematico, sebbene risolubile elementarmente, risulta essere di grosse dimensioni, cioè coinvolge un elevato numero di incognite, e quindi il tempo richiesto per determinarne la soluzione sarebbe troppo elevato. Si presenta quindi la necessità di risolvere il problema matematico mediante l'uso di opportuni *metodi numerici*. Il *Calcolo Numerico* si occupa della definizione e dell'analisi dei metodi numerici e studia gli algoritmi dedotti dai metodi stessi.

Considerato che ogni metodo numerico fornisce una *approssimazione* della soluzione esatta del problema posto, risulta importante saper quantificare l'errore commesso al fine di stabilire la "bontà" della soluzione calcolata. L'insorgenza di tale errore si può attribuire, almeno in linea di principio, alle seguenti cause:

- sostituizione del problema continuo assegnato con un opportuno problema discreto (**errore di discretizzazione** (o analitico)).

  Esempio 0.1 Si supponga che il problema da risolvere consista nel calcolo della derivata prima di una funzione derivabile $f(x)$ in un punto assegnato x_0. Dalla formula di Taylor si ha:

  $$f(x_0+h) = f(x_0) + hf'(x_0) + \frac{h^2}{2}f''(\xi), \quad \xi \in (x_0, x_0+h),$$

  da cui

  $$f'(x_0) = \frac{f(x_0+h) - f(x_0)}{h} - \frac{h}{2}f''(\xi).$$

Il valore di $f'(x_0)$ può quindi essere approssimato, a meno di un errore di discretizzazione che tende a zero con h, dal rapporto incrementale:

$$\varphi(x_0) = \frac{f(x_0 + h) - f(x_0)}{h}.$$

- rappresentazione dei dati sul calcolatore mediante un numero finito di cifre (**errore inerente**).

  Esempio 0.2 I numeri irrazionali quali, ad esempio, π e $\sqrt{2}$ necessitano di una informazione infinita per essere rappresentati esattamente. Pertanto, all'interno di un calcolatore, numeri di questo tipo saranno rappresentati solo in modo approssimato.

- aritmetica finita implementata dal calcolatore dovuta alla memoria finita di cui esso dispone (**errore algoritmico**).

  Esempio 0.3 Siano a un numero reale positivo e $b = (1 + a)^2 - 1$. Se $a = 10^{-6}$, il risultato fornito dal calcolatore è $\hat{b} = 2.000000999924367 \cdot 10^{-6}$, mentre il valore di b è dato da

  $$b = a^2 + 2a = 10^{-12} + 2 \cdot 10^{-6} = (10^{-6} + 2) \cdot 10^{-6} = 2.000001 \cdot 10^{-6}.$$

Dal precedente esempio si evince che per uno stesso problema esistono procedimenti di risoluzione che producono diversi errori algoritmici. Esistono cioè algoritmi più stabili e algoritmi meno stabili. Si ricorda che un algoritmo si dice *stabile* quando la sequenza delle singole operazioni elementari che lo definiscono è tale da non amplificare i piccoli errori dovuti al fatto che dette operazioni non sono effettuate correttamente.

Esistono anche problemi per i quali l'errore generato nel risultato risulta elevato indipendentemente dall'algoritmo utilizzato. Questo fenomeno è una particolarità intrinseca del problema. Per tali problemi, detti *malcondizionati*, piccole variazioni nei dati inducono grosse variazioni sui risultati.

Esempio 0.4 Si consideri il seguente sistema di equazioni lineari:

$$\begin{cases} x + y = 2 \\ 1001x + 1000y = 2001 \end{cases}$$

che ha soluzione $x = y = 1$. Si alteri dell'1% il coefficiente della x nella prima equazione e si consideri il nuovo sistema perturbato

$$\begin{cases} (1 + 1/100)x + y = 2 \\ 1001x + 1000y = 2001 \end{cases}$$

che ha soluzione $\hat{x} = -1/9 \approx -0.1111, \hat{y} = 1901/900 \approx 2.1122$. La soluzione del sistema perturbato presenta, rispetto allo soluzione del sistema non perturbato, una variazione maggiore del 110% sia nella x sia nella y (dare interpretazione geometrica).

Nel caso di malcondizionamento, per trattare il problema conviene aumentare la precisione della rappresentazione dei dati, aumentando il numero di cifre al fine di ridurre l'errore inerente. In tal caso, tuttavia, si presentano due inconvenienti: per memorizzare un numero occorre una *maggior quantità di memoria*; il tempo necessario per eseguire una operazione su due numeri aumenta all'aumentare del numero di cifre, con conseguente incremento del *tempo di elaborazione*.

Una scelta accorta di un metodo per la risoluzione di un problema consente di utilizzare in maniera efficiente la memoria disponibile riducendo al minimo possibile i tempi di esecuzione.

Esempio 0.5 Si consideri il problema del calcolo della soluzione di un sistema di n equazioni lineari in n incognite

$$A\,\mathbf{x} = \mathbf{b},$$

dove A è una matrice di ordine n non singolare, $\mathbf{x}$ è il vettore delle incognite e $\mathbf{b}$ è il vettore dei termini noti.
Un metodo di risoluzione consiste nell'applicare la regola di Cramer mediante la quale le componenti della soluzione vengono espresse come quozienti di determinanti di matrici di ordine n

$$x_j = \frac{\det(A_j)}{\det(A)}, \quad j = 1, 2, \ldots, n,$$

dove A_j è la matrice ottenuta da A sostituendo la j-esima colonna con il vettore $\mathbf{b}$. Tali determinanti sono calcolati con la regola di Laplace

$$\det(A) = \sum_{j=1}^{n} (-1)^{j+1} a_{1j} \det(A_{1j})$$

dove A_{1j} è la matrice di ordine $(n-1)$ ottenuta da A eliminando la prima riga e la j-esima colonna. Indichiamo con C_n il numero di moltiplicazioni richieste per il calcolo del determinante di una matrice di ordine n con la regola di Laplace. Allora vale

$$\begin{aligned} C_2 &= 2 \\ C_n &= n\,C_{n-1} + n, \qquad n \geq 3. \end{aligned}$$

Ne segue che

$$C_n \geq n\,C_{n-1} \geq n(n-1)\,C_{n-2} \geq \cdots \geq n(n-1)\cdots 3\,C_2 = n!$$

Quindi risolvere un sistema di n equazioni ed n incognite con il metodo di Cramer costa almeno $(n+1)\cdot n!$ moltiplicazioni.
Se il sistema viene risolto con il metodo di eliminazione di Gauss: per $k = 1, 2, \ldots, n-1$, si ha

$$\begin{cases} a_{ij}^{(k+1)} = a_{ij}^{(k)} - m_{ik}\, a_{kj}^{(k)}, & i = k+1, \ldots, n,\ j = k, \ldots, n, \\ b_i^{(k+1)} = b_i^{(k)} - m_{ik}\, b_k^{(k)}, & m_{ik} = \dfrac{a_{ik}^{(k)}}{a_{kk}^{(k)}}, \quad i = k+1, \ldots, n \end{cases}$$

e il numero di moltiplicazioni richieste è dell'ordine di $n^3/3$.

Supponendo di lavorare con processori di potenza superiore a 1 Gflops/s (10^9 operazioni elementari al secondo) con i due metodi in esame si avrebbero i seguenti tempi

n	metodo di Cramer	metodo di Gauss
13	1.45 minuti	$1.22 \cdot 10^{-8}$ secondi
14	21.79 minuti	$1.52 \cdot 10^{-8}$ secondi
15	5.81 ore	$1.88 \cdot 10^{-8}$ secondi
20	$1.62 \cdot 10^{3}$ anni	$4.44 \cdot 10^{-8}$ secondi
50	$4.92 \cdot 10^{49}$ anni	$6.94 \cdot 10^{-7}$ secondi

Capitolo 1

Analisi dell'errore

1.1 Numeri reali: rappresentazione in base

L'utilizzo in modo corretto del calcolatore per fare calcoli di tipo scientifico richiede la conoscenza di come sono rappresentati i numeri e degli errori che derivano da questa rappresentazione.

L'uso dei numeri reali richiede una attenzione particolare in quanto essi necessitano di una informazione infinita per essere rappresentati esattamente, mentre all'interno del calcolatore qualunque quantità numerica viene rappresentata mediante una informazione finita.

La rappresentazione dei numeri che utilizziamo comunemente è detta *posizionale*. Essa può essere fatta risalire ai Babilonesi, che usavano la base 60. Tale base è usata ancora oggi, ad esempio, nelle misure del tempo (ore, minuti, secondi). La notazione posizionale in base 10 venne introdotta in Europa dagli Arabi ed è particolarmente comoda per il calcolo (si provi a fare una semplice addizione utilizzando la rappresentazione adottata dagli antichi Romani). Poiché la base del nostro sistema di numerazione è 10 (*sistema decimale*) quando scriviamo il numero $x = 185.2$ intendiamo indicare la quantità

$$x = 1 \cdot 10^2 + 8 \cdot 10^1 + 5 \cdot 10^0 + 2 \cdot 10^{-1}.$$

Non esiste, tuttavia, una ragione particolare per assumere 10 come base del sistema di numerazione. Scelto un qualunque numero $b \in \mathbb{N}$ $(b > 1)$ come base della rappresentazione vale il seguente

Teorema 1.1 (di rappresentazione in base) *Sia $x \in \mathbb{R}\backslash\{0\}$. Allora esistono e sono unici il numero intero η e la successione $\{\alpha_i\}_{i=1,2,\ldots}$ di numeri interi,*

$0 \leq \alpha_i \leq b-1$, $\alpha_1 \neq 0$, *non definitivamente uguali a* $b-1$, *tali che*

$$x = \pm\, b^{\eta} \sum_{i=1}^{\infty} \alpha_i b^{1-i}. \tag{1.1}$$

La rappresentazione (1.1) viene detta *rappresentazione in base del numero* x.

Poiché $\alpha_1 \neq 0$ la rappresentazione (1.1) è detta *normalizzata*. La rappresentazione (1.1) in base b viene indicata con la notazione posizionale

$$x = \pm\, (\alpha_1.\alpha_2\, \alpha_3 \ldots)\, b^{\eta}.$$

La notazione normalizzata, oltre ad essere necessaria per l'unicità, è utile quando si hanno a disposizione solo un numero finito di cifre per rappresentare un numero (ciò che avviene appunto sul calcolatore!).

Esempio 1.1 Il numero $x = \frac{1}{7000}$ può essere equivalentemente rappresentato in base 10 come segue:

$$\begin{array}{ll} x = (1.42857142857\ldots) \cdot 10^{-4} & \text{(rappresentazione normalizzata)} \\ x = (0.000142857142\ldots) & \text{(rappresentazione } \textbf{non} \text{ normalizzata).} \end{array}$$

Se si tronca la rappresentazione a 6 cifre si ottiene nel primo caso il numero $x_1 = (1.42857) \cdot 10^{-4}$ e nel secondo caso il numero $x_2 = (.000142) = (1.40000) \cdot 10^{-4}$. È evidente che x_1 fornisce una migliore approssimazione, rispetto a x_2, del valore di x.

La base utilizzata dalla maggior parte dei calcolatori è $b = 2$ (*sistema binario*). Ogni cifra della rappresentazione ha il valore 0 oppure 1 e si dice *bit* (binary digit).

Esempio 1.2 Sia $y = 0.0101$. Allora

$$y = 0 \cdot 2^{-1} + 1 \cdot 2^{-2} + 0 \cdot 2^{-3} + 1 \cdot 2^{-4}$$

ed in forma normalizzata

$$y = 1.01 \cdot 2^{-2} = 1 \cdot 2^{-2} + 0 \cdot 2^{-3} + 1 \cdot 2^{-4}.$$

L'uso della base 2, o di basi che sono potenze intere di 2, deriva dalla natura fisica dei dipositivi di memoria dell'hardware di un calcolatore; ciascuna locazione di memoria fa riferimento ad un dispositivo che può assumere due stati fisici differenti.
Poiché la rappresentazione è normalizzata, risulta sempre $\alpha_1 = 1$. Quindi non è necessario memorizzare il primo bit della mantissa. Ciò garantisce il risparmio di una locazione di memoria nel rappresentare i numeri al calcolatore.

1.1.1 Conversione di base

Sia x un numero reale espresso in base 10 tramite la (1.1). Per poter memorizzare tale numero su un calcolatore che opera in base 2 bisogna esprimere x in cifre binarie. A tale scopo si distinguono due casi:

(i) x è intero;

(ii) x è decimale (ovvero $0 < x < 1$).

Se x è un numero **intero** si devono determinare le cifre $\gamma_0, \gamma_1, \ldots, \gamma_n$ tali che:

$$x = \gamma_n 2^n + \gamma_{n-1} 2^{n-1} + \cdots + \gamma_1 2^1 + \gamma_0 2^0.$$

Se consideriamo la seguente uguaglianza

$$x = 2(\gamma_n 2^{n-1} + \cdots + \gamma_1 2^0) + \gamma_0,$$

si nota che il valore di γ_0 può essere ottenuto come resto della divisione di x per 2. Infatti:

$$x = 2\, x_1 + \gamma_0$$

con $x_1 = \gamma_n 2^{n-1} + \cdots + \gamma_1 2^0$. Ripetendo il procedimento su x_1 si determina γ_1 e quindi, analogamente, tutte le restanti cifre.

Esempio 1.3 $(123)_{10} = (1111011)_2$. Infatti,

$$\begin{array}{rcll} 123 & = & 2 \cdot 61 + 1 & \Rightarrow \quad \gamma_0 = 1 \\ 61 & = & 2 \cdot 30 + 1 & \Rightarrow \quad \gamma_1 = 1 \\ 30 & = & 2 \cdot 15 + 0 & \Rightarrow \quad \gamma_2 = 0 \\ 15 & = & 2 \cdot 7 + 1 & \Rightarrow \quad \gamma_3 = 1 \\ 7 & = & 2 \cdot 3 + 1 & \Rightarrow \quad \gamma_4 = 1 \\ 3 & = & 2 \cdot 1 + 1 & \Rightarrow \quad \gamma_5 = 1 \\ 1 & = & 2 \cdot 0 + 1 & \Rightarrow \quad \gamma_6 = 1 \\ 0 & & & \Rightarrow \quad \text{stop} \end{array}$$

Se x è un numero **decimale** bisogna trovare le cifre $\beta_1, \beta_2, \ldots, \beta_p$ tali che:

$$x = \beta_1 2^{-1} + \beta_2 2^{-2} + \cdots + \beta_{p-1} 2^{1-p} + \beta_p 2^{-p}.$$

Moltiplicando x per 2 si ha:

$$2 \cdot x = \beta_1 + \beta_2 2^{-1} + \cdots + \beta_{p-1} 2^{2-p} + \beta_p 2^{1-p}$$

e quindi β_1 risulta essere la parte intera di $2x$. Il procedimento si può iterare applicandolo a

$$x_1 = \beta_2 2^{-1} + \cdots + \beta_{p-1} 2^{2-p} + \beta_p 2^{1-p}$$

per determinare β_2 e così via fino a che non si determinano tutte le rimanenti cifre.

Esempio 1.4 $(0.625)_{10} = (0.101)_2$. Infatti,

$$\begin{array}{rcrcl} 2 \cdot 0.625 & = & 1.25 & \Rightarrow & \beta_1 = 1 \\ 2 \cdot 0.25 & = & 0.5 & \Rightarrow & \beta_2 = 0 \\ 2 \cdot 0.5 & = & 1 & \Rightarrow & \beta_3 = 1 \\ 2 \cdot 0 & = & 0 & \Rightarrow & \text{stop} \end{array}$$

Osservazione 1.1 *Se $x > 1$ non è intero, si pone $x = \chi_1 + \chi_2$, con $\chi_1 = \lfloor x \rfloor$, la parte intera di x, e $\chi_2 \in (0, 1)$, e si utilizzano le due procedure appena descritte.*

Osservazione 1.2 *I ragionamenti precedenti si applicano anche nel caso in cui si voglia passare ad una base N, con $N > 2$.*

Si osservi che un numero può avere rappresentazione finita in una base ed una rappresentazione non finita in un'altra base.

Esercizio 1.1 Esprimere 1/10 in base 2.

1.2 Numeri di macchina

Si distinguono essenzialmente i seguenti due casi:

- numeri interi
- numeri reali

1.2.1 Numeri interi

Un numero intero è memorizzato, nella memoria di un calcolatore, mediante la stringa

$$\alpha_0 \, \alpha_1 \, \ldots\ldots \, \alpha_N$$

Assegnata la base $b \in \mathbb{N}$ α_0 rappresenta il segno e $\alpha_i \in \{0, 1, 2, \ldots, b-1\}, i = 1, 2, \ldots, N$. A questa stringa corrisponde il numero intero

$$n = \begin{cases} \sum_{i=1}^{N} \alpha_i \, b^{N-i} & \text{se } \alpha_0 = + \\ \sum_{i=1}^{N} \alpha_i \, b^{N-i} - b^N & \text{se } \alpha_0 = - \end{cases}$$

Esempio 1.5 *Nel caso in cui $b = 2$ e $N = 2$ si ha:*

$$n = \begin{cases} \alpha_1 \cdot 2 + \alpha_2 & \text{se } \alpha_0 = + \\ \alpha_1 \cdot 2 + \alpha_2 - 4 & \text{se } \alpha_0 = - \end{cases}$$

ovvero i numeri interi rappresentabili sono $\{-4, -3, -2, -1, 0, 1, 2, 3\}$.

In generale, i numeri interi rappresentabili appartengono all'insieme

$$\{-b^N, \ldots, b^N - 1\}.$$

Infatti, i valori minimo si ottiene per $\alpha_0 = -$ e $\alpha_i = 0, i = 1, 2, \ldots, N$. Il valore massimo invece si ha per $\alpha_0 = +$ e $\alpha_i = b - 1, i = 1, 2, \ldots, N$.

1.2.2 Numeri reali

Un numero reale viene memorizzato nella memoria di un calcolatore mediante una stringa del tipo:

$$\alpha_0\, \alpha_1\, \ldots\ldots\, \alpha_m\, \beta_1 \ldots \beta_s$$

in cui, assegnata la base $b \in \mathbb{N}$, α_0 rappresenta il segno e $\alpha_i, \beta_j \in \{0, 1, \ldots, b-1\}, i = 1, 2, \ldots, m, j = 1, 2, \ldots, s$, con $\alpha_0 \neq 0$. Tale rappresentazione di un numero reale definisce la notazione scientifica normalizzata in base b. A questa stringa corrisponde il numero reale:

$$\alpha_0\, \alpha_1\, \ldots\ldots\, \alpha_m\, \beta_1 \ldots \beta_s \quad \Leftrightarrow \quad r = \pm \left(\sum_{i=1}^{m} \alpha_i b^{1-i} \right) b^{e-\nu},\ e = \sum_{j=1}^{s} \beta_j b^{s-j}$$

Le quantità che compaiono nella relazione precedente sono denominate, rispettivamente,

α_i	cifre della rappresentazione
$\pm \left(\sum_{i=1}^{m} \alpha_i b^{1-i}\right)$	mantissa
e	esponente
$\nu \in \mathbb{N}$	shift (o *bias*)

L'esponente può essere rappresentato come un intero relativo, ma generalmente si usa la rappresentazione in traslazione. L'esponente $\eta = (e - \nu)$ si chiama *caratteristica*.

Definizione 1.1 *Si definisce insieme dei* numeri di macchina *(o* numeri floating-point*) con rappresentazione normalizzata, in base b con m cifre significative, l'insieme*

$$\mathcal{M} = \{0\} \cup \left\{ r \in \mathbb{R} : r = \pm\, b^{\eta} \sum_{i=1}^{m} \alpha_i b^{1-i} \right\}$$

con $0 \leq \alpha_i \leq b - 1, i = 1, 2, \ldots, m, \alpha_1 \neq 0$.

Si osservi che l'insieme dei numeri macchina contiene per definizione lo zero che non è rappresentabile con una mantissa normalizzata e quindi non è rappresentabile in modo univoco. Inoltre, l'insieme dei numeri macchina contiene un numero finito di elementi e la distribuzione di tali numeri non è uniforme. In particolare, la distanza tra due numeri consecutivi aumenta allontanandosi dall'origine. Sia

$$r_1 = \left(\sum_{i=1}^{m} \alpha_i b^{1-i}\right) b^{\eta}, \quad r_2 = \left(\sum_{i=1}^{m} \alpha_i b^{1-i} + b^{1-m}\right) b^{\eta}$$

allora

$$r_2 - r_1 = b^{1-m+\eta}$$

Teorema 1.2 *Il minimo ed il massimo (in valore assoluto) tra i numeri macchina appartenenti a* $\mathcal{M}$ *e diversi da* 0 *sono, rispettivamente,* $r_{\min} = b^{-\nu}$ *e* $r_{\max} = (1 - b^{-m}) \cdot b^{M}$, *con* $M = b^s - \nu$.

Dim. $r_{\min}$ si ottiene ponendo $\alpha_1 = 1, \alpha_i = 0, i = 2, 3, \ldots, m$ e $\beta_j = 0, j = 1, 2, \ldots, s$. Allora, si ha

$$r_{\min} = (1.\underbrace{0 \ldots 0}_{m-1})b^{-\nu} = b^{-\nu}.$$

Inoltre, $r_{\max}$ si ottiene ponendo $\alpha_i = b-1, i = 1, 2, 3, \ldots, m$ e $\beta_j = b-1, j = 1, 2, \ldots, s$. Allora, si ha

$$\begin{aligned} r_{\max} &= b^{b^s-1-\nu} \sum_{i=1}^{m} (b-1) b^{1-i} \\ &= b^{b^s-1-\nu}(b-1)(b^0 + b^{-1} + b^{-2} + \cdots + b^{2-m} + b^{1-m}) \\ &= b^{b^s-1-\nu}(b + b^0 + b^{-1} + \cdots + b^{2-m} - b^0 - b^{-1} - \cdots - b^{2-m} - b^{1-m}) \\ &= b^{b^s-1-\nu}(b - b^{1-m}). \end{aligned}$$

□

Osservazione 1.3 *In* `Matlab` *le variabili predefinite* `realmin` *e* `realmax` *contengono (in valore assoluto) il più grande ed il più piccolo numero di macchina.*

1.2.3 Lo standard IEEE 754

La rappresentazione dei numeri reali usata dalla maggior parte dei calcolatori segue lo standard ANSI/IEEE 754-1985, definito dall'Instituto degli

Ingegneri Elettrici ed Elettronici. Tale standard è stato definito per poter garantire che stessi programmi, anche se eseguiti su piattaforme di calcolo differenti, producano gli stessi risultati. Tale standard che utilizza il sistema binario ($b = 2$) prevede i seguenti due formati per memorizzare i dati: singola precisione e doppia precisione. In particolare,

- la **singola precisione** corrisponde a 32 bit (4 byte) così ripartiti:

1 bit	8 bit	23 bit
segno	esponente	mantissa

In particolare,

$$\text{segno} = \begin{cases} 0 & \text{se il numero è positivo,} \\ 1 & \text{se il numero è negativo,} \end{cases}$$

l'esponente è rappresentato in forma intera ed essendo costituito da 8 bit, permette di rappresentare 256 valori. Ai valori 0 e 255 vengono riservati funzioni speciali (descritte in seguito); gli altri permettono di rappresentare 254 valori per i numeri in forma normale, compresi tra -126 e 127. Per evitare di dover utilizzare un bit per memorizzare il segno dell'esponente si utilizza una traslazione pari a 127. In fase di decodifica del numero, il bias viene nuovamente sottratto per recuperare il valore originale. I valori assunti dall'esponente e dalla mantissa determinano l'appartenenza del numero ad una di queste categorie:

Categoria	**Esponente**	**Mantissa**
zeri	0	0
numeri denormalizzati	0	non zero
numeri normalizzati	1-254	qualunque
infiniti	255	0
NaN[1]	255	non zero

Tutte le mantisse sono normalizzate. Pertanto il numero prima della virgola risulta essere sempre il numero binario 1. Allora esso viene omesso e viene chiamato *bit nascosto*, o *bit implicito*.

- la **doppia precisione** corrisponde a 64 bit (8 byte) così ripartiti:

1 bit	11 bit	52 bit
segno	esponente	mantissa

Per i numeri normalizzati il bias è pari a 1023.

Osservazione 1.4 *Attualmente lo standard per il calcolo in virgola mobile IEEE 754 è in revisione ed il progetto di revisione si chiama IEEE 754r. Lo scopo di tale revisione è rivedere lo standard ed unirlo nell'IEEE 854 (lo standard per il calcolo in virgola mobile non posizionale).*

Osservazione 1.5 `Matlab` *usa lo standard IEEE 754 in doppia precisione.*

1.3 Troncamento ed Arrotondamento

Dal Teorema 1.2 segue che i numeri di macchina sono contenuti nel sottoinsieme della retta reale:

$$\mathcal{I} = [-r_{\max}, -r_{\min}] \cup \{0\} \cup [r_{\min}, r_{\max}]. \tag{1.2}$$

Sia x un numero reale, diverso da zero ed appartenente all'insieme $\mathcal{I}$. Se $x \notin \mathcal{M}$ si pone il problema di associare ad x un numero di macchina che denoteremo con $fl(x)$. Per semplicità supponiamo che x sia un numero positivo. Si possono verificare i seguenti casi:

(1) $x \in \mathcal{I}$, ma $\alpha_i \neq 0$ per $i > m$. In tal caso si possono utilizzare due diverse tecniche di rappresentazione:

- **troncamento**:
$$fl(x) = b^{\eta} \sum_{i=1}^{m} \alpha_i b^{1-i};$$
- **arrotondamento**:
$$fl(x) = \begin{cases} b^{\eta} \sum_{i=1}^{m} \alpha_i b^{1-i} & \text{se } \alpha_{m+1} < \frac{b}{2} \\ b^{\eta} \left(\sum_{i=1}^{m} \alpha_i b^{1-i} + b^{1-m}\right) & \text{se } \alpha_{m+1} \geq \frac{b}{2}. \end{cases}$$

(2) $x \notin \mathcal{I}$:

- $0 < x < r_{\min}$: in tal caso si incorre in una condizione di errore denominata **underflow**. Esistono, generalmente, due tipi di *recovery*:
  - nella prima, denominata *store* 0, si pone $fl(x) = 0$;
  - nella seconda, denominata *gradual underflow*, si denormalizza la mantissa del numero rappresentato, ovvero parte dei bit della mantissa passano all'esponente. Il processo si arresta quando tutti i bit della mantissa sono passati all'esponente. Come si capirà fra breve, con tale procedimento si perdono cifre significative.

- $x > r_{\max}$: in tal caso si incorre in una condizione di errore denominata **overflow**. La sua *recovery* dipende dal sistema di calcolo utilizzato. Lo standard IEEE 754 contempla una quantità indicata con Inf che rappresenta, virtualmente, ∞. In questo caso, quindi, $fl(x) =$Inf.

Esercizio 1.2 *In* `Matlab` *si calcoli* `realmin`$/10^{-q}$ *facendo variare* q *tra* 1 *e* 16 *e* `realmax`$*10$. *Commentare i risultati.*

1.3.1 Errori di rappresentazione

Cerchiamo ora di capire qual è l'errore che si commette nell'approssimare x con un numero di macchina tramite il processo di troncamento o di arrotondamento.
Si considerino innanzitutto le seguenti quantità:

$$E_a = |fl(x) - x|, \qquad E_r = \left|\frac{fl(x) - x}{x}\right|,$$

dette, rispettivamente, *errore assoluto* ed *errore relativo*. Inoltre, supponendo per semplicità che $x > 0$, siano r_1 e r_2 due numeri di macchina consecutivi:

$$r_1 = b^\eta \sum_{i=1}^{m} \alpha_i b^{1-i}, \qquad r_2 = b^\eta \left(\sum_{i=1}^{m} \alpha_i b^{1-i} + b^{1-m} \right),$$

tali che $r_1 \leq x < r_2$. Consideriamo ora separatamente i due casi:

- **rappresentazione con troncamento**: $fl(x) = r_1$.
  In particolare si ha $x - fl(x) < r_2 - r_1 = b^{\eta+1-m}$ e quindi

  $$|x - fl(x)| < b^{\eta+1-m}.$$

  Inoltre, poiché $\alpha_1 \neq 0$ si ha $x \geq b^\eta$, ne segue che

  $$\left|\frac{x - fl(x)}{x}\right| < \frac{b^{\eta+1-m}}{b^\eta} = b^{1-m}. \tag{1.3}$$

- **rappresentazione con arrotondamento**:

  $$fl(x) = \begin{cases} r_1 & \text{se } x < \dfrac{r_1 + r_2}{2} \\ \\ r_2 & \text{se } x \geq \dfrac{r_1 + r_2}{2}. \end{cases}$$

Allora si ha:

$$|x - fl(x)| \leq \frac{r_2 - r_1}{2} = \frac{1}{2} b^{\eta+1-m} \tag{1.4}$$

con l'uguaglianza che vale se e solo se $x = (r_1 + r_2)/2$. Infatti, in tal caso $\alpha_{m+1} = b/2$ e quindi $fl(x) = r_2$. Ne segue che:

$$\left| \frac{r_1 + r_2}{2} - r_2 \right| = \left| \frac{r_1 + r_2 - 2r_2}{2} \right| = \left| \frac{r_1 - r_2}{2} \right| = \frac{r_2 - r_1}{2}.$$

Inoltre, essendo $x > b^{\eta}$, da (1.4) si ha:

$$\left| \frac{x - fl(x)}{x} \right| \leq \frac{1}{2} b^{1-m}. \tag{1.5}$$

Osservazione 1.6 *In situazione di underflow, cioè nel caso in cui* $0 < x < r_{\min}$*, l'errore relativo è*

$$\left| \frac{x - 0}{x} \right| = 1.$$

Dagli argomenti su esposti si evince che, a parità di cifre significative nella mantissa, l'arrotondamento comporta un errore di rappresentazione pari alla metà di quello che deriva dal troncamento. È questo il motivo per cui la rappresentazione con arrotondamento è di gran lunga la più utilizzata.

Si definisca la quantità

$$u = \begin{cases} b^{1-m}, & \text{in caso di troncamento,} \\ \\ \frac{1}{2} b^{1-m}, & \text{in caso di arrotondamento,} \end{cases} \tag{1.6}$$

detta *precisione di macchina* e si ponga

$$\varepsilon_x = \frac{fl(x) - x}{x}.$$

Da (1.3), (1.5) e (1.6) si ottiene

$$|\varepsilon_x| < u,$$

ovvero, l'errore relativo commesso nel rappresentare il numero x sul calcolatore è maggiorato dalla precisione di macchina u. Inoltre, dalle precedenti segue che

$$fl(x) = x(1 + \varepsilon_x), \qquad |\varepsilon_x| < u.$$

Osservazione 1.7 `Matlab` *utilizza la rappresentazione con arrotondamento. Il valore della precisione di macchina è quindi* $u = 2^{-52} \approx 2.22 \cdot 10^{-16}$ *ed esso è contenuto nella variabile predefinita* `eps`.

1.4 Operazioni di macchina

È improbabile che il risultato di una qualunque operazione algebrica elementare tra due numeri di macchina sia ancora un numero di macchina. Occorre perciò definire una *aritmetica finita*, ossia individuare le operazioni su $\mathcal{M}$. Dati x, y due numeri di macchina si dice che una aritmetica di macchina è ben definita se

$$fl(x \text{ op } y) = (x \text{ op } y)(1 + \varepsilon_x), \quad |\varepsilon_x| < u,$$

dove op $\in \{+, -, \times, /\}$ denota una delle operazioni esatte. In seguito utilizzeremo la seguente notazione:

$$fl(x+y) = x \oplus y, \quad fl(x-y) = x \ominus y, \quad fl(x \times y) = x \otimes y, \quad fl(x/y) = x \oslash y.$$

L'aritmetica di macchina non soddisfa, in generale, le proprietà dell'aritmetica definita nei numeri reali. Le seguenti proprietà:

(1) associativa dell'addizione: $(x + y) + z = x + (y + z)$,

(2) associativa della moltiplicazione: $(xy)z = x(yz)$,

(3) legge di cancellazione: $xy = yz, y \neq 0 \Rightarrow x = z$

(4) distributiva: $x(y + z) = xy + xz$,

(5) semplificazione: $y\dfrac{x}{y} = x$,

in generale non valgono.

Esempio 1.6 Sia $\mathcal{M}$ con $b = 10, m = 2$ e aritmetica con arrotondamento:

(1) Posto $x = (1.1)10^{-1}, y = (1.3)10^{-2} = (0.13)10^{-1}, z = (1.4)10^{-2} = (0.14)10^{-1}$, si ha $x + y = (1.23)10^{-1}, y + z = (0.27)10^{-1}$. Allora

$$\begin{aligned} (x \oplus y) \oplus z &= (1.2)10^{-1} \oplus (0.14)10^{-1} = (1.34)10^{-1} = (1.3)10^{-1} \\ x \oplus (y \oplus z) &= (1.1)10^{-1} \oplus (0.3)10^{-1} = (1.4)10^{-1}. \end{aligned}$$

Quindi non vale la proprietà associativa dell'addizione.

(2) Posto $x = (1.1)10^{0}, y = (3.1)10^{0}, z = (2.5)10^{0}$, si ha $x \times y = (3.41)10^{0}, y \times z = (7.75)10^{0}$ e quindi

$$\begin{aligned} (x \otimes y) \otimes z &= (3.4)10^{0} \otimes (2.5)10^{0} = (8.5)10^{0} \\ x \otimes (y \otimes z) &= (1.1)10^{0} \otimes (7.8)10^{0} = (8.6)10^{0}. \end{aligned}$$

Quindi non vale la proprietà associativa della moltiplicazione.

(3) Posto $x = (5.1)10^0, y = (2.2)10^0, z = (5.2)10^0$, si ha

$$\begin{aligned} x \otimes y &= (1.1)10^1 \\ z \otimes y &= (1.1)10^1. \end{aligned}$$

Quindi

$$x \otimes y = z \otimes y, \quad y \neq 0,\ x \neq z.$$

Si conclude che non vale la legge di cancellazione.

(4) Posto $x = (1.1)10^0, y = (2.3)10^0, z = (2.4)10^0$, si ha $y + z = (4.7)10^0$. Allora

$$\begin{aligned} x \otimes (y \oplus z) &= (1.1)10^0 \otimes (4.7)10^0 = fl((5.17)10^0) = (5.2)10^0 \\ (x \otimes y) \oplus (x \otimes z) &= fl((2.53)10^0) \oplus fl((2.64)10^0) = (2.5)10^0 \oplus (2.6)10^0 = (5.1)10^0. \end{aligned}$$

Quindi non vale la proprietà distributiva.

(5) Sia $x = (1.0)10^0, y = (3.0)10^0$. Si ha $x/y = (3.333\ldots)10^{-1}$ e quindi

$$y \otimes (x \oslash y) = (3.0)10^0 \otimes (3.3)10^{-1} = (9.9)10^{-1}.$$

Quindi non vale la semplificazione.

1.5 Calcolo del valore di una funzione

Sia $f : \mathbb{R}^n \to \mathbb{R}$ una generica funzione di cui si vuole calcolare il valore in corrispondenza ad un vettore $\mathbf{x} = (x_1, x_2, \ldots, x_n)$. A tale scopo bisogna tenere presente che il valore calcolato $y = f(\mathbf{x})$ può essere affetto da errori introdotti da vari fenomeni:

- errore generato dalla rappresentazione dei dati $x_1, x_2, \ldots, x_n$ come numeri di macchina (**errore inerente**);
- errore generato dal fatto che le operazioni sono effettuate in aritmetica finita (**errore algoritmico**);
- se la funzione f non è razionale, errore generato dalla sua approssimazione con una funzione razionale (**errore analitico**).

