Aritmetica di macchina

S. Maset
Dipartimento di Matematica e Geoscienze
Università di Trieste
maset@units.it

May 18, 2021

Analizzeremo qui la discretizzazione dei numeri reali con i numeri di macchina. In particolare, studieremo gli errori introdotti nell'approssimare un numero reale con un numero di macchina (detti *errori di arrotondamento*) e l'effetto di tali errori nelle computazioni.

Però, prima di introdurre i numeri di macchina, è necessario presentare la rappresentazione normalizzata in base dei numeri reali.

1 Rappresentazione normalizzata in base

Sia $B > 1$ un numero naturale che chiameremo *base di rappresentazione*. Inoltre, chiameremo i numeri naturali $0, 1, 2, \ldots, B-1$ *cifre* nella base B.

Teorema 1 (Rappresentazione normalizzata in base B) *Per ogni numero reale $x \neq 0$ esistono unici $p \in \mathbb{Z}$ e una successione di cifre $b_0, b_{-1}, b_{-2}, b_{-3}, \ldots$ nella base B con $b_0 \neq 0$ e non definitivamente uguale a $B-1$ tali che*

$$x = \pm \left(b_0.b_{-1}b_{-2}b_{-3}\ldots\right)_B \cdot B^p. \tag{1}$$

La (1) è la *rappresentazione normalizzata in base B* di x, dove:

- $\pm$ è il segno di x;
- p è detto l'*esponente* della rappresentazione;
- $$\begin{aligned}\left(b_0.b_{-1}b_{-2}b_{-3}\ldots\right)_B &:= b_0 + b_{-1}B^{-1} + b_{-2}B^{-2} + b_{-3}B^{-3} + \cdots \\ &= \sum_{k=0}^{\infty} b_{-k}B^{-k}\end{aligned}$$

  è detto la *mantissa* o *caratteristica* della rappresentazione.

La rappresentazione normalizzata in base 10 è anche detta *rappresentazione (o notazione) scientifica.*

Osservare che la successione $b_0, b_{-1}, b_{-2}, b_{-3}, \ldots$ è richiesta essere non definitivamente uguale a $B-1$ per avere l'unicità della rappresentazione: osservare ad esempio che nella base $B=10$ si ha $1.2 = 1.999\ldots = 1.\overline{9}$.

Esempio 2 *Ecco alcuni esempi di rappresentazione normalizzata:*

$$\begin{aligned}
&0.0938 = 9.38 \cdot 10^{-2}\\
&0.0938 = \frac{1}{16} + \frac{1}{32} = (0.00011)_2 = (1.1)_2 \cdot 2^{-4}\\
&1023 = 1.023 \cdot 10^3\\
&1023 = 2^{10} - 1 = (1111111111)_2 = (1.111111111)_2 \cdot 2^9.
\end{aligned}$$

Per la mantissa si ha

$$1 \le (b_0.b_{-1}b_{-2}b_{-3}\ldots)_B < B. \tag{2}$$

Infatti,

$$(b_0.b_{-1}b_{-2}b_{-3}\ldots)_B = b_0 + b_{-1}B^{-1} + b_{-2}B^{-2} + b_{-3}B^{-3} + \cdots \ge b_0 \ge 1$$

essendo b_0 una cifra diversa da zero e

$$\begin{aligned}
&(b_0.b_{-1}b_{-2}b_{-3}\ldots)_B\\
&= b_0 + b_{-1}B^{-1} + b_{-2}B^{-2} + b_{-3}B^{-3} + \cdots\\
&< B - 1 + (B-1)B^{-1} + (B-1)B^{-2} + (B-1)B^{-3} + \cdots\\
&\quad \text{avendo } b_{-k} \le B-1 \text{ per } k \in \{0,1,2,3,\ldots\}\\
&\quad \text{e non avendo } b_0 = b_{-1} = b_{-2} = b_{-3} = \cdots = B-1\\
&= (B-1)\left(1 + B^{-1} + B^{-2} + B^{-3} + \cdots\right)\\
&= (B-1)\,\frac{1}{1-B^{-1}} = B.
\end{aligned}$$

Da (2) si ottiene

$$B^p \le |x| = (b_0.b_{-1}b_{-2}b_{-3}\ldots)_B \cdot B^p < B^{p+1}.$$

Per tale motivo, si dice che x ha *ordine di grandezza* B^p nella base B.

In ogni base B, si ha

$$0 = (0.000\ldots)_B.$$

Quindi 0 non ha una rappresentazione normalizzata, non potendo disporre di una prima cifra non nulla con la quale cominciare la rappresentazione.

2 Numeri di macchina

Nel calcolatore si possono rappresentare solo un numero finito di numeri reali e, per ciascuno di questi numeri reali che sono rappresentati, si possono rappresentare solo un numero finito delle cifre di una sua rappresentazione in base.

I numeri reali che vengono rappresentati nel calcolatore sono detti *numeri di macchina*.

Vi sono due tipi di rappresentazione dei numeri di macchina nel calcolatore: la *rappresentazione floating-point* e la *rappresentazione fixed-point*.

La rappresentazione floating point garantisce un piccolo errore relativo quando un numero reale è approssimato con un numero di macchina, mentre la rappresentazione fixed point garantisce un piccolo errore assoluto.

Per tale motivo, la rappresentazione floating-point è importante in ambito scientifico e ingegneristico, mentre la rappresentazione fixed point è importante in altri ambiti, ad esempio quello bancario o amministrativo.

Qui, noi siamo interessati solo alla rappresentazione floating-point.

Un *insieme dei numeri di macchina* in rappresentazione floating point è definito da quattro parametri. Questi parametri riguardano la rappresentazione normalizzata dei numeri di macchina e sono:

1) La base B di rappresentazione.
2) Il minimo esponente m ammesso.
3) Il massimo esponente M ammesso.
4) Il numero t di cifre della mantissa dopo il punto " . " che vengono considerate, nel senso che dalla $t+1-$ esima in poi le cifre sono uguali a zero.

I corrispondenti numeri di macchina sono i numeri reali non nulli x la cui rappresentazione normalizzata in base B è del tipo

$$x = \pm \left(b_0.b_{-1}b_{-2}b_{-3}\dots b_{-t}000\cdots\right)_B \cdot B^p = \pm \left(b_0.b_{-1}b_{-2}b_{-3}\dots b_{-t}\right)_B \cdot B^p,$$

dove $m \le p \le M$.

Inoltre, anche lo zero (che non ha una rappresentazione normalizzata) è un numero di macchina.

Osserviamo che i possibili numeri di macchina sono in numero finito di

$$\underbrace{2}_{\text{scelta di + o }-} \cdot \underbrace{(B-1)}_{\text{scelte di } b_0 \neq 0} \cdot \underbrace{B}_{\text{scelte di } b_{-1}} \cdot \cdots \cdot \underbrace{B}_{\text{scelte di } b_{-t}} \cdot \underbrace{(M-m+1)}_{\text{scelte di } p} + \underbrace{1}_{\text{lo zero}}$$

$$= 2\,(B-1)\,B^t\,(M-m+1) + 1.$$

Inoltre, ogni numero di macchina è rappresentato dalla seguente informazione finita:

- il segno + o −;
- l'esponente p, che è un numero in $\{m, m+1, \ldots, M-1, M\}$;
- le cifre della mantissa $b_0, b_{-1}, \ldots, b_{-t}$ che sono cifre nella base B, cioè numeri in $\{0, 1, \ldots, B-1\}$.

Esempio 3 Esempio didattico.

Un esempio didattico di insieme di numeri di macchina è quello definito dai parametri

$$B = 10,\ m = -1,\ M = 1,\ t = 2.$$

I numeri di macchina positivi sono

$$1.00 \cdot 10^{-1}, 1.01 \cdot 10^{-1}, 1.02 \cdot 10^{-1}, \ldots, 9.99 \cdot 10^{-1}, \quad p = -1,$$

$$1.00 \cdot 10^{0}, 1.01 \cdot 10^{0}, 1.02 \cdot 10^{0}, \ldots, 9.99 \cdot 10^{0}, \quad p = 0,$$

$$1.00 \cdot 10^{1}, 1.01 \cdot 10^{1}, 1.02 \cdot 10^{1}, \ldots, 9.99 \cdot 10^{1}, \quad p = 1,$$

cioè

$$.100, .101, .102, \ldots, .999,$$

$$1.00, 1.01, 1.02, \ldots, 9.99, \tag{3}$$

$$10.0, 10.1, 10.2, \ldots, 99.9.$$

La (3) spiega perchè il modo di rappresentare i numeri reali che stiamo considerando è detto rappresentazione floating point: sono sempre rappresentate tre cifre, ma la posizione del punto si sposta a seconda dell'ordine di grandezza del numero.

Invece, nella rappresentazione fixed point, la posizione del punto non si sposta con l'ordine di grandezza: i numeri sono in rappresentazione non normalizzata e con un certo numero assegnato di cifre prima e dopo il punto.

Importanti insiemi di numeri di macchina sono quelli definiti dallo *Standard IEEE* (IEEE è l'acronimo di Institute of Electrical and Electronics Engineers).

Lo standard IEEE prevede che, nel calcolatore, i numeri di macchina siano rappresentati nella base 2 in:

- *semplice precisione* usando per ogni numero di macchina una parola di memoria di 32 bit;
- *doppia precisione* usando una parola di memoria 64 bit;
- o *quadrupla precisione* usando una parola di memoria 128 bit.

Nella doppia precisione dello standard IEEE, quella a cui siamo interessati in quanto usata dai computer e dalle calcolatrici scientifiche, i parametri che definiscono l'insieme dei numeri di macchina sono

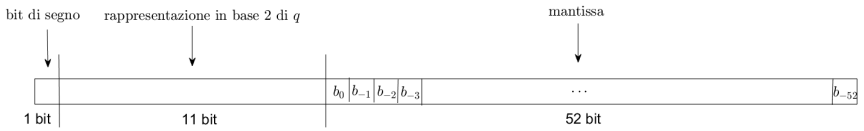
$$B = 2,\ m = -1022,\ M = 1023,\ t = 52.$$

I numeri di macchina non zero sono i numeri del tipo

$$x = \pm\,(1.b_{-1}b_{-2}b_{-3}\ldots b_{-52})_2 \cdot 2^p$$

con $-1022 \le p \le 1023$ e $b_{-1}, b_{-2}, b_{-3}, \ldots, b_{-52}$ sono cifre binarie 0 o 1. Si noti che $b_0 = 1$, dovendo essere $b_0 \neq 0$.

La parola di memoria di 64 bit per il numero x è così suddivisa (vedi figura):



- Il primo bit della parola contiene l'informazione del segno $\pm$: 0 per + e 1 per $-$.
- I bit dal secondo al dodicesimo contengono l'informazione dell'esponente p: in questi 11 bit si trova la rappresentazione in base 2 (con eventuali zeri in testa) dell'esponente traslato $q = 1023 + p$. I possibili valori per q sono $1, 2, 3, \ldots, 2046$ e servono appunto 11 bit per rappresentarli in base 2 (con 11 bit si possono rappresentare tutti i numeri da 0 a $2047 = 2^{11} - 1$).
- I bit dal tredicesimo al sessantaquatresimo contengono l'informazione della mantissa $(1.b_{-1}b_{-2}b_{-3}\cdots b_{-52})_2$: questi 52 bit sono $b_{-1}, b_{-2}, b_{-3}, \ldots, b_{-52}$.

Esempio 4 *Il numero*

$$10 = (1010)_2 = (1.010)_2 \cdot 2^3$$

è un numero di macchina nello standard IEEE in doppia precisione. Essendo

$$q = 1023 + p = 1026 = 1024 + 2 = 2^{10} + 2 = (10000000010)_2\,,$$

esso viene rappresentato con la parola di memoria

$$\underbrace{0}_{\textit{segno: 1 bit}}\quad\underbrace{10000000010}_{\textit{esponente: 11 bit}}\quad\underbrace{010\ldots0}_{\textit{mantissa: 52 bit}}\ .$$

2.1 Esercizi

Esercizio. Scrivere i numeri di macchina positivi dell'insieme di numeri di macchina definito dai parametri:

$$B = 3,\ m = -2,\ M = 2, t = 1.$$

Esercizio. Scrivere la parola di memoria di 64 bit corrispondente ai numeri 45 e -2^{-100} nello standard IEE in doppia precisione.

Esercizio. Se tutti i numeri dello Standard IEEE in doppia precisione fossero memorizzati in qualche supporto, quale sarebbe l'occupazione complessiva di memoria?

2.2 Numero di macchina più piccolo e numero di macchina più grande

In un insieme di numeri di macchina, il più piccolo numero positivo è

$$\left(1.\underbrace{000\ldots0}_{t \text{ cifre}}\right)_B \cdot B^m = B^m.$$

Esempio 5 *Nell' esempio didattico dove $B = 10$ e $m = -1$, il piu piccolo numero di macchina positivo è $10^{-1} = 0.1$.*

Nello standard IEEE in doppia precisione dove $B = 2$ e $m = -1022$, il più piccolo numero di macchina positivo è 2^{-1022}, il cui ordine di grandezza è 10^{-308}.

Invece, il più grande numero di macchina positivo è

$$\begin{aligned}
&\left((B-1).\underbrace{(B-1)\,(B-1)\ldots(B-1)}_{t}\right)_B \cdot B^M \\
&= \left((B-1) + (B-1)B^{-1} + (B-1)B^{-2} + \cdots + (B-1)B^{-t}\right) \cdot B^M \\
&= (B-1)\left(1 + B^{-1} + B^{-2} + \cdots + B^{-t}\right) \cdot B^M \\
&= (B-1)\,\frac{1 - B^{-(t+1)}}{1 - B^{-1}} \cdot B^M \\
&= B^{M+1} \cdot \left(1 - B^{-(t+1)}\right) \approx B^{M+1} \text{ essendo } B^{-(t+1)} \ll 1.
\end{aligned}$$

Esempio 6 *Nell'esempio didattico dove $B = 10$ e $M = 1$, il piu grande numero di macchina positivo è 99.9 che è circa $10^2 = 100$.*

Nello standard IEEE in doppia precisione dove $B = 2$ e $M = 1023$, il piu grande numero di macchina positivo è circa 2^{1024}, il cui ordine di grandezza è 10^{308}.

2.3 Epsilon di macchina

Consideriamo l'esempio didattico i cui numeri di macchina positivi sono

$$.100, .101, .102, \ldots, .999,$$

$$1.00, 1.01, 1.02, \ldots, 9.99,$$

$$10.0, 10.1, 10.2, \ldots, 99.9.$$

Essi sono suddivisi lungo le tre righe a seconda del loro ordine di grandezza. Osservare che:

- i numeri di macchina della prima riga, quelli con ordine di grandezza 10^{-1}, sono distanziati uno dall'altro di 0.001;
- i numeri della seconda riga, quelli con ordine di grandezza 10^0, sono distanziati l'uno dall'altro di 0.01,
- i numeri della terza riga, quelli con ordine di grandezza 10^1, sono distanziati l'uno dall'altro di 0.1.

In generale, in un insieme di numeri di macchina in rappresentazione floating point, la distanza tra due numeri di macchina consecutivi varia a seconda dell'ordine di grandezza dei numeri.

I numeri di macchina nell'intervallo $[B^p, B^{p+1})$, $p \in \{m, m+1, \ldots, M\}$, vale a dire quelli di ordine di grandezza B^p, che hanno la forma

$$(b_0.b_{-1}b_{-2}b_{-3}\ldots b_{-t})_B \cdot B^p,$$

sono distanziati l'uno dall'altro di

$$\left(0.\underbrace{0\ldots01}_{t \text{ cifre}}\right)_B \cdot B^p = B^{-t} \cdot B^p = B^{p-t}.$$

Il numero

$$\text{eps} := B^{-t}$$

(eps è un'abbreviazione di epsilon) è chiamato *epsilon di macchina*.

La distanza B^{p-t} tra due numeri di macchina consecutivi nell'intervallo $\left[B^p, B^{p+1}\right)$ è quindi $\text{eps} \cdot B^p$.

In particolare, eps è la distanza tra due numeri di macchina consecutivi nell'intervallo $[1, B)$.

Esempio 7 *Nell' esempio didattico dove* $B = 10$ *e* $t = 2$*, si ha* $\text{eps} = 10^{-2}$.

Nello standard IEEE in doppia precisione dove $B = 2$ *e* $t = 52$*, si ha* $\text{eps} = 2^{-52}$*, il cui ordine di grandezza è* 10^{-16}*. Questo significa che, se i numeri di macchina in tale standard fossero rappresentati in base* 10*, per passare da un numero di macchina a quello successivo bisognerebbe aggiungere* 1 *nella sedicesima cifra dopo il punto.*

Esercizio. Quanti sono i numeri macchina nell'intervallo $[B^p, B^{p+1})$, dove $p \in \{m, m+1, \ldots, M\}$ è fissato? Esprimere tale numero in termini di eps e B.

3 Approssimazione di numeri reali con numeri di macchina

Assumiamo di avere un insieme di numeri di macchina definito dai parametri B (base di rappresentazione), m (minimo esponente), M (massimo esponente) e t (numero di cifre della mantissa dopo il punto).

Ogni numero reale viene approssimato nel calcolatore con un numero di macchina.

Il numero di macchina che approssima un numero reale x viene denotato con $\mathrm{fl}(x)$ (fl sta per "floating").

Chiaramente, se x è un numero di macchina, si ha $\mathrm{fl}(x) = x$. Così, in particolare, $\mathrm{fl}(0) = 0$.

Nel seguito descriviamo $\mathrm{fl}(x)$ solo per un numero reale positivo x, in quanto, per un numero reale negativo x, si pone

$$\mathrm{fl}(x) = -\mathrm{fl}(|x|).$$

Sia x un numero reale positivo e sia

$$x = (b_0.b_{-1}b_{-2}b_{-3}\ldots b_{-t}b_{-t-1}b_{-t-2}\ldots)_B \cdot B^p$$

la sua rappresentazione normalizzata in base B.

Se $p < m$, allora il numero che si deve rappresentare è piu piccolo del più piccolo numero di macchina. Questa è una situazione di *underflow*. In questo caso, lo standard IEEE prevede che $\mathrm{fl}(x)$ sia zero.

Se $p > M$, allora il numero che si deve rappresentare è piu grande del più grande numero di macchina. Questa è una situazione di *overflow*. In questo caso, lo standard IEEE prevede che $\mathrm{fl}(x)$ sia un particolare numero di macchina, da aggiungere a quelli precedentemente descritti, che viene interpretato come $+\infty$ e che si comporta nelle espressioni aritmetiche come $+\infty$.

Supponiamo ora $p \in [m, M]$. Vi sono due modi di approssimare x con un numero di macchina: il *troncamento* e l'*arrotondamento* che ora descriviamo.

A tal proposito, si considerino i numeri di macchina consecutivi x_1 e x_2 che, rispettivamente, immediamente precedono e immediatamente seguono

$$x = (b_0.b_{-1}b_{-2}b_{-3}\ldots b_{-t}b_{-t-1}b_{-t-2}\ldots)_B \cdot B^p,$$

vale a dire i numeri di macchina consecutivi x_1 e x_2 tali che $x_1 \le x < x_2$.

Essi sono

$$x_1 = (b_0.b_{-1}b_{-2}\ldots b_{-t})_B \cdot B^p$$

e

$$\begin{aligned} x_2 &= \text{numero di macchina successivo a } x_1 \\ &= (b_0.b_{-1}\ldots b_{t-1}(b_{-t}+1))_B \cdot B^p \\ &= \left(b_0 + b_{-1}B^{-1} + \cdots + b_{-t}B^{-t} + B^{-t}\right) \cdot B^p. \end{aligned}$$

Infatti si ha

$$\begin{aligned} x_1 &= \left(b_0 + b_{-1}B^{-1} + \cdots + b_{-t}B^{-t}\right) \cdot B^p \\ &\leq \left(b_0 + b_{-1}B^{-1} + \cdots + b_{-t}B^{-t} + b_{-t-1}B^{-t-1} + b_{-t-2}B^{-t-2} + \cdots\right) \cdot B^p \\ &= x \end{aligned}$$

e

$$\begin{aligned} x &= \left(b_0 + b_{-1}B^{-1} + \ldots + b_{-t}B^{-t} + b_{-t-1}B^{-t-1} + b_{-t-2}B^{-t-2} + \cdots\right) \cdot B^p \\ &< \left(b_0 + b_{-1}B^{-1} + \cdots + b_{-t}B^{-t} + (B-1)\,B^{-t-1} + (B-1)\,B^{-t-2} + \cdots\right) \cdot B^p \\ &= \left(b_0 + b_{-1}B^{-1} + \cdots + b_{-t}B^{-t} + (B-1)\,B^{-t-1}\left(1 + B^{-1} + B^{-2} + \cdots\right)\right) \cdot B^p \\ &= \left(b_0 + b_{-1}B^{-1} + \cdots + b_{-t}B^{-t} + (B-1)\,B^{-t-1}\frac{1}{1-B^{-1}}\right) \cdot B^p \\ &= \left(b_0 + b_{-1}B^{-1} + \cdots + b_{-t}B^{-t} + B^{-t}\right) \cdot B^p \\ &= x_2. \end{aligned}$$

3.1 Troncamento

Nel *troncamento* si definisce

$$\mathrm{fl}\,(x) = x_1 = (b_0.b_{-1}b_{-2}\ldots b_{-t})_B \cdot B^p,$$

cioè $\mathrm{fl}\,(x)$ è il numero di macchina che immediatamente precede x.