Osservazione 1.8 *Nella stima dell'errore commesso per calcolare il valore della funzione in un dato vettore si preferisce considerare l'errore relativo*

invece di quello assoluto. Questo perché, in generale, l'errore relativo fornisce una informazione più completa rispetto all'errore assoluto. A titolo di esempio si considerino $z = 1000$, $\tilde{z} = 1000.5$, $w = 0.01$ *e* $\tilde{w} = 0.51$. *Allora*

$$E_a = |\tilde{z} - z| = (5.0)10^{-1}, \qquad E_r = \left|\frac{\tilde{z} - z}{z}\right| = \left|\frac{.5}{1000}\right| = (5.0) \cdot 10^{-4};$$

$$E_a = |\tilde{w} - w| = (5.0)10^{-1}, \qquad E_r = \left|\frac{\tilde{w} - w}{w}\right| = \left|\frac{.5}{0.01}\right| = (5.0) \cdot 10^{1}.$$

Si osservi come in entrambe i casi l'errore assoluto sia lo stesso, pur avendo $\tilde{z}$ 4 *cifre significative e* $\tilde{w}$ *neanche una!*

Sia $\tilde{\mathbf{x}} = (\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_n)$ il vettore le cui componenti sono i numeri di macchina che rappresentano le corrispondenti componenti in $\mathbf{x}$, i.e., $\tilde{x}_i = x_i(1 + \epsilon_i), |\epsilon_i| < u, i = 1, 2, \ldots, n$, ψ la funzione effettivamente calcolata e g la funzione analitica che approssima f nel caso in cui essa non sia razionale (ovviamente, se f è razionale, $g \equiv f$). Allora, posto

$$\begin{aligned}
E_{tot} &= \frac{\psi(\tilde{\mathbf{x}}) - f(\mathbf{x})}{f(\mathbf{x})} && \text{(errore totale relativo)} \\
E_{in} &= \frac{g(\tilde{\mathbf{x}}) - g(\mathbf{x})}{g(\mathbf{x})} && \text{(errore inerente relativo)} \\
E_{alg} &= \frac{\psi(\tilde{\mathbf{x}}) - g(\tilde{\mathbf{x}})}{g(\tilde{\mathbf{x}})} && \text{(errore algoritmico relativo)} \\
E_{an} &= \frac{g(\mathbf{x}) - f(\mathbf{x})}{f(\mathbf{x})} && \text{(errore analitico relativo),}
\end{aligned}$$

si ha

$$E_{tot} \approx E_{in} + E_{alg} + E_{an},$$

Infatti,

$$\begin{aligned}
E_{tot} &= \frac{\psi(\tilde{\mathbf{x}}) - f(\mathbf{x})}{f(\mathbf{x})} = \frac{\psi(\tilde{\mathbf{x}})}{f(\mathbf{x})} - 1 = \frac{\psi(\tilde{\mathbf{x}})}{g(\tilde{\mathbf{x}})}\frac{g(\tilde{\mathbf{x}})}{g(\mathbf{x})}\frac{g(\mathbf{x})}{f(\mathbf{x})} - 1 \\
&= (E_{alg} + 1)(E_{in} + 1)(E_{an} + 1) - 1 \\
&= E_{in} + E_{alg} + E_{an} + E_{in}E_{alg} + E_{in}E_{an} + E_{alg}E_{an} + E_{in}E_{alg}E_{an}.
\end{aligned}$$

Se tali errori sono dell'ordine della precisione di macchina è ragionevole condurre un'analisi dell'errore al primo ordine. Ciò implica che

$$E_{tot} \approx E_{in} + E_{alg} + E_{an}.$$

Supponiamo ora, per semplicità, che f sia una funzione razionale. In tal caso si ha immediatamente che $E_{an} = 0$ e quindi $E_{tot} \approx E_{in} + E_{alg}$. Inoltre, nel caso in cui f sia "sufficientemente" regolare, dallo sviluppo in serie di Taylor risulta

$$f(\tilde{\mathbf{x}}) - f(\mathbf{x}) = \sum_{i=1}^{n} (\tilde{x}_i - x_i) \frac{\partial f(\mathbf{x})}{\partial x_i} + \text{termini di ordine superiore.}$$

Allora, ricordando che $\tilde{x}_i = x_i(1+\epsilon_i), |\epsilon_i| < u$,

$$E_{in} = \frac{f(\tilde{\mathbf{x}}) - f(\mathbf{x})}{f(\mathbf{x})} \approx \sum_{i=1}^{n} \frac{x_i \epsilon_i}{f(\mathbf{x})} \frac{\partial f(\mathbf{x})}{\partial x_i}.$$

Posto

$$\gamma_i = \frac{x_i}{f(\mathbf{x})} \frac{\partial f(\mathbf{x})}{\partial x_i}, \qquad i = 1, 2, \ldots, n, \tag{1.7}$$

l'errore inerente può riscriversi come segue:

$$E_{in} = \sum_{i=1}^{n} \gamma_i \epsilon_i.$$

I coefficienti γ_i, detti *coefficienti di amplificazione*, danno una misura di quanto gli errori relativi ϵ_i influiscano sul risultato: se essi sono di modulo elevato anche piccoli ϵ_i inducono grossi errori su f. In questo caso il problema del calcolo di f è detto *problema malcondizionato*.
Per concludere, si osserva che l'errore algoritmico E_{alg} è generato dal calcolo della funzione ψ che è esprimibile come composizione di un numero finito di operazioni di macchina. In tal caso si dirà che l'algoritmo è tanto più *stabile* in corrispondenza ad un insieme di dati quanto più piccolo è l'errore generato eseguendo l'algoritmo.

Esempio 1.7 Sia $f : \mathbb{R}^2 \to \mathbb{R}$, $\mathbf{x} = (x_1, x_2)$ e $\tilde{\mathbf{x}} = (\tilde{x}_1, \tilde{x}_2)$. Si vogliono determinare gli errori inerenti relativi commessi effettuando le operazioni di macchina.

- **Somma**: $f(\mathbf{x}) = x_1 + x_2$.
  Ricordando che

$$\begin{aligned} \tilde{x}_1 &= x_1(1+\epsilon_1), \quad |\epsilon_1| < u, \\ \tilde{x}_2 &= x_2(1+\epsilon_2), \quad |\epsilon_2| < u, \end{aligned}$$

si ha

$$E_{in} = \frac{(\tilde{x}_1+\tilde{x}_2)-(x_1+x_2)}{x_1+x_2} = \frac{(x_1(1+\epsilon_1)+x_2(1+\epsilon_2))-(x_1+x_2)}{x_1+x_2}$$
$$= \frac{x_1}{x_1+x_2}\epsilon_1 + \frac{x_2}{x_1+x_2}\epsilon_2 = \gamma_1\epsilon_1+\gamma_2\epsilon_2,$$

con (vedi (1.7))

$$\gamma_1 = \frac{x_1}{x_1+x_2}, \qquad \gamma_2 = \frac{x_2}{x_1+x_2}.$$

Osservazione 1.9 *Se $x_1x_2 < 0$ e $|x_1| \approx |x_2|$, γ_1 e γ_2 sono quantità molto grandi. Quando ciò accade si verifica il cosiddetto fenomeno della* cancellazione numerica. *Ad esempio, siano $x_1 = 0.123456$, $x_2 = -0.123454$. Allora*

$$\gamma_1 = 6.1728 \cdot 10^4, \qquad \gamma_2 = -6.1727 \cdot 10^4.$$

- **Sottrazione**: $f(\mathbf{x}) = x_1 - x_2$.

$$E_{in} = \frac{(\tilde{x}_1-\tilde{x}_2)-(x_1-x_2)}{x_1-x_2} = \frac{(x_1(1+\epsilon_1)-x_2(1+\epsilon_2))-(x_1-x_2)}{x_1-x_2}$$
$$= \frac{x_1}{x_1-x_2}\epsilon_1 - \frac{x_2}{x_1-x_2}\epsilon_2 = \gamma_1\epsilon_1+\gamma_2\epsilon_2,$$

con (vedi (1.7))

$$\gamma_1 = \frac{x_1}{x_1-x_2}, \qquad \gamma_2 = -\frac{x_2}{x_1-x_2}.$$

Osservazione 1.10 *Se $x_1x_2 > 0$ e $|x_1| \approx |x_2|$, γ_1 e γ_2 sono quantità molto grandi. Quando ciò accade si verifica il cosiddetto fenomeno della* cancellazione numerica.

- **Moltiplicazione**: $f(\mathbf{x}) = x_1x_2$.

$$E_{in} = \frac{(\tilde{x}_1\tilde{x}_2)-(x_1x_2)}{x_1x_2} = \frac{(x_1(1+\epsilon_1)x_2(1+\epsilon_2))-(x_1x_2)}{x_1x_2}$$
$$= \epsilon_1+\epsilon_2+\epsilon_1\epsilon_2 = \gamma_1\epsilon_1+\gamma_2\epsilon_2+\epsilon_1\epsilon_2 \approx \gamma_1\epsilon_1+\gamma_2\epsilon_2,$$

con

$$\gamma_1 = 1, \qquad \gamma_2 = 1.$$

La moltiplicazione è dunque una operazione sempre ben condizionata.

- **Divisione**: $f(\mathbf{x}) = x_1/x_2$.

$$\begin{aligned} E_{in} &= \frac{(\tilde{x}_1/\tilde{x}_2) - (x_1/x_2)}{x_1/x_2} = \frac{((x_1(1+\epsilon_1))/(x_2(1+\epsilon_2))) - (x_1/x_2)}{x_1/x_2} \\ &= \frac{1+\epsilon_1}{1+\epsilon_2} - 1 = \frac{\epsilon_1 - \epsilon_2}{1+\epsilon_2}\frac{1-\epsilon_2}{1-\epsilon_2} = \frac{\epsilon_1 - \epsilon_2 - \epsilon_1\epsilon_2 + \epsilon_2^2}{1-\epsilon_2^2} \approx \gamma_1\epsilon_1 + \gamma_2\epsilon_2, \end{aligned}$$

con

$$\gamma_1 = 1, \qquad \gamma_2 = -1.$$

La divisione è dunque una operazione sempre ben condizionata.

Capitolo 2

Equazioni non lineari

Il problema che ci accingiamo ad affrontare è la ricerca di soluzioni di equazioni non lineari, ovvero:

Problema: Data una funzione $f : [a, b] \subseteq \mathbb{R} \to \mathbb{R}$, trovare $x \in [a, b]$, se esiste, tale che $f(x) = 0$.

Introdurremo ed esamineremo alcuni metodi classici. Alcuni di essi, come il metodo di Newton si generalizzano, almeno formalmente, senza difficoltà al caso pluridimensionale; altri, come il primo metodo che considereremo, ossia il metodo di bisezione, sono essenzialmente *ad hoc* per il caso unidimensionale.

2.1 Metodo di bisezione

In questo metodo ad ogni passo si costruisce un intervallo contenente uno zero di $f(x)$ e si assume come approssimazione di tale zero l'ascissa del punto medio del detto intervallo.

Assumiamo che f sia continua in $[a, b]$ e sia tale che $f(a) \cdot f(b) < 0$. Allora, per il teorema degli zeri per funzioni continue, sappiamo che esiste almeno un punto $x^* \in (a, b)$ tale che $f(x) = 0$. Per semplicità, supporremo che (a, b) contenga una radice semplice di f.

Poniamo $x_0 = a$, $x_1 = b$. Il numero

$$x_2 = \frac{x_0 + x_1}{2},$$

cioè l'ascissa del punto medio di (x_0, x_1), sarà certamente una approssimazione di x^* migliore di almeno una delle precedenti (i.e., x_0 e x_1). Se si verifica

$f(x_2) = 0$ abbiamo trovato la radice. Altrimenti si confronta il segno di $f(x_2)$ con quello di $f(x_1)$; se risulta $f(x_2)\,f(x_1) < 0$, allora $x^* \in (x_2, x_1)$, in caso contrario sarà $x^* \in (x_0, x_2)$. Quindi la nuova approssimazione x_3 sarà data da

$$x_3 = \begin{cases} \dfrac{x_2 + x_1}{2} & \text{se } f(x_2)\,f(x_1) < 0, \\ \dfrac{x_2 + x_0}{2} & \text{altrimenti.} \end{cases}$$

Indicando con $\hat{x}_2$ una variabile che può assumere i valori x_1 o x_0, possiamo unificare i due casi nella sola formula:

$$x_3 = \frac{x_2 + \hat{x}_2}{2} \qquad \text{dove } \hat{x}_2 = \begin{cases} x_1 & \text{se } f(x_2)\,f(x_1) < 0, \\ x_0 & \text{altrimenti.} \end{cases}$$

Ripetendo il ragionamento si determinano x_4, x_5, ... secondo la formula generale

$$x_{n+1} = \frac{x_n + \hat{x}_n}{2}, \qquad n = 1, 2, 3, \ldots,$$

dove per $n = 1$, $\hat{x}_1 = x_0$, mentre per $n > 1$ si pone

$$\hat{x}_n = \begin{cases} x_{n-1} & \text{se } f(x_n)\,f(x_{n-1}) < 0, \\ \hat{x}_{n-1} & \text{altrimenti.} \end{cases}$$

Poiché ad ogni passo l'intervallo contenente x^* è dimezzato, dopo n passi si ha una approssimazione x_{n+1} tale che

$$|x_{n+1} - x^*| \le \frac{b-a}{2^n}$$

da cui segue che

$$\lim_{n \to +\infty} |x_{n+1} - x^*| = 0$$

ovvero la convergenza del metodo alla radice x^*. Tale espressione fornisce anche una maggiorazione a priori dell'errore assoluto presente nell'iterata x_{n+1}. Inoltre suggerisce come **criterio di arresto** la condizione

$$\frac{b-a}{2^n} < \varepsilon$$

con $\varepsilon > 0$ numero prefissato. La stessa condizione permette di conoscere a priori il numero n di iterazioni necessario per ridurre il modulo dell'errore assoluto al di sotto di ε. Infatti

$$\frac{b-a}{2^n} < \varepsilon \quad \Rightarrow \quad \frac{b-a}{\varepsilon} < 2^n \quad \Rightarrow \quad n > \log_2 \left(\frac{b-a}{\varepsilon} \right).$$

Sia $e_n = |x_n - x^*|$ il modulo dell'errore assoluto che si commette approssimando la radice x^* mediante il valore x_n fornito da un metodo numerico al passo n.

Definizione 2.1 *Diremo che il metodo numerico è convergente se*

$$\lim_{n\to+\infty} e_n = 0.$$

Definizione 2.2 *Sia $\{x_n\}$ una successione generata da un metodo numerico e convergente a x^*. Se esiste un numero reale $p > 0$ e una costante $C \neq 0$ tale che*

$$\lim_{n\to+\infty} = \frac{e_{n+1}}{e_n^p} = C < \infty,$$

allora di dice che il metodo numerico ha ordine di convergenza p *con* costante asintotica dell'errore *pari a* C.

Considerato che per il metodo di bisezione si ha

$$|x_{n+1} - x^*| \simeq \frac{1}{2}|x_n - x^*|,$$

ne segue che

$$\frac{e_{n+1}}{e_n} \simeq \frac{1}{2}.$$

Pertanto, il metodo di bisezione ha ordine di convergenza $p = 1$ con costante asintotica dell'errore pari a $C = \frac{1}{2}$.

Nel caso in cui $p = 1$ si parla di convergenza lineare; nel caso $p = 2$ si parla di convergenza quadratica, ecc.. Tuttavia, bisogna sottolineare che, in generale, l'ordine di convergenza p può assumere un valore non intero, sebbene si debba necessariamente avere $p \geq 1$ affinché il metodo sia convergente.

Osservazione 2.1 *Nel caso di convergenza lineare, per n sufficientemente grande, si ha*

$$e_{n+1} \approx Ce_n,$$

ovvero

$$e_{n+k} \approx C^k e_n.$$

Pertanto il metodo è convergente se e solo se $0 \leq C < 1$.

In generale, tanto più elevato è l'ordine di convergenza di un metodo, tanto più velocemente le approssimazioni da questo generate convergono verso la radice x^*. Ad esempio, si supponga di avere due metodi convergenti alla stessa radice x^*, l'uno di ordine $p = 1$ e l'altro di ordine $p = 2$ ed aventi la stessa costante asintotica dell'errore $C = 0.2$. Si supponga inoltre che in entrambi i casi $|e_0| = 0.1$ e l'approssimazione iniziale x_0 sia sufficientemente vicina a x^*. Si ottiene, pertanto, la seguente tabella per gli errori e_n :

n	$p = 1$	$p = 2$
0	10^{-1}	10^{-1}
1	$2 \cdot 10^{-2}$	$2 \cdot 10^{-3}$
2	$4 \cdot 10^{-3}$	$8 \cdot 10^{-7}$
3	$8 \cdot 10^{-4}$	$128 \cdot 10^{-15}$
4	$16 \cdot 10^{-5}$	$32768 \cdot 10^{-31}$

Siccome la convergenza del metodo di bisezione è lenta, di solito questo metodo è usato per ottenere una prima approssimazione che consente l'uso di altri metodi più efficienti che andiamo a definire nel paragrafo successivo.

2.2 Metodi delle corde, delle secanti e delle tangenti (o di Newton)

Partendo da una approssimazione iniziale x_0 graficamente possiamo pensare di generare i valori successivi $x_1, x_2, \ldots, x_n$ conducendo dal punto iniziale (x_0, y_0), $y_0 = f(x_0)$, sulla curva $y = f(x)$ una retta con pendenza k_0, e prendendo come nuova (e migliore) approssimazione x_1 l'intersezione di questa retta con l'asse x. Ripartiamo poi dal nuovo punto (x_1, y_1), $y_1 = f(x_1)$ con una seconda retta con pendenza k_1 e determiniamo l'intersezione x_2 di quest'ultima con l'asse x. In altri termini, ad ogni passo linearizziamo localmente il problema iniziale e come approssimazione della radice x^* prendiamo la radice dell'equazione lineare

$$y_n + k_n(x - x_n) = 0, \qquad n = 0, 1, 2, \ldots$$

cioè

$$x_{n+1} = x_n - \frac{f(x_n)}{k_n}, \qquad n = 0, 1, 2, \ldots.$$

Le direzioni k_n possono essere scelte in molti modi. Vediamo alcuni esempi.

- *metodo delle corde*: $k_n = f'(x_0)$ (costante).
  Questo metodo è definito partendo dal presupposto che la derivata

prima della funzione f vari poco in prossimità della radice x^*. Il costo per iterazione è una valutazione della f, ovvero analogo al metodo di bisezione. Al pari di quest'ultimo si dimostra che anch'esso ha ordine di convergenza lineare.

- *metodo delle secanti*: $k_n = \dfrac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}$.
  Geometricamente si verifica che x_{n+1} costituisce l'intersezione della retta secante il grafico nei punti $(x_{n-1}, f(x_{n-1}))$ e $(x_n, f(x_n))$ con l'asse delle ascisse. Si dimostra che il suo ordine di convergenza è
  $$p = \frac{1+\sqrt{5}}{2},$$
  nel caso di radici semplici, mentre è lineare nel caso di radici multiple. Il corrispondente costo per iterazione è dato da una sola valutazione della funzione f ed è quindi analogo a quello del metodo di bisezione e del metodo delle corde, sebbene abbia un ordine di convergenza più elevato di questi ultimi.

- *metodo delle tangenti (o metodo di Newton)*: $k_n = f'(x_n)$.
  Nel caso particolare in cui f sia lineare, ad esempio $f(x) = cx + d$, il metodo di Newton converge in una sola iterazione, ovvero fornisce la soluzione in un solo passo.
  A differenza del metodo di bisezione, che ad ogni passo richiede solo una valutazione della funzione f, il metodo di Newton richiede anche una valutazione della sua derivata prima f'. Inoltre, il metodo di bisezione assumeva la sola continuità della funzione, laddove il metodo di Newton ne richiede anche la derivabilità. Tuttavia, questo costo per iterazione più elevato ed i maggiori requisiti sulla regolarità della funzione sono ripagati dal più elevato ordine di convergenza del metodo di Newton. Vale, infatti, il seguente risultato.

Teorema 2.1 *Sia $f \in C^2(I)$, dove I è un intervallo contenente una radice semplice x^* di f. Allora il metodo di Newton converge almeno quadraticamente verso x^*.*

Dim. Per ipotesi si ha che per un opportuno ξ_n compreso tra x^* e x_n :
$$f(x^*) = f(x_n) + (x^* - x_n)f'(x_n) + \frac{1}{2}(x^* - x_n)^2 f''(\xi_n).$$
Considerato che $f(x^*) = 0$, dalla precedente si ottiene:
$$0 = f'(x_n)\left(\frac{f(x_n)}{f'(x_n)} - x_n + x^*\right) + \frac{1}{2}f''(\xi_n)(x^* - x_n)^2.$$

Tenendo conto del fatto che

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

che è ben definita in un intorno di x^* essendo, per ipotesi, $f'(x^*) \neq 0$ e posto $e_n = |x_n - x^*|$, si ottiene

$$\frac{e_{n+1}}{e_n^2} = \frac{1}{2}\frac{|f''(\xi_n)|}{|f'(x_n)|}.$$

Se il metodo è convergente segue che

$$\lim_{n\to+\infty} \frac{e_{n+1}}{e_n^2} = \frac{1}{2}\frac{|f''(x^*)|}{|f'(x^*)|}$$

e quindi, se $f''(x^*) \neq 0$, il metodo ha ordine di convergenza $p = 2$. Altrimenti, l'ordine è $p > 2$. □

Teorema 2.2 (di convergenza globale) *Sia $f \in C^2([a,b])$. Se*

1) $f(a)\,f(b) < 0$;

2) $f'(x) \neq 0$, $\forall\, x \in [a,b]$;

3) $f''(x) \geq 0$ *oppure* $f''(x) \leq 0$, $\forall\, x \in [a,b]$;

4) $\left|\dfrac{f(a)}{f'(a)}\right| < b-a; \quad \left|\dfrac{f(b)}{f'(b)}\right| < b-a.$

Allora il metodo di Newton converge all'unica soluzione $x^* \in [a,b]$ qualunque sia la scelta di $x_0 \in [a,b]$.

Definizione 2.3 *Una radice x^* dell'equazione $f(x) = 0$ ha molteplicità esatta $m \geq 1$ se*

$$f(x^*) = f'(x^*) = \cdots = f^{(m-1)}(x^*) = 0, \quad f^{(m)}(x^*) \neq 0.$$

Se $m = 1$ la radice si dice semplice; si dice invece multipla se $m \geq 2$.

Teorema 2.3 *Se f è sufficientemente regolare, il metodo di Newton converge linearmente verso una radice di molteplicità $r > 1$ con costante asintotica d'errore pari a $\frac{r-1}{r}$.*

Dim. Per ipotesi si sa che $f(x^*) = f'(x^*) = \cdots = f^{(r-1)}(x^*) = 0$ e $f^{(r)}(x^*) \neq 0$. Sviluppando in serie di Taylor sia f sia f' in un intorno di x^* si ottiene:

$$\begin{aligned} f(x_n) &= \sum_{j=0}^{r-1} \frac{(x_n - x^*)^j}{j!} f^{(j)}(x^*) + \frac{(x_n - x^*)^r}{r!} f^{(r)}(\xi_n) = \frac{(x_n - x^*)^r}{r!} f^{(r)}(\xi_n), \\ f'(x_n) &= \sum_{j=0}^{r-2} \frac{(x_n - x^*)^j}{j!} f^{(j+1)}(x^*) + \frac{(x_n - x^*)^{r-1}}{(r-1)!} f^{(r)}(\eta_n) \\ &= \frac{(x_n - x^*)^{r-1}}{(r-1)!} f^{(r)}(\eta_n). \end{aligned}$$

Dalla ricorrenza che definisce il metodo di Newton e utilizzando le uguaglianze appena ottenute, segue che

$$\begin{aligned} x_{n+1} - x^* &= x_n - \frac{f(x_n)}{f'(x_n)} - x^* \\ &= x_n - \frac{(x_n - x^*)^r f^{(r)}(\xi_n)}{r!} \frac{(r-1)!}{(x_n - x^*)^{r-1} f^{(r)}(\eta_n)} - x^* \\ &= (x_n - x^*) \left(1 - \frac{1}{r} \frac{f^{(r)}(\xi_n)}{f^{(r)}(\eta_n)}\right), \end{aligned}$$

da cui si ha:

$$\frac{x_{n+1} - x^*}{x_n - x^*} = 1 - \frac{1}{r} \frac{f^{(r)}(\xi_n)}{f^{(r)}(\eta_n)}.$$

Pertanto, considerato che

$$\lim_{n \to +\infty} f^{(r)}(\xi_n) = f^{(r)}(x^*), \quad \lim_{n \to +\infty} f^{(r)}(\eta_n) = f^{(r)}(x^*),$$

se il metodo è convergente, si ottiene

$$\lim_{n \to +\infty} \frac{e_{n+1}}{e_n} = 1 - \frac{1}{r}. \ \square$$

2.2.1 Metodo di Newton e radici multiple

Nel caso di radici multiple, abbiamo visto che la convergenza del metodo di Newton risulta essere solo lineare. È possibile, tuttavia, ripristinare la convergenza la convergenza quadratica modificando opportunamente il metodo. Si distinguono, a riguardo, due casi:

- **la molteplicità della radice è nota**.
  Per esaminare questo caso supponiamo, per semplicità che

$$f(x) = (x - x^*)^r.$$

  Applicando il metodo di Newton per determinare la radice si ottiene,

$$x_{n+1} = x_n - \frac{1}{r}\frac{(x_n - x^*)^r}{(x_n - x^*)^{r-1}}, \quad n = 0, 1, 2, \ldots,$$

  ovvero

$$x_{n+1} = x_n - \frac{1}{r}(x_n - x^*), \quad n = 0, 1, 2, \ldots.$$

  È pertanto evidente che, conoscendo la molteplicità r della radice possiamo ottenere la soluzione esatta in un solo passo moltiplicando per r il termine di *correzione* ad x_n. Più in generale, si può dimostrare che lo schema iterativo

$$x_{n+1} = x_n - r\frac{f(x_n)}{f'(x_n)}, \quad n = 0, 1, 2, \ldots.$$

  ripristina la convergenza quadratica del metodo di Newton, qualora questo converga verso una radice di molteplicità esatta r.

- **la molteplicità della radice è incognita**.
  In questo caso, sapendo che la convergenza del metodo di Newton è, nel caso di radici multiple, solo lineare, segue che al generico passo n-simo,

$$e_n \approx Ce_{n-1}, \qquad e_{n+1} \approx Ce_n,$$

  dove C è la costante asintotica (incognita) dell'errore. Combinando queste due relazioni è possibile eliminare C ottenendo

$$\frac{e_n}{e_{n-1}} \approx \frac{e_{n+1}}{e_n},$$

  ovvero

$$e_{n+1}e_{n-1} \approx e_n^2.$$

  Supponendo, senza perdita di generalità, che $x_{n+1} > x^*$ e $x_{n-1} > x^*$

$$e_{n+1}e_{n-1} \approx e_n^2, \qquad \Leftrightarrow \qquad (x_{n+1} - x^*)(x_{n-1} - x^*) \approx (x_n - x^*)^2.$$

  Si perviene infine alla seguente approssimazione della radice,

$$x^* \approx x_n^* \equiv \frac{x_{n+1}x_{n-1} - x_n^2}{x_{n+1} - 2x_n + x_{n-1}}. \tag{2.1}$$

  L'intera procedura è quindi reiterata a partire da x_n^*. Si ottiene, pertanto, una procedura a due livelli:

1. al livello interno vengono eseguiti due passi del metodo di Newton,
2. al livello esterno viene eseguito il passo (2.1) di *accelerazione* che consente di estrapolare una approssimazione più accurata della radice. Quest'ultima fornirà, a sua volta, il nuovo punto iniziale per il livello interno.

Questa procedura è nota come *metodo di accelerazione di Aitken*. Si può dimostrare che la successione delle approssimazioni ottenute mediante accelerazione converge quadraticamente verso la radice x^*. Pertanto, al prezzo di avere una iterazione a due livelli, con un costo doppio per iterata rispetto al metodo di Newton, siamo in grado di ripristinare la convergenza quadratica di tale metodo anche nel caso di radici multiple con molteplicità incognita.

2.3 Metodo di iterazione funzionale

Nelle applicazioni, spesso una equazione non lineare si presenta nella forma di un *problema di punto fisso*:

$$\text{Trovare } x \text{ tale che } x = g(x). \tag{2.2}$$

Un numero x^* che soddisfa tale equazione è detto *punto fisso* di g. Se x_0 è una approssimazione iniziale del punto fisso x^* di (2.2) l'*iterazione di punto fisso* genera una successione approssimante nel modo seguente:

$$x_{n+1} = g(x_n), \qquad n = 0, 1, 2, \ldots.$$

La funzione g è detta *funzione di iterazione* ed è definita a partire dalla funzione f e dalle derivate di f.

Data una equazione $f(x) = 0$ esistono diverse funzioni g tali che un punto fisso di g sia una soluzione dell'equazione. A scopo illustrativo si consideri il seguente esempio.

Esempio 2.1 *Sia* $f(x) = x^2 - x - 2$. *Allora*

$g(x) = x^2 - 2$

$g(x) = \sqrt{x + 2}$

$g(x) = 1 + \frac{2}{x}$

$g(x) = x - \frac{x^2 - x - 2}{m}, \quad m \neq 0.$

Alcuni metodi analizzati in precedenza (bisezione, corde, Newton) possono essere considerati come casi particolari di iterazioni di punto fisso. Il metodo delle secanti, invece, NON rientra in questa categoria in quanto per ogni n, x_{n+1} dipende non solo da x_n, ma anche da x_{n-1}. In generale, se il procedimento iterativo per la ricerca del punto fisso di una trasformazione è dato da

$$x_{n+1} = g(x_n, x_{n-1}, \ldots, x_{n-r+1})$$

anche il metodo delle secanti vi rientra!

Teorema 2.4 (convergenza locale) *Sia x^* un punto fisso di g interno ad un intervallo I sul quale g sia derivabile con continuità. Si supponga che esistano due numeri positivi γ e K, $0 < K < 1$, tali che $\forall x \in [x^* - \gamma, x^* + \gamma] \subset I$ si verifichi la condizione*

$$|g'(x)| \leq K.$$

Allora per il metodo

$$x_{n+1} = g(x_n)$$

valgono le seguenti proposizioni:

1. *se $x_0 \in (x^* - \gamma, x^* + \gamma)$ allora è anche $x_n \in (x^* - \gamma, x^* + \gamma)$ per $n = 1, 2, \ldots$;*
2. *per la successione $\{x_n\}$, con $x_0 \in (x^* - \gamma, x^* + \gamma)$ si ha $\lim_{n \to +\infty} x_n = x^*$;*
3. *x^* è l'unico punto fisso di g in $[x^* - \gamma, x^* + \gamma]$;*
4. *l'errore alla n-esima iterata verifica la relazione*

$$|x_n - x^*| \leq \frac{K}{1 - K} |x_n - x_{n-1}|, \quad n > 1.$$

Dim. La proposizione 1 si dimostra per induzione. Scelto un $x_0 \in (x^* - \gamma, x^* + \gamma)$ si ammette per ipotesi che sia per un certo $x_n \in (x^* - \gamma, x^* + \gamma)$ o, equivalentemente, $|x_n - x^*| < \gamma$ e si deduce che deve essere $x_{n+1} \in (x^* - \gamma, x^* + \gamma)$ ovvero $|x_{n+1} - x^*| < \gamma$. Infatti, considerato che $x_{n+1} = g(x_n)$ e dal teorema del valor medio si ha:

$$x_{n+1} - x^* = g(x_n) - x^* = g'(\xi)(x_n - x^*),$$

dove ξ è compreso tra x_n e x^*. Dall'ipotesi fatta su x_n e da quelle del teorema segue poi

$$|x_{n+1} - x^*| = |g'(\xi)||x_n - x^*| \leq K|x_n - x^*| < \gamma.$$

La proposizione 2 segue dall'ipotesi $0 < K < 1$ e dalla disuguaglianza

$$|x_{n+1} - x^*| \leq K^{n+1}|x_0 - x^*|$$

che si ottiene dalla precedente disuguaglianza.
La proposizione 3 si dimostra per assurdo. Infatti se in $(x^* - \gamma, x^* + \gamma)$ esistesse un altro punto fisso $x^\star \neq x^*$ si avrebbe

$$|x^\star - x^*| = |g(x^\star) - g(x^*)| = |g'(\xi)||x^\star - x^*| \leq K|x^\star - x^*| < |x^\star - x^*|.$$

Infine, per dimostrare la proposizione 4 osserviamo che (vedi teorema del valor medio):

$$|x_{n+1} - x_n| = |g(x_n) - g(x_{n-1})| \leq K|x_n - x_{n-1}|. \tag{2.3}$$

D'altra parte, ricordando che $|u + v| \leq |u| + |v|$,

$$|x_n - x^*| \leq |x_n - x_{n+1}| + |x_{n+1} - x^*| \leq |x_n - x_{n+1}| + K|x_n - x^*|,$$

da cui

$$(1 - K)|x_n - x^*| \leq |x_n - x_{n+1}|$$

che, unita alla (2.3), prova la tesi. □
Nel caso di metodi iterativi di punto fisso stazionari (funzione di iterazione non dipende da n) ad un punto è possibile individuare il corrispondente ordine di convergenza tenendo conto del seguente teorema.

Teorema 2.5 *Un metodo iterativo ad un punto la cui funzione di iterazione g sia sufficientemente derivabile, ha ordine di convergenza uguale ad un numero intero positivo p. Precisamente se il metodo*

$$x_{n+1} = g(x_n)$$

converge a x^*, la convergenza è di ordine p allora e solo che si abbia

$$g(x^*) = x^*, g^{(i)}(x^*) = 0, \text{ per } 1 \leq i < p, \ g^{(p)}(x^*) \neq 0.$$

Dim. Dalla formula di Taylor:

$$x_{n+1} - x^* = g(x_n) - x^* = \sum_{j=1}^{p-1} \frac{(x_n - x^*)^j}{j!} g^{(j)}(x^*) + \frac{(x_n - x^*)^p}{p!} g^{(p)}(\xi)$$

dove ξ è compreso tra x_n e x^*. Allora, per l'ipotesi fatta si ha

$$\lim_{n \to +\infty} \frac{|x_{n+1} - x^*|}{|x_n - x^*|^p} = \frac{g^{(p)}(x^*)}{p!},$$

cioè l'ordine di convergenza è p ed il fattore di convergenza è $C = \frac{g^{(p)}(x^*)}{p!}$. Viceversa, se l'ordine è p sia $g^{(i)}(x^*)$ la prima derivata non nulla nel precedente sviluppo di Taylor intorno al punto x^*; se fosse $i \neq p$ per il ragionamento diretto anche l'ordine sarebbe $i \neq p$ contro l'ipotesi. Allora deve essere $i = p$.
□

2.4 Criteri di arresto

Vorrei arrestare le iterazioni quando

$$|x_n - x^*| < \varepsilon,$$

con $\varepsilon > 0$ una tolleranza prefissata. Tuttavia, siccome x^* è una quantità incognita, devo usare un criterio basato su quantità che so calcolare. I criteri più usati sono

- il **criterio del residuo**: $|f(x_n)| < \varepsilon$
- il **criterio dell'incremento**: $|x_{n+1} - x_n| < \varepsilon$

Dal teorema del valor medio

$$f(x_n) = f(x^*) + f'(\xi_n)(x_n - x^*), \quad \xi_n \in (x_n, x^*)$$

allora

$$|f(x_n)| = |f'(\xi_n)||x_n - x^*|.$$

Allora, se uso come criterio di arresto il criterio del residuo

$$|x_n - x^*| = \frac{|f(x_n)|}{|f'(\xi_n)|} < \frac{\varepsilon}{|f'(\xi_n)|}$$

e quindi il controllo dell'errore assoluto dipende dal valore $f'(\xi_n)$. Pertanto, considerato che $f'(\xi_n) \to f'(x^*)$ per $n \to +\infty$ si possono verificare i seguenti tre casi:

1. se $|f'(x^*)| \simeq 1$ il criterio del residuo permette di controllare l'errore assoluto;
2. se $|f'(x^*)| \ll 1$ l'errore assoluto potrebbe essere molto grande;
3. se $|f'(x^*)| \gg 1$ il criterio del residuo potrebbe essere restrittivo.

Se la successione $\{x_n\}$ è ottenuta con il metodo di Newton (caso radice semplice) è meglio utilizzare il criterio dell'incremento. Infatti dal teorema del valor medio

$$x_{n+1} - x_n = -\frac{f(x_n)}{f'(x_n)} \simeq -\frac{f(x_n)}{f'(\xi_n)} = x_n - x^*.$$

In tal caso, in virtù della precedente relazione, il criterio dell'incremento permette di controllare l'errore assoluto.

Tuttavia, se la successione $\{x_n\}$ è generata da un metodo iterativo di punto fisso $x_{n+1} = g(x_n)$ la cui funzione di iterazione g sia tale per cui $g'(x^*) \simeq 1$ si ha

$$\begin{aligned} x_{n+1} - x_n &= x_{n+1} - x^* + x^* - x_n = g(x_n) - g(x^*) - (x_n - x^*) \\ &= g'(\xi_n)(x_n - x^*) - (x_n - x^*). \end{aligned}$$

Allora, essendo

$$|x_n - x^*| = \frac{|x_{n+1} - x_n|}{|g'(\xi_n) - 1|} < \frac{\varepsilon}{|g'(\xi_n) - 1|}$$

il criterio dell'incremento non permette di controllare l'errore assoluto.

Nel caso di convergenza solo lineare il criterio dell'incremento può essere modificato come segue. Considerato che $e_{n+1} \approx Ce_n$ si ottiene:

$$|x_n - x_{n+1}| = |(x_n - x^*) - (x_{n+1} - x^*)| \approx e_n - e_{n+1} \approx e_n(1 - C).$$

Pertanto,

$$e_{n+1} \approx Ce_n \approx \frac{C}{1 - C}|x_n - x_{n+1}|.$$

In tal caso risulta più appropriato usare il criterio:

$$|x_n - x_{n+1}| \leq \frac{1 - C}{C}\varepsilon.$$

Per quanto riguarda la costante asintotica dell'errore C, una sua stima può essere ottenuta considerando che

$$|x_0 - x_1| \approx (1 - C)e_0, \qquad |x_1 - x_2| \approx (1 - C)e_1 \approx (1 - C)Ce_0$$

da cui si ricava:

$$C \approx \frac{|x_1 - x_2|}{|x_0 - x_1|}.$$

Evidentemente tale stima richiede che siano eseguite almeno due iterazioni del metodo. Inoltre, essa può essere aggiornata dinamicamente durante il corso delle iterazioni, considerando le tre iterate più recenti.

Capitolo 3

Sistemi di Equazioni Lineari

In questo capitolo tratteremo la risoluzione di sistemi di equazioni lineari:

$$A\,\mathbf{x} = \mathbf{b}, \qquad A \in \mathbb{C}^{m\times n},\ \mathbf{b} \in \mathbb{C}^{m}, \tag{3.1}$$

con $m \geq n$ e $\text{rank}(A) = n$ (A è una matrice avente rango massimo).
In particolare, nel caso in cui $m = n$, ovvero A è una matrice quadrata, avendosi $\text{rank}(A) = n$, segue che A è una matrice nonsingolare. Pertanto la soluzione del sistema lineare esiste ed è unica:

$$\mathbf{x} = A^{-1}\,\mathbf{b}.$$

Tuttavia, questa espressione formale della soluzione non induce, generalmente, un metodo di risoluzione efficiente, in quanto il costo richiesto risulta essere assai più elevato dei metodi numerici che studieremo nelle prossime sezioni. Tali metodi di risoluzione possono essere suddivisi in due classi: *metodi diretti* e *metodi iterativi*. In assenza di errori di rappresentazione dei dati e di arrotondamento, un metodo diretto calcola la soluzione esattamente in un numero finito di passi indipendentemente dalla precisione richiesta dall'utente, un metodo iterativo, se convergente, fornisce sempre una soluzione approssimata ottenuta bloccando il procedimento quando la precisione desiderata è stata ottenuta, commettendo un errore (errore analitico o di troncamento).

Se A è densa un metodo diretto è di solito più efficiente. Se A è sparsa e di ordine elevato è più indicato l'utilizzo di un metodo iterativo. Infatti, considerando che i metodi diretti trasformano la matrice:

- ▷ se l'ordine della matrice è elevato ci potrebbero essere problemi di saturazione dello spazio disponibile nella memoria centrale del calcolatore;
- ▷ se la matrice è sparsa nelle matrici intermedie si possono generare elementi diversi da zero in corrispondenza ad elementi nulli della matrice iniziale. Tale fenomeno si chiama *fill-in*.

3.1 Analisi dell'errore

Qualunque metodo si usi per la risoluzione di un sistema lineare non si può prescindere dagli errori di rappresentazione dei dati (errore inerente) e di arrotondamento nei calcoli (errore algoritmico). Una maggiorazione dell'errore da cui è affetta la soluzione effettivamente calcolata può essere rappresentata, a meno di termini di ordine superiore, da due termini distinti, uno dovuto agli errore di rappresentazione dei dati, che non dipende dal particolare metodo usato (errore inerente) e l'altro dovuto agli errori di arrotondamento nei calcoli (errore algoritmico).

Lo studio dell'errore inerente può essere fatto perturbando i dati (la matrice A e/o il vettore $\mathbf{b}$) ed esaminando gli effetti indotti da queste perturbazioni sulla soluzione $\mathbf{x}$. Dato un sistema di equazioni lineari:

$$A\,\mathbf{x} = \mathbf{b}, \qquad A \in \mathbb{C}^{n\times n},\ \mathbf{b} \neq \mathbf{0} \tag{3.2}$$

ci chiediamo come le perturbazioni sui dati del problema (ovvero A e $\mathbf{b}$) si ripercuotano sulla soluzione. Siano $\delta A \in \mathbb{C}^{n\times n}$ e $\delta\mathbf{b} \in \mathbb{C}^n$ rispettivamente la matrice ed il vettore delle perturbazioni sui dati del sistema (3.2). Si vuole determinare $\delta\mathbf{x} \in \mathbb{C}^n$, perturbazione sul risultato. Studieremo quindi il sistema lineare perturbato:

$$(A + \delta A)(\mathbf{x} + \delta\mathbf{x}) = \mathbf{b} + \delta\mathbf{b}. \tag{3.3}$$

Se A è non singolare e se, per una qualunque norma matriciale indotta $\|A^{-1}\|\|\delta A\| < 1$, allora anche la matrice $(A + \delta A)$ è non singolare. Infatti

$$A + \delta A = A(I + A^{-1}\delta A)$$

ma

$$\|A^{-1}\delta A\| \leq \|A^{-1}\|\|\delta A\| < 1$$

e quindi $I + A^{-1}\delta A$ è non singolare ed essendo anche A non singolare ne segue che $A + \delta A$ è non singolare.

Sottraendo membro a membro (3.2) da (3.3) si ottiene

$$(A + \delta A)\delta\mathbf{x} = -(\delta A)\mathbf{x} + \delta\mathbf{b}.$$

Moltiplicando ambo i membri per A^{-1} si ha

$$(I + A^{-1}\delta A)\delta\mathbf{x} = A^{-1}(-(\delta A)\mathbf{x} + \delta\mathbf{b}).$$

da cui

$$\delta\mathbf{x} = (I + A^{-1}\delta A)^{-1}A^{-1}(-(\delta A)\mathbf{x} + \delta\mathbf{b}).$$

Passando alle norme e utilizzando il risultato in Proposizione A.2 si ottiene

$$\|\delta\mathbf{x}\| \leq \frac{\|A^{-1}\|(\|(\delta A)\|\|\mathbf{x}\| + \|\delta\mathbf{b}\|)}{1 - \|A^{-1}\|\|\delta A\|}$$

Poiché per ipotesi $\mathbf{b} \neq \mathbf{0}$ e A è non singolare, risulta $\|\mathbf{x}\| > 0$ per cui posso dividere ambo i membri della relazione precedente:

$$\frac{\|\delta\mathbf{x}\|}{\|\mathbf{x}\|} \leq \frac{\|A^{-1}\|(\|(\delta A)\| + \|\delta\mathbf{b}\|/\|\mathbf{x}\|)}{1 - \|A^{-1}\|\|\delta A\|}.$$

Ora, tenendo conto che $\|A\mathbf{x}\| = \|\mathbf{b}\| \leq \|A\|\|\mathbf{x}\|$ si ha

$$\begin{aligned}\frac{\|\delta\mathbf{x}\|}{\|\mathbf{x}\|} &\leq \frac{\|A\|\|A^{-1}\|(\|(\delta A)\|/\|A\| + \|\delta\mathbf{b}\|/(\|A\|\|\mathbf{x}\|))}{1 - \|A^{-1}\|\|\delta A\|} \\ &\leq \frac{\|A\|\|A^{-1}\|}{1 - \|A^{-1}\|\|\delta A\|}\left(\frac{\|(\delta A)\|}{\|A\|} + \frac{\|\delta\mathbf{b}\|}{\|\mathbf{b}\|}\right)\end{aligned}$$

Introdotta la quantità

$$\kappa(A) = \|A\|\|A^{-1}\|,$$

detta *numero di condizionamento della matrice* A, la precedente diventa:

$$\left\|\frac{(\mathbf{x} + \delta\mathbf{x}) - \mathbf{x}}{\mathbf{x}}\right\| \leq \frac{\kappa(A)}{1 - \kappa(A)\frac{\|\delta A\|}{\|A\|}}\left(\frac{\|(A + \delta A) - A\|}{\|A\|} + \frac{\|(\mathbf{b} + \delta\mathbf{b}) - \mathbf{b}\|}{\|\mathbf{b}\|}\right)$$

Si osservi che il numero di condizionamento è sempre maggiore o uguale a 1; infatti

$$\kappa(A) = \|A\|\left\|A^{-1}\right\| \geq \left\|A\,A^{-1}\right\| = \|I\| = 1.$$

Pertanto si parlerà di problema ben posto se $\kappa(A) \approx 1$: in tal caso la matrice del sistema di dice *ben condizionata* e di problema mal posto quando $\kappa(A) \gg 1$: in tal caso la matrice del sistema di dice *mal condizionata*.

Osservazione 3.1 *In* `Matlab` *il numero di condizionamento di una matrice è realizzato con il comando* `cond`.

Per analizzare l'errore algoritmico viene usata la tecnica cosiddetta di *analisi all'indietro* (backward analysis), in cui la soluzione effettivamente calcolata $\mathbf{y}$ viene considerata come soluzione esatta di un problema perturbato del tipo

$$(A + \Delta A)\mathbf{y} = \mathbf{b} + \Delta\mathbf{b}.$$

A differenza dell'analisi fatta prima per l'errore inerente, ora abbiamo supposto che gli elementi di A e di $\mathbf{b}$ fossero numeri macchina e ΔA e $\Delta\mathbf{b}$ **non** sono

perturbazioni sui dati iniziali, ma sono legate agli errori commessi durante i calcoli e quindi alla precisione con cui vengono eseguite le operazioni. Un metodo risulta più stabile di un altro se è meno sensibile agli errori indotti dai calcoli. Si tenga presete che lo studio della stabilità di un metodo può perdere di significatività quando il problema è fortemente malcondizionato. In tal caso, l'errore inerente prevale sull'errore algoritmico.

3.2 Metodi diretti

Talvolta il sistema lineare (3.1) possiede una matrice dei coefficienti con particolari strutture e/o proprietà che ne rendono assai semplice la risoluzione. I casi che approfondiremo sono quelli in cui la matrice è *diagonale, triangolare, ortogonale*. Essi serviranno, successivamente, per definire opportuni *metodi di fattorizzazione* che consentiranno di risolvere il problema generale.

3.2.1 Matrici diagonali

In questo caso la matrice A ha tutti gli elementi al di fuori della diagonale principale nulli. Pertanto, si avrà:

$$A = \begin{pmatrix} a_{11} & & \\ & \ddots & \\ & & a_{nn} \end{pmatrix}$$

e quindi il sistema lineare (3.1) assume la forma:

$$\begin{cases} a_{11}x_1 &= b_1 \\ & \vdots \\ a_{nn}x_n &= b_n \end{cases}.$$

Ne segue che le componenti della soluzione possono essere facilmente ottenute come

$$x_i = \frac{b_i}{a_{ii}}, \quad i = 1, 2, \ldots, n.$$

Osserviamo che:

- A è nonsingolare, ovvero $a_{ii} \neq 0, i = 1, 2, \ldots, n$ e, pertanto, le operazioni precedenti sono ben definite;
- il costo computazionale è di n divisioni;
- il costo in termini di occupazione di memoria è lineare in quanto gli elementi diagonali di A possono essere memorizzati in un vettore di lunghezza n.

3.2.2 Matrici triangolari

In questo caso gli elementi significativi di A si trovano in una porzione *triangolare* della matrice. Si distinguono due casi:

- A triangolare *inferiore*,

$$A = \begin{pmatrix} a_{11} & & \\ \vdots & \ddots & \\ a_{n1} & \dots & a_{nn} \end{pmatrix};$$

- A triangolare *superiore*,

$$A = \begin{pmatrix} a_{11} & \dots & a_{1n} \\ & \ddots & \vdots \\ & & a_{nn} \end{pmatrix}.$$

Nel caso in cui A sia tringolare inferiore il sistema lineare (3.1) assume la forma:

$$\begin{cases} a_{11}x_1 & = & b_1 \\ a_{21}x_1 + a_{22}x_2 & = & b_2 \\ & \vdots & \\ a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n & = & b_n \end{cases}$$

e quindi gli elementi della soluzione possono essere ottenuti mediante *sostituzioni successive in avanti*,

$$\begin{cases} x_1 & = & b_1/a_{11} \\ x_2 & = & \left(b_2 - a_{21}x_1\right)/a_{22} \\ & \vdots & \\ x_n & = & \left(b_n - \sum_{j=1}^{n-1} a_{nj}x_j\right)/a_{nn} \end{cases}.$$

Osserviamo che:

- A è nonsingolare, ovvero $a_{ii} \neq 0, i = 1, 2, \dots, n$, e, pertanto, le operazioni precedenti sono ben definite;
- per quanto riguarda il costo computazionale si deve tener presente che
  - ▷ per calcolare x_1 si deve fare 1 divisione;

▷ per calcolare ciascun $x_i, i = 2, 3, \ldots, n$, si devono fare $(i-1)$ moltiplicazioni e 1 divisione.

Pertanto il numero di operazioni (moltiplicazioni e divisioni) da fare è:

$$\sum_{i=1}^{n}(i-1)+n = \sum_{r=1}^{n-1} r + n = \frac{n(n-1)}{2} + n \approx \frac{n^2}{2};$$

- A è triangolare e quindi deve essere memorizzata solo la porzione della matrice con elementi significativi per un totale di

$$\sum_{i=1}^{n} i = \frac{n(n+1)}{2} \approx \frac{n^2}{2}$$

posizioni di memoria.

Nel caso in cui la matrice A sia triangolare superiore il sistema lineare (3.1) assume la forma:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1,n-1}x_{n-1} + a_{1n}x_n &= b_1 \\ a_{22}x_2 + \cdots + a_{2,n-1}x_{n-1} + a_{2n}x_n &= b_2 \\ \vdots & \\ a_{n-1,n-1}x_{n-1} + a_{n-1,n}x_n &= b_{n-1} \\ a_{nn}x_n &= b_n \end{cases}$$

e quindi gli elementi della soluzione possono essere ottenuti mediante *sostituzioni successive all'indietro*,

$$\begin{cases} x_n &= b_n/a_{nn} \\ x_{n-1} &= \left(b_{n-1} - a_{n-1,n}x_n\right)/a_{n-1,n-1} \\ \vdots & \\ x_1 &= \left(b_1 - \sum_{j=2}^{n} a_{1j}x_j\right)/a_{11} \end{cases}.$$

Considerazioni analoghe a quelle fatte per il caso triangolare inferiore valgono riguardo alla buona definizione delle operazioni richieste, al costo computazionale e all'occupazione di memoria.

3.2.3 Matrici unitarie

Se la matrice dei coefficienti A in (3.1) è una matrice unitaria allora $A^{-1} = A^H$. Pertanto la soluzione si trova immediatamente come

$$\mathbf{x} = A^H\,\mathbf{b},$$

ovvero al costo di un prodotto matrice-vettore. Pertanto il corrispondente costo computazionale è $\approx n^2$ moltiplicazioni e $\approx n^2$ posizioni di memoria.

3.2.4 Metodi di fattorizzazione

Il fatto che i precedenti casi si possano trattare in modo efficiente e relativamente semplice ha indotto la derivazione, per il problema generale (3.1), di opportuni *metodi di fattorizzazione*. L'idea di base è quella di ottenere una fattorizzazione della matrice dei coefficienti

$$A = F_1 F_2 \cdots F_k, \tag{3.4}$$

per un opportuno (ed in verità assai piccolo) valore di k, dove i fattori

$$F_1, F_2, \ldots, F_k \in \mathbb{C}^{n\times n}$$

sono matrici nonsingolari appartenenti ad una delle categorie precedentemente esaminate. Tenendo conto della fattorizzazione (3.4), la soluzione del sistema (3.1) può essere calcolata risolvendo (nell'ordine specificato), i seguenti sistemilineari:

$$F_1\mathbf{x}_1 = \mathbf{b}, \quad F_2\mathbf{x}_2 = \mathbf{x}_1, \quad \ldots \quad F_k\mathbf{x} = \mathbf{x}_{k-1}.$$

Tre fattorizzazioni classiche sono le seguenti:

- *la fattorizzazione* LU : L è una matrice triangolare inferiore a diagonale unitaria ed U è una matrice triangolare superiore. Tale fattorizzazione è associata al metodo di eliminazione di Gauss;
- *la fattorizzazione* LL^H : L è una matrice triangolare inferiore con elementi positivi sulla diagonale principale. Tale fattorizzazione è associata al metodo di Cholesky;
- *la fattorizzazione* QR : Q è una matrice unitaria ed R è una matrice triangolare superiore. Tale fattorizzazione è associata al metodo di Householder e al metodo di Givens.

La fattorizzazione QR esiste per ogni matrice A, mentre non sempre è possibile ottenere le fattorizzazioni LU ed LL^H.

3.3 Fattorizzazione LU di una matrice

Se la matrice A in (3.1) può essere scritta come il prodotto di due fattori

$$A = LU, \tag{3.5}$$

con L triangolare inferiore a diagonale unitaria e U triangolare superiore, allora essa si dice *fattorizzabile* LU. Vale il seguente risultato:

Teorema 3.1 (unicità della fattorizzazione LU) *Se la fattorizzazione (3.5) esiste e A è nonsingolare, allora essa è unica.*

Dim. Supponiamo che la fattorizzazione non sia unica, ovvero $A = LU = L_1U_1$. Allora si ha che

$$0 \neq \det(A) = \det(L_1U_1) = \det(L_1)\det(U_1) = \det(U_1).$$

Pertanto U_1 è nonsingolare e quindi

$$UU_1^{-1} = L^{-1}L_1 \equiv D.$$

Tuttavia, essendo UU_1^{-1} triangolare superiore e $L^{-1}L_1$ triangolare inferiore segue che D è diagonale. Inoltre, essendo la diagonale di $L^{-1}L_1$ unitaria, $D = I$. Quindi, immediatamente discende che $L_1 = L$ e $U_1 = U$. □

Resta ancora da stabilire l'esistenza della fattorizzazione LU. A tale scopo, consideriamo il seguente problema: dato un vettore

$$\mathbf{v} = (v_1, \ldots, v_n)^H \in \mathbb{C}^n,$$

si supponga di voler azzerare tutte le componenti dalla $(k+1)$-esima in poi lasciando, al contempo, invariate le prime k mediante moltiplicazione a sinistra per una matrice $\mathcal{L} \in \mathbb{C}^{n\times n}$ che sia triangolare inferiore e a diagonale unitaria. Se $v_k \neq 0$ è possibile definire il *vettore elementare di Gauss*,

$$\mathbf{g} = \frac{1}{v_k}(\underbrace{0, \ldots, 0}_{k}, v_{k+1}, \ldots, v_n)^H \tag{3.6}$$

e, successivamente, la *matrice elementare di Gauss*

$$\mathcal{L} = I - \mathbf{g}\mathbf{e}_k^H, \tag{3.7}$$

dove $\mathbf{e}_k$ denota il k-mo vettore della base canonica in $\mathbb{C}^n$, ovvero $\mathbf{e}_k = (\underbrace{0, \ldots, 0}_{k-1}, 1, 0, \ldots, 0)^H$. Si ha che

$$\mathcal{L} = \begin{pmatrix} 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & -\frac{v_{k+1}}{v_k} & 1 & \\ & & \vdots & & \ddots \\ & & -\frac{v_n}{v_k} & & 1 \end{pmatrix}, \qquad \mathcal{L}\mathbf{v} = \begin{pmatrix} v_1 \\ \vdots \\ v_k \\ 0 \\ \vdots \\ 0 \end{pmatrix}.$$

Osserviamo che il vettore (3.6) e quindi la matrice (3.7) risultano essere definiti se e solo se $v_k \neq 0$. Inoltre, a causa della struttura della matrice $\mathcal{L}$, le cui prime k righe coincidono con quelle della matrice identità, le prime k componenti del vettore $\mathcal{L}\mathbf{v}$ coincidono, come richiesto, con quelle del vettore $\mathbf{v}$. Infine, l'inversa della matrice $\mathcal{L}$ si ottiene semplicemente come

$$\mathcal{L}^{-1} = I + \mathbf{g}\mathbf{e}_k^H. \tag{3.8}$$

Infatti,

$$\mathcal{L}^{-1}\mathcal{L} = \left(I + \mathbf{g}\mathbf{e}_k^H\right)\left(I - \mathbf{g}\mathbf{e}_k^H\right) = I - \mathbf{g}\mathbf{e}_k^H + \mathbf{g}\mathbf{e}_k^H - \mathbf{g}(\mathbf{e}_k^H\mathbf{g})\mathbf{e}_k^H = I$$

in quanto $\mathbf{e}_k^H\mathbf{g}$, ovvero la k-esima componente del vettore $\mathbf{g}$, è nulla (vedi (3.6)).

Ritorniamo ora al problema di fattorizzare una data matrice nonsingolare nella forma (3.5). Otterremo tale fattorizzazione, se possibile, utilizzando la procedura denominata **metodo di eliminazione di Gauss** che in $n-1$ passi, mediante moltiplicazione a sinistra di opportune matrici elementari di Gauss, porterà la matrice A a forma triangolare superiore. Più in dettaglio, definiamo

$$A \equiv A^{(1)} = \begin{pmatrix} a_{11}^{(1)} & \dots & a_{1n}^{(1)} \\ \vdots & & \vdots \\ a_{n1}^{(1)} & \dots & a_{nn}^{(1)} \end{pmatrix} \tag{3.9}$$

la matrice al primo passo. L'indice superiore di ciascun elemento servirà a mettere in evidenza il passo più recente in cui il valore dell'elemento sarà stato formalmente modificato dalla procedura.

Poniamoci ora l'obiettivo di definire una matrice triangolare inferiore a diagonale unitaria che, moltiplicando a sinistra $A^{(1)}$, renda la prima colonna del prodotto strutturalmente uguale alla prima colonna di una matrice triangolare superiore, ovvero abbia tutte le componenti dalla seconda in poi nulle. Se

$$a_{11}^{(1)} \neq 0, \tag{3.10}$$

possiamo definire il *primo* vettore elementare di Gauss:

$$\mathbf{g}_1 = \frac{1}{a_{11}^{(1)}}(0, a_{21}^{(1)}, \dots, a_{n1}^{(1)})^H \tag{3.11}$$

e la corrispondente *prima* matrice elementare di Gauss:

$$L_1 = I - \mathbf{g}_1 \mathbf{e}_1^H = \begin{pmatrix} 1 & & & \\ -\frac{a_{21}^{(1)}}{a_{11}^{(1)}} & 1 & & \\ \vdots & & \ddots & \\ -\frac{a_{n1}^{(1)}}{a_{11}^{(1)}} & & & 1 \end{pmatrix}, \tag{3.12}$$

tali che

$$L_1 A = \begin{pmatrix} a_{11}^{(1)} & \dots & \dots & a_{1n}^{(1)} \\ 0 & a_{22}^{(2)} & \dots & a_{2n}^{(2)} \\ \vdots & \vdots & & \vdots \\ 0 & a_{n2}^{(2)} & \dots & a_{nn}^{(2)} \end{pmatrix} \equiv A^{(2)}.$$

Lo stesso procedimento potrà essere ripetuto per la seconda colonna di $A^{(2)}$ purché $a_{22}^{(2)} \neq 0$. In generale, al passo i-esimo, se $a_{jj}^{(j)} \neq 0$ per ogni $j < i$, avremo ottenuto

$$L_{i-1} \cdots L_2 L_1 A = \begin{pmatrix} a_{11}^{(1)} & \dots & \dots & \dots & \dots & a_{1n}^{(1)} \\ 0 & \ddots & & & & \vdots \\ \vdots & \ddots & a_{i-1,i-1}^{(i-1)} & \dots & \dots & a_{i-1,n}^{(i-1)} \\ \vdots & & 0 & a_{ii}^{(i)} & \dots & a_{in}^{(i)} \\ \vdots & & \vdots & \vdots & & \vdots \\ 0 & \dots & 0 & a_{ni}^{(i)} & \dots & a_{nn}^{(i)} \end{pmatrix} \equiv A^{(i)}. \tag{3.13}$$

Se, a sua volta,

$$a_{ii}^{(i)} \neq 0, \tag{3.14}$$

possiamo definire l'i-esimo vettore elementare di Gauss:

$$\mathbf{g}_i = \frac{1}{a_{ii}^{(i)}} (\underbrace{0, \dots, 0}_{i}, a_{i+1,i}^{(i)}, \dots, a_{ni}^{(i)})^H \tag{3.15}$$

e la corrispondente matrice elementare di Gauss:

$$L_i = I - \mathbf{g}_i \mathbf{e}_i^H = \begin{pmatrix} 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & -\frac{a_{i+1,i}^{(i)}}{a_{ii}^{(i)}} & \ddots & \\ & & \vdots & & \ddots \\ & & -\frac{a_{ni}^{(i)}}{a_{ii}^{(i)}} & & 1 \end{pmatrix}, \tag{3.16}$$

tali che

$$\begin{aligned} L_i A^{(i)} &= L_i L_{i-1} \cdots L_2 L_1 A \\ &= \begin{pmatrix} a_{11}^{(1)} & \dots & \dots & \dots & \dots & a_{1n}^{(1)} \\ 0 & \ddots & & & & \vdots \\ \vdots & \ddots & a_{ii}^{(i)} & \dots & \dots & a_{in}^{(i)} \\ \vdots & & 0 & a_{i+1,i+1}^{(i+1)} & \dots & a_{i+1,n}^{(i+1)} \\ \vdots & & \vdots & \vdots & & \vdots \\ 0 & \dots & 0 & a_{n,i+1}^{(i+1)} & \dots & a_{nn}^{(i+1)} \end{pmatrix} \equiv A^{(i+1)}. \end{aligned} \quad (3.17)$$

Se è possibile iterare i passi (3.13)-(3.17) fino a $i = n - 1$ avremo infine ottenuto che

$$L_{n-1} \cdots L_2 L_1 A = \begin{pmatrix} a_{11}^{(1)} & \dots & \dots & a_{1n}^{(1)} \\ & \ddots & & \vdots \\ & & a_{n-1,n-1}^{(n-1)} & a_{n-1,n}^{(n-1)} \\ & & & a_{nn}^{(n)} \end{pmatrix} \equiv A^{(n)} \equiv U. \quad (3.18)$$

A questo punto osserviamo che la matrice

$$L_{n-1} \cdots L_2 L_1$$

è il prodotto di matrici triangolari inferiori a diagonale unitaria e quindi è a sua volta una matrice triangolare inferiore a diagonale unitaria. Inoltre, in virtù del fatto che

$$\mathbf{e}_k^H \mathbf{g}_i \equiv g_{ki} = 0, \quad \text{per } k \leq i,$$

posto

$$L^{-1} = L_{n-1} \cdots L_2 L_1$$

si ottiene:

$$\begin{aligned} L &= (L_{n-1} \cdots L_2 L_1)^{-1} = L_1^{-1} L_2^{-1} \cdots L_{n-1}^{-1} \\ &= (I + \mathbf{g}_1 \mathbf{e}_1^H) \cdots (I + \mathbf{g}_{n-1} \mathbf{e}_{n-1}^H) = I + \mathbf{g}_1 \mathbf{e}_1^H + \cdots + \mathbf{g}_{n-1} \mathbf{e}_{n-1}^H \\ &= \begin{pmatrix} 1 & & & \\ g_{21} & 1 & & \\ \vdots & \ddots & \ddots & \\ g_{n1} & \dots & g_{n,n-1} & 1 \end{pmatrix}. \end{aligned} \quad (3.19)$$

Dalle considerazioni appena fatte si ottiene la fattorizzazione (3.5).

Vogliamo ora derivare dalle (3.18)-(3.19) condizioni sufficienti a garantire l'esistenza della fattorizzazione LU. Ricordiamo che, per quanto detto prima, il metodo di eliminazione di Gauss è definito se e solo se è possibile costruire al generico passo i la corrispondente matrice di Gauss. Questo si è visto essere equivalente a richiedere che la (3.14) sia verificata. Dovendo tale condizione essere soddisfatta per $i = 1, 2, \ldots, n-1$, e considerando che la matrice A è nonsingolare, otteniamo il seguente risultato.

Lemma 3.1 *Se A è nonsingolare, la fattorizzazione (3.5) è definita se e solo se $a_{ii}^{(i)} \neq 0$, $i = 1, 2, \ldots, n$, ovvero se e solo se la matrice U è nonsingolare.*

Si possono dimostrare le seguenti proprietà:

Lemma 3.2 *Una matrice triangolare è nonsingolare se e solo se tutti i suoi minori principali di testa sono non nulli.*

Lemma 3.3 *Il minore principale di testa di ordine k di A in (3.5) coincide con il minore principale di testa di ordine k di U in (3.18).*

Possiamo enunciare il seguente risultato.

Teorema 3.2 (Esistenza della fattorizzazione LU)
Se A è nonsingolare, la fattorizzazione (3.5) esiste se e solo se tutti i minori principali di testa di A sono non nulli.

Dim. Dal Lemma 3.1 segue che la fattorizzazione esiste se e solo se U è nonsingolare. Dal Lemma 3.2, segue che questo equivale a richiedere che tutti i minori principali di testa di U siano non nulli. Dal Lemma 3.3 segue infine che questo equivale a richiedere che tutti i minori principali di testa di A siano non nulli. □

Esaminiamo ora il costo del metodo di eliminazione di Gauss sia in termini di occupazione di memoria sia in termini di numero di operazioni necessarie. Riguardo alla occupazione di memoria osserviamo che, al passo i-esimo, azzereremo le componenti $i+1, \ldots, n$ in colonna i della matrice $A^{(i)}$ (vedi (3.13)). Di conseguenza, $n - i$ locazioni di memoria si rendono disponibili: queste possono essere riscritte con le ultime $n - i$ componenti del vettore di Gauss (3.22) che sono le sole componenti significative del vettore. Pertanto, così facendo, alla fine del processo avremo riscritto la matrice A con:

- le componeneti significative del fattore U nella parte triangolare superiore (vedi (3.18)),

- le componenti significative del fattore L (vedi (3.19)) nella porzione strettamente triangolare inferiore. Infatti, la diagonale principale di L, avendo tutti elementi unitari, non necessita di una esplicita memorizzazione.

Si conclude che, dal punto di vista dell'occupazione di memoria, il metodo di eliminazione di Gauss non ne richiede di addizionale in quanto la matrice di ingresso A può essere riscritta con l'informazione relativa ai suoi fattori L ed U.
Per quanto riguarda le operazioni richieste, al generico passo i-esimo dobbiamo costruire il vettore (3.22) e, successivamente, il prodotto $L_i A^{(i)}$.

▷ per costruire il vettore $\mathbf{g}_i$ definito in (3.22) si devono fare $(n-i)$ divisioni.
Allora per costruirli tutti:

$$\sum_{i=1}^{n-1}(n-i) = \sum_{j=1}^{n-1} j = \frac{(n-1)n}{2};$$

▷ per calcolare il prodotto $L_i A^{(i)} = (I - \mathbf{g}_i \mathbf{e}_i^H) A^{(i)} = A^{(i)} - \mathbf{g}_i(\mathbf{e}_i^H A^{(i)})$ si devono fare $(n-i)^2$ prodotti.
Allora per costruirli tutti:

$$\sum_{i=1}^{n-1}(n-i)^2 = \sum_{j=1}^{n-1} j^2 = \frac{(n-1)n(2n-1)}{6}.$$

Il costo computazionale (moltiplicazioni e divisioni) del metodo di eliminazione di Gauss di una matrice di ordine n è : $\approx \frac{2}{6}\, n^3 + \frac{n^2}{2} \approx \frac{n^3}{3}$.

Osservazione 3.2 *Considerato che $A = LU$, si ha*

$$det\ A = det\ (LU) = (det\ L)(det\ U) = 1 \cdot det\ U.$$

Essendo U una matrice triangolare superiore, il suo determinante è uguale al prodotto degli elementi sulla diagonale principale.

Osservazione 3.3 *In* `Matlab` *la fattorizzazione LU si ottiene mediante la* function `lu`.

3.3.1 Metodo di eliminazione di Gauss per la risoluzione del sistema lineare

Se A soddisfa le ipotesi del Teorema 3.2, il metodo di eliminazione di Gauss può essere utilizzato per la risoluzione del sistema lineare $A\mathbf{x} = \mathbf{b}$. La soluzione del sistema

$$L\mathbf{y} = \mathbf{b}$$

viene calcolata durante i passi del procedimento della fattorizzazione LU, in quanto il vettore $\mathbf{y}$ viene costruito moltiplicando il vettore $\mathbf{b}$ successivamente per le matrici L_i così come si fa con la matrice A. Per questo si considera la matrice

$$[A^{(1)}|\mathbf{b}^{(1)}] \equiv [A|\mathbf{b}]$$

e si costruisce la successione

$$[A^{(1)}|\mathbf{b}^{(1)}],\ [A^{(2)}|\mathbf{b}^{(2)}],\ \ldots,\ [A^{(n)}|\mathbf{b}^{(n)}] \equiv [U|\mathbf{y}],$$

tale che

$$[A^{(i+1)}|\mathbf{b}^{(i+1)}] = L_i[A^{(i)}|\mathbf{b}^{(i)}], \quad i = 1, 2, \ldots, n-1.$$

Riassumendo, per risolvere un sistema mediante il metodo di eliminazione di Gauss si deve:

1) fattorizzare la matrice $A = L\,U$;

2) risolvere il sistema lineare $U\,\mathbf{x} = \mathbf{y}$ con matrice dei coefficienti triangolare superiore.

Ne segue che il costo computazionale risulta essere:

$$\left(\frac{n^3}{3} + \frac{n^2}{2}\right) \approx \frac{n^3}{3}.$$

Si osservi come la parte più costosa della procedura sia la fattorizzazione della matrice.

Osservazione 3.4 *Il metodo di eliminazione di Gauss risulta essere particolarmente utile quando si vuole determinare la matrice inversa di A. In tal caso, infatti, basta risolvere*

$$A\,X = I \qquad \Leftrightarrow \qquad A\mathbf{x}_i = \mathbf{e}_i, \quad i = 1, 2, \ldots, n,$$

ovvero n sistemi con termine noto l'i-esimo vettore della base canonica in $\mathbb{C}^n$. Risulta quindi che il costo totale è $\approx \frac{5}{6}n^3$ (costo della fattorizzazione LU, $\approx n^3/3$, e costo della risolzione degli n sistemi risultanti con matrice dei coefficienti triangolare superiore, $\approx n(n^2/2) = n^3/2$.)

Osservazione 3.5 *In* `Matlab` *l'inversa di una matrice A di ordine n si ottiene utilizzando la* function `inv` *oppure mediante l'istruzione* `A\eye(size(A,1))`.

Sappiamo che se una matrice è a predominanza diagonale forte allora è nonsingolare (vedi Proposizione A.1). Inoltre, tutti i suoi minori principali di testa sono a predominanza diagonale forte. Pertanto si ha il seguente risultato.

Teorema 3.3 *Se A è a predominanza diagonale forte, allora è fattorizzabile LU.*

Un'altra classe significativa di matrici, che risultano essere sempre fattorizzabili LU, è quella delle matrici hermitiane e definite positive o, più brevemente, *hdp*. Esaminiamola più nel dettaglio nella prossima sezione.