Esempio 8 *Nell'esempio didattico, con il troncamento si ha*

$$\mathrm{fl}\left(\frac{2}{3}\right) = \mathrm{fl}\left(0.\overline{6}\right) = \mathrm{fl}\left(6.\overline{6} \cdot 10^{-1}\right) = 6.66 \cdot 10^{-1}$$

e

$$\mathrm{fl}\,(\mathrm{e}) = \mathrm{fl}\,(2.71828\ldots) = 2.71.$$

3.2 Arrotondamento

Nell'*arrotondamento* si definisce

$$\mathrm{fl}\,(x) = \begin{cases} x_1 \ \text{ se } \ x < \frac{x_1+x_2}{2} = \ \text{punto medio dell'intervallo } [x_1, x_2] \\ \\ x_2 \ \text{ se } \ x \geq \frac{x_1+x_2}{2}, \end{cases}$$

cioè $\mathrm{fl}\,(x)$ è il numero di macchina più vicino ad x. Osservare che quando x è equidistante da x_1 e x_2, $\mathrm{fl}(x)$ è x_2.

Per conoscere se valga o no la condizione

$$x < \frac{x_1 + x_2}{2},$$

viene in aiuto la seguente Proposizione.

Proposizione 9 *Si ha*

$$x < \frac{x_1 + x_2}{2} \quad \Leftrightarrow \quad y < \frac{B}{2},$$

dove

$$y := b_{-t-1} + b_{-t-2}B^{-1} + b_{-t-3}B^{-2} + \cdots = (b_{-t-1}.b_{-t-2}b_{-t-3}\ldots)_B\,.$$

Dimostrazione. Si ha

$$\begin{aligned} x_1 + x_2 &= (b_0 + b_{-1}B^{-1} + \cdots + b_{-t}B^{-t}) \cdot B^p \\ &\quad +(b_0 + b_{-1}B^{-1} + \cdots + b_{-t}B^{-t} + B^{-t}) \cdot B^p \\ &= \left(2(b_0 + b_{-1}B^{-1} + \cdots + b_{-t}B^{-t}) + B^{-t}\right) \cdot B^p \end{aligned}$$

e quindi

$$\frac{x_1 + x_2}{2} = (b_0 + b_{-1}B^{-1} + \cdots + b_{-t}B^{-t} + \frac{B^{-t}}{2}) \cdot B^p.$$

Per cui,

$$\begin{aligned} x &= (b_0 + b_{-1}B^{-1} + \cdots + b_{-t}B^{-t} + b_{-t-1}B^{-t-1} + b_{-t-2}B^{-t-2} + \cdots) \cdot B^p \\ &< \frac{x_1 + x_2}{2} = (b_0 + b_{-1}B^{-1} + \cdots + b_{-t}B^{-t} + \frac{B^{-t}}{2}) \cdot B^p \end{aligned}$$

se e solo se

$$b_{-t-1}B^{-t-1} + b_{-t-2}B^{-t-2} + \cdots < \frac{B^{-t}}{2},$$

vale a dire, moltiplicando per B^{t+1},

$$y := b_{-t-1} + b_{-t-2}B^{-1} + b_{-t-3}B^{-2} + \cdots < \frac{B}{2}.$$

■

3.2.1 Il caso B pari

Supponiamo che B sia pari. Allora $\frac{B}{2}$ è una cifra nella base B e quindi

$$\frac{B}{2} = \left(\frac{B}{2}.00\ldots\right)_B.$$

Per cui

$$y = (b_{-t-1}.b_{-t-2}b_{-t-3}\ldots)_B < \frac{B}{2} = \left(\frac{B}{2}.00\ldots\right)_B$$

sussiste se e solo se

$$b_{-t-1} < \frac{B}{2}.$$

Nel caso di B pari, si ha quindi la formula per l'arrotondamento

$$\mathrm{fl}\,(x) = \begin{cases} x_1 = (b_0.b_{-1}b_{-2}\ldots b_{-t})_B \cdot B^p \ \text{ se } \ b_{-t-1} < \frac{B}{2} \\ \\ x_2 = (b_0.b_{-1}b_{-2}\ldots (b_{-t}+1))_B \cdot B^p \ \text{ se } \ b_{-t-1} \geq \frac{B}{2}. \end{cases}$$

Esempio 10 *Nell'esempio didattico con $B = 10$, la formula per l'arrotondamento risulta*

$$\mathrm{fl}(x) = \begin{cases} x_1 = b_0.b_{-1}b_{-2} \cdot 10^p \quad se \;\; b_{-3} < 5 \\ \\ x_2 = b_0.b_{-1}(b_{-2}+1) \cdot 10^p \quad se \;\; b_{-3} \geq 5. \end{cases}$$

e così

$$\mathrm{fl}\left(\frac{2}{3}\right) = \mathrm{fl}\left(0.\overline{6}\right) = \mathrm{fl}\left(6.\overline{6} \cdot 10^{-1}\right) = 6.67 \cdot 10^{-1}$$

e

$$\mathrm{fl}\,(\mathrm{e}) = \mathrm{fl}\,(2.71828\cdots) = 2.72.$$

Nello standard IEEE in doppia precisione con $B = 2$, la formula per l'arrotondamento risulta

$$\mathrm{fl}(x) = \begin{cases} x_1 = (1.b_{-1}b_{-2}\cdots b_{-52})_2 \cdot 2^p \quad se \;\; b_{-53} = 0 \\ \\ x_2 = (1.b_{-1}b_{-2}\cdots (b_{-52}+1))_2 \cdot 2^p \quad se \;\; b_{-53} = 1. \end{cases}$$

3.2.2 Il caso B dispari

Assumiamo ora che B sia dispari, vale a dire $B = 2k+1$ per qualche intero positivo k: si ha $k = \frac{B-1}{2}$. Allora k è una cifra nella base B e risulta

$$\frac{B}{2} = \frac{2k+1}{2} = k + \frac{1}{2} = \left(k.\overline{k}\right)_B$$

avendosi

$$\begin{aligned} 0.\overline{k} &= kB^{-1} + KB^{-2} + kB^{-3} + \cdots \\ &= kB^{-1}\left(1 + B^{-1} + B^{-2} + \cdots\right) \\ &= kB^{-1}\frac{1}{1-B^{-1}} \\ &= \frac{k}{B-1} = \frac{1}{2}. \end{aligned}$$

Per cui

$$y = (b_{-t-1}.b_{-t-2}b_{-t-3}\cdots)_B < \frac{B}{2} = \left(k.\overline{k}\right)_B$$

sussiste se e solo se la prima cifra in $b_{-t-1}, b_{-t-2}, b_{-t-3}, \ldots$ diversa da k è minore di k.

Nel caso di B dispari, si ha quindi la formula per l'arrotondamento

$$\mathrm{fl}(x) = \begin{cases} x_1 \;\; \text{se la prima cifra in } b_{-t-1}, b_{-t-2}, b_{-t-3}, \ldots \text{ diversa da } \frac{B-1}{2} \text{ è minore di } \frac{B-1}{2} \\ \\ x_2 \;\; \text{se la prima cifra in } b_{-t-1}, b_{-t-2}, b_{-t-3}, \ldots \text{ diversa da } \frac{B-1}{2} \text{ è maggiore di } \frac{B-1}{2}. \end{cases}$$

Osserviamo che, a differenza del caso B pari, al fine di arrotondare a $\mathrm{fl}(x)$ potrebbe non essere sufficiente esaminare solo la cifra b_{t-1}: se $b_{t-1} = \frac{B-1}{2}$ occorre esaminare le cifre successive.

Esempio 11 *Per $B = 3$, si ha*

$$\mathrm{fl}\,(x) = \begin{cases} x_1 & \textit{se la prima cifra in } b_{-t-1}, b_{-t-2}, b_{-t-3}, \ldots \textit{ diversa da } 1 \textit{ è } 0 \\ \\ x_2 & \textit{se la prima cifra in } b_{-t-1}, b_{-t-2}, b_{-t-3}, \ldots \textit{ diversa da } 1 \textit{ è } 2. \end{cases}$$

3.3 Esercizi

Esercizio. Si consideri l'insieme di numeri di macchina definito dai parametri

$$B = 16,\ m = -3,\ M = 3,\ t = 2.$$

Trovare $\mathrm{fl}\left(\frac{1}{13}\right)$ e $\mathrm{fl}(15005)$ sia nel caso del troncamento che in quello dell'arrotondamento.

Esercizio. Si consideri l'insieme di numeri di macchina definito dai parametri

$$B = 5,\ m = -2,\ M = 2,\ t = 2.$$

Nel caso dell'arrotondamento, trovare $\mathrm{fl}(x)$ per $x = (441.301)_5$. Fare lo stesso ora per $B = 3$ e $x = (11.1b1)_3$ con $b = 0, 1, 2$.

3.4 Errore nel troncamento

L'errore assoluto dell'approssimazione $\mathrm{fl}(x)$ di x è

$$\varepsilon_a = \mathrm{fl}\,(x) - x = x_1 - x$$

Osservare che $\varepsilon_a \leq 0$. Ricordando che numeri di macchina consecutivi nell'intervallo $\left[B^p, B^{p+1}\right)$ distano $\mathrm{eps} \cdot B^p$, risulta

$$|\varepsilon_a| = |x_1 - x| < x_2 - x_1 = \mathrm{eps} \cdot B^p,$$

essendo $|x_1 - x|$ limitato dall'ampiezza dell'intervallo $[x_1, x_2]$ e mai uguale a tale ampiezza in quanto $x < x_2$.

Si osservi che la maggiorazione $\mathrm{eps} \cdot B^p$ per $|\varepsilon_a|$ è la migliore possibile per numeri $x \in \left[B^p, B^{p+1}\right)$, dal momento che x può essere vicino quanto si vuole a x_2.

L'errore relativo dell'approssimazione $\mathrm{fl}(x)$ di x è

$$\varepsilon_r = \frac{\varepsilon_a}{x}.$$

Risulta

$$|\varepsilon_r| = \left|\frac{\varepsilon_a}{x}\right| = \frac{|\varepsilon_a|}{x} < \frac{\mathrm{eps} \cdot B^p}{B^p} = \mathrm{eps},$$

essendo $|\varepsilon_a| < \mathrm{eps} \cdot B^p$ e $x \geq B^p$.

Si ha quindi

$$|\varepsilon_r| < \mathrm{eps}.$$

Tale maggiorazione è ottimale: per ogni $c \in (0,1)$ esiste un numero reale x tale che

$$|\varepsilon_r| = (1-c)\text{eps} + O(\text{eps}^2).$$

Infatti, fissato $c \in (0,1)$, consideriamo il numero

$$x = B^p + (1-c)\text{eps}B^p.$$

Risulta

$$\begin{aligned} x_1 &= B^p \\ x_2 &= \text{"numero di macchina successivo a } x_1\text{"} = B^p + \text{eps}B^p, \end{aligned}$$

ricordando che i numeri di macchina nell'intervallo $\left[B^p, B^{p+1}\right)$ distano $\text{eps}B^p$. Per cui $\text{fl}(x) = x_1 = B^p$ e

$$\begin{aligned} |\varepsilon_r| &= \left|\frac{\text{fl}(x) - x}{x}\right| = \left|\frac{B^p - (B^p + (1-c)\,\text{eps}B^p)}{B^p + (1-c)\,\text{eps}B^p}\right| \\ &= \left|\frac{1-(1+(1-c)\,\text{eps})}{1+(1-c)\,\text{eps}}\right| = \frac{(1-c)\,\text{eps}}{1+(1-c)\,\text{eps}} = (1-c)\,\text{eps}\cdot\frac{1}{1+(1-c)\,\text{eps}} \\ &\quad \text{con } \frac{1}{1+(1-c)\,\text{eps}} = \frac{1}{1-z} = 1 + z + z^2 + \cdots = 1 + O(\text{eps}) \\ &\quad \text{dove } z = -(1-c)\,\text{eps} \\ &= (1-c)\,\text{eps}\cdot(1+O(\text{eps})) = (1-c)\,\text{eps} + O\left(\text{eps}^2\right). \end{aligned}$$

3.5 Errore nell'arrotondamento

L'errore assoluto $\varepsilon_a = \text{fl}(x) - x$ può essere, a differenza del troncamento, sia positivo che negativo e risulta

$$|\varepsilon_a| = |\text{fl}(x) - x| \leq \frac{x_2 - x_1}{2} = \frac{\text{eps}\cdot B^p}{2},$$

essendo $|\text{fl}(x) - x|$ limitato da metà dell'ampiezza dell'intervallo $[x_1, x_2]$.

Come nel caso del troncamento, la maggiorazione $\frac{\text{eps}\cdot B^p}{2}$ per $|\varepsilon_a|$ risulta essere la migliore possibile per numeri $x \in \left[B^p, B^{p+1}\right)$, dal momento che per x punto medio dell'intervallo $[x_1, x_2]$ si ha $|\text{fl}(x) - x|$ uguale a metà dell'ampiezza dell'intervallo $[x_1, x_2]$.

Per l'errore relativo $\varepsilon_r = \frac{\varepsilon_a}{x}$, risulta

$$|\varepsilon_r| = \frac{|\varepsilon_a|}{x} \leq \frac{\frac{\text{eps}\cdot B^p}{2}}{B^p} = \frac{\text{eps}}{2}.$$

essendo $|\varepsilon_a| \leq \frac{\text{eps}\cdot B^p}{2}$ e $x \geq B^p$. Si ha quindi

$$|\varepsilon_r| \leq \frac{\text{eps}}{2}.$$

Tale maggiorazione è ottimale: esiste un numero reale x tale che

$$|\varepsilon_r| = \frac{\text{eps}}{2} + O(\text{eps}^2).$$

Infatti, consideriamo il numero

$$x = B^p + \frac{\text{eps}}{2} B^p,$$

per il quale

$$\begin{aligned} x_1 &= B^p \\ x_2 &= \text{"numero di macchina successivo a } x_1\text{"} = B^p + \text{eps} B^p. \end{aligned}$$

Si ha $\text{fl}(x) = x_2 = B^p + \text{eps} B^p$ e

$$\begin{aligned} |\varepsilon_r| &= \left| \frac{\text{fl}(x) - x}{x} \right| = \left| \frac{B^p + \text{eps} B^p - \left(B^p + \frac{\text{eps}}{2} B^p \right)}{B^p + \frac{\text{eps}}{2} B^p} \right| \\ &= \left| \frac{1 + \text{eps} - \left(1 + \frac{\text{eps}}{2} \right)}{1 + \frac{\text{eps}}{2}} \right| = \frac{\frac{\text{eps}}{2}}{1 + \frac{\text{eps}}{2}} = \frac{\text{eps}}{2} \cdot \frac{1}{1 + \frac{\text{eps}}{2}} \\ &\quad \text{con } \frac{1}{1 + \frac{\text{eps}}{2}} = \frac{1}{1 - z} = 1 + z + z^2 + \cdots = 1 + O(\text{eps}) \\ &\quad \text{dove } z = -\frac{\text{eps}}{2} \\ &= \frac{\text{eps}}{2} \cdot (1 + O(\text{eps})) = \frac{\text{eps}}{2} + O\left(\text{eps}^2\right). \end{aligned}$$

La maggiorazione che si ottiene nell'arrotondamento è metà di quella del troncamento. D'altra parte, questa dimezzamanto della maggiorazione viene pagato con il dover esaminare la cifra b_{-t-1} (o eventualmente quelle successive per una base dispari).

3.6 Considerazioni finali sull'errore nell'approssimare con numeri di macchina

Concludendo, nell'approssimare i numeri reali con numeri di macchina in rappresentazione floating point utilizzando il troncamento o l'arrotondamento si può dire che:

- il massimo errore assoluto $\text{eps} B^p$ o $\frac{\text{eps}}{2} B^p$ cresce con l'ordine di grandezza;
- l'errore relativo rimane invece sempre maggiorato dall'epsilon di macchina eps (da eps per il troncamento e da $\frac{\text{eps}}{2}$ per l'arrotondamento);

Nel caso della rappresentazione fixed point si ha invece che l'errore assoluto rimane sempre maggiorato da una piccola quantità, mentre l'errore relativo cresce (decresce) con il decrescere (crescere) dell'ordine di grandezza di x.

Osserviamo, infine, che nel caso di underflow nello standard IEEE si ha un errore assoluto

$$\varepsilon_a = \mathrm{fl}\,(x) - x = 0 - x = -x$$

piccolo essendo x piccolo, ma un errore relativo

$$\varepsilon_r = \frac{\varepsilon_a}{x} = -1.$$

3.7 Esercizi

Esercizio. Sia nel caso del troncamento che in quello dell'arrotondamento, trovare l'errore relativo ε_r dell'approssimazione $\mathrm{fl}(x)$ di

$$x = B^{p+1} - c\mathrm{eps}B^p,$$

dove $c \in (0,1)$, e scriverlo nella forma

$$|\varepsilon_r| = C\mathrm{eps} + O\left(\mathrm{eps}^2\right)$$

per un'opportuna costante C.

Esercizio. Qual è l'errore relativo di $\mathrm{fl}(x)$ in caso di overflow nello standard IEEE?

4 Aritmetica di macchina

Un calcolatore esegue sui numeri di macchina le operazioni aritmetiche $+$, $-$, $\cdot$ e $/$. Sia $\circ$ una di queste operazioni.

In generale, non è detto che $a \circ b$, con a e b numeri di macchina, sia un numero di macchina.

Esempio 12 *Nell'esempio didattico, si ha che*

$$a = 30 = 3.0 \cdot 10 \quad e \quad b = 0.11 = 1.1 \cdot 10^{-1}$$

sono numeri di macchina ma

$$a + b = 30.11 = 3.011 \cdot 10$$

non è un numero di macchina. Ancora, $a = 6.1$ *è un numero di macchina ma*

$$a \cdot a = 6.1 \cdot 6.1 = 37.21 = 3.721 \cdot 10$$

non è un numero di macchina.

Pertanto, in generale, il risultato dell'operazione $\circ$ dovrà essere approssimato con un numero di macchina e quindi non sarà $a \circ b$, bensì $\mathrm{fl}\,(a \circ b)$.

Così, associata all'operazione $\circ$ vi è una corrispondente *operazione di macchina* $\widetilde{\circ}$ definita da

$$a \,\widetilde{\circ}\, b = \mathrm{fl}\,(a \circ b)\,,$$

che è quella che viene effettivamente eseguita sul calcolatore.

Ricordando le maggiorazioni per l'errore relativo quando si approssima un numero reale con un numero di macchina, si ha

$$a \,\widetilde{\circ}\, b = (a \circ b)\,(1+\varepsilon)\,,\ \text{con}\ |\varepsilon| \leq \begin{cases} \text{eps per il troncamento} \\ \\ \frac{\text{eps}}{2}\ \text{per l'arrotondamento,} \end{cases}$$

essendo

$$\varepsilon = \frac{a \,\widetilde{\circ}\, b - a \circ b}{a \circ b}$$

l'errore relativo dell'approssimazione $a \,\widetilde{\circ}\, b$ di $a \circ b$.

Un calcolatore esegue sui numeri di macchina anche le funzioni matematiche elementari: radici, funzioni trigonometriche e loro inverse, funzione esponenziale e logaritmo.

Come nel caso delle operazioni aritmetiche, ad ogni funzione matematica elementare g resta associata una corrispondente funzione di macchina $\widetilde{g}$, che è quella che viene effettivamente eseguita sul calcolatore.

Come nel caso delle operazioni aritmetiche, si garantisce che

$$\widetilde{g}\,(a) = g\,(a)\,(1+\varepsilon)\,,\ \text{con}\ \varepsilon\ \text{al più dell' ordine di grandezza di eps,}$$

per ogni numero di macchina a nel dominio di g, essendo

$$\varepsilon = \frac{\widetilde{g}(a) - g(a)}{g(a)}$$

l'errore relativo dell'approssimazione $\widetilde{g}(a)$ di $g(a)$.

Gli errori dovuti all'approssimazione di numeri reali con numeri di macchina e gli errori dovuti all'uso di operazioni di macchina o di funzioni matematiche elementari di macchina sono noti come *errori di arrotondamento* (anche nel caso in cui si usi il troncamento per approssimare con numeri di macchina).

Esercizio. Si consideri l'insieme di numeri di macchina

$$B = 2,\ m = -10,\ M = 10,\ t = 3.$$

Calcolare, con un tale insieme di numeri di macchina, $a \,\widetilde{\circ}\, b$, per $\circ = +, -, \cdot, /$, nel caso $a = 12$ e $b = 15$ come pure nel caso $a = 25$ e $b = 31$. Ricordare che nel caso in cui a e b non siano numeri di macchina, essi vanno approssimati con numeri di macchina prima di effettuare l'operazione $\circ$.

5 Propagazione degli errori di arrotondamento

Consideriamo ora un problema matematico caratterizzato da una funzione dato-risultato $f : D \to \mathbb{R}$, dove $D \subseteq \mathbb{R}^n$ è un aperto. Si assume f di classe C^2. In questo modo, si può usare per f la teoria del condizionamento vista in precedenza.

Sia $x = (x_1, \ldots, x_n) \in D$ un dato e si voglia calcolare $y = f(x)$ utilizzando un calcolatore. Supporremo $x_1, \ldots, x_n \neq 0$ and $y \neq 0$ per poter parlare di errori relativi su $x_1, \ldots, x_n \neq 0$ and y.

In generale, ancora prima di inserirlo su un calcolatore, il dato x non sarà noto in maniera esatta, ma si avrà a disposizione solo una sua approssimazione $\widehat{x} = (\widehat{x}_1, \ldots, \widehat{x}_n)$ con errori relativi sulle componenti

$$\widehat{\varepsilon}_i = \frac{\widehat{x}_i - x_i}{x_i},\ i \in \{1, \ldots, n\}\,.$$

Infatti, spesso $x_1, \ldots, x_n$ sono ottenuti attraverso delle misurazioni, le quali sono inevitabilmente affette da errori.