3.4 Matrice hdp: fattorizzazione LL^H

Definizione 3.1 *Una matrice $A \in \mathbb{C}^{n\times n}$ è hdp se è hermitiana (cioè, $A = A^H$) e, per ogni $\mathbf{x} \in \mathbb{C}^n$, $\mathbf{x} \neq \mathbf{0}$, risulta*

$$\mathbf{x}^H A\mathbf{x} > 0.$$

Valgono le seguenti proprietà.

Lemma 3.4 *Tutte le sottomatrici principali di testa di una matrice hdp sono hdp.*

Dim. Sia $A \in \mathbb{C}^{n\times n}$ hdp e sia A_k la sua sottomatrice principale di testa di ordine k. È evidente che se $A = A^H$ anche $A_k = A_k^H$. Rimane quindi da dimostrare che se $\mathbf{y} \in \mathbb{C}^k$, $\mathbf{y} \neq \mathbf{0}$, allora $\mathbf{y}^H A_k \mathbf{y} > 0$. Definendo il vettore

$$\mathbf{x} = \begin{pmatrix} \mathbf{y} \\ \mathbf{0} \end{pmatrix} \in \mathbb{C}^n,$$

e considerando la seguente partizione a blocchi della matrice A,

$$A = \begin{pmatrix} A_k & B \\ B^H & C \end{pmatrix}$$

si ottiene che $\mathbf{x} \neq \mathbf{0}$ e, di conseguenza,

$$0 < \mathbf{x}^H A\mathbf{x} = \begin{pmatrix} \mathbf{y}^H & \mathbf{0}^H \end{pmatrix} \begin{pmatrix} A_k & B \\ B^H & C \end{pmatrix} \begin{pmatrix} \mathbf{y} \\ \mathbf{0} \end{pmatrix} = \mathbf{y}^H A_k \mathbf{y}. \qquad \square$$

Lemma 3.5 *Una matrice hdp è nonsingolare.*

Dim. Se, per assurdo, A fosse hdp e singolare, allora esisterebbe un vettore $\mathbf{x} \neq \mathbf{0}$ tale che $A\mathbf{x} = \mathbf{0}$. Ne segue che $\mathbf{x}^H A\mathbf{x} = 0$, contraddicendo l'ipotesi A hdp. □

Teorema 3.4 *Se A è hdp, allora è fattorizzabile LU.*

Dim. Dal Lemma 3.4 tutte le sottomatrici principali di testa sono hdp e quindi dal Lemma 3.5 segue che i corrispondenti minori principali di testa sono tutti non nulli. □

Teorema 3.5 *Tutti gli elementi diagonali di una hdp sono reali e positivi.*

Dim. Infatti, se A è hdp allora $a_{ii} = \mathbf{e}_i^H A\mathbf{e}_i = (\mathbf{e}_i^H A\mathbf{e}_i)^H > 0$. □

Vale inoltre il seguente risultato.

Teorema 3.6 *A è hdp se e solo se*

$$A = \hat{L}D\hat{L}^H, \tag{3.20}$$

con

- *$\hat{L}$ matrice triangolare inferiore a diagonale unitaria,*
- *D matrice diagonale con elementi sulla diagonale principale tutti reali e positivi.*

Dim. Lasciamo come esercizio la dimostrazione che (3.20) implica che A è hdp. Dimostriamo, invece, l'implicazione inversa. Se A è hdp, allora in virtù del Teorema 3.4, $A = \hat{L}\hat{U}$. Inoltre il fattore $\hat{U}$ può essere scritto nella forma

$$\hat{U} = D\tilde{U}$$

con D diagonale e $\tilde{U}$ triangolare superiore a diagonale unitaria. Inoltre, essendo $A = A^H$, segue che:

$$\hat{L}\hat{U} = (\hat{L}D\tilde{U})^H = \tilde{U}^H D^H \hat{L}^H.$$

Per l'unicità della "fattorizzazione LU" essendo $\tilde{U}^H$ triangolare inferiore a diagonale unitaria e $D^H\hat{L}^H$ triangolare superiore, segue che $\tilde{U}^H = \hat{L}$ e $D^H\hat{L}^H = D\tilde{U}$. Ne segue che $D = D^H$. Resta da dimostrare che gli elementi diagonali di D sono positivi. In virtù del Teorema 3.5 basta provare che D è hdp. Comunque si fissi $\mathbf{x} \neq \mathbf{0}$ esiste ed è unico il vettore $\mathbf{y} \neq \mathbf{0}$ tale che $\hat{L}^H\mathbf{y} = \mathbf{x}$. Segue pertanto che

$$\mathbf{x}^H D\mathbf{x} = (\hat{L}^H\mathbf{y})^H D(\hat{L}^H\mathbf{y}) = \mathbf{y}^H \hat{L}D\hat{L}^H\mathbf{y} = \mathbf{y}^H A\mathbf{y} > 0,$$

essendo A hdp. $\square$

Quando la matrice è hdp la fattorizzazione (3.20) risulta essere più efficiente, dal punto di vista computazionale, della fattorizzazione (3.5). Infatti dal punto di vista dell'occupazione di memoria si osserva che una matrice hdp può essere memorizzata in forma compatta memorizzandone solo la parte triangolare inferiore o superiore. Anche dal punto di vista delle operazioni, non dovendo calcolare anche il fattore triangolare superiore, è lecito aspettarsi un costo inferiore. Al fine di stabilire quest'ultima conclusione, andiamo a considerare le formule che permettono di ottenere in modo efficiente la fattorizzazione (3.20). Dal Teorema 3.6 si ha che D ha elementi reali e positivi. Quindi è ben definita la matrice $D^{1/2}$ i cui elementi sono le radici quadrate degli elementi di D. Pertanto, posto

$$L = \hat{L} D^{1/2}$$

i suoi elementi

$$L = \begin{pmatrix} \ell_{11} & 0 & \dots & 0 & 0 \\ \ell_{21} & \ell_{22} & 0 & \dots & 0 \\ \ell_{31} & \ell_{32} & \ell_{33} & \dots & 0 \\ \vdots & & & \ddots & \\ \ell_{n1} & \ell_{n2} & \ell_{n3} & \dots & \ell_{nn} \end{pmatrix}, \qquad \ell_{ii} > 0, i = 1, 2, \dots, n$$

si possono determinare imponendo che (vedi (3.20))

$$A = LL^H.$$

Per $i \geq j$ si ha:

$$\begin{aligned} a_{ij} &= \mathbf{e}_i^H A \mathbf{e}_j = \mathbf{e}_i^H L L^H \mathbf{e}_j = (\mathbf{e}_i^H L)(\mathbf{e}_j^H L)^H \\ &= \sum_{k=1}^{j} \ell_{ik} \bar{\ell}_{jk} = \sum_{k=1}^{j-1} \ell_{ik} \bar{\ell}_{jk} + \ell_{ij} \ell_{jj} \end{aligned}$$

Si ottengono le seguenti espressioni valide per $j = 1, 2, \dots, n$:

$$\begin{aligned} \ell_{jj} &= \left(a_{jj} - \sum_{k=1}^{j-1} |\ell_{jk}|^2 \right)^{1/2}, \\ \ell_{ij} &= \left(a_{ij} - \sum_{k=1}^{j-1} \ell_{ik} \bar{\ell}_{jk} \right) / \ell_{jj}, \qquad i = j+1, 3, \dots, n. \end{aligned}$$

Queste relazioni definiscono il **metodo di Cholesky**. Si può dimostrare che per essere determinate esse richiedono

$$\approx \frac{n^3}{6} \quad \text{moltiplicazioni e divisioni,}$$

ovvero circa la metà di quelle necessarie per fattorizzare una matrice LU.

Osservazione 3.6 *In* `Matlab` *la fattorizzazione di Cholesky di una matrice* A *hdp di ordine* n *si ottiene utilizzando la* function `chol`.

3.5 Pivoting

Esaminiamo ora il caso in cui le ipotesi del Teorema 3.2 non siano soddisfatte, sebbene la matrice A sia nonsingolare. In tal caso la fattorizzazione LU non esiste. Tuttavia, cerchiamo di capire come sia possibile modificare opportunamente tale fattorizzazione in modo da definirne una nuova che esista sotto la sola ipotesi di nonsingolarità di A. Supponiamo di essere al primo passo del metodo di Gauss e che la condizione $a_{11}^{(1)} \neq 0$ non sia verificata. Certamente, essendo $A^{(1)}$ nonsingolare, esisterà sulla sua prima colonna un elemento non nullo. Pertanto si avrà

$$|a_{k_1 1}^{(1)}| = \max_{k \geq 1} a_{k1}^{(1)} > 0.$$

Definiamo la matrice elementare di permutazione P_1 tale che applicata ad un generico vettore $\mathbf{v} = (v_1, v_2, \ldots, v_n)^T$ dia

$$P_1 \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_{k_1-1} \\ v_{k_1} \\ \vdots \\ v_n \end{pmatrix} = \begin{pmatrix} v_{k_1} \\ v_2 \\ \vdots \\ v_{k_1-1} \\ v_1 \\ \vdots \\ v_n \end{pmatrix}.$$

Segue che $P_1 A^{(1)}$ è la matrice ottenuta da $A^{(1)}$ permutando le righe 1 e k_1. Pertanto il primo passo di eliminazione di Gauss è ora possibile per la matrice $P_1 A^{(1)}$ e il vettore (3.11) diviene

$$\mathbf{g}_1 = \frac{1}{a_{k_1 1}^{(1)}} (\underbrace{0, a_{21}^{(1)}, \ldots,}_{k_1 - 1} a_{11}^{(1)}, \ldots, a_{n1}^{(1)})^H.$$

È quindi possibile definire la corrispondente prima matrice elementare di Gauss:

$$L_1 = I - \mathbf{g}_1 \mathbf{e}_1^H \tag{3.21}$$

che consente di ottenere

$$L_1 P_1 A^{(1)} = \begin{pmatrix} a_{k_1 1}^{(1)} & \dots & \dots & a_{k_1 n}^{(1)} \\ 0 & a_{22}^{(2)} & \dots & a_{2n}^{(2)} \\ \vdots & \vdots & & \vdots \\ 0 & a_{n2}^{(2)} & \dots & a_{nn}^{(2)} \end{pmatrix} \equiv A^{(2)}.$$

La procedura prosegue quindi in modo analogo a quanto visto in Sezione 3.3: al passo i-esimo si avrà la matrice

$$L_{i-1} P_{i-1} \cdots L_1 P_1 A = \begin{pmatrix} a_{k_1 1}^{(1)} & \dots & \dots & \dots & \dots & a_{k_1 n}^{(1)} \\ 0 & \ddots & & & & \vdots \\ \vdots & \ddots & a_{k_{i-1},i-1}^{(i-1)} & \dots & \dots & a_{k_{i-1},n}^{(i-1)} \\ \vdots & & 0 & a_{i,i}^{(i)} & \dots & a_{i,n}^{(i)} \\ \vdots & & \vdots & \vdots & & \vdots \\ 0 & \dots & 0 & a_{n,i}^{(i)} & \dots & a_{nn}^{(i)} \end{pmatrix} \equiv A^{(i)}.$$

Se

$$|a_{k_i i}^{(i)}| \equiv \max_{k \geq i} a_{ki}^{(i)}$$

dovrà aversi $a_{k_i i}^{(i)} \neq 0$, se A è nonsingolare. Pertanto, definendo la matrice di permutazione P_i che permuta le righe i e k_i si avrà che l'elemento in posizione (i, i) della matrice $P_i A^{(i)}$ diverrà $a_{k_i i}^{(i)}$. Possiamo quindi definire l'i-esimo vettore elementare di Gauss:

$$\mathbf{g}_i = \frac{1}{a_{k_i i}^{(i)}} (\underbrace{0, \dots, 0}_{i}, a_{i+1,i}^{(i)}, \dots, a_{ii}^{(i)}, \dots, a_{ni}^{(i)})^H \tag{3.22}$$

e la corrispondente matrice elementare di Gauss:

$$L_i = I - \mathbf{g}_i \mathbf{e}_i^H \tag{3.23}$$

tali che

$$L_i P_i A^{(i)} = \begin{pmatrix} a_{k_1 1}^{(1)} & \dots & \dots & \dots & \dots & a_{k_1 n}^{(1)} \\ 0 & \ddots & & & & \vdots \\ \vdots & \ddots & a_{k_i,i}^{(i)} & \dots & \dots & a_{k_i,n}^{(i)} \\ \vdots & & 0 & a_{i+1,i+1}^{(i+1)} & \dots & a_{i+1,n}^{(i+1)} \\ \vdots & & \vdots & \vdots & & \vdots \\ 0 & \dots & 0 & a_{n,i+1}^{(i+1)} & \dots & a_{nn}^{(i+1)} \end{pmatrix} \equiv A^{(i+1)}. \quad (3.24)$$

Procedendo in questo modo fino a $i = n - 1$ si ottiene:

$$L_{n-1} P_{n-1} L_{n-2} P_{n-2} \cdots L_1 P_1 A = A^{(n)} \equiv U.$$

Ponendo

$$\begin{aligned} \hat{L}_{n-1} &= L_{n-1} \\ \hat{L}_i &= P_{n-1} \cdots P_{i+1} L_i P_{i+1} \cdots P_{n-1}, \quad i = 1, 2, , \dots, n-2, \\ P &= P_{n-1} \cdots P_1, \end{aligned}$$

la precedente può essere riscritta come

$$\hat{L}_{n-1} \hat{L}_{n-2} \cdots \hat{L}_1 P A = U.$$

Osserviamo ora che

$$\begin{aligned} \hat{L}_i &= P_{n-1} \cdots P_{i+1} (I - \mathbf{g}_i \mathbf{e}_i^H) P_{i+1} \cdots P_{n-1} \\ &= I - (P_{n-1} \cdots P_{i+1} \mathbf{g}_i)(\mathbf{e}_i^H P_{i+1} \cdots P_{n-1}) = I - \hat{\mathbf{g}}_i \mathbf{e}_i^H. \end{aligned}$$

Nell'ultimo passaggio si è tenuto conto del fatto che $\mathbf{e}_i^H P_j = \mathbf{e}_i^H$, per $j > i$. Di conseguenza la struttura della matrice $\hat{L}_i$ è analoga a quella della matrice elementare di Gauss (3.16). Se ne conclude che $\hat{L}_{n-1} \hat{L}_{n-2} \cdots \hat{L}_1 \equiv L^{-1}$ è una matrice triangolare inferiore a diagonale unitaria. Pertanto si ottiene la seguente *fattorizzazione LU con pivoting parziale*

$$PA = LU.$$

I precedenti argomenti si possono riassumere nel seguente

Teorema 3.7 *Se A è una matrice nonsingolare, allora esiste una matrice di permutazione P tale che PA è fattorizzabile LU.*

Nel caso in cui si debba risolvere il sistema lineare (3.1) con matrice A nonsingolare si ha:

$$\begin{aligned} A\mathbf{x} = \mathbf{b} \quad &\Leftrightarrow \quad PA\mathbf{x} = P\mathbf{b} \\ &\Leftrightarrow \quad L\mathbf{y} = P\mathbf{b} \quad \text{e} \quad U\mathbf{x} = \mathbf{y}. \end{aligned}$$

Pertanto, ottenuta la fattorizzazione $PA = LU$ la matrice di permutazione è utilizzata solo per permutare il vettore dei termini noti $\mathbf{b}$.

Si osservi che per calcolare i *pivot* ai passi $i = 1, 2, \ldots, n-1$, sono necessari $\approx n^2/2$ confronti. Pertanto, tale costo è trascurabile rispetto al costo della fattorizzazione.

3.6 Fattorizzazione QR

In molte applicazioni pratiche è necessario risolvere un sistema di equazioni lineari sovradeterminato, ovvero con più equazioni che incognite, in cui la matrice dei coefficienti ha rango massimo. Formalmente, quindi, si vuole risolvere il sistema lineare

$$A\mathbf{x} = \mathbf{b}, \quad A \in \mathbb{R}^{m\times n}, \quad m > n \equiv \text{rank}(A). \tag{3.25}$$

Una rilevante istanza del problema (3.25) sarà trattata in Sezione 3.6.2: per essa, in genere, è $m \gg n$.

Lo stumento principale che utilizzeremo per risolvere (3.25) è la *fattorizzazione* QR della matrice A.

Teorema 3.8 *Data la matrice A in (3.25) esistono:*

- $Q \in \mathbb{R}^{m\times m}$, *ortogonale,*
- $\hat{R} \in \mathbb{C}^{n\times n}$, *triangolare superiore e non singolare,*

tali che:

$$A = QR \equiv Q \begin{pmatrix} \hat{R} \\ O \end{pmatrix}.$$

Al fine di disporre degli strumenti per affrontare la dimostrazione del Teorema 3.8, si consideri il seguente problema: dato un vettore

$$\mathbf{z} = (z_1, \ldots, z_m)^T, \in \mathbb{R}^m, \quad \mathbf{z} \neq \mathbf{0},$$

determinare una matrice ortogonale H tale che

$$H\mathbf{z} = \alpha \mathbf{e}_1, \tag{3.26}$$

dove α è un'opportuna costante. Avendosi,

$$\|\mathbf{z}\|_2^2 = \mathbf{z}^T\mathbf{z} = \mathbf{z}^T(H^TH)\mathbf{z} = (H\mathbf{z})^T(H\mathbf{z}) = \alpha^2\mathbf{e}_1^T\mathbf{e}_1 = \alpha^2,$$

risulta

$$\alpha = \pm\|\mathbf{z}\|_2.$$

A questo punto consideriamo una matrice H avente la seguente forma:

$$H = I - \frac{2}{\|\mathbf{v}\|_2^2}\mathbf{v}\mathbf{v}^T, \qquad \mathbf{v} \neq \mathbf{0}$$

in cui il vettore $\mathbf{v}$ sarà scelto in modo da soddisfare la (3.26). Osserviamo che la matrice H è simmetrica per costruzione. Inoltre, essa è anche ortogonale:

$$H \cdot H^T = H^T \cdot H = H^2 = I - \frac{4}{\|\mathbf{v}\|_2^2}\mathbf{v}\mathbf{v}^T + \frac{4}{\|\mathbf{v}\|_2^4}\mathbf{v}(\mathbf{v}^T\mathbf{v})\mathbf{v}^T = I.$$

Dimostriamo adesso che la matrice H corrispondente al vettore

$$\mathbf{v} = \mathbf{z} - \alpha\mathbf{e}_1$$

soddisfa la (3.26). Infatti,

$$H\,\mathbf{z} = \left(I - \frac{2}{\|\mathbf{v}\|_2^2}\mathbf{v}\mathbf{v}^T\right)\mathbf{z} = \mathbf{z} - \frac{2}{\|\mathbf{v}\|_2^2}\cdot(\mathbf{v}^T\mathbf{z})\mathbf{v}$$

con

$$\begin{aligned}
\mathbf{v}^T\mathbf{z} &= (\mathbf{z}-\alpha\mathbf{e}_1)^T\mathbf{z} = \mathbf{z}^T\mathbf{z} - \alpha\mathbf{e}_1^T\mathbf{z} = \|\mathbf{z}\|_2^2 - \alpha z_1 = \alpha^2 - \alpha z_1 \\
&= \alpha(\alpha - z_1) \\
\frac{2}{\|\mathbf{v}\|_2^2} &= \frac{2}{(\mathbf{z}-\alpha\mathbf{e}_1)^T(\mathbf{z}-\alpha\mathbf{e}_1)} = \frac{2}{(\mathbf{z}-\alpha\mathbf{e}_1)^T\mathbf{z} - \alpha(\mathbf{z}^T - \alpha\mathbf{e}_1^T)\mathbf{e_1}} = \\
&= \frac{2}{\alpha(\alpha - z_1) - \alpha(z_1 - \alpha)}
\end{aligned}$$

e quindi

$$H\,\mathbf{z} = \mathbf{z} - \frac{1}{\alpha(\alpha - z_1)}\alpha(\alpha - z_1)(\mathbf{z} - \alpha\mathbf{e}_1) = \alpha\mathbf{e}_1.$$

La matrice H così definita è detta *matrice elementare di Householder*.

Osservazione 3.7 *Al fine di rendere sempre ben condizionata la costruzione del vettore* $\mathbf{v}$ *e, quindi, della matrice* H*, la scelta ottimale del segno di* α *sarà quella per cui* z_1 *e* $-\alpha$ *sono di segno concorde. In tal modo, si evita il rischio di produrre un errore di cancellazione numerica.*

Proposizione 3.1 *Sia H la matrice di Householder definita dal vettore $\mathbf{v}$. Allora, la matrice di Householder definita dal vettore $(\beta\mathbf{v})$, $\forall\beta \neq 0$, coincide con H.*

Dim.

$$I - \frac{2}{\|\beta\mathbf{v}\|_2^2}(\beta\mathbf{v})(\beta\mathbf{v})^T = I - 2\frac{(\beta\mathbf{v})(\beta\mathbf{v})^T}{(\beta\mathbf{v})^T(\beta\mathbf{v})} = I - 2\frac{\beta^2\mathbf{v}\mathbf{v}^T}{\beta^2\mathbf{v}^T\mathbf{v}} = H.$$

□

Questa proprietà di invarianza per scalamento del corrispondente vettore di Householder sarà convenientemente usata per ottimizzare il costo computazionale, in termini di occupazione di memoria, per ottenere la fattorizzazione QR.

Osservazione 3.8 *Sia $\mathbf{x}$ un vettore non nullo. Denotiamo con $\mathbf{u}$ la sua componente parallela al vettore $\mathbf{v}$ e con $\mathbf{w}$ la sua componente ortogonale ad $\mathbf{v}$. Posto quindi $\mathbf{x} = \mathbf{u} + \mathbf{w}$, calcoliamo*

$$H\,\mathbf{x} = \left(I - \frac{2}{\|\mathbf{v}\|_2^2}\mathbf{v}\mathbf{v}^T\right)(\mathbf{u} + \mathbf{w}) = \mathbf{u} + \mathbf{w} - 2\,\frac{\mathbf{v}^T\mathbf{u}}{\|\mathbf{v}\|_2^2}\,\mathbf{v} - 2\,\frac{\mathbf{v}^T\mathbf{w}}{\|\mathbf{v}\|_2^2}\,\mathbf{v}.$$

Ma $\mathbf{v}^T\mathbf{w} = 0$. Inoltre, essendo $\mathbf{u} = \beta\mathbf{v}$, la precedente diventa:

$$H\,\mathbf{x} = \beta\mathbf{v} + \mathbf{w} - 2\beta\mathbf{v} = -\beta\mathbf{v} + \mathbf{w} = -\mathbf{u} + \mathbf{w}.$$

Pertanto, geometricamente, la matrice H opera una riflessione del vettore $\mathbf{x}$ rispetto all'asse perpendicolare al vettore $\mathbf{v}$ e passante per l'origine. Per questo motivo le matrici di Householder sono anche dette matrici di riflessione.

3.6.1 Metodo di Householder per la fattorizzazione QR

Il procedimento di fattorizzazione della matrice A con matrici di Householder è **sempre** applicabile. Poniamo

$$A = \begin{pmatrix} a_{11}^{(0)} & a_{12}^{(0)} & \dots & a_{1n}^{(0)} \\ a_{21}^{(0)} & a_{22}^{(0)} & & a_{2n}^{(0)} \\ \vdots & \vdots & & \vdots \\ a_{m1}^{(0)} & a_{m2}^{(0)} & \dots & a_{mn}^{(0)} \end{pmatrix} \equiv A^{(0)}.$$

Nella notazione usata l'indice superiore serve ad indicare il passo più recente in cui il corrispondente elemento è stato modificato. Al primo passo, se

indichiamo con $\mathbf{a}_1$ il vettore formato dalla prima colonna di $A^{(0)}$, possiamo definire la matrice elementare di Householder H_1 tale che

$$H_1\mathbf{a}_1 = \begin{pmatrix} a_{11}^{(1)} \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

con $a_{11}^{(1)} = -segno(a_{11}^{(0)})\|\mathbf{a}_1\|_2$. Osserviamo che se la matrice A ha rango massimo, dovrà necessariamente aversi

$$a_{11}^{(1)} \neq 0.$$

Diversamente la prima colonna di A sarebbe nulla. Si ottiene pertanto:

$$H_1A^{(0)} = \begin{pmatrix} a_{11}^{(0)} + segno(a_{11}^{(0)})\|\mathbf{a}_1\|_2 & a_{12}^{(1)} & \dots & a_{1n}^{(1)} \\ 0 & a_{22}^{(1)} & & a_{2n}^{(1)} \\ \vdots & \vdots & & \vdots \\ 0 & a_{m2}^{(1)} & \dots & a_{mn}^{(1)} \end{pmatrix} \equiv A^{(1)}.$$

A questo punto consideriamo la porzione della seconda colonna a partire dalla seconda riga in poi, ovvero

$$\mathbf{a}_2 = \begin{pmatrix} a_{22}^{(1)} \\ 0 \\ \vdots \\ a_{m2}^{(1)} \end{pmatrix}$$

e definiamo la matrice di Householder $H^{(2)}$ (di dimensione $m-1$) tale che

$$H^{(2)}\mathbf{a}_2 = \begin{pmatrix} a_{22}^{(2)} \\ 0 \\ \vdots \\ 0 \end{pmatrix}.$$

Definendo la matrice

$$H_2 = \begin{pmatrix} 1 & \\ & H^{(2)} \end{pmatrix}$$

otteniamo che essa sarà ancora ortogonale e, inoltre,

$$H_2A^{(1)} = \begin{pmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \dots & \dots & a_{1n}^{(1)} \\ 0 & a_{22}^{(2)} & \dots & \dots & a_{2n}^{(2)} \\ \vdots & 0 & a_{33}^{(2)} & \dots & a_{3n}^{(2)} \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & a_{m3}^{(2)} & \dots & a_{mn}^{(2)} \end{pmatrix} \equiv A^{(2)}.$$

Osserviamo che anche in questo caso se A ha rango massimo si deve avere $a_{22}^{(2)} \neq 0$. Così facendo, dopo n passi

$$H_n H_{n-1} H_2 H_1 A = R.$$

Pertanto, posto $Q^T = H_n H_{n-1} H_2 H_1$ si ottiene

$$A = Q\,R.$$

Questo procedimento descrive in sostanza il metodo di fattorizzazione QR di Householder. Consideriamo ora gli aspetti implementativi. Utilizzando uno scalamento dei vettori di Householder, scegliendo $\beta = 1/v_1$ si avrà $\mathbf{e}_1^T(\beta\mathbf{v}) = 1$ pertanto non serve memorizzare la prima componente del vettore $\mathbf{v}$. Ad ogni passo, memorizzo le rimanenti componenti del vettore $\mathbf{v}$ nella colonna della matrice in cui dovrebbero starci gli zeri. Inoltre, considerato che

$$I - 2\frac{\mathbf{v}\mathbf{v}^T}{\mathbf{v}^T\mathbf{v}} = I - 2\frac{\mathbf{v}\mathbf{v}^T}{-2\alpha v_1} = I + \frac{v_1^2}{\alpha v_1}\left(\frac{\mathbf{v}}{v_1}\right)\left(\frac{\mathbf{v}}{v_1}\right)^T$$

e posto $\gamma = -v_1^2/\alpha v_1$ il prodotto per una generica matrice M si ottiene come segue:

$$\left(I - \gamma\left(\frac{\mathbf{v}}{v_1}\right)\left(\frac{\mathbf{v}}{v_1}\right)^T\right) M = M - \left(\gamma\frac{\mathbf{v}}{v_1}\right)\left(\frac{\mathbf{v}^T}{v_1}M\right)$$

La fattorizzazione QR di Householder ha un costo computazionale dell'ordine di $\frac{n^2}{3}(3m-n)$. Si potrebbe usare per risolvere sistemi lineari quadrati ($m = n$) ma in tal caso il costo da sostenere sarebbe doppio rispetto alla fattorizzazione LU.

3.6.2 Soluzione di sistemi lineari nel senso dei minimi quadrati

Si vuole risolvere il sistema lineare

$$A\mathbf{x} = \mathbf{b}, \quad A \in \mathbb{R}^{m\times n}, \quad m > n \equiv \text{rank}(A). \tag{3.27}$$

Osserviamo che tale sistema ammette soluzione se e solo se $\mathbf{b} \in I(A)$. Tuttavia, considerato che $\mathbf{b} \in \mathbb{R}^m$ e che la dimensione dell'immagine di A è uguale a n, un semplice argomento dimensionale ci dice che, in generale, il sistema non ammette soluzione. In alternativa, si cerca il vettore $\mathbf{x}$ tale che il vettore residuo

$$\mathbf{r} = A\mathbf{x} - \mathbf{b}$$

sia tale che $\|\mathbf{r}\|_2$ risulti minima. In tal caso si parla di soluzione del sistemi lineari *nel senso dei minimi quadrati*.

Considerato che

$$\begin{aligned}\|\mathbf{r}\|_2^2 &= \|A\mathbf{x}-\mathbf{b}\|_2^2 = (A\mathbf{x}-\mathbf{b})^T(A\mathbf{x}-\mathbf{b}) = (\mathbf{x}^TA^T-\mathbf{b}^T)(A\mathbf{x}-\mathbf{b}) \\ &= \mathbf{x}^TA^TA\mathbf{x}-\mathbf{x}^TA^T\mathbf{b}-\mathbf{b}^TA\mathbf{x}+\mathbf{b}^T\mathbf{b}.\end{aligned}$$

Condizione necessaria per l'esistenza del minimo è porre uguale a zero le derivate del residuo $\mathbf{r}$ rispetto alle componenti del vettore $\mathbf{x}$. Si ottiene così

$$A^TA\mathbf{x} = A^T\mathbf{b}$$

detto sistema delle *equazioni normali*. Dalla precedente si ottiene che

$$A^T(A\mathbf{x}-\mathbf{b}) = 0$$

ovvero il residuo è ortogonale alle colonne di A.

Considerato che le colonne di A sono linearmente indipendenti per ipotesi, la matrice $B = A^TA$ è simmetrica e definita positiva. Pertanto, le equazioni normali ammettono un'unica soluzione. Tuttavia, se le colonne di A sono "quasi" linearmente dipendenti ci possono essere dei problemi. Ad esempio, sia

$$A = \begin{pmatrix} 1 & 1 & 1 \\ \epsilon & 0 & 0 \\ 0 & \epsilon & 0 \\ 0 & 0 & \epsilon \end{pmatrix}, \qquad \mathbf{b} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}.$$

Allora

$$A^TA = \begin{pmatrix} 1+\epsilon^2 & 1 & 1 \\ 1 & 1+\epsilon^2 & 1 \\ 1 & 1 & 1+\epsilon^2 \end{pmatrix}, \qquad A^T\mathbf{b} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}.$$

Pertanto, se $|\epsilon| < u$, la precisione di macchina, per il calcolatore la matrice

$$A^TA = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix},$$

ovvero è una matrice singolare e di rango 1.

Un modo alternativo per risolvere problemi di tipo (3.27) consiste nel ricorrere alla fattorizzazione QR della matrice A. In tal caso,

$$\|\mathbf{r}\|_2^2 = \|A\mathbf{x}-\mathbf{b}\|_2^2 = \|QR\mathbf{x}-\mathbf{b}\|_2^2 = \|Q(R\mathbf{x}-Q^T\mathbf{b})\|_2^2 = \|R\mathbf{x}-Q^T\mathbf{b}\|_2^2.$$

Posto

$$\mathbf{g} = Q^T\mathbf{b} = \begin{pmatrix} \mathbf{g_1} \\ \mathbf{g_2} \end{pmatrix}, \qquad \mathbf{g_1} \in \mathbb{R}^n,$$

si ottiene quindi che

$$\|\mathbf{r}\|_2^2 = \left\| \begin{pmatrix} \hat{R} \\ O \end{pmatrix} \mathbf{x} - \begin{pmatrix} \mathbf{g_1} \\ \mathbf{g_2} \end{pmatrix} \right\|_2^2 = \left\| \begin{pmatrix} \hat{R}\mathbf{x} - \mathbf{g_1} \\ -\mathbf{g_2} \end{pmatrix} \right\|_2^2 = \|\hat{R}\mathbf{x} - \mathbf{g_1}\|_2^2 + \|\mathbf{g_2}\|_2^2.$$

Allora, scegliendo $\mathbf{x}$ come soluzione del sistema

$$\hat{R}\mathbf{x} = \mathbf{g_1} \tag{3.28}$$

minimizzo $\|\mathbf{r}\|_2^2$. A questo proposito si osserva che:

- essendo $\hat{R}$ non singolare il sistema lineare (3.28) ammette un'unica soluzione;
- essendo $\hat{R}$ una matrice triangolare superiore (3.28) è facilmente risolubile con una sostituzione all'indietro;
- il fattore Q non è esplicitamente richiesto se è possibile effettuare il prodotto $Q^T\mathbf{b}$ per ottenere $\mathbf{g}$.

3.7 Metodi iterativi

Sia $A \in \mathbb{C}^{n\times n}$ non singolare. In generale, da un punto di vista di costo computazionale è conveniente usare i metodi iterativi quando la matrice dei coefficienti è di grosse dimensioni e sparsa (ovvero il numero di elementi non nulli è $O(n)$). Si consideri la decomposizione di A nella forma:

$$A = M - N, \qquad \det M \neq 0.$$

Allora

$$A\mathbf{x} = \mathbf{b} \Leftrightarrow (M-N)\mathbf{x} = \mathbf{b} \Leftrightarrow M\mathbf{x} = N\mathbf{x} + \mathbf{b} \Leftrightarrow \mathbf{x} = M^{-1}N\mathbf{x} + M^{-1}\mathbf{b}.$$

Posto

$$P = M^{-1}N, \qquad \mathbf{q} = M^{-1}\mathbf{b},$$

dalle precedenti relazioni si ottiene il seguente sistema:

$$\mathbf{x} = P\mathbf{x} + \mathbf{q}$$

che è equivalente ad $A\mathbf{x} = \mathbf{b}$.
Dato un vettore iniziale $\mathbf{x}^{(0)}$, si considera la successione $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots$ così definita:

$$\mathbf{x}^{(k)} = P\mathbf{x}^{(k-1)} + \mathbf{q}, \qquad k = 1, 2, \ldots. \tag{3.29}$$

Se la successione $\{\mathbf{x}^{(k)}\}$ è convergente, cioè se $\lim_{k\to\infty}\mathbf{x}^{(k)} = \mathbf{x}^*$, passando al limite in (3.29) si ottiene

$$\mathbf{x}^* = P\mathbf{x}^* + \mathbf{q}, \tag{3.30}$$

ovvero $\mathbf{x}^*$ è soluzione di $\mathbf{x} = P\mathbf{x} + \mathbf{q}$ e quindi di $A\mathbf{x} = \mathbf{b}$.
La relazione (3.29) individua un *metodo iterativo*. La matrice P è detta *matrice di iterazione* del metodo.

Al variare del vettore iniziale $\mathbf{x}^{(0)}$ si ottengono dalla (3.29) diverse successioni $\{\mathbf{x}^{(k)}\}$, alcune delle quali possono essere convergenti ed altre no.
Un metodo iterativo è detto *convergente* se qualunque sia il vettore iniziale $\mathbf{x}^{(0)}$, la successione $\{\mathbf{x}^{(k)}\}$ è convergente.

Esempio 3.1 *Consideriamo un metodo iterativo la cui matrice di iterazione ed il vettore dei termini noti sono:*

$$P = \begin{pmatrix} 1/3 & & \\ & 1/2 & \\ & & 5 \end{pmatrix}, \qquad \mathbf{q} = \mathbf{0}.$$

La soluzione esatta di tale sistema è $\mathbf{x}^* = \mathbf{0}$. *Il metodo iterativo è quindi:*

$$\mathbf{x}^{(k)} = P\mathbf{x}^{(k-1)} = P^k\mathbf{x}^{(0)}.$$

Considerato che

$$P^k = \begin{pmatrix} (1/3)^k & & \\ & (1/2)^k & \\ & & 5^k \end{pmatrix}$$

per $k = 1, 2, 3, \ldots,$ *si ha che se*

$$\begin{array}{lcllcl} \mathbf{x}^{(0)} & = & (1,\,0,\,0)^T & \Rightarrow & \mathbf{x}^{(k)} = ((1/3)^k,\,0,\,0)^T & \Rightarrow \quad \mathbf{x}^{(k)} \to \mathbf{x}^* \\ \mathbf{x}^{(0)} & = & (0,\,1,\,1)^T & \Rightarrow & \mathbf{x}^{(k)} = (0,\,(1/2)^k,\,5^k)^T & \Rightarrow \quad \mathbf{x}^{(k)} \not\to \mathbf{x}^* \end{array}$$

Allora il metodo iterativo proposto non è convergente.

Quali condizioni deve soddisfare la matrice P affinché vi sia convergenza? La risposta più semplice si ottiene sottraendo le due espressioni precedenti (vedi (3.29)-(3.30)):

$$\mathbf{x}^{(k)} - \mathbf{x}^* = P(\mathbf{x}^{(k-1)} - \mathbf{x}^*), \qquad k = 1, 2, \ldots,$$

da cui si ottiene

$$\mathbf{x}^{(k)} - \mathbf{x}^* = P^k(\mathbf{x}^{(0)} - \mathbf{x}^*). \tag{3.31}$$

Passando alle norme

$$\|\mathbf{x}^{(k)} - \mathbf{x}^*\| \leq \|P\|^k \|\mathbf{x}^{(0)} - \mathbf{x}^*\|.$$

Si può quindi asserire che se $\|P\| < 1$ allora il metodo iterativo (3.29) è convergente.
La condizione appena trovata è però solo sufficiente. Esiste una condizione necessaria e sufficiente la quale fa uso delle nozioni di autovalore ed autovettore.

Teorema 3.9 *Il metodo iterativo (3.29) è convergente se e solo se* $\rho(P) < 1$.

Dim. Posto $\mathbf{e}^{(k)} = \mathbf{x}^{(k)} - \mathbf{x}^*$, (3.31) diventa

$$\mathbf{e}^{(k)} = P^k \mathbf{e}^{(0)}.$$

Se $\rho(P) < 1$, si ha

$$\lim_{k \to +\infty} P^k = O$$

e quindi, per ogni vettore $\mathbf{e}^{(0)}$ segue che

$$\lim_{k \to +\infty} \mathbf{e}^{(k)} = 0.$$

Viceversa, se il metodo è convergente, la precedente vale per ogni $\mathbf{x}^{(0)}$ ed, in particolare, deve valere se $\mathbf{x}^{(0)}$ è tale che $\mathbf{e}^{(0)}$ sia un autovettore di P corrispondente ad un autovalore λ di modulo massimo, i.e., $|\lambda| = \rho(P)$. In questo caso si ha

$$P\mathbf{e}^{(0)} = \lambda\, \mathbf{e}^{(0)}$$

e quindi

$$\mathbf{e}^{(k)} = P^k \mathbf{e}^{(0)} = \lambda^k \mathbf{e}^{(0)}.$$

Ne segue che

$$\lim_{k \to \infty} [\rho(P)]^k = 0$$

e quindi $\rho(P) < 1$. $\square$

Ora ci poniamo la seguente domanda: quante iterazioni sono necessarie per ridurre l'errore di un fattore 10^{-m}?
Si osserva che $\|\mathbf{e}^{(k)}\|$ si riduce a $\|\mathbf{e}^{(0)}\| \cdot 10^{-m}$ dopo un numero di iterazioni k tali che

$$[\rho(P)]^k \leq 10^{-m} \quad \Leftrightarrow \quad \log_{10}[\rho(P)]^k \leq \log_{10} 10^{-m} \quad \Leftrightarrow \quad k \log_{10} \rho(P) \leq -m.$$

Ne segue quindi che il *numero di iterazioni* richieste per ridurre l'errore di 10^{-m} è

$$k \approx -\frac{m}{\log_{10} \rho(P)}.$$

La quantità

$$\frac{m}{k} \approx -\log_{10} \rho(P) \qquad (\log_{10} \rho(P) < 0 \text{ perché } \rho(P) < 1)$$

è detta *velocità asintotica di convergenza*. Il raggio spettrale della matrice di iterazione, indipendentemente dalla norma utilizzata e dall'indice di iterazione, è assunta come misura della velocità di convergenza del metodo iterativo.

3.7.1 Criteri di arresto

Considerato che stiamo lavorando con metodi iterativi, dobbiamo stabilire a quale iterata arrestare la procedura.
Ovviamente vorremmo farlo quando la distanza dell'approssimazione $\mathbf{x}^{(k)}$ dalla soluzione $\mathbf{x}^*$ diventa minore di una tolleranza prefissata ε, cioè quando

$$\|\mathbf{x}^{(k)} - \mathbf{x}^*\| < \varepsilon.$$

Ma la soluzione $\mathbf{x}^*$ è ignota. È necessario quindi controllare una quantità nota connessa alla precedente. Tenendo conto che

$$\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)} = P\,\mathbf{x}^{(k)} + \mathbf{q} - \mathbf{x}^{(k)} = (P - I)\,\mathbf{x}^{(k)} + \mathbf{q}$$

e (vedi (3.30)):

$$\mathbf{0} = (P - I)\,\mathbf{x}^* + \mathbf{q}$$

sottraendo l'ultima relazione da quella precedente si ottiene

$$\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)} = (P - I)\,(\mathbf{x}^{(k)} - \mathbf{x}^*).$$

Passando alle norme:

$$\|\mathbf{x}^{(k)} - \mathbf{x}^*\| \leq \|(P - I)^{-1}\| \|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\|.$$

Posso scegliere come criterio di arresto:

$$\|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\| < tol, \qquad \text{CRITERIO DELL'INCREMENTO,}$$

dove *tol* rappresenta una quantità stabilita dall'utente. Si osservi però che se $\|(P-I)^{-1}\|$ è molto grande, questo criterio non garantisce un buon controllo per l'errore, ovvero non è detto che $\|(P-I)^{-1}\| \cdot tol < \varepsilon$.

Un altro criterio di arresto molto usato è quello che prende in considerazione il *residuo* al passo k :

$$\mathbf{r}^{(k)} = \mathbf{b} - A\mathbf{x}^{(k)} = A\mathbf{x}^* - A\mathbf{x}^{(k)} = A(\mathbf{x}^* - \mathbf{x}^{(k)}). \tag{3.32}$$

In tal caso il procedimento viene arrestato quando

$$\frac{\|\mathbf{r}^{(k)}\|}{\|\mathbf{b}\|} < tol, \qquad \text{CRITERIO DEL RESIDUO.}$$

Da (3.32) si ottiene che

$$\mathbf{x}^* - \mathbf{x}^{(k)} = A^{-1}\,\mathbf{r}^{(k)}$$

e quindi

$$\|\mathbf{x}^* - \mathbf{x}^{(k)}\| \leq \|A^{-1}\|\,\|\mathbf{r}^{(k)}\|.$$

Essendo $A\,\mathbf{x}^* = \mathbf{b}$, $\|\mathbf{b}\| = \|A\,\mathbf{x}^*\| \leq \|A\|\,\|\mathbf{x}^*\|$ segue che

$$\frac{\|\mathbf{x}^{(k)} - \mathbf{x}^*\|}{\|\mathbf{x}^*\|} \leq \|A^{-1}\|\|A\|\frac{\|\mathbf{r}^{(k)}\|}{\|\mathbf{b}\|} \leq \kappa(A) \cdot tol.$$

Pertanto, se A è una matrice malcondizionata questo criterio fallisce!

Osservazione 3.9 *In un programma che implementa un metodo iterativo deve essere comunque previsto un controllo sul numero massimo di iterazioni che l'utente è disposto a fare. Infatti, se il criterio di arresto dovesse fallire, si riesce a bloccare l'esecuzione del programma.*

Costo computazionale: Ad ogni iterazione è principalmente determinato da un prodotto matrice-vettore (n^2 operazioni moltiplicative). Se A è sparsa il costo si riduce ad un $O(n)$. Inoltre, se la matrice ha particolari proprietà di struttura, è possibile anche ridurre l'ingombro della memoria.

Si consideri la decomposizione della matrice A :

$$A = D - L - U$$

dove $D = \text{diag}(a_{11}, a_{22}, \ldots, a_{nn})$,

$$l_{ij} = \begin{cases} -a_{ij} & i > j \\ 0 & \text{altrimenti} \end{cases}, \qquad u_{ij} = \begin{cases} -a_{ij} & i < j \\ 0 & \text{altrimenti} \end{cases}.$$

Scegliendo

$$\begin{aligned} M &= D, \quad N = L + U && \textit{metodo di Jacobi} \\ M &= D - L, \quad N = U && \textit{metodo di Gauss-Seidel}. \end{aligned}$$

Per queste decomposizioni risulta

$$\det M \neq 0 \quad \Leftrightarrow \quad \text{elementi diagonali di } A \neq 0.$$

Sia

$$J = D^{-1}(L + U)$$

la matrice di iterazione del metodo di Jacobi. Il procedimento iterativo in questo caso diventa

$$\mathbf{x}^{(k)} = J\,\mathbf{x}^{(k-1)} + D^{-1}\mathbf{b}$$

e, in componenti:

$$x_i^{(k)} = \frac{1}{a_{ii}}\left(b_i - \sum_{j=1, j\neq i}^{n} a_{ij} x_j^{(k-1)}\right), \qquad i = 1, 2, \ldots, n.$$

Tale metodo è anche detto *metodo degli spostamenti simultanei* in quanto le componenti del vettore $\mathbf{x}^{(k)}$ sostituiscono simultaneamente al termine dell'iterazione le componenti del vettore $\mathbf{x}^{(k-1)}$.
Sia

$$G = (D - L)^{-1} U$$

la matrice di iterazione del metodo di Gauss-Seidel. Il procedimento iterativo in questo caso diventa

$$\mathbf{x}^{(k)} = G\,\mathbf{x}^{(k-1)} + (D - L)^{-1}\mathbf{b}$$

ovvero

$$(D - L)\,\mathbf{x}^{(k)} = U\,\mathbf{x}^{(k-1)} + \mathbf{b}$$

e quindi $D\,\mathbf{x}^{(k)} = L\,\mathbf{x}^{(k)} + U\,\mathbf{x}^{(k-1)} + \mathbf{b}$ che implica

$$\mathbf{x}^{(k)} = D^{-1}\left(L\,\mathbf{x}^{(k)} + U\,\mathbf{x}^{(k-1)} + \mathbf{b}\right)$$

e, in componenti:

$$x_i^{(k)} = \frac{1}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k-1)}\right), \qquad i = 1, 2, \ldots, n.$$

Tale metodo è anche detto *metodo degli spostamenti successivi.*

Osservazione 3.10 *Per l'implementazione del metodo di Jacobi è necessario disporre, contemporaneamente, dei due vettori* $\mathbf{x}^{(k)}$ *e* $\mathbf{x}^{(k-1)}$; *mentre nel metodo di Gauss-Seidel è sufficiente disporre di un solo vettore.*

Osservazione 3.11 *In molte applicazioni il metodo di Gauss-Seidel, che utilizza immediatamente i valori calcolati nella iterazione corrente, risulta più veloce del metodo di Jacobi. Esistono però anche dei casi in cui il metodo di Jacobi converge ed è più veloce del metodo di Gauss-Seidel. In generale, la convergenza del metodo di Gauss-Seidel non implica quella del metodo di Jacobi e viceversa.*

Osservazione 3.12 *Se* A *è diagonale dominante in senso forte allora convergono entrambe i metodi (Jacobi e Gauss-Seidel). Nel caso di matrici tridiagonali* $\rho(G) = (\rho(J))^2$.

Capitolo 4

Interpolazione ed approssimazione

In molti problemi si ha a che fare con una funzione $f(x)$ di forma non elementare, o addirittura sconosciuta, di cui si possiede solo una tabulazione in un numero finito di punti (sovente si tratta di misurazioni sperimentali). In questi casi la stima di un valore di $f(x)$, in un punto diverso da quelli in cui è data, può essere fatta utilizzando i dati disponibili. Questa operazione, detta *interpolazione,* di solito si effettua sostituendo a $f(x)$ una funzione che sia facilmente calcolabile come, per esempio, un polinomio. C'è quindi connessione tra il problema dell'interpolazione e quello più generale della *approssimazione* di una funzione $f(x)$, cioè della sostituzione di $f(x)$ con una funzione più semplice e che si discosti da $f(x)$ il meno possibile. Per misurare lo scostamento da $f(x)$ esistono vari criteri che danno luogo ad altrettanti metodi di approssimazione. In questo capitolo si descrivono alcune tecniche di interpolazione e l'approssimazione di una funzione con il metodo dei minimi quadrati nel caso discreto.

4.1 Interpolazione polinomiale

Siano dati $k+1$ punti reali $x_0, x_1, \ldots, x_k \in I$, due a due distinti, in corrispondenza dei quali siano noti i $k+1$ valori reali $f(x_0), f(x_1), \ldots, f(x_k)$. L'*interpolazione polinomiale* (o *parabolica*) consiste nel determinare un polinomio di grado al più k

$$p_k(x) = a_k\, x^k + a_{k-1}\, x^{k-1} + \cdots + a_1\, x + a_0 \tag{4.1}$$

tale che

$$p_k(x_i) = f(x_i), \quad i = 0, 1, \ldots, k; \tag{4.2}$$

il polinomio $p_k(x)$ si chiama *polinomio di interpolazione*. Nell'insieme dei polinomi (4.1) ne esiste uno ed uno solo che verifica (4.2). Infatti, imponendo che il polinomio (4.1) verifichi le condizioni (4.2) si ottiene il sistema lineare di $k+1$ equazioni nelle $k+1$ incognite a_i, $i = 0, 1, \ldots, k$:

$$\begin{cases} a_0 + a_1\, x_0 + a_2\, x_0^2 + \cdots + a_{k-1}\, x_0^{k-1} + a_k x_0^k &= f(x_0) \\ \qquad\qquad \vdots & \qquad\quad \vdots \\ a_0 + a_1\, x_k + a_2\, x_k^2 + \cdots + a_{k-1}\, x_k^{k-1} + a_k x_k^k &= f(x_k) \end{cases}$$

o, equivalentemente, ponendo

$$V = \begin{pmatrix} 1 & x_0 & \ldots & x_0^{k-1} & x_0^k \\ 1 & x_1 & \ldots & x_1^{k-1} & x_1^k \\ \vdots & \vdots & & \vdots & \vdots \\ 1 & x_k & \ldots & x_k^{k-1} & x_k^k \end{pmatrix}$$

$\mathbf{a} = (a_0, a_1, \ldots, a_k)^T$, $\mathbf{f} = (f(x_0), f(x_1), \ldots, f(x_k))^T$,

$$V\,\mathbf{a} = \mathbf{f}.$$

La matrice V è detta *matrice di Vandermonde*; il suo determinante è

$$\det V = \prod_{0 \le j < i \le k} (x_i - x_j)$$

ed è diverso da zero essendo i punti x_i distinti a due a due. Il sistema ha quindi una unica soluzione e quindi il polinomio cercato è unico.

Per la costruzione di $p_k(x)$ esistono procedimenti più pratici che non la risoluzione del sistema precedente. Una seconda forma del polinomio di interpolazione si può ottenere utilizzando le seguenti funzioni polinomiali di grado k :

$$L_r(x) = \frac{(x - x_0) \cdots (x - x_{r-1})(x - x_{r+1}) \cdots (x - x_k)}{(x_r - x_0) \cdots (x_r - x_{r-1})(x_r - x_{r+1}) \cdots (x_r - x_k)}, \quad r = 0, 1, \ldots, k.$$

Tali polinomi, detti *polinomi (fondamentali della interpolazione) di Lagrange*, godono della proprietà

$$L_r(x_s) = \begin{cases} 1 & \text{se } s = r \\ 0 & \text{altrimenti,} \end{cases} \qquad r, s = 0, 1, \ldots, k.$$

Di conseguenza il polinomio

$$p_k(x) = \sum_{r=0}^{k} L_r(x)\, f(x_r)$$

verifica le condizioni (4.2). In tal caso il polinomio $p_k(x)$ si chiama *polinomio di interpolazione di Lagrange* poiché definito attraverso la base di Lagrange. Questa tecnica è di difficile estensione: se si aggiungono nuovi punti bisogna rifare tutti i conti! In tal caso è possibile esprimere il polinomio interpolante in un'altra base più opportuna in tali casi. A tal fine si devono introdurre le differenze divise.

Definizione 4.1 *Sia* $f : I \subseteq \mathbb{R} \to \mathbb{R}$ *e siano* $x_0, x_1, \ldots, x_{k-1} \in I$, *con* $x_i \neq x_j$ *se* $i \neq j$. *La funzione*

$$f[x_0, x_1, \ldots, x_{k-1}, x] = \frac{f[x_0, x_1, \ldots, x_{k-2}, x] - f[x_0, x_1, \ldots, x_{k-1}]}{x - x_{k-1}}$$

ove, per $k = 1$, $f[x_0, x] = \frac{f(x)-f(x_0)}{x-x_0}$ *è definita per ogni* $x \in I, x \neq x_i, i = 0, 1, \ldots, k-1$, *e si chiama* differenza divisa di ordine k.

Il polinomio interpolante si può quindi scrivere usando la *base di Newton*:

$$1,\ (x - x_0),\ (x - x_0)(x - x_1),\ \ldots,\ (x - x_0)(x - x_1)\cdots(x - x_{k-1})$$

nel seguente modo:

$$\begin{aligned} p_k(x) \quad = \quad & f(x_0) + (x - x_0)f[x_0, x_1] + (x - x_0)(x - x_1)f[x_0, x_1, x_2] + \\ & + (x - x_0)(x - x_1)\cdots(x - x_{k-1})f[x_0, x_1, \ldots, x_k]. \end{aligned}$$

Tale polinomio è detto *polinomio di interpolazione di Newton.* È facile provare che esso verifica le condizioni di interpolazione (4.2).

Esempio 4.1 *Data la tabella di valori*

x	-2	0	-1	2	α
$f(x)$	-13	5	3α	15α	5

determinare per quali α *il polinomio di interpolazione risulta di grado minimo.*
Si costruisce il quadro delle differenze divise:

x	$f(x)$	*DD1*	*DD2*	*DD3*
x_0	$f(x_0)$			
x_1	$f(x_1)$	$f[x_0, x_1]$		
x_2	$f(x_2)$	$f[x_0, x_2]$	$f[x_0, x_1, x_2]$	
x_3	$f(x_3)$	$f[x_0, x_3]$	$f[x_0, x_1, x_3]$	$f[x_0, x_1, x_2, x_3]$

Osservazione 4.1 *Nel caso in cui gli elementi della colonna delle differenze divise di ordine r (DDr) risultino uguali fra loro, gli elementi delle colonne successive sono nulli, perciò il polinomio di interpolazione ha grado r.*

x	$f(x)$	$DD1$	$DD2$	$DD3$
-2	-13			
0	5	9		
-1	3α	$3\alpha+13$	$-(3\alpha+4)$	
2	15α	$\frac{15\alpha+13}{4}$	$\frac{15\alpha-23}{8}$	$\frac{13\alpha+3}{8}$
α	5	$\frac{18}{\alpha+2}$	$-\frac{9}{\alpha+2}$	$\frac{3\alpha^2+10\alpha-1}{(\alpha+2)(\alpha+1)}$

Dalla tabella precedente si ricava che il polinomio di interpolazione ha grado minimo per i valori di α che rendono uguali le ultime due differenze divise:

$$\frac{13\alpha+3}{8} = \frac{3\alpha^2+10\alpha-1}{(\alpha+2)(\alpha+1)}$$

da cui

$$\begin{aligned}(13\alpha+3)(\alpha+2)(\alpha+1) &= 8(3\alpha^2+10\alpha-1)\\ (13\alpha+3)(\alpha^2+3\alpha+2) &= 24\alpha^2+80\alpha-8\\ 13\alpha^3+39\alpha^2+26\alpha+3\alpha^2+9\alpha+6 &= 24\alpha^2+80\alpha-8.\end{aligned}$$

Allora

$$13\alpha^3+18\alpha^2-45\alpha+14=0$$

le cui radici sono $\alpha_1 = 1$, i.e., $p(1) = 0$:

	13	18	-45	14
$-(1)$		-13	-31	14
	13	31	-14	0

$$(\alpha-1)(13\alpha^2+31\alpha-14)=0$$

e quindi

$$\alpha_{2,3} = \frac{-31\pm\sqrt{1689}}{26}.$$

Per $\alpha = 1$ il polinomio di interpolazione è:

$$\begin{aligned}p(x) &= -13+(x-(-2))\cdot 9+(x+2)x\cdot(-7)+(x+2)x(x+1)\cdot 2\\ &= -13+9(x+2)-7(x^2+2x)+2(x^3+x^2+2x^2+2x)\\ &= -13+9x+18-7x^2-14x+2x^3+6x^2+4x\\ &= 2x^3-x^2-x+5.\end{aligned}$$

4.1.1 Errore nella interpolazione polinomiale

Tranne nel caso in cui $f(x)$ sia un polinomio, $p_k(x)$ non coinciderà con $f(x)$. Cerchiamo di ricavare l'errore nell'interpolazione. Si osservi che

$$\begin{aligned} f(x) &= f(x_0) + (x - x_0)\frac{f(x) - f(x_0)}{x - x_0} \\ &\equiv f(x_0) + (x - x_0)f[x_0, x] \end{aligned}$$

$$\begin{aligned} f[x_0, x] &= f[x_0, x_1] + (x - x_1)\frac{f[x_0, x] - f[x_0, x_1]}{x - x_1} \\ &\equiv f[x_0, x_1] + (x - x_1)f[x_0, x_1, x]. \end{aligned}$$

Sostituendo la seconda relazione nella prima si ottiene che

$$f(x) = f(x_0) + (x - x_0)f[x_0, x_1] + (x - x_0)(x - x_1)f[x_0, x_1, x].$$

Iterando il procedimento precedente si perviene alla identità:

$$\begin{aligned} f(x) &= \sum_{j=0}^{k} f[x_0, \ldots, x_j](x - x_0)(x - x_1)\cdots(x - x_{j-1}) + \\ &\quad + f[x_0, \ldots, x_k, x](x - x_0)(x - x_1)\cdots(x - x_k) \\ &\equiv p_k(x) + E_k(x) \end{aligned}$$

dove si è posto

$$E_k(x) = (x - x_0)(x - x_1)\cdots(x - x_k)f[x_0, \ldots, x_k, x].$$

L'errore $E_k(x)$ è nullo nei nodi; infatti $f(x_i) = p_k(x_i), i = 0, 1, \ldots, k$. Essendo il polinomio interpolante che passa per i punti $(x_i, f(x_i))$ unico, allora anche l'errore $E_k(x)$ è UNICO.

Supponiamo che $f(x)$ sia derivabile $k+1$ volte in $[a, b] \equiv I$, intervallo che contiene tutti i nodi x_i. Ha quindi senso cercare in $[a, b]$ una rappresentazione del tipo

$$E_k(x) = (x - x_0)(x - x_1)\cdots(x - x_k)R_k(x)$$

così che

$$f(x) = p_k(x) + (x - x_0)(x - x_1)\cdots(x - x_k)R_k(x).$$

Definiamo la funzione ausiliaria

$$G(t) = f(t) - p_k(t) - (t - x_0)(t - x_1)\cdots(t - x_k)R_k(x)$$

e supponiamo che $x \neq x_i$. La funzione $G(t)$ si annulla in $k+2$ punti: $x, x_0, x_1, \ldots, x_k$. Il teorema di Rolle assicura l'esistenza di $k+1$ punti distinti in cui la $G'(t) = 0$. Lo stesso teorema applicato a $G'(t)$ assicura che esistono k punti tale che la derivata seconda $G''(t) = 0$. Quindi, applicando il teorema di Rolle ripetutamente si ottiene che esiste un punto ξ tale che $G^{(k+1)}(\xi) = 0$, ovvero

$$G^{(k+1)}(\xi) = f^{(k+1)}(\xi) - (k+1)!R_k(x) = 0.$$

Da quest'ultima espressione si ricava che

$$R_k(x) = \frac{f^{(k+1)}(\xi)}{(k+1)!}$$

e quindi

$$f(x) = p_k(x) + \frac{(x-x_0)(x-x_1)\cdots(x-x_k)}{(k+1)!} f^{(k+1)}(\xi), \qquad \xi \in (a,b).$$

Tale rappresentazione dell'errore riveste un'importanza teorica in quanto il suo utilizzo richiede non solo la determinazione di $f^{(k+1)}(x)$, ma anche la sua valutazione in un punto ξ non noto!

Osservazione 4.2 *Se $f(x)$ è un polinomio di grado k, $f^{(k+1)}(x) = 0$ e dunque $f(x) = p_k(x)$, cioè il polinomio interpolante è il polinomio stesso.*

L'errore nell'interpolazione polinomiale:

$$E_k(x) = (x-x_0)(x-x_1)\cdots(x-x_k) f[x_0, \ldots, x_k, x]$$

è quindi costituito da due fattori:

1. $f[x_0, \ldots, x_k, x]$ che dipende dalla funzione e dalla sua regolarità. Fissata $f(x)$ si può fare ben poco per modificarlo;
2. $(x-x_0)(x-x_1)\cdots(x-x_k)$ che dipende dalla distribuzione dei nodi di interpolazione.

Dunque

$$\|E_k(x)\|_\infty \leq \|(x-x_0)(x-x_1)\cdots(x-x_k)\|_\infty \|f[x_0, \ldots, x_k, x]\|_\infty.$$

È possibile scegliere i nodi $x_i, i = 0, 1, \ldots, k$ in modo da minimizzare $\|(x-x_0)(x-x_1)\cdots(x-x_k)\|_\infty$? La risposta è affermativa. Basta prendere come nodi gli zeri dei polinomi di Chebyshev:

$$\begin{cases} T_k(x) &= 2\,x\,T_{k-1} - T_{k-2}, \qquad k \geq 2 \\ T_0(x) &= 1 \\ T_1(x) &= x. \end{cases}$$

4.2 Approssimazione polinomiale

Non sempre i polinomi di interpolazione sono adatti per approssimare una funzione continua con una data accuratezza su tutto un intervallo. Infatti, per un noto *teorema di Bernstein,* dato un intervallo $[a, b]$ e fissati in esso $k + 1$ punti esiste certamente qualche funzione $f(x)$ continua su $[a, b]$ con la proprietà che la successione dei polinomi interpolanti $p_1(x), p_2(x), \ldots,$ di grado pari all'indice, non converga uniformemente ad $f(x)$. Tuttavia se, per esempio, si considerano i polinomi di grado k

$$B_k(f; x) = \sum_{i=0}^{k} \frac{k!}{i!\,(k-i)!} x^i (1-x)^{k-i} f\left(\frac{i}{k}\right)$$

si dimostra che la successione di polinomi $\{B_k(f; x)\}$ converge uniformemente a $f(x)$ su $[0, 1]$.

L'approssimazione (polinomiale o no) di una funzione $f(x)$ può anche ottenersi richiedendo, ad esempio, che la funzione approssimante minimizzi una opportuna norma euclidea. Con questo criterio, detto dei *minimi quadrati*, si possono costruire approssimazioni che utilizzano un insieme discreto di punti dati $(x_i, f(x_i)), i = 0, 1, \ldots, k$, senza necessariamente imporre al grafico della funzione approssimante di passare per quei punti, come nel caso dell'interpolazione.

4.2.1 Metodo dei minimi quadrati nel discreto

Siano date $m+1$ funzioni $\phi_i(x), i = 0, 1, \ldots, m$, continue almeno su un insieme I contenente $k + 1$ punti $x_j, j = 0, 1, \ldots, k$, con $k \geq m$, e si abbiano i valori $f(x_j)$ per ogni x_j. Si consideri la funzione combinazione lineare delle $\phi_i(x)$

$$\Phi(x) = c_0\phi_0(x) + c_1\phi_1(x) + \cdots + c_m\phi_m(x)$$

dove $c_i \in \mathbb{R}, i = 0, 1, \ldots, m$. Si vuole approssimare $f(x)$ con $\Phi(x)$. Evidentemente, la funzione $\Phi(x)$ dipende dalla scelta dei coefficienti c_i. Il metodo dei minimi quadrati consiste nello scegliere i coefficienti c_i per i quali la funzione

$$\Psi(c_0, c_1, \ldots, c_m) = \sum_{j=0}^{k} \left[\sum_{i=0}^{m} c_i\, \phi_i(x_j) - f(x_j) \right]^2$$

assume valore minimo. Ponendo $\mathbf{c} = (c_0, c_1, \ldots, c_m)^T$ tale funzione si può equivalentemente riscrivere nel seguente modo:

$$\Psi(\mathbf{c}) = \left\| \begin{pmatrix} \Phi(x_0) - f(x_0) \\ \Phi(x_1) - f(x_1) \\ \vdots \\ \Phi(x_k) - f(x_k) \end{pmatrix} \right\|_2^2 . \tag{4.3}$$

Pertanto, $\Psi(\mathbf{c})$ rappresenta la somma degli *scarti quadratici* tra la funzione $\Phi(x)$ e la funzione $f(x)$ nei punti x_j e coincide con il quadrato della norma euclidea del vettore di componenti $\Phi(x_j) - f(x_j)$. Da qui la denominazione del metodo. In forma matriciale, ponendo $\mathbf{b} = (f(x_0), f(x_1), \ldots, f(x_k))^T$

$$A = \begin{pmatrix} \phi_0(x_0) & \phi_1(x_0) & \ldots & \phi_m(x_0) \\ \phi_0(x_1) & \phi_1(x_1) & \ldots & \phi_m(x_1) \\ \ldots & \ldots & \ldots & \ldots \\ \phi_0(x_k) & \phi_1(x_k) & \ldots & \phi_m(x_k) \end{pmatrix}_{(k+1)\times(m+1)}$$

la (4.3) può scriversi:

$$\Psi(\mathbf{c}) \;=\; \|A\mathbf{c} - \mathbf{b}\|_2^2.$$

Si consideri la fattorizzazione QR di A con Q matrice ortogonale quadrata di ordine $k+1$ ed R matrice triangolare superiore di dimensione $(k+1)\times(m+1)$. In particolare

$$R = \begin{pmatrix} \hat{R} \\ O \end{pmatrix}, \qquad \hat{R} \in \mathbb{R}^{(m+1)\times(m+1)}.$$

Essendo $\|Q\mathbf{v}\|_2^2 = \mathbf{v}^T Q^T Q \mathbf{v} = \mathbf{v}^T\mathbf{v} = \|\mathbf{v}\|_2^2$, si ha

$$\Psi(\mathbf{c}) \;=\; \|Q(R\mathbf{c} - Q^T\mathbf{b})\|_2^2 = \|R\mathbf{c} - Q^T\mathbf{b}\|_2^2.$$

Posto

$$Q^T\mathbf{b} = \begin{pmatrix} \hat{\mathbf{b}} \\ \tilde{\mathbf{b}} \end{pmatrix}$$

con $\hat{\mathbf{b}} \in \mathbb{R}^{m+1}, \tilde{\mathbf{b}} \in \mathbb{R}^{k-m}$, la precedente diventa:

$$\Psi(\mathbf{c}) \;=\; \left\| \begin{pmatrix} \hat{R}\mathbf{c} - \hat{\mathbf{b}} \\ \mathbf{0} - \tilde{\mathbf{b}} \end{pmatrix} \right\|_2^2 = \|\hat{R}\mathbf{c} - \hat{\mathbf{b}}\|_2^2 + \|\tilde{\mathbf{b}}\|_2^2.$$

Allora basta scegliere $\mathbf{c}$ tale che $\hat{R}\mathbf{c} = \hat{\mathbf{b}}$ (ricordo che $\hat{R}$ è triangolare superiore!).

Le scelte più comuni per le funzioni $\phi_i(x)$ sono:

- $\phi_0(x) = 1$, $\phi_1(x) = x$, $\phi_2(x) = x^2, \ldots$;
- $\phi_0(x) = \frac{1}{2}$, $\phi_1(x) = \cos\, x$, $\phi_2(x) = \sin\, x$, $\phi_3(x) = \cos\,(2x), \ldots$;
- $\phi_0(x) = e^{\alpha_0 x}$, $\phi_1(x) = e^{\alpha_1 x}$, $\phi_2(x) = e^{\alpha_2 x}, \ldots$, $\alpha_0, \alpha_1, \alpha_2, \cdots \in \mathbb{R}, \alpha_r \neq \alpha_s$, se $r \neq s$.

Si ha così la *migliore approssimazione* $\Phi(x)$ *nel senso dei minimi quadrati* rispettivamente di tipo polinomiale, trigonometrica ed esponenziale.

Capitolo 5

Integrazione numerica

Il problema che andiamo ad affrontare in questo capitolo è il calcolo di integrali definiti del tipo

$$I(\rho f) = \int_a^b \rho(x) f(x)\, dx, \qquad a < b, \tag{5.1}$$

in cui $\rho(x)$ è una funzione positiva in $[a, b]$, detta *funzione peso* e $[a, b]$ è un intervallo limitato. La trattazione sarà qui confinata al caso in cui la funzione f è sufficientemente regolare, nel senso che non ha singolarità in $[a, b]$. In particolare, nel seguito assumeremo che f sia almeno continua in $[a, b]$.

I metodi che andiamo ad esaminare sono definiti mediante l'integrale di una approssimazione polinomiale (o polinomiale a tratti) di f in quanto l'integrale di un polinomio si può calcolare, senza approssimazioni, in modo assai semplice.

5.1 Formule di tipo interpolatorio

Scegliamo $(n + 1)$ punti nell'intervallo $[a, b]$ in modo tale che risulti:

$$x_0 < x_1 < \cdots < x_n.$$

Consideriamo il polinomio interpolante i nodi $(x_i, f(x_i)), i = 0, 1, \ldots, n$, ed espresso nella base di Lagrange per approssimare la funzione f :

$$f(x) = \sum_{i=0}^{n} f(x_i) L_i(x) + E_n(x).$$

Allora

$$\begin{aligned} I(\rho f) &= \int_a^b \rho(x) f(x)\,dx = \int_a^b \rho(x)\left[\sum_{i=0}^{n} f(x_i)L_i(x) + E_n(x)\right] dx \\ &= \int_a^b \rho(x)\sum_{i=0}^{n} f(x_i)L_i(x)\,dx + \int_a^b \rho(x)E_n(x)\,dx \\ &= \sum_{i=0}^{n} f(x_i)\int_a^b \rho(x)L_i(x)\,dx + \int_a^b \rho(x)E_n(x)\,dx \\ &:= \sum_{i=0}^{n} f(x_i)a_i + E_n(\rho f). \end{aligned}$$

I valori $a_i = \int_a^b \rho(x)L_i(x)\,dx$ sono detti *pesi* o *coefficienti*. Posto

$$J_n(\rho f) = \sum_{i=0}^{n} f(x_i)a_i,$$

si ha

$$I(\rho f) = J_n(\rho f) + E_n(\rho f).$$

La quantità $J_n(\rho f)$ è detta *formula di quadratura* (o di integrazione) *di tipo interpolatorio*, x_i sono detti *nodi*, mentre

$$\begin{aligned} E_n(\rho f) &= I(\rho f) - J_n(\rho f) = \int_a^b \rho(x)E_n(x)\,dx \\ &= \int_a^b \rho(x)(x-x_0)(x-x_1)\cdots(x-x_n)\frac{f^{(n+1)}(\xi)}{(n+1)!}\,dx \end{aligned}$$

è detto *errore* della formula di integrazione.

Definizione 5.1 *La formula di integrazione* $J_n(\rho f) = \sum_{i=0}^{n} f(x_i)a_i$ *ha* grado di precisione $d \in \mathbb{N}$ *se risulta esatta per tutti i polinomi di grado minore od uguale a* d*, cioè se si verifica*

$$E_n(\rho \cdot 1) = E_n(\rho \cdot x) = E_n(\rho \cdot x^2) = \cdots = E_n(\rho \cdot x^d) = 0, \quad E_n(\rho \cdot x^{d+1}) \neq 0.$$

Osserviamo che la formula $J_n(\rho f)$ è individuata una volta che lo siano i nodi x_i ed i pesi a_i. Una prima distinzione tra le varie formule riguarda la scelta dei nodi x_i. Essi possono essere:

1) fissati a priori;

2) scelti in modo da minimizzare l'errore $E_n(\rho f)$.

5.2 Formule di Newton-Cotes

Queste formule si ottengono fissando:

1) $\rho(x) = 1$;

2) $x_i = a + ih$, $i = 0, 1, \ldots, n$, con $h = b - a/n$ detto *passo della formula*.