Quando si calcola il risultato $y = f(x)$ con un calcolatore in realtà si calcolerà

$$\widetilde{y} = \widetilde{f}\left(\mathrm{fl}\left(\widehat{x}\right)\right),$$

dove

$$\mathrm{fl}\left(\widehat{x}\right) = \left(\mathrm{fl}\left(\widehat{x}_1\right), \ldots, \mathrm{fl}\left(\widehat{x}_n\right)\right)$$

è l'approssimazione di $\widehat{x} = (\widehat{x}_1, \ldots, \widehat{x}_n)$ con numeri di macchina del calcolatore e $\widetilde{f}$ è la funzione che viene effettivamente impiegata dal calcolatore al posto di f.

Le funzioni f e $\widetilde{f}$ sono diverse in quanto nel calcolatore le operazioni aritmetiche e le funzioni matematiche elementari sono sostituite dalle corrispondenti operazioni e funzioni di macchina.

La funzione f viene calcolata utilizzando un algoritmo che a partire dal dato di input x produce l'output $f(x)$. In generale, una stessa funzione può essere calcolata con più di un algoritmo.

Nel seguito per illustrare si considereranno i seguenti due esempi di f:

- Esempio A:
$$f(x) = \sqrt{x+1} - \sqrt{x},\ x > 0.$$
- Esempio B:
$$f(x) = x_1 x_2 + x_1,\ x \in \mathbb{R}^2.$$

Nell'Esempio A vi sono i seguenti due algoritmi per calcolare i valori di f basati sulle due diverse espressioni per $f(x)$:

$$f(x) = \sqrt{x+1} - \sqrt{x} = \frac{1}{\sqrt{x+1} + \sqrt{x}}.$$

ALGORITMO 1
$a = x + 1$
$b = \sqrt{a}$
$c = \sqrt{x}$
$y = b - c$

ALGORITMO 2
$a = x + 1$
$b = \sqrt{a}$
$c = \sqrt{x}$
$d = b + c$
$y = 1/d$

Nell'Esempio B vi sono i seguenti due algoritmi per calcolare i valori di f basati sulle due diverse espressioni per $f(x)$:

$$f(x) = x_1 x_2 + x_1 = x_1 (x_2 + 1).$$

ALGORITMO 1
$a = x_1 \cdot x_2$
$y = a + x_1$

ALGORITMO 2
$a = x_2 + 1$
$y = x_1 \cdot a$

Come appare dai due esempi, un algoritmo è costituito da una sequenza di istruzioni, ognuna delle quali è l'esecuzione di un'operazione aritmetica o il calcolo di una funzione matematica elementare (realizzate nel calcolatore con operazioni e funzioni di macchina).

Osserviamo anche che il dato di input dell'algoritmo è indicato con la variabile $x = (x_1, \ldots, x_n)$, ma il il reale dato di input su cui opera l'algoritmo è $\mathrm{fl}(\widehat{x}) = (\mathrm{fl}(\widehat{x}_1), \ldots, \mathrm{fl}(\widehat{x}_n))$.

Esercizio. Scrivere due algoritmi per il calcolo di

$$f(x) = x_1^2 - x_2^2, \ x \in \mathbb{R}^2,$$

e tre algoritmi per il calcolo di

$$f(x) = x_1^4 - x_2^4, \ x \in \mathbb{R}^2.$$

La funzione $\widetilde{f}$ viene a dipendere dal particolare algoritmo usato per calcolare i valori della funzione f.

Esempio 13 *Usando l'esempio didattico di numeri di macchina (ora con $m = -2$) e la funzione dell'Esempio B con $\mathrm{fl}(\widehat{x}) = (7.47, -0.99)$, l'Algoritmo 1 fornisce*

$$a = 7.47 \ \widetilde{\cdot} \ (-0.99) = -\mathrm{fl}(7.3953) = -7.40$$
$$y = -7.40 \ \widetilde{+} \ 7.47 = 0.07,$$

mentre l'Algoritmo 2 fornisce

$$a = -0.99 \ \widetilde{+} \ 1 = 0.01$$
$$y = 7.47 \ \widetilde{\cdot} \ 0.01 = 0.0747.$$

Le funzioni $\widetilde{f}$ corrispondenti a due espressioni diverse della f sono diverse, perché le operazioni aritmetiche di macchina e le funzioni matematiche elementari di macchina non soddisfano quelle proprietà delle operazioni aritmetiche e delle funzioni matematiche elementari che permettono di dire che le due espressioni forniscono gli stessi valori. Nell'esempio B, le due espressioni sono uguali perché la moltiplicazione è distributiva rispetto all'addizione. Invece, come l'esempio appena visto sopra dimostra, la moltiplicazione di macchina non è distributiva rispetto all'addizione di macchina.

5.1 Errori inerente, di macchina e algoritmico

Il nostro obiettivo è ora quello di analizzare l'errore sul risultato

$$\varepsilon_y := \frac{\widetilde{y} - y}{y} = \frac{\widetilde{f}\left(\mathrm{fl}\left(\widehat{x}\right)\right) - f(x)}{f(x)}.$$

A tal fine introduciamo le quantità

$$\widehat{y} := f\left(\widehat{x}\right) \quad \text{e} \quad \overline{y} := f\left(\mathrm{fl}\left(\widehat{x}\right)\right)$$

che sono intermedie tra $y = f(x)$ e $\widetilde{y} = \widetilde{f}(\mathrm{fl}(\widehat{x}))$: si ha

$$y = f(x) \ \rightarrow \ \widehat{y} = f\left(\widehat{x}\right) \ \rightarrow \ \overline{y} = f\left(\mathrm{fl}\left(\widehat{x}\right)\right) \ \rightarrow \ \widetilde{y} = \widetilde{f}(\mathrm{fl}(\widehat{x})).$$

Associati alle quantità $\widehat{y}$, $\overline{y}$ e $\widetilde{y}$, consideriamo i seguenti tre errori:

- l' *errore inerente*
$$\varepsilon_{\text{in}} := \frac{\widehat{y} - y}{y} = \frac{f(\widehat{x}) - f(x)}{f(x)}$$
che è l'errore che si ottiene sostituendo in $f\left(x\right)$ il dato x con la sua approssimazione $\widehat{x}$; questo errore è indipendente dal calcolatore e dall'algoritmo per calcolare i valori di f (non essendo coinvolte né le approssimazioni $\mathrm{fl}(\cdot)$ né la funzione $\widetilde{f}$);

- l' *errore di macchina*
$$\varepsilon_{\text{mac}} := \frac{\overline{y} - \widehat{y}}{\widehat{y}} = \frac{f(\mathrm{fl}(\widehat{x})) - f(\widehat{x})}{f(\widehat{x})}$$
che è l'errore che si ottiene sostituendo in $f\left(\widehat{x}\right)$ il dato approssimato $\widehat{x}$ con la sua approssimazione di macchina $\mathrm{fl}\left(\widehat{x}\right)$; questo errore dipende dal calcolatore ma è indipendente dall'algoritmo per calcolare i valori di f (essendo coinvolte le approssimazioni $\mathrm{fl}(\cdot)$ ma non la funzione $\widetilde{f}$).

- l' *errore algoritmico*

$$\varepsilon_{\text{alg}} := \frac{\widetilde{y} - \overline{y}}{\overline{y}} = \frac{\widetilde{f}(\text{fl}(\widehat{x})) - f(\text{fl}(\widehat{x}))}{f(\text{fl}(\widehat{x}))}$$

  che è l'errore che si ottiene sostituendo in $f(\text{fl}(\widehat{x}))$ la funzione f con la sua approssimazione di macchina $\widetilde{f}$; questo errore dipende dal calcolatore e dall' algoritmo per calcolare i valori di f (essendo coinvolta $\widetilde{f}$ che dipende dal calcolatore e dall'algoritmo).

Vogliamo ora mettere in relazione ε_y con ε_{in}, ε_{mac} e ε_{alg}.
Usando le quantità intermedie $\widehat{y}$ e $\overline{y}$, scriviamo l'errore assoluto $\widetilde{y} - y$ come

$$\widetilde{y} - y = \widetilde{y} - \overline{y} + \overline{y} - \widehat{y} + \widehat{y} - y$$

e quindi l'errore relativo ε_y è dato da

$$\begin{aligned}
\varepsilon_y &= \frac{\widetilde{y} - y}{y} = \frac{\widetilde{y} - \overline{y} + \overline{y} - \widehat{y} + \widehat{y} - y}{y} \\
&= \frac{\widetilde{y} - \overline{y}}{y} + \frac{\overline{y} - \widehat{y}}{y} + \frac{\widehat{y} - y}{y} \\
&= \frac{\widetilde{y} - \overline{y}}{\overline{y}} \cdot \frac{\overline{y}}{\widehat{y}} \cdot \frac{\widehat{y}}{y} + \frac{\overline{y} - \widehat{y}}{\widehat{y}} \cdot \frac{\widehat{y}}{y} + \frac{\widehat{y} - y}{y} \\
&= \varepsilon_{\text{alg}} \cdot \frac{\overline{y}}{\widehat{y}} \cdot \frac{\widehat{y}}{y} + \varepsilon_{\text{mac}} \cdot \frac{\widehat{y}}{y} + \varepsilon_{\text{in}}.
\end{aligned}$$

Avendosi

$$\frac{\widehat{y}}{y} = 1 + \varepsilon_{\text{in}} \quad \text{e} \quad \frac{\overline{y}}{\widehat{y}} = 1 + \varepsilon_{\text{mac}},$$

si ottiene

$$\begin{aligned}
\varepsilon_y &= \varepsilon_{\text{alg}}(1 + \varepsilon_{\text{mac}})(1 + \varepsilon_{\text{in}}) + \varepsilon_{\text{mac}}(1 + \varepsilon_{\text{in}}) + \varepsilon_{\text{in}} \\
&= \varepsilon_{\text{alg}} + \varepsilon_{\text{alg}}\varepsilon_{\text{in}} + \varepsilon_{\text{alg}}\varepsilon_{\text{mach}} + \varepsilon_{\text{alg}}\varepsilon_{\text{mac}}\varepsilon_{\text{in}} + \varepsilon_{\text{mac}} + \varepsilon_{\text{mac}}\varepsilon_{\text{in}} + \varepsilon_{\text{in}}.
\end{aligned}$$

Trascurando i monomi di grado ≥ 2 negli errori si ha

$$\varepsilon_y \doteq \varepsilon_{\text{in}} + \varepsilon_{\text{mac}} + \varepsilon_{\text{alg}}, \tag{4}$$

dove $\doteq$ significa uguale a meno di un termine il cui modulo è minore o oguale di una costante volte $\max\{|\varepsilon_{\text{in}}|, |\varepsilon_{\text{mac}}|, |\varepsilon_{\text{alg}}|\}^2$ per $\max\{|\varepsilon_{\text{in}}|, |\varepsilon_{\text{mac}}|, |\varepsilon_{\text{alg}}|\}$ sufficientemente piccolo, vale a dire uguale a meno di un termine che è

$$O\left(\max\{|\varepsilon_{\text{in}}|, |\varepsilon_{\text{mac}}|, |\varepsilon_{\text{alg}}|\}^2\right), \quad \text{per} \max\{|\varepsilon_{\text{in}}|, |\varepsilon_{\text{mac}}|, |\varepsilon_{\text{alg}}|\} \to 0.$$