Allora, considerato che $a = x_0 < x_1 < \cdots < x_{n-1} < x_n = b$, le formule di Newton-Cotes qui definite sono dette *chiuse* per distinguerle da formule analoghe, dette *aperte*, nelle quali si ha $a < x_0 < \cdots < x_n < b$. Quindi, in questo caso, si vuole calcolare

$$I(f) = \int_a^b f(x)\,dx, \tag{5.2}$$

considerando che

$$f(x) = \sum_{i=0}^{n} f(x_i)L_i(x) + (x - x_0)(x - x_1)\cdots(x - x_n)\frac{f^{(n+1)}(\xi)}{(n+1)!}.$$

Allora, usando la trasformazione

$$x = x_0 + th$$

$$L_i(x) = \prod_{\substack{j=0 \\ j\neq i}}^{n} \frac{x - x_j}{x_i - x_j} = \prod_{\substack{j=0 \\ j\neq i}}^{n} \frac{(x_0 + th) - (x_0 + jh)}{(x_0 + ih) - (x_0 + jh)} = \prod_{\substack{j=0 \\ j\neq i}}^{n} \frac{t - j}{i - j}$$

e quindi

$$\begin{aligned} I(f) &= \int_a^b \sum_{i=0}^{n} f(x_i)L_i(x)\,dx + \int_a^b \prod_{j=0}^{n}(x - x_j)\frac{f^{(n+1)}(\xi)}{(n+1)!}\,dx \\ &= \sum_{i=0}^{n} f(x_i)\int_a^b L_i(x)\,dx + \frac{1}{(n+1)!}\int_a^b f^{(n+1)}(\xi)\prod_{j=0}^{n}(x - x_j)\,dx. \end{aligned}$$

Essendo

$$dx = h\,dt, \quad x = a \equiv x_0 \Rightarrow t = 0, \quad x = b \equiv x_n \Rightarrow t = n,$$

$$\begin{aligned} I(f) &= \sum_{i=0}^{n} f(x_i)\int_0^n \prod_{\substack{j=0 \\ j\neq i}}^{n} \frac{t - j}{i - j} \cdot h\,dt + \frac{1}{(n+1)!}\int_0^n f^{(n+1)}(\xi)\prod_{j=0}^{n}(t - j)h \cdot h\,dt \\ &\equiv h\sum_{i=0}^{n} f(x_i)\cdot c_i + \frac{h^{n+2}}{(n+1)!}\int_0^n f^{(n+1)}(\xi)\prod_{j=0}^{n}(t - j)\,dt = J(f) + E(f). \end{aligned}$$

Quando $n = 1$ si ha

$$x_0 = a,\ x_1 = b, \qquad h = b - a$$

e quindi

$$L_0(x) = \frac{x - x_1}{x_0 - x_1} = \frac{x - b}{a - b}, \qquad L_1(x) = \frac{x - x_0}{x_1 - x_0} = \frac{x - a}{b - a}$$

Allora

$$\begin{aligned} c_0 &= \int_0^1 \frac{t-1}{0-1}\,dt = -\frac{t^2}{2}\Big|_0^1 + t|_0^1 = -\frac{1}{2} + 1 = \frac{1}{2} \\ c_1 &= \int_0^1 \frac{t-0}{1-0}\,dt = \frac{t^2}{2}\Big|_0^1 = \frac{1}{2} \end{aligned}$$

Ne segue che

$$J(f) = h\left(c_0 f(x_0) + c_1 f(x_1)\right) = \frac{b-a}{2}\left(f(a) + f(b)\right).$$

Questa formula di quadratura è nota come *formula dei trapezi* (o formula trapezoidale). Il nome di questa formula deriva dal suo evidente significato geometrico di approssimare l'area sottesa dal grafico di f con quella del trapezio di vertici $(a, 0), (a, f(a)), (b, 0), (b, f(b))$.

Per quanto riguarda l'errore di tale formula di quadratura, bisogna innanzitutto richiamare il seguente risultato:

Teorema 5.1 (della media integrale) *Siano f e $g \in C([a, b])$. Inoltre supponiamo che g non cambi segno in $[a, b]$. Allora $\exists\, \eta \in [a, b]$ tale che*

$$\int_a^b f(x)g(x)\,dx = f(\eta)\int_a^b g(x)\,dx.$$

Allora si ha che

$$\begin{aligned} E(f) &= \frac{h^3}{2!}\int_0^1 f''(\xi)\prod_{j=0}^{1}(t-j)\,dt = \frac{(b-a)^3}{2} f''(\eta)\int_0^1 t(t-1)\,dt \\ &= \frac{h^3}{2} f''(\eta)\left(\frac{t^3}{3}\Big|_0^1 - \frac{t^2}{2}\Big|_0^1\right) = -\frac{1}{12} f''(\eta) h^3. \end{aligned}$$

Si conclude dunque che, quando $n = 1$, si ha:

$$I(f) = \frac{b-a}{2}\left(f(a) + f(b)\right) - \frac{1}{12} f''(\eta)(b-a)^3.$$

È immediato verificare che, posto $f(x) = x^r, r = 0, 1, 2, \ldots$:

$$E(1) = E(x) = 0, \quad E(x^2) = -\frac{h^3}{6},$$

ovvero il grado di precisione del metodo dei trapezi è 1.

Quando $n = 2$ si ha

$$x_0 = a,\ x_1 = a + h,\ x_2 = b, \qquad h = \frac{b-a}{2}.$$

Pertanto

$$\begin{aligned} c_0 &= \int_0^2 \prod_{j=1}^{2} \frac{t-j}{-j} dt = \int_0^2 \frac{(t-1)(t-2)}{2} dt = \frac{1}{3} \\ c_1 &= \int_0^2 \prod_{\substack{j=0 \\ j\neq 1}}^{2} \frac{t-j}{1-j} dt = \int_0^2 t(-t+2) dt = \frac{4}{3} \\ c_2 &= \int_0^2 \prod_{j=0}^{1} \frac{t-j}{2-j} dt = \int_0^2 \frac{t}{2}(t-1) dt = \frac{1}{3}. \end{aligned}$$

Inoltre, considerato che

$$x_1 = a + \frac{b-a}{2} = \frac{a+b}{2},$$

si ottiene la *formula di Simpson*:

$$J(f) = \frac{b-a}{2}\left(\frac{1}{3}f(a) + \frac{4}{3}f\left(\frac{a+b}{2}\right) + \frac{1}{3}f(b)\right).$$

Si può dimostrare che quando $n = 2$

$$E(f) = -\frac{1}{90} f^{(iv)}(\xi)\left(\frac{b-a}{2}\right)^5.$$

È immediato verificare che, posto $f(x) = x^r, r = 0, 1, 2, \ldots$:

$$E(1) = E(x) = E(x^2) = E(x^3) = 0, \quad E(x^4) = -\frac{4}{15}h^5,$$

ovvero il grado di precisione del metodo dei trapezi è 3.

5.2.1 Condizionamento del problema

Studiamo ora il condizionamento del problema (5.2) in cui la perturbazione è da intendersi sulla funzione integranda f. Denotando con $\tilde{f}$ la funzione perturbata, si ottiene quindi

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

Considerando che il termine $\|f - \tilde{f}\|$ misura l'errore sui dati iniziali e che l'errore sul risultato è ottenuto da questo amplificato di un fattore $(b-a)$, si conclude che $\kappa = b - a$ definisce il numero di condizionamento del problema (5.2).

Abbiamo visto che per $n = 1$ e $n = 2$ i coefficienti $c_i > 0, i = 0, 1, \ldots, n$. Più in generale, si può verificare che per $n \leq 6$ si ha che i pesi sono positivi. Al contrario, per $n > 6$ compaiono dei pesi negativi. Inoltre,

$$\frac{1}{n} \sum_{i=0}^{n} c_i = 1.$$

Infatti, considerando $f(x) = 1$, per ogni $n \geq 1$ si ha:

$$\begin{aligned} I(1) &= \int_a^b 1\, dx = b - a, \\ J(1) &= \frac{b-a}{n} \sum_{i=0}^{n} c_i. \end{aligned}$$

Sulla scorta di questi fatti, è possibile condurre l'analisi del condizionamento del calcolo della formula di quadratura. Al solito, denotando con $\tilde{f}$ la funzione perturbata, si ottiene

$$\begin{aligned} |J(f) - J(\tilde{f})| &= \frac{b-a}{n} \left| \sum_{i=0}^{n} (f(x_i) - \tilde{f}(x_i)) c_i \right| \leq \frac{b-a}{n} \sum_{i=0}^{n} \left| f(x_i) - \tilde{f}(x_i) \right| |c_i| \\ &\leq \left(\frac{b-a}{n} \sum_{i=0}^{n} |c_i| \right) \|f - \tilde{f}\|. \end{aligned}$$

Si conclude pertanto che

$$\kappa_n = \frac{b-a}{n} \sum_{i=0}^{n} |c_i|$$

definisce il numero di condizionamento del problema. Per quanto detto in precedenza $\kappa_n = \kappa$ per $n = 1, 2, \ldots, 6$, mentre $\kappa_n > \kappa$ per $n \geq 7$. Quindi solo per $n \leq 6$ i problemi del calcolo di $I(f)$ e $J(f)$ hanno lo stesso numero di condizionamento. Se ne conclude che le formule di Newton-Cotes sono convenientemente utilizzabili solo fino a $n = 6$.

Contrariamente a quanto si potrebbe pensare non conviene usare formule di Newton-Cotes di grado di precisione via via crescente. I pesi tendono a crescere e ad essere a segni alterni dando luogo a rilevanti errori di arrotondamento. Conviene invece usare un valore di n basso e un h piccolo, ossia suddividere l'intervallo $[a, b]$ in un certo numero di sottointervalli uguali e su ciascuno di essi applicare una formula di quadratura con basso grado (di precisione).

5.2.2 Formule composite o generalizzate

Le formule precedenti usano un unico polinomio interpolante su tutto l'intervallo $[a, b]$. Ciò presenta talvolta un inconveniente: se n è grande il polinomio può presentare forti oscillazioni!!!
Allora si preferisce usare formule che si ottengono dividendo l'intervallo $[a, b]$ in m parti uguali, mediante i punti $a = z_0 < z_1 < \cdots < z_m = b$ e utilizzando la proprietà:

$$I(f) = \sum_{r=0}^{m-1} \int_{z_r}^{z_{r+1}} f(x)\, dx.$$

Si ottengono le cosiddette *formule di Newton-Cotes generalizzate* (o composite) applicando una stessa formula a $n+1$ punti per ognuno degli m integrali a secondo membro. Si hanno $n \cdot m + 1$ nodi con un passo $h = (b-a)/(n \cdot m)$.

Caso $n = 1$ - *formula trapezoidale generalizzata*:

$$\begin{aligned} I(f) &= \int_a^b f(x)\, dx = \sum_{r=0}^{m-1} \int_{z_r}^{z_{r+1}} f(x)\, dx \\ &= \sum_{r=0}^{m-1} \left(\frac{z_{r+1} - z_r}{2} \left(f(z_r) + f(z_{r+1}) \right) - \frac{1}{12} f''(\xi)(z_{r+1} - z_r)^3 \right). \end{aligned}$$

Considerato che

$$[z_r, z_{r+1}] = [a + r(b-a)/m, a + (r+1)(b-a)/m]$$

si ha che

$$z_{r+1} - z_r = \frac{b-a}{m}$$

e quindi la precedente diventa:

$$\begin{aligned} I(f) &= \sum_{r=0}^{m-1} \frac{b-a}{2m}(f(z_r)+f(z_{r+1})) - \sum_{r=0}^{m-1} \frac{1}{12} f''(\xi) \left(\frac{b-a}{m}\right)^3 \\ &= \frac{b-a}{2m}\left(f(a)+2\sum_{r=1}^{m-1} f(z_r)+f(b)\right) - \frac{1}{12} f''(\xi)\left(\frac{b-a}{m}\right)^2 (b-a) \end{aligned}$$

Caso $n = 2$ - *formula di Simpson generalizzata*:

$$\begin{aligned} I(f) &= \int_a^b f(x)\,dx = \sum_{r=0}^{m-1} \int_{z_r}^{z_{r+1}} f(x)\,dx \\ &= \sum_{r=0}^{m-1} \frac{z_{r+1}-z_r}{2}\left(\frac{1}{3}f(z_r)+\frac{4}{3}f\left(\frac{z_r+z_{r+1}}{2}\right)+\frac{1}{3}f(z_{r+1})\right)+ \\ &\quad -\sum_{r=0}^{m-1}\frac{1}{90}f^{(iv)}(\xi)\left(\frac{z_{r+1}-z_r}{2}\right)^5 \\ &= \frac{b-a}{6\,m}\left(f(a)+4\sum_{r=0}^{m-1} f\left(\frac{z_r+z_{r+1}}{2}\right)+2\sum_{r=1}^{m-1} f(z_r)+f(b)\right) \\ &\quad -\frac{1}{180}f^{(iv)}(\xi)\left(\frac{b-a}{2m}\right)^4 (b-a). \end{aligned}$$

Si noti come l'errore tenda a zero al crescere di m. Essendo $h = (b-a)/(n\cdot m)$ gli errori sono dell'ordine di h^2 per $n = 1$ ed h^4 per $n = 2$.

5.3 Formule di tipo gaussiano

Si premettono brevemente alcune definizioni e proprietà di una particolare classe di polinomi: i *polinomi ortogonali*.
Si indichi con Π lo spazio vettoriale dei polinomi a coefficienti reali. Per ogni coppia $r(x), s(x) \in \Pi$ si consideri il prodotto scalare:

$$< r, s >=< s, r >= I(\rho\, r s) = \int_a^b \rho(x)\, r(x) s(x)\, dx. \tag{5.3}$$

Si definisce la classe dei polinomi ortogonali Π^* l'insieme dei polinomi, a coefficienti reali, ortogonali rispetto al prodotto scalare (5.3), cioè

$\Pi^* = \{q_i : q_i$ è un polinomio di grado di $i, < q_i, q_j >= h_i\delta_{ij}, i, j = 0, 1, 2, \dots\}$.

I numeri h_i sono dette costanti di normalizzazione e δ_{ij} denota il simbolo di Kronecker.

Teorema 5.2 *Data la funzione peso $\rho(x)$ e l'intervallo $[a,b]$ esiste un sistema di polinomi ortogonali, univocamente determinato a meno del coefficiente di grado massimo. Essi soddisfano alla relazione di ricorrenza a tre termini:*

$$\begin{cases} q_{i+1} & = \alpha_i(x-\beta_i)q_i - \gamma_i q_{i-1} \\ q_{-1} & = 0 \qquad q_0 = \alpha_{-1} \end{cases} \tag{5.4}$$

Quindi i coefficienti β_i, γ_i sono univocamente determinati una volta fissati gli α_i, i quali sono evidentemente legati ai coefficienti di grado massimo dei polinomi q_i.

Ecco alcuni polinomi ortogonali:

Polinomi	α_n	β_n	γ_n	$\rho(x)$	$[a,b]$
Chebychev di 1^a specie T_n $(T_0=1, T_1=x)$	2	0	1	$\frac{1}{\sqrt{1-x^2}}$	$[-1,1]$
Chebychev di 2^a specie U_n $(U_0=1, U_1=2x)$	2	0	1	$\sqrt{1-x^2}$	$[-1,1]$
Legendre $(P_0=1, P_1=x)$	$\frac{2n+1}{n+1}$	0	$\frac{n}{n+1}$	1	$[-1,1]$

Osservazione 5.1 *Per ragioni di convenienza, per gli integrali estesi ad un intervallo $[a,b]$, si usano polinomi ortogonali definiti in $[-1,1]$. In effetti ogni intervallo di integrazione $a \le t \le b$ può ricondursi all'intervallo $-1 \le x \le 1$ con la trasformazione*

$$t = \frac{b-a}{2}x + \frac{b+a}{2}$$

e la funzione $\rho(x)$ può essere comunque introdotta. Risulta infatti

$$\int_a^b g(t)\,dt = \int_{-1}^1 \rho(x) f(x)\,dx$$

ove si assuma $f(x) = \frac{b-a}{2\,\rho(x)} g\left(\frac{b-a}{2}x + \frac{b+a}{2}\right)$.

In definitiva, data la funzione peso e l'intervallo di integrazione, mediante la formula (5.4) si costituiscono i polinomi ortogonali fino all'n-mo, si determinano gli zeri di quest'ultimo e questi saranno i nodi della formula di integrazione. Naturalmente è necessario che questi zeri siano tutti reali ed interni all'intervallo $[a,b]$. Difatti è quello che avviene perché sussiste il seguente teorema.

Teorema 5.3 *Gli zeri dei polinomi ortogonali sono tutti reali ed interni ad* $[a, b]$.

Dim. Supponiamo per assurdo che gli zeri di un polinomio ortogonale di grado n in $[a, b]$ siano $k < n$, cioè che

$$q_n(x) = \prod_{i=1}^{k}(x - x_i)\,\phi(x)$$

ove $\phi(x) \neq 0$ in $[a, b]$. Il prodotto

$$\prod_{i=1}^{k}(x - x_i)\,q_n(x) = \prod_{i=1}^{k}(x - x_i)^2\,\phi(x)$$

non cambia segno in $[a, b]$ e quindi il prodotto scalare

$$< \prod_{i=1}^{k}(x - x_i), q_n(x) >= \int_a^b \rho(x) \prod_{i=1}^{k}(x - x_i)^2\,\phi(x)\,dx$$

deve essere non nullo. Ciò però è assurdo in quanto $\prod_{i=1}^{k}(x - x_i)$ è un polinomio di grado inferiore ad n e quindi è ortogonale a $q_n(x)$. □

Il calcolo degli zeri dei polinomi ortogonali può farsi in maniera efficiente calcolando gli autovalori di una opportuna matrice. Posto

$$\sigma_i = \frac{\gamma_i}{\alpha_i}, \qquad \tau_i = \frac{1}{\alpha_i},$$

riscriviamo (5.4) nella forma

$$xq_i = \tau_i q_{i+1} + \beta_i q_i + \sigma_i q_{i-1}, \qquad \text{per } i = 0, 1, \ldots, n-1,$$

cioè

$$\begin{aligned} xq_0 &= \tau_0 q_1 + \beta_0 q_0 \\ xq_1 &= \tau_1 q_2 + \beta_1 q_1 + \sigma_1 q_0 \\ &\vdots \\ xq_{n-1} &= \tau_{n-1} q_n + \beta_{n-1} q_{n-1} + \sigma_{n-1} q_{n-2}. \end{aligned}$$

Introducendo la matrice

$$T = \begin{pmatrix} \beta_0 & \tau_0 & & & \\ \sigma_1 & \beta_1 & \tau_1 & & \\ & \sigma_2 & \beta_2 & \ddots & \\ & & \ddots & \ddots & \tau_{n-2} \\ & & & \sigma_{n-1} & \beta_{n-1} \end{pmatrix}$$

si ha:

$$x \begin{pmatrix} q_0 \\ q_1 \\ \vdots \\ q_{n-1} \end{pmatrix} = T \begin{pmatrix} q_0 \\ q_1 \\ \vdots \\ q_{n-1} \end{pmatrix} + \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \tau_{n-1} q_n \end{pmatrix}.$$

Quindi se x è uno zero di $q_n(x)$, esso è anche autovalore di T e viceversa. Ne segue che gli zeri di $q_n(x)$, cioè i nodi della formula di quadratura, si possono ottenere calcolando gli autovalori di T.

Fissati $\rho(x)$, $[a, b]$ ed n risulta univocamente determinata la formula di quadratura di grado di precisione almeno $2n + 1$

$$I(\rho f) = \int_a^b \rho(x) f(x)\, dx \simeq \sum_{i=0}^{n} a_i f(x_i) = J_n(f),$$

che propriamente dicesi *formula di quadratura gaussiana.*

Osservazione 5.2 *Nodi e pesi delle formule gaussiane sono numeri irrazionali e sono stati calcolati per vari valori di n.*

Osservazione 5.3 *La positività dei pesi consente di dimostrare la convergenza di $J_n(f)$. Nel caso di intervalli limitati $[a, b]$ e per formule di grado $2n + 1$ la continuità di f è condizione sufficiente affinché*

$$\lim_{n\to\infty} J_n(f) = I(\rho f).$$

Pertanto l'errore $E_n(f)$ tende a zero per $n \to +\infty$ anche nel caso in cui f non sia derivabile e quindi non si abbia la formula esplicita dell'errore.

Confrontando tra loro le formule di Newton-Cotes e quelle gaussiane si possono fare le seguenti considerazioni:

- a parità di numero di nodi le formule gaussiane hanno grado di precisione che è circa il doppio di quello delle formule di Newton-Cotes;
- i coefficienti delle formule gaussiane sono sempre positivi, mentre quelli delle formule di Newton-Cotes lo sono solamente per $n \leq 6$;
- i nodi di una formula gaussiana a $n+1$ punti NON sono un sottoinsieme dei nodi di alcuna formula gaussiana a $m+1$ punti con $m > n$. Questo è un notevole inconveniente poiché non è possibile costruire formule gaussiane a $m+1$ punti utilizzando i valori di formule gaussiane a $n+1$ punti precedentemente calcolati, cosa che è invece possibile fare con le formule di Newton-Cotes.

Capitolo 6

Metodi numerici per equazioni differenziali ordinarie

Le equazioni differenziali sono equazioni che esprimono un legame tra alcune funzioni incognite e le loro derivate. Quando le derivate si riferiscono ad una sola variabile si parla di equazioni differenziali ordinarie, nel caso di più variabili si hanno le equazioni differenziali alle derivate parziali. Si riportano qui di seguito un esempio di

- equazioni differenziali ordinarie:

$$\frac{d}{dt}y(t) = ky(t), \quad k \in \mathbb{R},$$

  con t variabile indipendente e y funzione incognita;

- equazioni differenziali alle derivate parziali:

$$\frac{\partial}{\partial t}u(x,t) - a\frac{\partial^2}{\partial x^2}u(x,t) = 0$$

  con t, x variabili indipendenti e u funzione incognita.

Definizione 6.1 *Si dice che un'equazione differenziale, ordinaria o alle derivate parziali, ha* ordine p *se p è l'ordine massimo delle derivate della funzione incognita che compare nell'equazione.*

Esempio 6.1 *L'equazione differenziale ordinaria*

$$y'(t) = y^2 + 1$$

è del primo ordine; l'equazione differenziale ordinaria

$$y''(t) = F(t, y)$$

è invece del secondo ordine.

Introducendo il vettore $\mathbf{z} = (z_1, z_2)^T$ definito da $z_1 = y, z_2 = y'$ e $\mathbf{g} = (z_2, F(t, z_1))^T$ si ha che

$$y''(t) = F(t, y) \qquad \Longleftrightarrow \qquad \mathbf{z}' = \mathbf{g}(t, \mathbf{z})$$

ovvero problemi differenziali di ordine superiore al primo possono essere trasformati in problemi equivalenti del primo ordine con una opportuna sostituzione. Pertanto, senza perdita di generalità, ci limiteremo al caso di equazioni differenziali ordinarie di ordine 1.

In particolare, se l'equazione differenziale è accoppiata con un'equazione che esprime il valore che y assume in un punto $t_0 \in [a, b]$, ovvero:

$$\begin{cases} y'(t) = f(t, y(t)), & t \in [a, b] \\ y(t_0) = y_0, \end{cases} \tag{6.1}$$

si ha un *problema ai valori iniziali*, o di Cauchy, anche detto *problema di evoluzione* in quanto descrive l'evoluzione di un fenomeno nel tempo. Il problema di cui ci occuperemo nel presente capitolo riguarda l'approssimazione numerica di una funzione $y(t) : [a, b] \subseteq \mathbb{R} \longrightarrow \mathbb{R}^m$ soluzione del problema ai valori iniziali del primo ordine (6.1) dove $f : [a, b] \times \mathbb{R}^m \longrightarrow \mathbb{R}^m$, $y_0 \in \mathbb{R}^m$.

Osservazione 6.1 *Per semplicità di esposizione nel seguito supporremo sempre* $m = 1$.

Prima di presentare alcune delle principali tecniche di approssimazione nel discreto, premettiamo qualche considerazione teorica sul problema continuo.

Esempio 6.2 *Consideriamo il problema ai valori iniziali*

$$\begin{cases} y'(t) = f(y), & t \in [0, b] \\ y(0) = 0, \end{cases}$$

con

$$f(y) = \begin{cases} 0, & y < 0 \\ 2\sqrt{y}, & y \geq 0. \end{cases}$$

Questo ammette due soluzioni:

1. $y(t) = 0$;
2. $y(t) = \begin{cases} 0, & t < 0 \\ t^2, & t \geq 0. \end{cases}$

Questo esempio mostra che richiedere solo la continuità di f non è sufficiente per assicurare l'unicità della soluzione. L'unicità è invece garantita se si richiede che $f(t, y)$ verifichi la condizione di Lipschitz rispetto al suo secondo argomento. Si può infatti dimostrare il seguente risultato.

Teorema 6.1 *Sia $f(t,y)$ definita e continua nell'insieme*

$$D = \{(t,y) : -\infty < a \leq t \leq b < +\infty, \|y\| < +\infty\}$$

e sia inoltre lipschitziana rispetto al suo secondo argomento, ovvero esista una costante $L > 0$ tale che

$$\|f(t,y) - f(t,y^*)\| \leq L\|y - y^*\|$$

per ogni $(t,y),(t,y^*) \in D$. Allora esiste un'unica soluzione $y(t) \in C^1([a,b])$ del problema ai valori iniziali (6.1), per ogni y_0 assegnato.

Sfortunatamente solo un limitato numero di equazioni differenziali ordinarie può essere risolto per via analitica. Ad esempio, l'equazione

$$y'(t) = y^2(t) + 1$$

ha come soluzione analitica $y(t) = \tan(t + c)$, con c parametro da determinare imponendo la condizione iniziale. Tuttavia, se modifichiamo un po' l'equazione ovvero consideriamo l'equazione differenziale

$$y'(t) = y^2(t) + t$$

la sua soluzione analitica è:

$$y(t) = \frac{c_1 A_i'(-t) + B_i'(-t)}{c_1 A_i(-t) + B_i(-t)}$$

con $A_i(t)$ e $B_i(t)$ funzioni di Airy. Infine, nel caso si consideri l'equazione differenziale

$$y'(t) = y^3(t) + t^2$$

non si riesce a trovare una espressione analitica per la soluzione. In vista di ciò e del fatto che nei modelli matematici $f(t,y)$ può essere anche assai complicata, è opportuno sapere risolvere i problemi differenziali numericamente piuttosto che analiticamente.

Cerchiamo quindi dei metodi numerici in grado di approssimare la soluzione per ogni classe di equazioni differenziali ordinarie che ammettano una soluzione.

La strategia generale dei metodi numerici considerati in seguito consiste nel dividere l'intervallo $[a,b]$ in N sottointervalli $[t_n, t_{n+1}], n = 0, 1, \ldots, N-1$, di ampiezza h con

$$t_n = t_0 + nh, \quad n = 0, 1, \ldots, N, \quad t_0 = a, t_N = b.$$

Il parametro h è detto *passo della discretizzazione*. In corrispondenza di ogni nodo t_n si vuole trovare una approssimazione della soluzione $y(t_n), n = 0, 1, \ldots, N$ che indichiamo con y_n. L'insieme $\{t_i\}_{i=0}^N$ è detta *griglia,* mentre l'insieme dei valori $\{y_i\}_{i=0}^N$ forma la soluzione numerica.

6.1 Metodi ad un passo

I metodi ad un passo sono della forma generale

$$y_{n+1} = y_n + h\Phi(h, t_n, y_n, y_{n+1}), \qquad n = 0, 1, \ldots, N-1,$$

in cui la funzione Φ dipende dalla funzione f che compare nel problema ai valori iniziali (6.1). Posto $y_0 = y(t_0)$, tale relazione ci permette di calcolare y_{n+1} conoscendo y_n. Se Φ dipende da y_{n+1} il metodo si dice *implicito* altrimenti esso è detto *esplicito*.

Si introducono ora alcune definizioni.

Definizione 6.2 *Si dice* errore globale di discretizzazione *nel punto t_{n+1} la differenza tra il valore esatto della soluzione in t_{n+1} ed il valore approssimato ottenuto mediante il metodo numerico, ossia*

$$e_{n+1} = y(t_{n+1}) - y_{n+1}, \quad n = 0, 1, \ldots, N-1.$$

Definizione 6.3 *Un metodo si dice* convergente *se, applicato ad un qualunque problema ai valori iniziali che soddisfa le ipotesi del Teorema 6.1, risulta per ogni $t \in [a, b]$*

$$\lim_{\substack{h \to 0 \\ t = t_{n+1}}} e_{n+1} = 0.$$

Inoltre, se $e_{n+1} = O(h^p)$ il metodo si dice convergente di ordine p.

Cerchiamo ora di capire cosa vuol dire che una equazione differenziale è approssimata mediante una discretizzazione con passo h. Consideriamo l'equazione differenziale ordinaria

$$y'(t) = f(t, y(t)).$$

Sia $y(t_n)$ il valore noto della soluzione al tempo t_n. Al tempo $t_{n+1} = t_n + h$ si avrà $y(t_n + h)$. Tuttavia, se applichiamo un metodo ad un passo, otterremo

$$u_{n+1} = y(t_n) + h\Phi(h, t_n, y(t_n), y(t_n + h))$$

che, in generale, sarà diverso da $y(t_n + h)$. Non è sufficiente richiedere che $u_{n+1} \to y(t_n + h)$ se $h \to 0$ poiché le direzioni dei vettori $(h, y(t_n + h) -$

$y(t_n))^T$ e $(h, u_{n+1} - y(t_n))^T$ potrebbero rimanere distinte per $h \to 0$. Quindi le quantità cruciali sono

$$\frac{y(t_n + h) - y(t_n)}{h}, \qquad \frac{u_{n+1} - y(t_n)}{h},$$

ovvero i coefficienti angolari dei due vettori. Pertanto,

Definizione 6.4 *Dicesi* errore locale di discretizzazione *la quantità*

$$\delta(y(t_n), h) = \frac{y(t_n + h) - y(t_n)}{h} - \frac{u_{n+1} - y(t_n)}{h} = \frac{y(t_n + h) - u_{n+1}}{h}.$$

Una richiesta ragionevole è che

$$\delta(y(t_n), h) \to 0, \quad \text{per } h \to 0.$$

Considerato che

$$\delta(y(t_n), h) = \frac{y(t_{n+1}) - (y(t_n) + h\Phi(h, t_n, y(t_n), y(t_n + h)))}{h},$$

l'errore $\tau_{n+1} \equiv h\delta(y(t_n), h)$ è la quantità che si ottiene andando a sostituire la soluzione esatta nel metodo numerico, i.e.,

$$\tau_{n+1} = y(t_{n+1}) - y(t_n) - h\Phi(h, t_n, y(t_n), y(t_n + h)).$$

Definizione 6.5 *La quantità τ_{n+1} è detta* errore locale di troncamento.

Definizione 6.6 *Un metodo ad un passo si dice* consistente *se vale la condizione*

$$\lim_{\substack{h \to 0 \\ t = t_{n+1}}} \frac{\tau_{n+1}}{h} = 0.$$

Si definisce ordine di consistenza del metodo il più grande intero positivo p tale per cui

$$\tau_{n+1} = O(h^{p+1}).$$

Il seguente teorema stabilisce condizioni necessarie e sufficienti affinché un metodo ad un passo esplicito sia convergente.

Teorema 6.2 *La funzione $\Phi(h, t, y)$ sia continua nella regione*

$$\mathcal{D} = \{(h, t, y) : 0 < h \leq h_0, -\infty < a \leq t \leq b < +\infty, \|y\| < +\infty\}$$

e soddisfi la condizione di Lipschitz

$$\|\Phi(h, t, y^*) - \Phi(h, t, y)\| \leq M\|y^* - y\|$$

per ogni $(h, t, y^*), (h, t, y) \in \mathcal{D}$. Allora il metodo ad un passo è convergente se e solo se è consistente.

6.1.1 Metodo di Eulero esplicito

L'idea alla base del metodo di Eulero esplicito è quella di approssimare localmente la soluzione del problema ai valori iniziali mediate la tangente ad essa in un punto specifico. In dettaglio, si consideri l'equazione della tangente alla curva $y(t)$ nel punto (t_0, y_0):

$$R(t) = y_0 + f(t_0, y_0)(t - t_0).$$

Per $t = t_1$ si ha $R(t_1) = \eta_1$ e quindi

$$\eta_1 = y_0 + hf(t_0, y_0).$$

Si consideri ora il problema ai valori iniziali

$$\begin{cases} y_1'(t) = f(t, y_1(t)), \\ y_1(t_1) = \eta_1, \end{cases}$$

e si tracci la tangente nel punto (t_1, η_1) alla curva $y_1(t)$. Per $t = t_2$ si ottiene

$$\eta_2 = \eta_1 + hf(t_1, \eta_1).$$

Procedendo in questo modo si ottiene la soluzione approssimata $\{\eta_i\}_{i=0}^N$ con

$$\begin{cases} \eta_{i+1} = \eta_i + hf(t_i, \eta_i), \quad i = 0, 1, \ldots, N-1 \\ \eta_0 = y_0 \end{cases}$$

Il comportamento della soluzione approssimata rispetto alla soluzione continua in corrispondenza al punto t_2 rappresenta, in realtà, la situazione al generico punto $t_i, i > 1$. Nell'errore globale di discretizzazione,

$$e_2 = y(t_2) - \eta_2,$$

si individuano due contributi:

1. l'errore introdotto dal metodo dovuto al fatto che la soluzione esatta passante per il punto (t_1, η_1) è stata sostituita con la tangente:

$$h\delta(y_1(t_1), h) = y_1(t_2) - \eta_2$$

2. accumulo di tutti gli errori commessi in precedenza.

Concentriamoci sul contributo 1.:

$$\begin{aligned} h\delta(y_1(t_1),h) &= y_1(t_2)-\eta_2 \\ &= y_1(t_2)-(\eta_1+hf(t_1,\eta_1)) \\ &= y_1(t_2)-y_1(t_1)-hf(t_1,y_1(t_1)) \\ &= y_1(t_2)-y_1(t_1)-h(y_1(t_1))' \end{aligned}$$

Ma

$$y_1(t_2)=y_1(t_1+h)=y_1(t_1)+hy_1'(t_1)+\frac{h^2}{2}(y_1(\xi))'', \qquad \xi\in(t_1,t_2).$$

Allora

$$h\delta(y_1(t_1),h)=\frac{h^2}{2}(y_1(\xi))''$$

e, in generale, per l'errore locale di troncamento si ha

$$\begin{aligned} h\delta(y_j(t_j),h) &= y_j(t_{j+1})-\eta_{j+1} \\ &= \frac{h^2}{2}(y_j(\xi))'', \qquad \xi\in(t_j,t_{j+1}). \end{aligned}$$

Allora se ne conclude che il metodo di Eulero esplicito è consistente di ordine 1.