In effetti in (4) si ha

$$\varepsilon_y = \varepsilon_{\text{in}} + \varepsilon_{\text{mac}} + \varepsilon_{\text{alg}} + \text{termine}$$

dove

$$\text{termine} = \varepsilon_{\text{alg}}\varepsilon_{\text{mac}} + \varepsilon_{\text{alg}}\varepsilon_{\text{in}} + \varepsilon_{\text{mac}}\varepsilon_{\text{in}} + \varepsilon_{\text{alg}}\varepsilon_{\text{mac}}\varepsilon_{\text{in}}$$

e, posto

$$\varepsilon = \max\{|\varepsilon_{\text{in}}|, |\varepsilon_{\text{mac}}|, |\varepsilon_{\text{alg}}|\}$$

e assumendo $\varepsilon \leq 1$, risulta

$$\begin{aligned}
&|\text{termine}| \\
&= |\varepsilon_{\text{alg}}\varepsilon_{\text{mac}} + \varepsilon_{\text{alg}}\varepsilon_{\text{in}} + \varepsilon_{\text{mac}}\varepsilon_{\text{in}} + \varepsilon_{\text{alg}}\varepsilon_{\text{mac}}\varepsilon_{\text{in}}| \\
&\leq |\varepsilon_{\text{alg}}\varepsilon_{\text{mac}}| + |\varepsilon_{\text{alg}}\varepsilon_{\text{in}}| + |\varepsilon_{\text{mac}}\varepsilon_{\text{in}}| + |\varepsilon_{\text{alg}}\varepsilon_{\text{mac}}\varepsilon_{\text{in}}| \\
&= |\varepsilon_{\text{alg}}|\,|\varepsilon_{\text{mac}}| + |\varepsilon_{\text{alg}}|\,|\varepsilon_{\text{in}}| + |\varepsilon_{\text{mac}}|\,|\varepsilon_{\text{in}}| + |\varepsilon_{\text{alg}}|\,|\varepsilon_{\text{mac}}|\,|\varepsilon_{\text{in}}| \\
&\leq \varepsilon^2 + \varepsilon^2 + \varepsilon^2 + \underbrace{\varepsilon^3}_{=\varepsilon^2\cdot\varepsilon\leq\varepsilon^2\cdot 1 \text{ essendo } \varepsilon\leq 1} \\
&\leq \varepsilon^2 + \varepsilon^2 + \varepsilon^2 + \varepsilon^2 \\
&= 4\varepsilon^2.
\end{aligned}$$

Si considerano solo i monomi di grado 1 negli errori in quanto si è interessati solo all'ordine di grandezza di ε_y e questo è determinato dai soli monomi di grado 1 in quanto gli errori ε_{in}, ε_{mac} e ε_{alg} sono quantità piccole.

Lo studio dell'errore ε_y sul risultato è quindi ridotto allo studio dei tre errori inerente, di macchina e algoritmico.

5.2 Analisi dell'errore inerente

L'errore inerente è l'errore di $f(x)$ quando il dato x viene perturbato in $\widehat{x}$.

Per quanto visto nella teoria del condizionamento per funzioni dato-risultato $D \subseteq \mathbb{R}^n \to \mathbb{R}$, D aperto, di classe C^2 (come lo è la funzione f) risulta

$$\varepsilon_{\text{in}} \doteq \sum_{i=1}^{n} K_i(x)\,\widehat{\varepsilon}_i,$$

dove i $K_i(x)$, $i \in \{1, \ldots, n\}$, sono gli indici di condizionamento di f sul dato x e $\doteq$ significa uguale a meno di un termine il cui modulo è minore o uguale di una costante volte $\left(\max_{i\in\{1,\ldots,n\}} |\widehat{\varepsilon}_i|\right)^2$ per $\max_{i\in\{1,\ldots,n\}} |\widehat{\varepsilon}_i|$ sufficientemente piccolo, vale a dire uguale a meno di un termine

$$O\left(\left(\max_{i\in\{1,\ldots,n\}} |\widehat{\varepsilon}_i|\right)^2\right) \quad \text{per} \max_{i\in\{1,\ldots,n\}} |\widehat{\varepsilon}_i| \to 0.$$

Nel seguito, questa notazione $\doteq$, coerentemente con l'uso fatto fino ad ora, sempre significherà uguale a meno di un termine il cui modulo è minore o uguale

di una costante volte ε^2 per ε sufficientemente piccolo, dove ε il massimo modulo degli errori coinvolti; in altre parole uguale a meno di un termine

$$O\left(\varepsilon^2\right) \quad \text{per } \varepsilon \to 0.$$

Dal momento che l'errore inerente può essere analizzato attraverso la teoria del condizionamento, possiamo dire quanto segue.

Se f è ben condizionata sul dato x, allora ε_{in} ha ordine di grandezza non superiore al massimo ordine di grandezza degli $\widehat{\varepsilon}_i$, vale a dire non si ha $\varepsilon_{\text{in}} \gg \max\limits_{i\in\{1,\ldots,n\}} |\widehat{\varepsilon}_i|$.

Se, invece, f è mal condizionata sul dato x, allora ε_{in} ha "in generale" ordine di grandezza superiore al massimo ordine di grandezza degli $\widehat{\varepsilon}_i$, vale a dire si ha $\varepsilon_{\text{in}} \gg \max\limits_{i\in\{1,\ldots,n\}} |\widehat{\varepsilon}_i|$ ("in generale" vuol dire che lo è per qualche n-upla di errori $\widehat{\varepsilon}_i$, ma non è detto che ciò accada per la n-upla di errori $\widehat{\varepsilon}_i$ che effettivamente si hanno).

Nell' Esempio A, dove

$$f(x) = \sqrt{x+1} - \sqrt{x},\ x > 0,$$

si ha

$$\begin{aligned} K(x) &= f'(x) \cdot \frac{x}{f(x)} \cdot = \left(\frac{1}{2\sqrt{x+1}} - \frac{1}{2\sqrt{x}}\right) \cdot \frac{x}{\sqrt{x+1} - \sqrt{x}} \cdot \\ &= \frac{\sqrt{x} - \sqrt{x+1}}{2\sqrt{x+1}\sqrt{x}} \cdot \frac{x}{\sqrt{x+1} - \sqrt{x}} \cdot \\ &= -\frac{\sqrt{x}}{2\sqrt{x+1}} = -\frac{1}{2}\sqrt{\frac{x}{x+1}} \end{aligned}$$

e quindi

$$|K(x)| \leq \frac{1}{2}.$$

Per cui, f è ben condizionata su ogni dato x e quindi ε_{in} ha ordine di grandezza non superiore a quello dell'errore $\widehat{\varepsilon}$ di $\widehat{x}$.

In particolare, si ha

$$\varepsilon_{\text{in}} \doteq -\frac{1}{2}\sqrt{\frac{x}{x+1}}\widehat{\varepsilon}.$$

Nell'Esempio B, dove

$$f(x) = x_1 x_2 + x_1,\ x \in \mathbb{R}^2,$$

si ha

$$\begin{aligned} K_1(x) &= \frac{\partial f}{\partial x_1}(x) \cdot \frac{x_1}{f(x)} = (x_2+1) \cdot \frac{x_1}{x_1(x_2+1)} \cdot = 1, \\ K_2(x) &= \frac{\partial f}{\partial x_2}(x) \cdot \frac{x_2}{f(x)} = x_1 \cdot \frac{x_2}{x_1(x_2+1)} = \frac{x_2}{x_2+1} \end{aligned}$$

e quindi f è ben condizionata se e solo se x_2 non è vicino a -1.

Si ha

$$\varepsilon_{\text{in}} \doteq \widehat{\varepsilon}_1 + \frac{x_2}{x_2+1}\widehat{\varepsilon}_2$$

e ε_{in} ha ordine di grandezza non superiore al massimo ordine di grandezza di $\widehat{\varepsilon}_1$ e $\widehat{\varepsilon}_2$, per x_2 non vicino a -1.

Esercizio. Quando x_2 è vicino a -1 si può concludere che ε_{in} ha ordine di grandezza superiore a quello di $\widehat{\varepsilon}_1$ e $\widehat{\varepsilon}_2$? Si assuma $\widehat{\varepsilon}_1$ e $\widehat{\varepsilon}_2$ non nulli e dello stesso ordine di grandezza.

5.3 Analisi dell'errore di macchina

L'errore di macchina è l'errore di $f(\widehat{x})$ quando $\widehat{x}$ viene perturbato in $\text{fl}(\widehat{x})$.

Avendosi

$$\text{fl}(\widehat{x}_i) = \widehat{x}_i(1+\delta_i),\ i \in \{1,\dots n\},$$

dove δ_i è al più dell'ordine di grandezza di eps (si ha $|\delta_i| \leq$ eps per il troncamento e $|\delta_i| \leq \frac{\text{eps}}{2}$ per l'arrotondamento), si ottiene, di nuovo dalla teoria del condizionamento

$$\varepsilon_{\text{mac}} \doteq \sum_{i=1}^{n} K_i(\widehat{x})\,\delta_i.$$

Si osservi che la precedente formula contiene gli indici di condizionamento $K_i(\widehat{x})$ relativi a $\widehat{x}$, non gli indici di condizionamento $K_i(x)$ relativi al dato originale x. Tuttavia, si ha anche

$$\varepsilon_{\text{mac}} \doteq \sum_{i=1}^{n} K_i(x)\,\delta_i$$

come mostriamo ora.

Per $i \in \{1,\dots,n\}$, con una sviluppo di Taylor al grado zero risulta

$$K_i(\widehat{x}) = K_i(x) + O\left(\|\widehat{x}-x\|_\infty\right).$$

Un tale sviluppo esiste in quanto K_i ha derivate prime continue, cosa che segue dall'essere K definito in termini di f e delle sue derivate prime, con f avente derivate prime e seconde continue.

Quindi

$$\begin{aligned}\varepsilon_{\text{mac}} &\doteq \sum_{i=1}^{n} K_i(\widehat{x})\,\delta_i = \sum_{i=1}^{n}\left(K_i(x) + O\left(\|\widehat{x}-x\|_\infty\right)\right)\delta_i \\ &= \sum_{i=1}^{n} K_i(x)\,\delta_i + \sum_{i=1}^{n} O\left(\|\widehat{x}-x\|_\infty\right)\cdot\delta_i. \qquad (5)\end{aligned}$$

con

$$\begin{aligned}\left\|\widehat{x}-x\right\|_{\infty} &= \max_{i\in\{1,\ldots,n\}}\left|\widehat{x}_i-x_i\right| = \max_{i\in\{1,\ldots,n\}}\left|\widehat{\varepsilon}_i x_i\right| \\ &\leq \max_{i\in\{1,\ldots,n\}}\left|\widehat{\varepsilon}_i\right| \max_{i\in\{1,\ldots,n\}}\left|x_i\right| = \max_{i\in\{1,\ldots,n\}}\left|\widehat{\varepsilon}_i\right| \left\|x\right\|_{\infty}.\end{aligned}$$

Per cui il secondo termine in (5) ha modulo minore o uguale di una costante volte

$$\max\left\{\max_{i\in\{1,\ldots,n\}}\left|\widehat{\varepsilon}_i\right|, \max_{i\in\{1,\ldots,n\}}\left|\delta_i\right|\right\}^2$$

per

$$\max_{i\in\{1,\ldots,n\}}\left|\widehat{\varepsilon}_i\right|$$

sufficientemente piccolo.