Quando $h\to 0$, il numero degli errori commessi ai passi precedenti tende all'infinito. Non è evidente, a priori, che la consistenza del metodo comporti necessariamente la convergenza a zero dell'errore globale. Affinché si abbia convergenza occorre che l'accumulo degli errori si mantenga limitato per $h\to 0$. Per verificare che il metodo di Eulero esplicito è convergente, scriviamo l'errore nel seguente modo:

$$\begin{aligned} e_{j+1} &= y(t_{j+1})-\eta_{j+1} \\ &= y(t_{j+1})-y_j(t_{j+1})+y_j(t_{j+1})-\eta_{j+1}. \end{aligned}$$

Consideriamo ora la differenza $y(t_{j+1})-y_j(t_{j+1})$ dove $y(t)$ è soluzione del problema

$$\begin{cases} y'(t)=f(t,y(t)), \\ y(t_j)=y(t_j), \end{cases}$$

mentre $y_j(t)$ è soluzione del problema

$$\begin{cases} (y_j(t))'=f(t,y_j(t)), \\ y_j(t_j)=\eta_j(=y(t_j)+z_j), \end{cases}$$

Allora se f è lipschitziana con costante di Lipschitz pari a L e ricordando che

$$|y(t_{j+1}) - y_j(t_{j+1})| \leq e^{L(t_{j+1}-t_j)}|z_j|$$

si ottiene

$$\begin{aligned} |y(t_{j+1}) - y_j(t_{j+1})| &\leq e^{L(t_{j+1}-t_j)}|z_j| = e^{Lh}|y(t_j) - \eta_j| \\ &= e^{Lh}|e_j|. \end{aligned}$$

Pertanto

$$|e_{j+1}| \leq e^{Lh}|e_j| + h\delta(y_j(t_j), h).$$

Essendo $e_0 = 0$, applicando ricorsivamente si ha:

$$\begin{aligned} |e_1| &\leq e^{Lh}|e_0| + h|\delta(y_0(t_0), h)| = h|\delta(y_0(t_0), h)| \\ |e_2| &\leq e^{Lh}|e_1| + h|\delta(y_1(t_1), h)| \\ &\leq e^{Lh}(h|\delta(y_0(t_0), h)|) + h|\delta(y_1(t_1), h)| \\ &\leq (e^{Lh} + 1)h \max_{\ell \leq 1} |\delta(y_\ell(t_\ell), h)| \\ |e_3| &\leq e^{Lh}|e_2| + h|\delta(y_2(t_2), h)| \\ &\leq e^{Lh}\left((e^{Lh} + 1)h \max_{\ell \leq 1} |\delta(y_\ell(t_\ell), h)|\right) | + h|\delta(y_2(t_2), h)| \\ &\leq \left(\left(e^{Lh}\right)^2 + e^{Lh} + 1\right) h \max_{\ell \leq 2} |\delta(y_\ell(t_\ell), h)| \end{aligned}$$

da cui

$$|e_{j+1}| \leq \sum_{r=0}^{j} \left(e^{Lh}\right)^r h \max_{\ell \leq j} |\delta(y_\ell(t_\ell), h)|$$

Considerato che

$$\sum_{r=0}^{j} \left(e^{Lh}\right)^r = \frac{\left(e^{Lh}\right)^{j+1} - 1}{e^{Lh} - 1}$$

e che $1 + Lh \leq e^{Lh}$, allora

$$\frac{\left(e^{Lh}\right)^{j+1} - 1}{e^{Lh} - 1} \leq \frac{\left(e^{Lh}\right)^{j+1} - 1}{Lh}.$$

Inoltre, essendo $(j+1)h = t_{j+1} - t_0 \leq b - a$, ne segue che

$$|e_{j+1}| \leq \frac{e^{L(b-a)} - 1}{L} \max_{\ell \leq j} |\delta(y_\ell(t_\ell), h)|.$$

Ricordando che $\delta(y_\ell(t_\ell), h) = h/2|(y_\ell(\xi))''|$, posto $M = \max_{\ell \le j} |(y_\ell(\xi))''|$ si ottiene

$$|e_{j+1}| \le \frac{e^{L(b-a)} - 1}{L} \frac{M}{2} h$$

da cui discende che il metodo di Eulero esplicito è convergente di ordine 1.

Osserviamo che si parla di convergenza quando $h \to 0$. Ma un metodo numerico ha senso se h è diverso da zero e non troppo piccolo. In tal caso si auspica che l'equazione che definisce il metodo numerico generi una soluzione approssimata che mimi il comportamento della soluzione del problema ai valori iniziali. Si consideri l'equazione test

$$y'(t) = \lambda y(t), \quad \lambda \in \mathbb{C}, \, \mathrm{Re}(\lambda) < 0, \quad t \in [0, T] \tag{6.2}$$

la cui soluzione esatta è $y(t) = ce^{\lambda t}$, con c costante arbitraria. Allora $y(t_i) = ce^{\lambda t_i}$ mentre la soluzione numerica ottenuta applicando il metodo di Eulero esplicito all'equazione test risulta essere:

$$\eta_i = \eta_{i-1} + h\lambda\eta_{i-1} = (1 + h\lambda)\eta_{i-1}, \qquad i = 1, 2, \ldots, N,$$

da cui

$$\eta_i = (1 + h\lambda)^i \eta_0.$$

Considerato che $|y(t)|$ ha un decadimento di tipo esponenziale per $t \to +\infty$ è naturale richiedere che la soluzione numerica abbia un comportamento analogo a quello della soluzione continua, cioè per ogni $i = 1, 2, \ldots, N$,

$$|\eta_i| < |\eta_{i-1}|.$$

L'insieme del piano complesso

$$\mathcal{S} = \{q = h\lambda \in \mathbb{C} : |1 + q| < 1\}.$$

si chiama *regione di assoluta stabilità* del metodo di Eulero esplicito.

Ma $|1 + q| = (1 + Re(q))^2 + (Im(q))^2 < 1$ ovvero è un cerchio centrato in $(-1, 0)$ e raggio unitario. Esplicitando rispetto ad h, dalla disuguaglianza $|1 + h\lambda| < 1$ si ottiene

$$h < -\frac{2Re(\lambda)}{|\lambda|^2}$$

che, nel caso di $\lambda \in \mathbb{R}$ si riduce a

$$h < -\frac{2}{\lambda}.$$

6.1.2 Metodi di Runge-Kutta

I metodi di Runge-Kutta costituiscono una importante classe di metodi ad un passo. La struttura generale di tali metodi è:

$$\begin{cases} y_{n+1} = y_n + h\sum_{i=1}^{s} b_i K_i \\ K_i = f(t_n + c_i h, y_n + h\sum_{j=1}^{s} a_{ij} K_j), \qquad i = 1, 2, \ldots, s. \end{cases}$$

I parametri reali b_i, c_i, a_{ij} definiscono il metodo ed s è detto *numero di stadi*. Pertanto un metodo di Runge-Kutta è un metodo ad un passo in cui risulta

$$\Phi(h, t_n, y_n, y_{n+1}) = \sum_{i=1}^{s} b_i K_i.$$

Una forma compatta per rappresentare un metodo di Runge-Kutta è fornita dall'uso della seguente *tabella di Butcher*

$$\begin{array}{c|c} \mathbf{c} & A \\ \hline & \mathbf{b}^T \end{array} = \begin{array}{c|ccc} c_1 & a_{11} & \ldots & a_{1s} \\ \vdots & \vdots & & \vdots \\ c_s & a_{s1} & \ldots & a_{ss} \\ \hline & b_1 & \ldots & b_s \end{array}.$$

Si distinguono due classi di metodi, riconoscibili dalla forma della matrice A:

- metodi espliciti: se $a_{ij} = 0$ per ogni coppia i, j con $1 \leq i \leq j \leq s$ (A è strettamente triangolare inferiore);
- metodi impliciti: se $a_{ij} \neq 0$ per qualche coppia i, j con $i \leq j$.

Di seguito si riportano alcuni esempi di metodi di Runge-Kutta:

- metodo di Eulero esplicito:

$$\begin{array}{c|c} 0 & 0 \\ \hline & 1 \end{array}$$

- metodo di Eulero implicito:

$$\begin{array}{c|c} 1 & 1 \\ \hline & 1 \end{array}$$

- metodo dei trapezi:

$$\begin{array}{c|cc} 0 & 0 & 0 \\ 1 & 1/2 & 1/2 \\ \hline & 1/2 & 1/2 \end{array}$$

Nel caso di metodi espliciti

$$K_i = f(t_n + c_i h, y_n + h \sum_{j=1}^{i-1} a_{ij} K_j), \qquad i = 1, 2, \ldots, s$$

e quindi ciascun valore K_i si può calcolare esplicitamente in funzione dei precedenti $K_j, j = 1, 2, \ldots, i-1$.

Si è interessati a caratterizzare metodi che, fissato h consentano di ottenere un valore accurato della soluzione $y(t)$. Si consideri il problema test

$$\begin{cases} y'(t) = \lambda y, & \mathrm{Re}(\lambda) < 0 \\ y(0) = 1, \end{cases}$$

la cui soluzione esatta è $y(t) = e^{\lambda t}$.

Applicando un metodo di Runge-Kutta al problema test si vorrebbe ottenere una soluzione numerica con un comportamento analogo a quello della soluzione continua, i.e.,

$$|y_{n+1}| < c|y_n|$$

per qualche costante $0 < c < 1$. Ma, il metodo di Runge-Kutta applicato al problema test è:

$$\begin{cases} y_{n+1} = y_n + h \sum_{i=1}^{s} b_i K_i \\ K_i = \lambda(y_n + h \sum_{j=1}^{s} a_{ij} K_j), \qquad i = 1, 2, \ldots, s. \end{cases}$$

Posto $\mathbf{K} = (K_1, K_2, \ldots, K_s)^T$ e $\mathbb{1} = (1, 1, \ldots, 1)^T \in \mathbb{R}^s$ si ha

$$\begin{cases} y_{n+1} = y_n + h\mathbf{b}^T\mathbf{K} \\ \mathbf{K} = \lambda y_n \mathbb{1} + h\lambda A\mathbf{K}. \end{cases}$$

e quindi, posto $q = h\lambda$,

$$\begin{cases} y_{n+1} = y_n + h\mathbf{b}^T\mathbf{K} \\ (I - qA)\mathbf{K} = \lambda y_n \mathbb{1} \end{cases}$$

o, equivalentemente, in forma matriciale:

$$\begin{pmatrix} I - qA & \mathbf{0} \\ -h\mathbf{b}^T & 1 \end{pmatrix} \begin{pmatrix} \mathbf{K} \\ y_{n+1} \end{pmatrix} = y_n \begin{pmatrix} \lambda\mathbb{1} \\ 1 \end{pmatrix}.$$

Usando la regola di Cramer si ottiene che

$$y_{n+1} = \frac{\det\begin{pmatrix} I - qA & \lambda\mathbb{1}y_n \\ -h\mathbf{b}^T & y_n \end{pmatrix}}{\det\begin{pmatrix} I - qA & \mathbf{0} \\ -h\mathbf{b}^T & 1 \end{pmatrix}} = \frac{\det\begin{pmatrix} I - qA & \lambda\mathbb{1} \\ -h\mathbf{b}^T & 1 \end{pmatrix}}{\det(I - qA)} y_n.$$

Ora, per calcolare il determinante che si trova a numeratore si osserva che

$$M(q) = \begin{pmatrix} I & -\lambda\mathbb{1} \\ & 1 \end{pmatrix}\begin{pmatrix} I - qA & \lambda\mathbb{1} \\ -h\mathbf{b}^T & 1 \end{pmatrix} = \begin{pmatrix} I - qA + q\mathbb{1}\mathbf{b}^T & \mathbf{0} \\ -h\mathbf{b}^T & 1 \end{pmatrix}$$

e quindi

$$\det M(q) = \det\begin{pmatrix} I - qA & \lambda\mathbb{1} \\ -h\mathbf{b}^T & 1 \end{pmatrix} = \det(I - qA + q\mathbb{1}\mathbf{b}^T).$$

Allora

$$y_{n+1} = \frac{\det(I - qA + q\mathbb{1}\mathbf{b}^T)}{\det(I - qA)} y_n.$$

La funzione

$$R(q) = \frac{\det(I - qA + q\mathbb{1}\mathbf{b}^T)}{\det(I - qA)} \tag{6.3}$$

è detta *funzione di stabilità* di un metodo di Runge-Kutta. Allora

$$|y_{n+1}| < c|y_n| \qquad \Leftrightarrow \qquad |R(q)| < 1.$$

Definizione 6.7 *Un metodo di Runge-Kutta si dice* assolutamente stabile *per un dato q se la funzione di stabilità soddisfa la condizione $|R(q)| < 1$.*

Definizione 6.8 *L'insieme del piano complesso*

$$\mathcal{S} = \{q \in \mathbb{C} : |R(q)| < 1\}$$

si chiama regione di assoluta stabilità *del metodo.*

Nella (6.3) $\det(I - qA + q\mathbb{1}\mathbf{b}^T)$ e $\det(I - qA)$ sono polinomi a coefficienti reali in q di grado al più s. Quindi $R(q)$ è una funzione razionale. Tuttavia, se il metodo di Runge-Kutta è esplicito, A è strettamente triangolare inferiore e quindi $\det(I - qA) = 1$. In tal caso $R(q)$ è un polinomio di grado al più s.

Definizione 6.9 *Un metodo di Runge-Kutta si dice* A-stabile *se*

$$\mathcal{S} \supseteq \mathbb{C}^- = \{q \in \mathbb{C} : Re(q) < 0\}$$

La regione di assoluta stabilità del metodo dei trapezi coincide con il semipiano $Re(q) < 0$. Tale metodo è quindi A-stabile.

Considerato che per un metodo di Runge-Kutta esplicito la funzione di stabilità coincide con un polinomio di grado al più s, si ottiene che

$$\lim_{Re(q)\to-\infty} |R(q)| = +\infty$$

e quindi non esistono metodi di Runge-Kutta espliciti A-stabili.

6.2 Metodi lineari a più passi

Un metodo lineare a più passi, o a k-passi, con $k \geq 1$, per approssimare la soluzione del problema ai valori iniziali (6.1) ha la seguente struttura

$$\sum_{j=0}^{k} \alpha_j y_{n+j} = h \sum_{j=0}^{k} \beta_j f_{n+j}, \qquad n = 0, 1, \ldots, N-k, \tag{6.4}$$

dove $y_{n+j} \approx y(t_{n+j})$, $f_{n+j} \equiv f(t_{n+j}, y_{n+j})$ e i coefficienti α_j e β_j sono costanti reali.

Osservazione 6.2 *Il metodo è detto lineare poiché è lineare in f. Ciò non significa che f è una funzione lineare di t e y.*

Si suppongono noti (dati o calcolati) i k valori iniziali $y_0, y_1, \ldots, y_{k-1}$. Per ogni valore di n, la relazione (6.4) fornisce y_{n+k} in funzione dei k valori precedenti $y_{n+k-1}, \ldots, y_n$. In altri termini, approssimiamo il problema ai valori iniziali continuo del primo ordine con un problema discreto di ordine k.

I coefficienti $\{\alpha_j\}_{j=0}^{k}$ e $\{\beta_j\}_{j=0}^{k}$ definiscono il metodo. Considerato che il metodo è definito a meno di una costante moltiplicativa, per eliminare scalature arbitrarie assumeremo che $\alpha_k = 1$. Inoltre, si ammette che sia $|\alpha_0| + |\beta_0| \neq 0$.

Un metodo lineare a k-passi può essere riscritto nel seguente modo:

$$y_{n+k} - h\beta_k f_{n+k} = -\sum_{j=0}^{k-1} \alpha_j y_{n+j} + h \sum_{j=0}^{k-1} \beta_j f_{n+j}, \qquad n = 0, 1, \ldots, N-k.$$

Se $\beta_k = 0$ il metodo si dice *esplicito* ed è evidente che il calcolo di y_{n+k} è diretto. Se $\beta_k \neq 0$ il metodo si dice *implicito*. In tal caso, posto

$$w \equiv -\sum_{j=0}^{k-1} \alpha_j y_{n+j} + h \sum_{j=0}^{k-1} \beta_j f_{n+j}$$

si calcola per $n = 0, 1, \ldots, N-k$ una approssimazione z^* della soluzione dell'equazione

$$z = h\beta_k f(t_{n+k}, z) + w$$

e si assume $y_{n+k} = z^*$. Se $f(t, y)$ è lineare rispetto a y questa equazione, in realtà, si riduce ad un sistema lineare. In generale si può utilizzare, ad esempio, il seguente procedimento iterativo

$$z^{(r+1)} = h\beta_k f(t_{n+k}, z^{(r)}) + w, \qquad r = 0, 1, \ldots$$

In tal caso, denotando con L la costante di Lipschitz della funzione $f(t, y)$,

$$\begin{aligned} |z^{(r+1)} - z^*| &= h|\beta_k||f(t_{n+k}, z^{(r)}) - f(t_{n+k}, z^*)| \\ &\leq h|\beta_k|L|z^{(r)} - z^*| \leq (h|\beta_k|L)^{r+1}\,|z^{(0)} - z^*|. \end{aligned}$$

Pertanto, la convergenza è garantita se

$$h|\beta_k|L < 1.$$

Definizione 6.10 *Si dice* errore globale di discretizzazione *nel punto* t_{n+k} *la differenza tra* $y(t_{n+k})$, *il valore esatto della soluzione in* t_{n+k} *e il valore approssimato ottenuto mediante il metodo lineare a* k*-passi, ossia*

$$e_{n+k} = y(t_{n+k}) - y_{n+k}.$$

In generale, i valori $y_1, \ldots, y_{k-1}$ dipendono da h e si dice che formano un *insieme compatibile di valori iniziali* se vale la proprietà

$$\lim_{h\to 0} y_i = y_0, \quad i = 1, 2, \ldots, k-1.$$

Definizione 6.11 *Un metodo lineare a* k*-passi si dice* convergente *se è tale che per ogni* $t \in [a, b]$ *si abbia*

$$\lim_{\substack{h\to 0 \\ t=t_{n+k}}} e_{n+k} = 0$$

per ogni insieme compatibile di valori iniziali.

Definizione 6.12 *Dicesi* errore locale di troncamento *di un metodo lineare a* k*-passi la quantità*

$$\tau_{n+k} = \sum_{j=0}^{k} \alpha_j y(t_{n+j}) - h\sum_{j=0}^{k} \beta_j f(t_{n+j}, y(t_{n+j})).$$

Definizione 6.13 *Un metodo lineare a k-passi si dice* consistente *se è tale che per ogni* $t = t_{n+k} \in [a, b]$ *si abbia*

$$\lim_{\substack{h\to 0 \\ t=t_{n+k}}} \frac{\tau_{n+k}}{h} = 0.$$

Si definisce ordine di consistenza del metodo il più grande intero p *tale per cui* $\tau_{n+k} = O(h^{p+1})$.

Nell'ipotesi in cui $y(t) \in C^\infty((a, b))$ l'ordine di consistenza di un metodo a k-passi si traduce in condizioni algebriche sui suoi coefficienti che ora andiamo ad esaminare. Considerato che $y'(t_{n+j}) = f(t_{n+j}, y(t_{n+j}))$ si ottiene:

$$\tau_{n+k} = \sum_{j=0}^{k} \alpha_j y(t_{n+j}) - h \sum_{j=0}^{k} \beta_j y'(t_{n+j}).$$

Sviluppando il secondo membro si può scrivere formalmente:

$$\begin{aligned}
\tau_{n+k} &= \sum_{j=0}^{k} \alpha_j \left(\sum_{\ell\geq 0} \frac{y^{(\ell)}(t_n)}{\ell!} (t_{n+j} - t_n)^\ell \right) - h \sum_{j=0}^{k} \beta_j \left(\sum_{\ell\geq 1} \frac{y^{(\ell)}(t_n)}{(\ell-1)!} (t_{n+j} - t_n)^{\ell-1} \right) \\
&= y(t_n) \left(\sum_{j=0}^{k} \alpha_j \right) + \sum_{\ell\geq 1} \frac{y^{(\ell)}(t_n)}{\ell!} h^\ell \left(\sum_{j=0}^{k} \alpha_j j^\ell - \beta_j \ell j^{\ell-1} \right)
\end{aligned}$$

ovvero

$$\tau_{n+k} = \sum_{j=0}^{+\infty} c_j y^{(j)}(t_n) h^j$$

con

$$\begin{aligned}
c_0 &= \sum_{j=0}^{k} \alpha_j \\
c_1 &= \sum_{j=0}^{k} (j\alpha_j - \beta_j) \\
c_r &= \sum_{j=0}^{k} \left(\alpha_j \frac{j^r}{r!} - \beta_j \frac{j^{r-1}}{(r-1)!} \right), \qquad r = 2, 3, \ldots
\end{aligned}$$

Ne segue che per un metodo lineare a k-passi di ordine p deve essere $c_0 = c_1 = \cdots = c_p = 0, c_{p+1} \neq 0$; quindi risulta

$$\tau_{n+k} = c_{p+1} y^{(p+1)}(t_n) h^{p+1} + O(h^{p+2})$$

dove c_{p+1} è denominato *coefficiente principale dell'errore* (o costante di errore del metodo).

Al metodo lineare a k-passi sono associati i seguenti due polinomi, detti *primo* e *secondo polinomio caratteristico*,

$$\rho(\mu) = \sum_{j=0}^{k} \alpha_j \mu^j, \qquad \sigma(\mu) = \sum_{j=0}^{k} \beta_j \mu^j.$$

Pertanto si ha che un metodo lineare a k-passi è consistente (di ordine 1) se e solo se

$$\rho(1) = 0, \qquad \rho'(1) = \sigma(1).$$

Si consideri ora il problema test

$$\begin{cases} y'(t) = 0 \\ y(t_0) = y_0, \end{cases} \tag{6.5}$$

la cui soluzione esatta è $y(t) = y_0$. Applicando un metodo lineare a k-passi a tale problema si ottiene l'equazione lineare omogenea alle differenze

$$\sum_{j=0}^{k} \alpha_j y_{n+j} = 0 \tag{6.6}$$

il cui polinomio caratteristico coincide con il primo polinomio caratteristico $\rho(\mu)$ del metodo stesso. La soluzione generale di tale equazione dipende dalle radici di $\rho(\mu)$ (a seconda che esse siano distinte oppure no!). Si può verificare che, per un insieme compatibile di valori iniziali, la soluzione numerica di (6.6) converge alla soluzione del problema continuo (6.5) solo se vale la seguente *condizione delle radici*

$$|\mu_i| \leq 1, \qquad i = 1, 2, \ldots, k$$

dove se $|\mu_i| = 1$ allora μ_i è semplice (il che equivale a dire che $\rho(\mu)$ è un polinomio di Von Neumann).

Definizione 6.14 *Un metodo lineare a k-passi si dice* zero-stabile *se gli zeri del polinomio $\rho(\mu)$ soddisfano la condizione delle radici.*

Si può dimostrare il seguente teorema.

Teorema 6.3 *Un metodo lineare a k-passi è convergente se e solo se è consistente e zero-stabile.*

In linea di principio è possibile costruire metodi lineari a k-passi fino ad un ordine massimo $p = 2k$ determinando le $(2k + 1)$ costanti $\alpha_0, \alpha_1, \ldots, \alpha_{k-1}$, $\beta_0, \beta_1, \ldots, \beta_k$ in modo che risulti $c_0 = c_1 = \cdots = c_{2k} = 0$. Tuttavia, è altresì evidente che richiedere che il polinomio $\rho(\mu)$ sia un polinomio di Von Neumann induca dei vincoli sui suoi coefficienti che, pertanto, non potranno essere utilizzati per incrementare l'ordine del metodo. Questo argomento, che abbiamo esposto in modo intuitivo, si formalizza nel seguente teorema detto *prima barriera di Dahlquist*.

Teorema 6.4 *L'ordine massimo di un metodo lineare a k-passi zero-stabile è $k + 1$ se k è dispari e $k + 2$ se k è pari.*

Consideriamo ora alcune classi di metodi lineari a k-passi derivate utilizzando opportuni polinomi di interpolazione.

6.2.1 Metodi di Adams

I metodi di Adams si ottengono integrando l'equazione differenziale $y'(t) = f(t, y(t))$ tra t_{n+k-1} e t_{n+k}, da cui

$$y(t_{n+k}) - y(t_{n+k-1}) = \int_{t_{n+k-1}}^{t_{n+k}} f(s, y(s))ds$$

e approssimando tale integrale mediante $\int_{t_{n+k-1}}^{t_{n+k}} p(s)ds$, essendo $p(t)$ il polinomio interpolante un certo insieme di punti. In base alla scelta dei punti si distinguono:

- i metodi di Adams-Moulton;
- i metodi di Adams-Bashfort.

I *metodi di Adams-Moulton* sono ottenuti richiedendo che

$$p(t_{n+j}) = f_{n+j}, \qquad j = 0, 1, \ldots, k.$$

Pertanto, esprimendo $p(t)$ nella base di Lagrange si ha

$$p(t) = \sum_{r=n}^{n+k} L_r(t) f_r, \qquad L_r(t) = \prod_{\substack{\ell=n \\ \ell \neq r}}^{n+k} \frac{t - t_\ell}{t_r - t_\ell},$$

da cui

$$y_{n+k} - y_{n+k-1} = \int_{t_{n+k-1}}^{t_{n+k}} \left(\sum_{r=n}^{n+k} L_r(s) f_r \right) ds.$$

Posto $r - n = j$ si ottiene

$$y_{n+k} - y_{n+k-1} = \sum_{j=0}^{k} \left(\int_{t_{n+k-1}}^{t_{n+k}} L_{n+j}(s)ds \right) f_{n+j}.$$

Utilizzando la trasformazione $s = t_0 + wh$ si ha che

$$\begin{aligned} \int_{t_{n+k-1}}^{t_{n+k}} L_{n+j}(s)ds &= \int_{n+k-1}^{n+k} \prod_{\substack{\ell=n \\ \ell \neq n+j}}^{n+k} \frac{(t_0 + wh) - (t_0 + \ell h)}{(t_0 + (n+j)h) - (t_0 + \ell h)} h dw \\ &= h \int_{n+k-1}^{n+k} \prod_{\substack{\ell=n \\ \ell \neq n+j}}^{n+k} \frac{w - \ell}{(n+j) - \ell} dw \\ &\equiv h\beta_j \end{aligned}$$

Esempio 6.3 *Quando $k = 1$ si ha*

$$y_{n+1} - y_n = h(\beta_0 f_n + \beta_1 f_{n+1}),$$

con

$$\beta_0 = \int_n^{n+1} \frac{w - (n+1)}{n - (n+1)} dw = \frac{1}{2}, \qquad \beta_1 = \int_n^{n+1} \frac{w - n}{(n+1) - n} dw = \frac{1}{2}$$

ovvero si ottiene il metodo dei trapezi:

$$y_{n+1} - y_n = \frac{h}{2}(f_n + f_{n+1}).$$

È facile verificare che in questo caso

$$\begin{aligned} c_0 &= \sum_{j=0}^{1} \alpha_j = -1 + 1 = 0 \\ c_1 &= \sum_{j=0}^{1} (j\alpha_j - \beta_j) = -\frac{1}{2} + \left(1 - \frac{1}{2}\right) = 0 \\ c_2 &= \sum_{j=0}^{1} \left(\alpha_j \frac{j^2}{2!} - \beta_j j \right) = \left(1 \cdot \frac{1}{2!} - \frac{1}{2}\right) = 0 \\ c_3 &= -\frac{1}{12} \end{aligned}$$

ovvero che il metodo dei trapezi ha ordine di consistenza uguale a 2.

I *metodi di Adams-Bashfort* sono ottenuti richiedendo che

$$p(t_{n+j}) = f_{n+j}, \qquad j = 0, 1, \ldots, k-1.$$

Pertanto, esprimendo $p(t)$ nella base di Lagrange si ha

$$p(t) = \sum_{r=n}^{n+k-1} L_r(t) f_r, \qquad L_r(t) = \prod_{\substack{\ell=n \\ \ell \neq r}}^{n+k-1} \frac{t - t_\ell}{t_r - t_\ell},$$

da cui

$$y_{n+k} - y_{n+k-1} = \int_{t_{n+k-1}}^{t_{n+k}} \left(\sum_{r=n}^{n+k-1} L_r(s) f_r \right) ds.$$

Posto $r - n = j$ si ottiene

$$y_{n+k} - y_{n+k-1} = \sum_{j=0}^{k-1} \left(\int_{t_{n+k-1}}^{t_{n+k}} L_{n+j}(s) ds \right) f_{n+j}.$$

Utilizzando la trasformazione $s = t_0 + wh$ si ha che

$$\begin{aligned} \int_{t_{n+k-1}}^{t_{n+k}} L_{n+j}(s) ds &= \int_{n+k-1}^{n+k} \prod_{\substack{\ell=n \\ \ell \neq n+j}}^{n+k-1} \frac{(t_0 + wh) - (t_0 + \ell h)}{(t_0 + (n+j)h) - (t_0 + \ell h)} h dw \\ &= h \int_{n+k-1}^{n+k} \prod_{\substack{\ell=n \\ \ell \neq n+j}}^{n+k-1} \frac{w - \ell}{(n+j) - \ell} dw \\ &\equiv h\beta_j \end{aligned}$$

Esempio 6.4 *Quando $k = 1$ si ha*

$$y_{n+1} - y_n = h\beta_0 f_n,$$

con

$$\beta_0 = \int_n^{n+1} 1 dw = 1$$

ovvero si ottiene il metodo di Eulero esplicito:

$$y_{n+1} - y_n = h f_n.$$

È facile verificare che in questo caso

$$\begin{aligned} c_0 &= \sum_{j=0}^{1} \alpha_j = -1 + 1 = 0 \\ c_1 &= \sum_{j=0}^{1} (j\alpha_j - \beta_j) = -1 + (1 - 0) = 0 \\ c_2 &= \frac{1}{2} \end{aligned}$$

ovvero che il metodo di Eulero esplicito ha ordine di consistenza uguale a 1.

Per costruzione i metodi di Adams hanno

$$\rho(\mu) = -\mu^{k-1}(1 - \mu)$$

ovvero sono zero-stabili.

Inoltre, si dimostra che i metodi a k-passi di tipo Adams-Moulton hanno ordine $(k + 1)$ mentre quelli di Adams-Bashfort hanno ordine k.

6.2.2 Metodi BDF

I metodi BDF (acronimo di Backward Differentiation Formulae) sono ottenuti differenziando il polinomio $p(t)$ che interpola $y(t)$ in $t_n, t_{n+1}, \ldots, t_{n+k}$ e imponendo che $p'(t_{n+k}) = f_{n+k}$. Considerando le condizioni di interpolazione:

$$p(t_{n+j}) = y_{n+j}, \qquad j = 0, 1, \ldots, k,$$

il polinomio $p(t)$ espresso in base di Lagrange è:

$$p(t) = \sum_{r=n}^{n+k} L_r(t) y_r, \qquad L_r(t) = \prod_{\substack{\ell=n \\ \ell \neq r}}^{n+k} \frac{t - t_\ell}{t_r - t_\ell},$$

da cui, posto $r - n = j$,

$$p'(t) = \sum_{j=0}^{k} \left(L_{n+j}(t)\right)' y_{n+j}.$$

Richiedendo che $p'(t_{n+k}) = f_{n+k}$ (ragionevole considerato che $p(t)$ interpola $y(t)$ e $y'(t) = f(t, y)$) e indicando con

$$h\beta_k = \frac{1}{\left(L_{n+k}(t)\right)' |_{t=t_{n+k}}}$$

ne deriva

$$y_{n+k} + \sum_{j=0}^{k-1} \frac{(L_{n+j}(t))'}{(L_{n+k}(t))'}\bigg|_{t=t_{n+k}} y_{n+j} = h\beta_k f_{n+k}.$$

Esempio 6.5 *Quando $k = 1$ si ha*

$$y_{n+1} + \alpha_0 y_n = h\beta_1 f_{n+1},$$

con

$$\alpha_0 = \frac{(L_n(t))'}{(L_{n+1}(t))'}\bigg|_{t=t_{n+1}} = \left(\frac{t - t_{n+1}}{t_n - t_{n+1}}\right)'\bigg|_{t=t_{n+1}} \frac{1}{\left(\frac{t-t_n}{t_{n+1}-t_n}\right)'\Big|_{t=t_{n+1}}} = -\frac{1}{h}\frac{1}{1/h} = -1$$

$$h\beta_1 = \frac{1}{1/h} \quad \Rightarrow \beta_1 = 1$$

ovvero si ottiene il metodo di Eulero implicito:

$$y_{n+1} - y_n = hf_{n+1}.$$

È facile verificare che in questo caso

$$c_0 = \sum_{j=0}^{1} \alpha_j = -1 + 1 = 0$$

$$c_1 = \sum_{j=0}^{1} (j\alpha_j - \beta_j) = (1 - 1) = 0$$

$$c_2 = -\frac{1}{2}$$

ovvero che il metodo di Eulero implicito ha ordine di consistenza uguale a 1.

Per costruzione le BDF hanno

$$\sigma(\mu) = \beta_k \mu^k$$

e sono zero-stabili solo per $k \leq 6$. Inoltre, si dimostra che le BDF a k-passi hanno ordine k.

Si è visto che la consistenza e la zero-stabilità garantiscono la convergenza di un metodo lineare a k-passi, ma essendo la convergenza una proprietà al limite per $h \to 0$ può accadere che un metodo convergente usato con un fissato passo $h > 0$, anche se piccolo, produca errori globali relativamente

grandi. Occorre pertanto definire un concetto di stabilità che garantisca non solo la convergenza del metodo per $h \to 0$ ma anche il contenimento degli errori per un dato h.

Si consideri il problema test

$$\begin{cases} y'(t) = \lambda y, & Re(\lambda) < 0 \\ y(0) = 1 \end{cases}$$

e ad esso si applichi il generico metodo lineare a k-passi:

$$\sum_{j=0}^{k} (\alpha_j - q\beta_j) y_{n+j} = 0, \qquad q = h\lambda. \tag{6.7}$$

A questa equazione alle differenze è associato il polinomio caratteristico detto *polinomio di stabilità*

$$\pi(q, \mu) = \rho(\mu) - q\sigma(\mu).$$

Se $\mu_i(q), i = 1, 2, \ldots, k$, sono le k radici semplici di $\pi(q, \mu)$ tali per cui $|\mu_i(q)| < 1$, al crescere di n la soluzione di (6.7) ha un comportamento analogo a quella del problema continuo.

Definizione 6.15 *Un metodo lineare a k-passi si dice* assolutamente stabile *per un dato $q \in \mathbb{C}$ se il polinomio di stabilità $\pi(q, \mu)$ è un polinomio di Schur (ovvero tutte le sue radici sono di modulo minore di 1).*

Definizione 6.16 *L'insieme del piano complesso*

$$\mathcal{S} = \{q \in \mathbb{C} : \pi(q, \mu) \text{ è un polinomio di Schur}\}$$

si chiama regione di assoluta stabilità *del metodo lineare a k-passi.*

Osservazione 6.3 *Un metodo lineare a k-passi si dice A-stabile se $\mathbb{C}^- \subseteq \mathcal{S}$ e perfettamente A-stabile se $\mathcal{S} \equiv \mathbb{C}^-$.*

Chiaramente la proprietà di A-stabilità lascia molta libertà nella scelta del passo h. Pertanto essa è una proprietà di rilievo. Tuttavia, vale il seguente risultato negativo.

Teorema 6.5 *Non esistono metodi lineari a k-passi espliciti A-stabili. Inoltre, l'ordine massimo di un metodo lineare a k-passi A-stabile non può superare due* (seconda barriera di Dahlquist).

Definizione 6.17 *Un metodo lineare a k-passi si dice* $A(\alpha)$-stabile *se*

$$C(\alpha) = \{q \in \mathbb{C} : |\pi - arg(q)| < \alpha, \alpha \in (0, \pi/2)\} \subset \mathcal{S}.$$

I metodi BDF sono A-stabili per $k = 1, 2$ e $A(\alpha)$-stabili per $3 \leq k \leq 6$.

In generale, per un metodo lineare a più passi potrebbe essere complicato determinare la corrispondente regione di assoluta stabilità. Per semplificare questo compito è conveniente ricorrere al concetto di *boundary locus*. Abbiamo visto che un punto $q \in \mathbb{C}$ appartiene a $\mathcal{S}$, regione di assoluta stabilità del metodo a k-passi, se il corrispondente polinomio $\pi(q, \mu)$ è un polinomio di Schur. Questo vuol dire che se $\bar{q} \in \partial\mathcal{S}$, la frontiera di $\mathcal{S}$, allora $\pi(\bar{q}, \mu)$ deve avere almeno una radice di modulo unitario. Indichiamo tale radice con $\mu_j(\bar{q})$. Allora possiamo scrivere $\mu_j(\bar{q}) = e^{i\bar{\theta}}$ per $\bar{\theta} \in [0, 2\pi)$. Essendo $\mu_j(\bar{q})$ radice di $\pi(q, \mu)$ si avrà:

$$\begin{aligned} \pi(\bar{q}, e^{i\bar{\theta}}) = 0 \quad &\Leftrightarrow \quad \rho(e^{i\bar{\theta}}) - \bar{q}\sigma(e^{i\bar{\theta}}) = 0 \\ &\Leftrightarrow \quad \bar{q} = \frac{\rho(e^{i\bar{\theta}})}{\sigma(e^{i\bar{\theta}})}. \end{aligned}$$

Poiché ogni punto sulla frontiera di $\mathcal{S}$ deve avere questa forma, possiamo disegnare la curva

$$q(e^{i\theta}) = \frac{\rho(e^{i\theta})}{\sigma(e^{i\theta})}, \qquad \theta \in [0, 2\pi)$$

per trovare il luogo dei punti che costituiscono la frontiera di $\mathcal{S}$.