Si conclude che

$$\varepsilon_{\text{mac}} \doteq \sum_{i=1}^{n} K_i\left(x\right)\delta_i + \sum_{i=1}^{n} O\left(\left\|\widehat{x}-x\right\|_{\infty}\right)\cdot\delta_i \doteq \sum_{i=1}^{n} K_i\left(x\right)\delta_i.$$

Come l'errore inerente, anche l'errore di macchina può essere analizzato tramite il condizionamento di f.

Se f è ben condizionata sul dato x, allora l'errore ε_{mac} ha ordine di grandezza non superiore al massimo ordine di grandezza degli errori δ_i e quindi ha ordine di grandezza non superiore a quello di eps.

Se, invece, f è mal condizionata sul dato x, allora l'errore ε_{mac} ha "in generale" ordine di grandezza superiore al massimo ordine di grandezza degli errori δ_i e quindi ha "in generale" ordine di grandezza superiore a quello di eps.

Nell'Esempio A, ε_{mac} ha ordine di grandezza non superiore a quello di eps per ogni dato.

Nell'Esempio B, ε_{mac} ha ordine di grandezza non superiore a quello di eps se il dato x ha x_2 non vicino a -1. Quando x_2 è vicino a -1, $\varepsilon_{\text{mach}}$ ha ordine di grandezza superiore a quello di eps se δ_1 e δ_2 sono non nulli e dello stesso ordine di grandezza di eps.

5.4 Analisi dell'errore algoritmico

L'errore algoritmico ε_{alg} è dovuto alla sostituzione di f con $\widetilde{f}$ nel calcolo di $f(\text{fl}(\widehat{x}))$ e dipende dal particolare algoritmo usato per calcolare i valori di f.

Si supponga che l'algoritmo consista di m istruzioni, ognuna delle quali è un'operazione aritmetica o il calcolo di una funzione matematica elementare.

5.4.1 Formule di propagazione degli errori nelle singole istruzioni

Si assuma che la j-esima istruzione, $j \in \{1, \ldots, m\}$, consista in un'operazione aritmetica

$$c = a \circ b,$$

dove $\circ$ è $+$, $-$, $\cdot$ o $/$.

L'operazione $\circ$ non è eseguita sui valori a e b ottenibili dai numeri di macchina di input $\mathrm{fl}(\widehat{x}_1), \ldots, \mathrm{fl}(\widehat{x}_n)$ con operazioni aritmetiche esatte e funzioni matematiche elementari esatte eseguite nelle istruzioni precedenti alla j-esima, ma su dalle approssimazioni $\widetilde{a}$ e $\widetilde{b}$ ottenute da $\mathrm{fl}(\widehat{x}_1), \ldots, \mathrm{fl}(\widehat{x}_n)$ con operazioni aritmetiche di macchina e funzioni matematiche elementari di macchina. Inoltre, l'operazione $\circ$ è approssimata dall'operazione di macchina $\widetilde{\circ}$.

Per cui, il risultato dell'operazione è

$$\widetilde{c} = \widetilde{a} \,\widetilde{\circ}\, \widetilde{b}$$

e non

$$c = a \circ b.$$

Siano

$$\beta_a = \frac{\widetilde{a} - a}{a} \quad \text{e} \quad \beta_b = \frac{\widetilde{b} - b}{b}$$

gli errori delle approssimazioni $\widetilde{a}$ di a e $\widetilde{b}$ di b. Vogliamo trovare come l'errore

$$\beta_c = \frac{\widetilde{c} - c}{c}$$

dell'approssimazione $\widetilde{c}$ di c è in relazione con β_a e β_b.

Si ha

$$\widetilde{c} = \widetilde{a} \,\widetilde{\circ}\, \widetilde{b} = \left(\widetilde{a} \circ \widetilde{b}\right)(1 + \gamma_j),$$

dove γ_j è al più dell'ordine di grandezza di eps (ricordare come sono definite le operazioni di macchina). Si consideri ora la funzione

$$h(\alpha, \beta) = \alpha \circ \beta, \ (\alpha, \beta) \in \mathbb{R}^2.$$

L'errore

$$\lambda_j = \frac{\widetilde{a} \circ \widetilde{b} - a \circ b}{a \circ b}$$

di $\widetilde{a} \circ \widetilde{b} = h(\widetilde{a}, \widetilde{b})$ rispetto ad $a \circ b = h(a, b)$ è l'errore nei valori della funzione h quando il dato (a, b) viene perturbato in $(\widetilde{a}, \widetilde{b})$. Pertanto, dalla teoria del condizionamento si ottiene

$$\lambda_j \doteq K_1(a, b)\, \beta_a + K_2(a, b)\, \beta_b,$$

dove $K_1(a,b) := K(h,(a,b))$ e $K_2(a,b) := K(h,(a,h))$ sono gli indici di condizionamento della funzione h sul dato (a,b). Essendo

$$\widetilde{a} \circ \widetilde{b} = h\left(\widetilde{a}, \widetilde{b}\right) = h(a,b)(1+\lambda_j) = (a \circ b)(1+\lambda_j),$$

si ha

$$\begin{aligned} \widetilde{c} &= \widetilde{a} \mathbin{\widetilde{\circ}} \widetilde{b} = \left(\widetilde{a} \circ \widetilde{b}\right)(1+\gamma_j) \\ &= (a \circ b)(1+\lambda_j)(1+\gamma_j) \\ &= c(1+\lambda_j+\gamma_j+\lambda_j\gamma_j) \end{aligned}$$

e quindi

$$\beta_c = \frac{\widetilde{c}-c}{c} = \lambda_j + \gamma_j + \lambda_j\gamma_j.$$

Trascurando il monomio di grado 2 negli errori si ottiene

$$\beta_c \doteq \lambda_j + \gamma_j \doteq K_1(a,b)\,\beta_a + K_2(a,b)\,\beta_b + \gamma_j. \tag{6}$$

La formula (6) dice come gli errori β_a e β_b sugli operandi a e b dell'operazione $a \circ b$ e l'errore γ_j dell'operazione si propagano sul risultato c.

Si assuma ora che la $j-$esima istruzione

$$c = g(a)$$

consista nel calcolo di una funzione matematica elementare g. Il risultato dell'operazione è

$$\widetilde{c} = \widetilde{g}(\widetilde{a})$$

e non

$$c = g(a),$$

dove $\widetilde{g}$ è la funzione di macchina che approssima g e $\widetilde{a}$ è ottenuta da $\mathrm{fl}(\widehat{x}_1), \ldots,$ $\mathrm{fl}(\widehat{x}_n)$ con operazioni aritmetiche di macchina e funzioni matematiche elementari di macchina eseguite nelle istruzioni precedenti la $j-$esima. Sia

$$\beta_a = \frac{\widetilde{a}-a}{a}$$

l'errore delle approssimazione $\widetilde{a}$ di a. Vogliamo trovare come l'errore

$$\beta_c = \frac{\widetilde{c}-c}{c}$$

dell'approssimazione $\widetilde{c}$ di c è in relazione con β_a.

Si ha

$$\widetilde{c} = \widetilde{g}(\widetilde{a}) = g(\widetilde{a})(1+\gamma_j),$$

dove γ_j è al più dell'ordine di eps (questo viene garantito nelle versioni di macchina delle funzioni matematiche elementari), e

$$g(\widetilde{a}) = g(a)(1+\lambda_j)$$

con

$$\lambda_j = \frac{g(\widetilde{a}) - g(a)}{g(a)} \doteq K(a)\,\beta_a,$$

dalla teoria del condizionamento, dove $K(a) := K(g,a)$ è l'indice di condizionamento di g sul dato a.

Per cui

$$\begin{aligned} \widetilde{c} &= g(\widetilde{a})(1+\gamma_j) \\ &= g(a)(1+\lambda_j)(1+\gamma_j) \\ &= c(1+\lambda_j+\gamma_j+\lambda_j\gamma_j) \end{aligned}$$

e quindi

$$\beta_c \doteq \lambda_j + \gamma_j \doteq K(a)\,\beta_a + \gamma_j. \tag{7}$$

La formula (7) dice come l'errore β_a sull'argomento a in $c = g(a)$ e l'errore γ_j nel calcolo di g si propagano sul risultato c.

5.4.2 Determinazione dell'errore algoritimico

Notiamo che l'errore algoritmico ε_{alg} è l'errore β_c quando l'istruzione

$$c = a \circ b \quad \text{oppure} \quad c = g(a)$$

è l'ultima dell'algoritmo.

Infatti, nel caso dell'ultima istruzione si ha

$$c = f(\text{fl}(\widehat{x})) = \overline{y}$$

e

$$\widetilde{c} = \widetilde{f}(\text{fl}(\widehat{x})) = \widetilde{y}.$$

Quindi

$$\varepsilon_{\text{alg}} = \frac{\widetilde{y} - \overline{y}}{\overline{y}} = \frac{\widetilde{c} - c}{c} = \beta_c.$$

Al fine di ottenere ε_{alg} occorre quindi applicare le precedenti formule di propagazione degli errori a tutte le istruzioni dell'algoritmo, procedendo in ordine dalla prima all'ultima istruzione. In particolare si applica:

- la formula
$$\beta_c \doteq K_1(a,b)\,\beta_a + K_2(a,b)\,\beta_b + \gamma_j$$
se l'istruzione è del tipo $c = a \circ b$ con $\circ$ operazione aritmetica.

- la formula

$$\beta_c \doteq K(a)\,\beta_a + \gamma_j$$

se l'istruzione è del tipo $c = g(a)$ con g funzione matematica elementare.

Nell'applicare le formule di propagazione bisogna procedere nell'ordine dalla prima all'ultima istruzione in quanto β_a e β_b sono quantità β_c di precedenti istruzioni oppure quantità $\beta_{x_1}, \ldots, \beta_{x_n}$, dove $x_1, \ldots, x_n$ sono le variabili dell'algoritmo che contengono i numeri di macchina di input $\mathrm{fl}(\widehat{x}_1), \ldots, \mathrm{fl}(\widehat{x}_n)$.

Si osservi che per tali variabili $x_1, \ldots, x_n$, si ha

$$\beta_{x_1} = \cdots = \beta_{x_n} = 0.$$

Infatti, per una variabile c dell'algoritmo, l'errore β_c sorge solo perché c è ottenuta in una istruzione $c = a \circ b$ come risultato di una operazione di macchina o in una istruzione $c = g(a)$ come risultato di una funzione matematica elementare di macchina. Invece, le variabili $x_1, \ldots, x_n$ non sono ottenute in nessuna istruzione come risultato, esse sono le variabili iniziali.