Definizione 6.18 *Si definisce* boundary locus *di un metodo lineare a k-passi l'insieme*

$$\partial\mathcal{S} = \{q \in \mathbb{C} : q = q(e^{i\theta}), \theta \in [0, 2\pi)\}.$$

Esempio 6.6 *Considerato che per il metodo di Eulero esplicito*

$$\rho(\mu) = \mu - 1, \qquad \sigma(\mu) = 1$$

il boundary locus in questo caso ha la seguente forma:

$$q(e^{i\theta}) = e^{i\theta} - 1, \qquad \theta \in [0, 2\pi),$$

ovvero $\partial\mathcal{S}$ è la circonferenza del cerchio di centro $(-1, 0)$ e raggio unitario.

Appendice A

Richiami di Algebra Lineare

A.1 Matrici

Con $A \in \mathbb{C}^{m\times n}$ si indica una *matrice* di m righe e n colonne costituita da $m \times n$ numeri complessi a_{ij}, $i = 1, 2, \ldots, m$, $j = 1, 2, \ldots, n$, detti *elementi* di A. Generalmente si scrive

$$A = \begin{pmatrix} a_{11} & a_{12} & \ldots & a_{1n} \\ a_{21} & a_{22} & \ldots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \ldots & a_{mn} \end{pmatrix}.$$

Gli interi m ed n si dicono le *dimensioni* di A. Se $m = n$ la matrice A si dice *quadrata di ordine* n. Altrimenti, se $m \neq n$, A si dice *rettangolare*.
Si definisce:

- la *trasposta* di A,

$$A^T = \begin{pmatrix} a_{11} & a_{21} & \ldots & a_{m1} \\ a_{12} & a_{22} & \ldots & a_{m2} \\ \vdots & \vdots & & \vdots \\ a_{1n} & a_{2n} & \ldots & a_{mn} \end{pmatrix};$$

- la *coniugata* di A, $\bar{A} = (\bar{a}_{ij})$, dove $\bar{a}_{ij}$ è il coniugato del numero complesso a_{ij} (se $A \in \mathbb{R}^{m\times n}$, $\bar{A} = A$);
- la *trasposta coniugata* di A, $A^H = (\bar{A})^T \equiv \overline{(A^T)}$ (se $A \in \mathbb{R}^{m\times n}$, $A^H = A^T$);

Una matrice $A \in \mathbb{C}^{m\times n}$ è:

- *diagonale* se $a_{ij} = 0$ per $i \neq j$;
- *triangolare superiore* (*inferiore*) se $a_{ij} = 0$ per $i > j$ $(i < j)$;
- *triangolare superiore* (*inferiore*) *in senso stretto* se $a_{ij} = 0$ per $i \geq j$ $(i \leq j)$;
- *tridiagonale* se $a_{ij} = 0$ per $|i - j| > 1$.

In particolare, la matrice diagonale di ordine n i cui elementi sono tutti uguali ad 1 si chiama matrice *identica* e si denota con I.
Si definiscono le seguenti operazioni tra matrici:

- *addizione di matrici*: siano $A, B, C \in \mathbb{C}^{m\times n}$. Allora $C = A + B \Leftrightarrow c_{ij} = a_{ij} + b_{ij}$;
- *prodotto di un numero per una matrice*: sia $\alpha \in \mathbb{C}$ e $A \in \mathbb{C}^{m\times n}$. Allora $B = \alpha A \Leftrightarrow b_{ij} = \alpha a_{ij}$;
- *prodotto righe per colonne* di due matrici: siano $A \in \mathbb{C}^{m\times n}$, $B \in \mathbb{C}^{n\times p}$. Allora $C = AB \in \mathbb{C}^{m\times p}$ e i suoi elementi sono $c_{ij} = \sum_{k=1}^{n} a_{ik}b_{kj}$.

La moltiplicazione fra matrici non gode della proprietà commutativa. Inoltre valgono le seguenti proprietà:

- $(A\,B\,C)^H = C^H\,B^H\,A^H$;
- $(A\,B\,C)^T = C^T\,B^T\,A^T$;
- $\overline{(A\,B\,C)} = \bar{A}\,\bar{B}\,\bar{C}$;
- $(A\,B\,C)^{-1} = C^{-1}\,B^{-1}\,A^{-1}$.

Una matrice A quadrata di ordine n si dice:

- *simmetrica* se $A = A^T$;
- *antisimmetrica* se $A = -A^T$;
- *hermitiana* se $A = A^H$;
- *antihermitiana* se $A = -A^H$;
- *ortogonale* se $AA^T = A^TA = I$;
- *unitaria* se $AA^H = A^HA = I$;

- *definita positiva* se $\mathbf{x}^H A\mathbf{x} > 0$ per qualsiasi $\mathbf{x} \in \mathbb{C}^n, \mathbf{x} \neq \mathbf{0}$;
- a *predominanza diagonale forte* se per ogni $i = 1, 2, \ldots, n$, risulta

$$|a_{ii}| > \sum_{\substack{j=1 \\ j \neq i}}^{n} |a_{ij}|;$$

- a *predominanza diagonale debole* se per ogni $i = 1, 2, \ldots, n$, risulta

$$|a_{ii}| \geq \sum_{\substack{j=1 \\ j \neq i}}^{n} |a_{ij}|$$

e per almeno un indice $r, 1 \leq r \leq n$, si ha

$$|a_{rr}| > \sum_{\substack{j=1 \\ j \neq r}}^{n} |a_{rj}|.$$

Data una matrice $A \in \mathbb{C}^{m \times n}$, una matrice $B \in \mathbb{C}^{k \times h}, 0 \leq k < m, 0 \leq h < n$, è detta *sottomatrice* di A se è ottenuta da A eliminando $m - k$ righe ed $n - h$ colonne. Data una matrice $A \in \mathbb{C}^{n \times n}$, una sottomatrice quadrata B di ordine $k \leq n$, di A è detta *sottomatrice principale* se gli elementi principali di B sono anche gli elementi principali di A. Una sottomatrice B principale di ordine k di A è detta *sottomatrice principale di testa* se è formata dagli elementi $a_{ij}, i, j = 1, 2, \ldots, k$.

Esempio A.1 Sia

$$A = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}.$$

La matrice

$$\begin{pmatrix} 1 & 3 \\ 4 & 6 \end{pmatrix}$$

è sottomatrice di ordine 2 di A, la matrice

$$\begin{pmatrix} 1 & 3 \\ 7 & 9 \end{pmatrix}$$

è sottomatrice principale di ordine 2 di A, la matrice

$$\begin{pmatrix} 1 & 2 \\ 4 & 5 \end{pmatrix}$$

è sottomatrice principale di testa di ordine 2 di A.

A.2 Vettori

Se $A \in \mathbb{C}^{m\times 1}$ ($A \in \mathbb{C}^{1\times m}$), la matrice si riduce ad una sola colonna (riga) e viene detta *vettore colonna* (*riga*) ad m *elementi* o *componenti*. Generalmente con il termine *vettore* si intende un vettore colonna e, se ha m componenti, si dice che $\mathbf{x} \in \mathbb{C}^m$. Si usa scrivere:

$$\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{pmatrix}, \qquad \mathbf{x} = \begin{pmatrix} x_1, x_2, \ldots, x_m \end{pmatrix}^T.$$

Si indica con $\mathbf{0}$ il vettore di componenti nulle.
Casi particolari del prodotto righe per colonne di matrici:

- *prodotto di una matrice per un vettore*: sia $A \in \mathbb{C}^{m\times n}$ e siano $\mathbf{y} \in \mathbb{C}^m$, $\mathbf{x} \in \mathbb{C}^n$. Allora $\mathbf{y} = A\,\mathbf{x}$, $\Leftrightarrow$ $y_i = \sum_{j=1}^{n} a_{ij}x_j, \quad i = 1, 2, \ldots, m$;
- *prodotto interno tra vettori*: siano $\mathbf{x}, \mathbf{y} \in \mathbb{C}^m$. Allora $\mathbf{x}^H\mathbf{y} = \alpha$, $\Leftrightarrow$ $\sum_{j=1}^{m} \bar{x}_j y_j = \alpha$.

Definizione A.1 *I vettori* $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n \in \mathbb{C}^m, n \leq m$, *si dicono* linearmente indipendenti *se dalla condizione*

$$\sum_{j=1}^{n} \alpha_j \mathbf{x}_j = \mathbf{0}, \quad \alpha_j \in \mathbb{C},$$

segue che

$$\alpha_j = 0, \quad j = 1, 2, \ldots, n.$$

Definizione A.2 *Sia S un sottospazio di $\mathbb{C}^n$. I vettori $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_k \in S$ costituiscono una* base *di S se ogni vettore $\mathbf{v} \in S$ si può esprimere, in modo unico, come una loro combinazione lineare:*

$$\mathbf{v} = \sum_{j=1}^{k} \alpha_j \mathbf{x}_j.$$

Si dice che S è generato dalla base $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_k$.

Definizione A.3 *I vettori non nulli $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n \in \mathbb{C}^m$ si dicono* ortogonali *se $\mathbf{x}_i^H\mathbf{x}_j = 0$ per $i \neq j$; si dicono* ortonormali *se sono ortogonali ed inoltre $\mathbf{x}_i^H\mathbf{x}_i = 1$.*

A.3 Determinante

Sia $A \in \mathbb{C}^{n\times n}$. Il determinante di A può essere espresso utilizzando la *regola di Laplace*. Indicata con A_{ij} la sottomatrice quadrata di ordine $n-1$ ottenuta dalla matrice A eliminando la i-esima riga e la j-esima colonna, si ha:

$$\det(A) = \begin{cases} a_{11} & \text{se } n = 1, \\ \sum_{j=1}^{n} (-1)^{i+j} a_{ij} \det A_{ij} & \text{se } n > 1. \end{cases}$$

Siano $A, B \in \mathbb{C}^{n\times n}$, $\alpha \in \mathbb{C}$; valgono le seguenti proprietà:

- $\det(A) = \prod_{j=1}^{n} a_{jj}$ se A è diagonale o triangolare;
- $\det(I) = 1$;
- $\det(A^T) = \det(A)$;
- $\det(A^H) = \overline{\det(A)}$;
- $\det(A^{-1}) = 1/\det(A)$;
- $\det(A\,B) = \det(A)\det(B)$ (regola di *Binet*);
- $\det(\alpha A) = \alpha^n \det(A)$;
- A è nonsingolare $\Leftrightarrow$ $\det(A) \neq 0$.

Una importante classe di matrici ortogonali è quella delle *matrici di permutazione*. Una matrice di permutazione si ottiene dalla matrice identità permutandone l'ordine delle colonne. In particolare, una matrice di permutazione si dice *elementare* se viene effettuata solo la permutazione di due colonne. Ad esempio, la matrice

$$P = \begin{pmatrix} 0 & 1 & & & \\ 1 & 0 & & & \\ & & 1 & & \\ & & & \ddots & \\ & & & & 1 \end{pmatrix}$$

è una matrice di permutazione elementare. Sono note le seguenti proprietà:

- se P è una matrice di permutazione elementare, $\det(P) = -1$ e, inoltre, $P = P^T = P^{-1}$.

A.4 Sistemi lineari

Sia $A \in \mathbb{C}^{m\times n}$ e si considerino i seguenti sottospazi:

- l'*immagine* di A (o *range* di A), $\mathrm{I}(A) = \{\mathbf{y} \in \mathbb{C}^m : \mathbf{y} = A\,\mathbf{x},\ \mathbf{x} \in \mathbb{C}^n\}$;
- il *nucleo* di A (o *spazio nullo* di A), $\mathrm{N}(A) = \{\mathbf{x} \in \mathbb{C}^n : A\,\mathbf{x} = \mathbf{0}\}$.

Si definisce il *rango* di A,

$$\mathrm{rank}(A) = \dim(\mathrm{I}(A)).$$

È possibile dimostrare che:

- $\mathrm{rank}(A) = \mathrm{rank}(A^T) = \mathrm{rank}(\bar{A}) = \mathrm{rank}(A^H)$;
- $\mathrm{rank}(A) \leq \min\{m, n\}$;
- $\mathrm{rank}(A) + \dim(\mathrm{N}(A)) = n$.

Se A è di ordine n ed il suo rango è pari ad n, allora A è nonsingolare. Inoltre A è nonsingolare se e solo se $\dim(\mathrm{N}(A)) = 0$, ovvero il nucleo di A è costituito dal solo vettore nullo.

Definizione A.4 *Siano* $A \in \mathbb{C}^{m\times n}, \mathbf{b} \in \mathbb{C}^m$; *si definisce* sistema lineare di m equazioni in n incognite *il sistema*

$$A\mathbf{x} = \mathbf{b},$$

dove $\mathbf{x} \in \mathbb{C}^n$ *è il vettore delle incognite,* A *è la matrice del sistema e* $\mathbf{b}$ *è il vettore dei termini noti. Il sistema si dice* consistente *se ha almeno una soluzione.*

Le seguenti condizioni sono equivalenti:

- il sistema $A\mathbf{x} = \mathbf{b}$ è consistente;
- $\mathbf{b} \in \mathrm{I}(A)$;
- $\mathrm{rank}(A) = \mathrm{rank}(A|\mathbf{b})$ (teorema di Rouché-Capelli).

Inoltre,

- se $m = n$:
  - se A è nonsingolare il sistema ammette la (unica) soluzione $\mathbf{x} = A^{-1}\mathbf{b}$. In particolare, se $\mathbf{b} = \mathbf{0}$ la soluzione è $\mathbf{x} = \mathbf{0}$;

  - se A è singolare il sistema può non essere consistente.

- se $m \neq n$:
  - se $m < n$, cioè se ho più incognite che equazioni, il sistema se è consistente ha infinite soluzioni in quanto $\dim(\mathrm{I}(A)) \leq m$.
  - se $m > n$, cioè se ho più equazioni che incognite, il sistema può essere consistente solo se vi sono almeno $m-n$ equazioni che sono combinazioni lineari delle altre.

A.5 Autovalori ed autovettori

Definizione A.5 *Siano* $A \in \mathbb{C}^{n\times n}, \lambda \in \mathbb{C}, \mathbf{x} \in \mathbb{C}^n, \mathbf{x} \neq \mathbf{0}$ *tali che valga la relazione*

$$A\mathbf{x} = \lambda\mathbf{x}.$$

Allora si dice che λ *è* autovalore *di* A *e* $\mathbf{x}$ *è un corrispondente* autovettore.

La relazione

$$A\mathbf{x} = \lambda\mathbf{x} \qquad \Leftrightarrow \qquad (A-\lambda I)\mathbf{x} = \mathbf{0}.$$

Per il teorema di Rouché-Capelli un sistema lineare omogeneo ammette sempre soluzione. Inoltre, ha soluzione non nulla se e solo se la matrice dei coefficienti del sistema è singolare:

$$\det(A-\lambda I) = 0. \tag{A.1}$$

Dal calcolo di $\det(A-\lambda I)$ risulta:

$$p(\lambda) \equiv \det\,(A-\lambda I) = (-1)^n\lambda^n + (-1)^{n-1}\sigma_1\lambda^{n-1} + \cdots - \sigma_{n-1}\lambda + \sigma_n,$$

dove ciascun coefficiente $\sigma_i, i = 1, 2, \ldots, n$, è la somma dei minori (determinanti delle sottomatrici) principali di A di ordine i. In particolare,

$$\begin{aligned} \sigma_1 &= \sum_{i=1}^{n} a_{ii} \equiv \mathrm{tr}(A) && \text{traccia di } A \\ \sigma_n &= \det(A) && \text{determinante di } A \end{aligned}$$

Il polinomio $p(\lambda)$ è detto *polinomio caratteristico* della matrice A, mentre l'equazione (A.1) ($\Leftrightarrow p(\lambda) = 0$) è detta *equazione caratteristica*.
Gli autovalori di una matrice $A \in \mathbb{C}^{n\times n}$ coincidono con le radici dell'equazione caratteristica. Allora

1. $\lambda = 0$ è autovalore se e solo se $p(0) = 0 \Leftrightarrow \sigma_n = 0 \Leftrightarrow \det(A) = 0$;
2. $\sum_{i=1}^{n} \lambda_i = \mathrm{tr}(A)$;
3. $\prod_{i=1}^{n} \lambda_i = \det(A)$.

L'insieme

$$\sigma(A) = \{\lambda \in \mathbb{C} : \lambda \text{ è autovalore di } A\}$$

si dice *spettro* di A. La quantità

$$\rho(A) = \max_{\lambda \in \sigma(A)} |\lambda|$$

si dice *raggio spettrale* di A.

Definizione A.6 *Una matrice $A \in \mathbb{C}^{n\times n}$ si dice* convergente *se*

$$\lim_{k\to\infty} A^k = O,$$

dove O è la matrice nulla.

Teorema A.1 *Una matrice $A \in \mathbb{C}^{n\times n}$ è convergente se e solo se $\rho(A) < 1$.*

A.5.1 Trasformazione per similitudine

Definizione A.7 *Due matrici $A, B \in \mathbb{C}^{n\times n}$ si dicono* simili *se esiste una matrice nonsingolare S per cui*

$$A = SBS^{-1}.$$

Teorema A.2 *Due matrici simili A e B hanno gli stessi autovalori. Inoltre, per ogni autovalore λ, se $\mathbf{x}$ è autovettore di A, allora $S^{-1}\mathbf{x}$ è autovettore di B.*

Dim. Poiché A e B sono simili, i.e.,

$$A = SBS^{-1},$$

si ha

$$\begin{aligned} \det(A - \lambda I) &= \det(SBS^{-1} - \lambda I) = \det(S(B - \lambda I)S^{-1}) \\ &= \det S \det(B - \lambda I) \det S^{-1} = \det(B - \lambda I). \end{aligned}$$

Siccome le due matrici hanno lo stesso polinomio caratteristico, ne segue che hanno anche gli stessi autovalori. Inoltre, se $\mathbf{x}$ è autovettore di A corrispondente all'autovalore λ risulta:

$$A\mathbf{x} = \lambda\mathbf{x} \qquad \Leftrightarrow \qquad SBS^{-1}\mathbf{x} = \lambda\mathbf{x}$$

e quindi

$$BS^{-1}\mathbf{x} = \lambda S^{-1}\mathbf{x}.$$

Perciò il vettore $S^{-1}\mathbf{x}$ è autovettore di B corrispondente a λ. □

Osservazione A.1 *Dal teorema precedente si ha che se due matrici sono simili hanno lo stesso determinante e la stessa traccia.*

Fra le trasformazioni per similitudine che associano alla matrice B la matrice $A = SBS^{-1}$ hanno particolare importanza quelle per cui S è unitaria. Il teorema seguente mostra come sia possibile, mediante una trasformazione per similitudine unitaria, ricondurre una qualsiasi matrice ad una forma triangolare superiore.

Teorema A.3 (Forma canonica di Schur)
Sia $A \in \mathbb{C}^{n\times n}$, $\lambda_1, \lambda_2, \ldots, \lambda_n$ i suoi autovalori. Allora esistono una matrice U unitaria e una matrice T triangolare superiore con elementi diagonali uguali a $\lambda_i, i = 1, 2, \ldots, n$, tale che

$$A = UTU^H.$$

Definizione A.8 *Una matrice A simile ad una matrice diagonale D si dice* diagonalizzabile.

Teorema A.4 *Una matrice A di ordine n è diagonalizzabile se e solo se ha n autovettori linearmente indipendenti.*

Teorema A.5 *Autovettori corrispondenti ad autovalori distinti sono linearmente indipendenti.*

A.5.2 Localizzazione degli autovalori

Definizione A.9 *Sia $A \in \mathbb{C}^{n\times n}$. Gli insiemi del piano complesso*

$$K_i = \{z \in \mathbb{C} : |z - a_{ii}| \le r_i,\ r_i \equiv \sum_{\substack{j=1 \\ j\neq i}}^{n} |a_{ij}|\}, \qquad i = 1, 2, \ldots, n,$$

sono dei cerchi di centro a_{ii} e raggio r_i. Essi sono detti cerchi di Gerschgorin.

Vale il seguente

Teorema A.6 (primo teorema di Gerschgorin)
Gli autovalori della matrice A di ordine n sono tutti contenuti in

$$\bigcup_{i=1,2,\ldots,n} K_i.$$

Dim. Sia λ un autovalore di A e $\mathbf{x}$ un autovettore corrispondente, ossia

$$A\mathbf{x} = \lambda\mathbf{x}, \quad \mathbf{x} \neq \mathbf{0}.$$

Allora si ha

$$\sum_{j=1}^{n} a_{ij}x_j = \lambda x_i, \quad i = 1, 2, \ldots, n,$$

da cui

$$(\lambda - a_{ii})x_i = \sum_{\substack{j=1 \\ j\neq i}}^{n} a_{ij}x_j \quad i = 1, 2, \ldots, n. \tag{A.2}$$

Sia x_m la componente di $\mathbf{x}$ di massimo modulo, cioè

$$|x_m| = \max_{j\,=\,1,\ldots,n} |x_j|.$$

Si osservi che $|x_m| \neq 0$ essendo $\mathbf{x} \neq \mathbf{0}$. Ponendo $i = m$ in (A.2) si ha

$$(\lambda - a_{mm})x_m = \sum_{\substack{j=1 \\ j\neq m}}^{n} a_{mj}x_j,$$

da cui, passando ai moduli:

$$|\lambda - a_{mm}|\,|x_m| \leq \sum_{\substack{j=1 \\ j\neq m}}^{n} |a_{mj}|\,|x_j| \leq \left(\sum_{\substack{j=1 \\ j\neq m}}^{n} |a_{mj}| \right) |x_m|.$$

Infine, dividendo per $|x_m| > 0$ si ottiene

$$|\lambda - a_{mm}| \leq \sum_{\substack{j=1 \\ j\neq m}}^{n} |a_{mj}|$$

e quindi $\lambda \in K_m$. Poiché a priori non è noto il valore dell'indice m, è possibile solo dire che λ appartiene all'unione di tutti i cerchi $K_i, i = 1, 2, \ldots, n$. □

Osservazione A.2 *Poiché il teorema precedente può essere applicato anche alla matrice A^T che ha gli stessi autovalori di A (dimostrare per esercizio), risulta che gli autovalori di A appartengono anche all'unione dei cerchi*

$$H_i = \{z \in \mathbb{C} : |z - a_{ii}| \leq r_i,\ r_i \equiv \sum_{\substack{j=1 \\ j \neq i}}^{n} |a_{ji}|\}, \qquad i = 1, 2, \ldots, n,$$

e quindi gli autovalori di A appartengono all'insieme

$$\left(\bigcup_{i=1}^{n} K_i\right) \bigcap \left(\bigcup_{i=1}^{n} H_i\right).$$

Esempio A.2 *Sia*

$$A = \begin{pmatrix} 15 & -2 & 2 \\ 1 & 10 & -3 \\ -2 & 1 & 0 \end{pmatrix}.$$

Allora

$$\begin{aligned} K_1 &= \{z \in \mathbb{C} : |z - 15| \leq |-2| + |2|\} = \{z \in \mathbb{C} : |z - 15| \leq 4\}, \\ K_2 &= \{z \in \mathbb{C} : |z - 10| \leq |1| + |-3|\} = \{z \in \mathbb{C} : |z - 10| \leq 4\}, \\ K_3 &= \{z \in \mathbb{C} : |z - 0| \leq |-2| + |0|\} = \{z \in \mathbb{C} : |z| \leq 3\}. \end{aligned}$$

Inoltre

$$A^T = \begin{pmatrix} 15 & 1 & -2 \\ -2 & 10 & 1 \\ 2 & 3 & 0 \end{pmatrix}.$$

Allora

$$\begin{aligned} H_1 &= \{z \in \mathbb{C} : |z - 15| \leq 3\}, \\ H_2 &= \{z \in \mathbb{C} : |z - 10| \leq 3\}, \\ H_3 &= \{z \in \mathbb{C} : |z| \leq 5\}. \end{aligned}$$

Gli autovalori sono quindi contenuti nell'insieme

$$(K_1 \cup K_2 \cup K_3) \cap (H_1 \cup H_2 \cup H_3) \equiv K_3 \cup H_1 \cup H_2;$$

infatti $\lambda_1 = 0.5121$, $\lambda_2 = 14.1026$, $\lambda_3 = 10.3854$.

Teorema A.7 (secondo teorema di Gerschgorin)
Se l'unione M_1 di k cerchi di Gerschgorin è disgiunta dall'unione M_2 dei rimanenti $n-k$, allora k autovalori appartengono ad M_1 ed $n-k$ appartengono a M_2.

Proposizione A.1 *Se la matrice A di ordine n è a predominanza diagonale forte allora A è non singolare.*

Dim. Se A è a predominanza diagonale forte ogni cerchio di Gerschgorin ha il centro ad una distanza dall'origine degli assi maggiore del raggio. Quindi ciascun cerchio non passa per l'origine. Pertanto, dal primo teorema di Gerschgorin risulta che $\lambda = 0$ non può essere autovalore ($\lambda = 0 \Leftrightarrow \det(A) = 0$) e dunque la tesi.□

Una interessante applicazione del primo teorema di Gerschgorin è la possibilità di localizzare le radici di una *equazione algebrica*. Sia

$$x^k + a_{k-1}x^{k-1} + \cdots + a_1 x + a_0 = 0, \qquad a_i \in \mathbb{C},\ i = 0, 1, \ldots, k-1. \quad \text{(A.3)}$$

Si consideri la matrice quadrata di ordine k :

$$F = \begin{pmatrix} 0 & 1 & 0 & \ldots & 0 & 0 \\ 0 & 0 & 1 & \ldots & 0 & 0 \\ 0 & 0 & 0 & \ldots & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & \ldots & 0 & 1 \\ -a_0 & -a_1 & -a_2 & \ldots & -a_{k-2} & -a_{k-1} \end{pmatrix}.$$

Tale matrice è detta *matrice di Frobenius o matrice compagna*.
Si verifica che (A.3) è l'equazione caratteristica della matrice F: quindi i suoi autovalori sono le radici dell'equazione assegnata. Allora è possibile localizzare le radici dell'equazione (A.3) facendo uso del primo teorema di Gerschgorin.

Esempio A.3 *Si consideri l'equazione algebrica:*

$$x^4 - 8x^3 + 16x^2 - 8x + 15 = 0.$$

Allora, essendo

$$F = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -15 & 8 & -16 & 8 \end{pmatrix}$$

si ha

$$\begin{aligned} K_1 = K_2 = K_3 &= \{z \in \mathbb{C} : |z| \leq 1\}, \\ K_4 &= \{z \in \mathbb{C} : |z - 8| \leq 39\}; \end{aligned}$$

$$
\begin{aligned}
H_1 &= \{z \in \mathbb{C} : |z| \leq 15\}, \\
H_2 &= \{z \in \mathbb{C} : |z| \leq 9\}, \\
H_3 &= \{z \in \mathbb{C} : |z| \leq 17\}, \\
H_4 &= \{z \in \mathbb{C} : |z - 8| \leq 1\}.
\end{aligned}
$$

Allora gli autovalori sono contenuti nell'insieme:

$$\left(\bigcup_{i=1}^{4} K_i\right) \bigcap \left(\bigcup_{i=1}^{4} H_i\right) = (K_1 \cup K_4) \cap (H_3 \cup H_4) = K_4 \cap H_3 = H_3.$$

Infatti, essi sono $\lambda_1 = 3$, $\lambda_2 = 5$, $\lambda_{3,4} = \pm i$.

A.6 Norme

A.6.1 Norme vettoriali

Definizione A.10 *Una funzione a valori reali non negativi*

$$
\begin{aligned}
\|\cdot\| : \mathbb{C}^n &\longrightarrow \mathbb{R}^+ \\
\mathbf{x} &\longmapsto \|\mathbf{x}\|
\end{aligned}
$$

che verifica le seguenti proprietà:

1) $\|\mathbf{x}\| = 0 \quad \Leftrightarrow \quad \mathbf{x} = \mathbf{0}$

2) $\|\alpha\mathbf{x}\| = |\alpha| \, \|\mathbf{x}\|, \quad \forall\, \alpha \in \mathbb{C}$

3) $\|\mathbf{x} + \mathbf{y}\| \leq \|\mathbf{x}\| + \|\mathbf{y}\|, \quad \forall\, \mathbf{y} \in \mathbb{C}^n$

è detta norma vettoriale.

Si introducono alcune delle norme vettoriali più usate:

$$
\begin{aligned}
\|\mathbf{x}\|_1 &= \sum_{i=1}^{n} |x_i| \qquad \text{norma 1 (o norma Manhattan)} \\
\|\mathbf{x}\|_2 &= \sqrt{\sum_{i=1}^{n} |x_i|^2} = \sqrt{\mathbf{x}^H \mathbf{x}} \qquad \text{norma 2 (o norma euclidea)} \\
\|\mathbf{x}\|_\infty &= \max_{i \leq 1 \leq n} |x_i| \qquad \text{norma } \infty \text{ (o norma del massimo).}
\end{aligned}
$$

L'insieme

$$S_p = \{\mathbf{x} \in \mathbb{R}^2 : \|\mathbf{x}\|_p \leq 1\}, \quad p = 1, 2, \infty,$$

detto *sfera unitaria* di $\mathbb{R}^2$, per le norme definite risulta:

$$\|\mathbf{x}\|_1 = |x_1| + |x_2| \leq 1; \quad \|\mathbf{x}\|_2 = \sqrt{|x_1|^2 + |x_2|^2} \leq 1; \quad \|\mathbf{x}\|_\infty = \max(|x_1|, |x_2|) \leq 1.$$

Osservazione A.3 *Si può dimostrare che per ogni* $\mathbf{x} \in \mathbb{C}^n$ *si ha:*

$$\|\mathbf{x}\|_\infty \leq \|\mathbf{x}\|_2 \leq \sqrt{n}\|\mathbf{x}\|_\infty.$$

Il concetto di norma può essere esteso al caso delle matrici.

Definizione A.11 *Una funzione a valori reali non negativi*

$$\|\cdot\| : \begin{array}{rcl} \mathbb{C}^{m\times n} & \longrightarrow & \mathbb{R}^+ \\ A & \longmapsto & \|A\| \end{array}$$

che verifica le seguenti proprietà:

1) $\|A\| = 0 \quad \Leftrightarrow \quad A = O$

2) $\|\alpha A\| = |\alpha|\,\|A\|, \quad \forall\, \alpha \in \mathbb{C}$

3) $\|A + B\| \leq \|A\| + \|B\|, \quad \forall\, B \in \mathbb{C}^{m\times n},$

4) $\|A\,B\| \leq \|A\|\,\|B\|, \quad \forall\, B \in \mathbb{C}^{n\times p},$

è detta norma matriciale.

Definizione A.12 *Una norma matriciale si dice* coerente *o* compatibile *con una norma vettoriale se*

$$\|A\,\mathbf{x}\| \leq \|A\|\,\|\mathbf{x}\|.$$

Ad una norma vettoriale è possibile associare una norma matriciale, definendo

$$\|A\| = \max_{\|\mathbf{x}\| = 1} \|A\,\mathbf{x}\|.$$

La norma così definita viene detta *norma matriciale indotta (o naturale)* dalla norma vettoriale.
Si può dimostrare che le norme matriciali indotte dalle norme vettoriali definite precedentemente sono le seguenti:

$$\begin{array}{rcll} \|A\|_1 & = & \max\limits_{j\,=\,1,2,\ldots,n} \sum\limits_{i=1}^{m} |a_{ij}| & \text{norma 1} \\ \|A\|_2 & = & \sqrt{\rho(A^H A)} & \text{norma 2} \\ \|A\|_\infty & = & \max\limits_{i\,=\,1,2,\ldots,m} \sum\limits_{j=1}^{n} |a_{ij}| & \text{norma } \infty. \end{array}$$

Osservazione A.4 *Se $A = A^H$ (A è una matrice Hermitiana) allora*

$$\begin{aligned} \|A\|_1 &= \|A\|_\infty \\ \|A\|_2 &= \sqrt{\rho(A^H A)} = \sqrt{\rho(A^2)} = \sqrt{(\rho(A))^2} = \rho(A). \end{aligned}$$

Per le norme considerate valgono le seguenti relazioni:

$$\begin{aligned} &\frac{1}{\sqrt{n}}\|A\|_\infty \le \|A\|_2 \le \sqrt{n}\|A\|_\infty, \\ &\frac{1}{\sqrt{n}}\|A\|_1 \le \|A\|_2 \le \sqrt{n}\|A\|_1, \\ &\|A\|_2 \le \sqrt{\|A\|_1\|A\|_\infty}. \end{aligned}$$

Esempio A.4 Sia

$$A = \begin{pmatrix} -1 & 0 & 1+2i \\ 0 & 2 & 1-i \\ 1-2i & 1+i & 0 \end{pmatrix}.$$

Essendo $A = A^H$ si ha $\|A\|_2 \equiv \rho(A) \approx 3.0791$ e $\|A\|_1 = \|A\|_\infty = \sqrt{5}+\sqrt{2} \approx 3.6503$. Quindi

$$\|A\|_2 \le \|A\|_\infty.$$

Osservazione A.5 *Una norma matriciale indotta da una norma vettoriale è coerente, i.e.,*

$$\|A\,\mathbf{x}\| \le \|A\|\,\|\mathbf{x}\|.$$

Teorema A.8 *Sia $A \in \mathbb{C}^{n\times n}$. Per ogni norma matriciale indotta vale la seguente relazione:*

$$\rho(A) \le \|A\|. \tag{A.4}$$

Dim. Sia λ un autovalore di A ed $\mathbf{x}$ il corrispondente autovettore associato a λ, ossia

$$A\mathbf{x} = \lambda\mathbf{x}, \qquad \mathbf{x} \neq \mathbf{0}.$$

Prendendo una qualunque norma dei due membri si ha

$$\|\lambda\,\mathbf{x}\| = \|A\mathbf{x}\|$$

da cui

$$|\lambda|\,\|\mathbf{x}\| = \|A\mathbf{x}\|.$$

Utilizzando la norma matriciale indotta da quella vettoriale, si ottiene

$$|\lambda|\,\|\mathbf{x}\| \le \|A\|\,\|\mathbf{x}\|.$$

Essendo $\mathbf{x} \neq \mathbf{0}$, posso dividere per $\|\mathbf{x}\|$ (poiché sarà sicuramente un valore diverso da zero) e quindi ottenere

$$|\lambda| \leq \|A\|.$$

Siccome λ è un qualunque autovalore di A, la relazione precedente è valida anche per quello di modulo massimo, da cui la tesi. □

Dalla Definizione A.6 e dal Teorema A.1 segue immediatamente che

Corollario A.1 *Sia $A \in \mathbb{C}^{n\times n}$. Se*

$$\|A\| < 1 \quad \Rightarrow \quad A \text{ è convergente.}$$

Un'altra norma frequentemente usata è quella così definita:

$$\|A\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n} |a_{ij}|^2} = \sqrt{\mathrm{tr}(A^H A)}.$$

Essa è detta *norma di Frobenius* (o di Schur).
Tale norma **non** è indotta da alcuna norma vettoriale in quanto:

$$\|I\|_F = \sqrt{n}$$

mentre

$$\|I\| = \max_{\|\mathbf{x}\|=1} \|I\,\mathbf{x}\| = 1$$

qualunque sia la norma vettoriale considerata.

Proposizione A.2 *Sia $\|A\| < 1$. Allora $I + A$ è non singolare e*

$$\|(I+A)^{-1}\| \leq \frac{1}{1-\|A\|}.$$

Dim. Essendo $\|A\| < 1$ da (A.4) risulta $\rho(A) < 1$. Quindi la matrice $I + A$ non può avere autovalori nulli e pertanto è non singolare.

Dalla relazione $(I + A)(I + A)^{-1} = I$ segue che

$$(I+A)^{-1} = I - A(I+A)^{-1}$$

e poiché $\|I\| = 1$ si ha

$$\|(I+A)^{-1}\| = \|I - A(I+A)^{-1}\| \leq 1 + \|A\|\|(I+A)^{-1}\|$$

e quindi

$$(1-\|A\|)\|(I+A)^{-1}\| \leq 1$$

da cui, essendo $\|A\| < 1$, segue la tesi. □