Applichiamo le formule di propagazione all'Algoritmo 1 dell'Esempio A.

Si ha

$$1 \quad a = x+1 \quad \beta_a \doteq \tfrac{x}{x+1} \cdot \underbrace{\beta_x}_{=0 \text{ essendo } x \text{ un dato iniziale}} + \tfrac{1}{x+1} \cdot \underbrace{\beta_1}_{=0 \text{ essendo 1 un numero di macchina}} + \gamma_1 = \gamma_1$$

$$2 \quad b = \sqrt{a} \qquad \beta_b \doteq \tfrac{1}{2}\beta_a + \gamma_2 \doteq \tfrac{1}{2}\gamma_1 + \gamma_2$$

$$3 \quad c = \sqrt{x} \qquad \beta_c \doteq \tfrac{1}{2} \cdot \underbrace{\beta_x}_{=0} + \gamma_3 = \gamma_3$$

$$4 \quad y = b - c \quad \varepsilon_{\text{alg}} = \beta_y \doteq \tfrac{b}{b-c}\beta_b - \tfrac{c}{b-c}\beta_c + \gamma_4 \doteq \tfrac{b}{b-c}\left(\tfrac{1}{2}\gamma_1 + \gamma_2\right) - \tfrac{c}{b-c}\gamma_3 + \gamma_4$$

$$\doteq \tfrac{1}{2} \cdot \tfrac{b}{b-c}\gamma_1 + \tfrac{b}{b-c}\gamma_2 - \tfrac{c}{b-c}\gamma_3 + \gamma_4.$$

Essendo

$$b = \sqrt{\mathrm{fl}\,(\widehat{x}) + 1} \text{ e } \; c = \sqrt{\mathrm{fl}\,(\widehat{x})},$$

si ottiene

$$\begin{aligned} \varepsilon_{\text{alg}} \quad \doteq \quad & \frac{1}{2} \cdot \frac{\sqrt{\mathrm{fl}\,(\widehat{x}) + 1}}{\sqrt{\mathrm{fl}\,(\widehat{x}) + 1} - \sqrt{\mathrm{fl}\,(\widehat{x})}}\gamma_1 + \frac{\sqrt{\mathrm{fl}\,(\widehat{x}) + 1}}{\sqrt{\mathrm{fl}\,(\widehat{x}) + 1} - \sqrt{\mathrm{fl}\,(\widehat{x})}}\gamma_2 \\ & - \frac{\sqrt{\mathrm{fl}\,(\widehat{x})}}{\sqrt{\mathrm{fl}\,(\widehat{x}) + 1} - \sqrt{\mathrm{fl}\,(\widehat{x})}}\gamma_3 + \gamma_4. \end{aligned}$$

5.4.3 Indici di stabilità e stabilità di un algoritmo

Su ogni algoritmo si ottiene per ε_{alg} un'espressione del tipo

$$\varepsilon_{\text{alg}} \doteq \sum_{j=1}^{m} M_j\left(\text{fl}\left(\widehat{x}\right)\right)\gamma_j,$$

dove, per $j \in \{1,\ldots,m\}$, $M_j : D \to \mathbb{R}$, essendo D il dominio della funzione dato-risultato f. In realtà le funzioni $M_j(x)$ sono definite per un dato $x \in D$ tale che $x_1,\ldots,x_n \neq 0$ e $y = f(x) \neq 0$ come stabilito all'inizio dello studio della propagazione degli errori di arrotondamento.

Si ha anche

$$\varepsilon_{\text{alg}} \doteq \sum_{j=1}^{m} M_j\left(x\right)\gamma_j.$$

come ora mostriamo.

Nella nostra analisi dell'errore algoritmico, si assume che la f sia una composizione di operazioni aritmetiche e funzioni matematiche elementari. Per cui, ne viene che f è di classe C^∞. Inoltre, anche gli M_j, essendo composizioni di indici di condizionamento di operazioni aritmetiche e funzioni matematiche elementari, sono di classe C^∞. Ora, per $j \in \{1,\ldots,m\}$, si ha

$$M_j\left(\text{fl}\left(\widehat{x}\right)\right) = M_j\left(x\right) + O\left(\left\|\text{fl}\left(\widehat{x}\right) - x\right\|_\infty\right)$$

con uno sviluppo di Taylor di grado zero (per l'esistenza un tale sviluppo basta assumere che M_j sia di classe C^1). Quindi

$$\begin{aligned}\varepsilon_{\text{alg}} &\doteq \sum_{j=1}^{m} M_j\left(\text{fl}\left(\widehat{x}\right)\right)\gamma_j = \sum_{j=1}^{m}\left(M_j\left(x\right) + O\left(\left\|\text{fl}\left(\widehat{x}\right) - x\right\|_\infty\right)\right)\gamma_j \\ &= \sum_{j=1}^{m} M_j\left(x\right)\gamma_j + \sum_{j=1}^{m} O\left(\left\|\text{fl}\left(\widehat{x}\right) - x\right\|_\infty\right)\gamma_j. \qquad (8)\end{aligned}$$

con

$$\begin{aligned}&\left\|\text{fl}\left(\widehat{x}\right) - x\right\|_\infty = \max_{i\in\{1,\ldots,n\}}\left|\text{fl}\left(\widehat{x}_i\right) - x_i\right| = \max_{i\in\{1,\ldots,n\}}\left|\text{fl}\left(\widehat{x}_i\right) - \widehat{x}_i + \widehat{x}_i - x_i\right| \\ &= \max_{i\in\{1,\ldots,n\}}\left|\delta_i\widehat{x}_i + \widehat{\varepsilon}_i x_i\right| = \max_{i\in\{1,\ldots,n\}}\left|\delta_i\left(1+\widehat{\varepsilon}_i\right)x_i + \widehat{\varepsilon}_i x_i\right| \\ &= \max_{i\in\{1,\ldots,n\}}\left|\delta_i\left(1+\widehat{\varepsilon}_i\right) + \widehat{\varepsilon}_i\right|\left|x_i\right| \leq \max_{i\in\{1,\ldots,n\}}\left|\delta_i\left(1+\widehat{\varepsilon}_i\right) + \widehat{\varepsilon}_i\right| \cdot \left\|x\right\|_\infty.\end{aligned}$$

Per cui il secondo termine in (8) ha modulo minore o uguale di una costante volte

$$\max\left\{\max_{i\in\{1,\ldots,n\}}\left|\widehat{\varepsilon}_i\right|, \max_{i\in\{1,\ldots,n\}}\left|\delta_i\right|, \max_{j\in\{1,\ldots,m\}}\left|\gamma_j\right|\right\}^2$$

per

$$\max\left\{\max_{i\in\{1,\ldots,n\}}\left|\widehat{\varepsilon}_i\right|, \max_{i\in\{1,\ldots,n\}}\left|\delta_i\right|\right\}$$

sufficientemente piccolo. Si conclude che

$$\varepsilon_{\text{alg}} \doteq \sum_{j=1}^{m} M_j(x)\,\gamma_j + \sum_{j=1}^{m} O\left(\|\text{fl}(\widehat{x}) - x\|_\infty\right)\gamma_j \doteq \sum_{j=1}^{m} M_j(x)\,\gamma_j.$$

Le funzioni M_j, $j \in \{1, \ldots m\}$, sono dette *indici di stabilità* dell'algoritmo. Per $x \in D$, il valore $M_j(x)$ dice quanto l'errore γ_j introdotto nell'esecuzione della $j-$esima operazione si propaghi sull'errore ε_{alg}, quando il dato è x.

L'indice di stabilità M_m dell'ultima istruzione è costantemente uguale a 1:

$$M_m(x) = 1,\ x \in D,$$

Infatti, γ_m compare solo nella formula di propagazione

$$\varepsilon_{\text{alg}} = \beta_c \doteq K_1(a,b)\beta_a + K_2(a,b)\beta_b + \gamma_m \ \text{ oppure } \ \varepsilon_{\text{alg}} = \beta_c \doteq K(a)\beta_a + \gamma_m$$

dell'ultima istruzione

$$c = a \circ b \ \text{ oppure } \ c = g(a).$$

Definizione 14 *L'algoritmo usato per calcolare i valori di f si dice* stabile *sul dato $x \in D$ se tutti gli indici di stabilità $M_j(x)$, $j \in \{1, \ldots, m-1\}$, hanno ordine di grandezza non superiore all'unità. L'algoritmo si dice* instabile *sul dato x se non è stabile, cioè esiste un indice di stabilità $M_j(x)$, $j \in \{1, \ldots, m-1\}$, che ha ordine di grandezza superiore all'unità.*

Pertanto, se l'algoritmo è stabile sul dato x, allora da

$$\varepsilon_{\text{alg}} \doteq \sum_{j=1}^{m} M_j(x)\,\gamma_j$$

segue che l'errore ε_{alg} ha ordine di grandezza non superiore al massimo ordine di grandezza degli errori γ_j e quindi ha ordine di grandezza non superiore a quello di eps. Questo naturalmente risulta vero supponendo che m non sia grande, come assumeremo nel seguito.

Se, invece, l'algoritmo è instabile sul dato x, allora ε_{alg} ha "in generale" ordine di grandezza superiore al massimo ordine di grandezza degli errori γ_j e quindi ha "in generale" ordine di grandezza superiore a quello di eps: come per il condizionamento, "in generale" vuol dire che lo è per qualche $m-$upla di errori γ_j, ma non è detto che ciò accada per la $m-$upla di errori γ_j che effettivamente si hanno.

Nell'algoritmo esaminato sopra, che è l'Algoritmo 1 dell'Esempio A, si ha, per $x > 0$,

$$\begin{aligned}
M_1(x) &= \frac{1}{2} \cdot \frac{\sqrt{x+1}}{\sqrt{x+1} - \sqrt{x}} = \frac{1}{2}\sqrt{x+1}\left(\sqrt{x+1} + \sqrt{x}\right), \\
M_2(x) &= \frac{\sqrt{x+1}}{\sqrt{x+1} - \sqrt{x}} = \sqrt{x+1}\left(\sqrt{x+1} + \sqrt{x}\right), \\
M_3(x) &= -\frac{\sqrt{x}}{\sqrt{x+1} - \sqrt{x}} = -\sqrt{x}\left(\sqrt{x+1} + \sqrt{x}\right).
\end{aligned}$$

Poichè

$$x = \frac{1}{2}\sqrt{x}\left(\sqrt{x}+\sqrt{x}\right) \leq M_1(x) \leq \frac{1}{2}\sqrt{x+1}\left(\sqrt{x+1}+\sqrt{x+1}\right) = x+1$$

e, analogamente,

$$\begin{aligned} 2x &\leq M_2(x) = \sqrt{x+1}\left(\sqrt{x+1}+\sqrt{x}\right) \leq 2(x+1) \\ 2x &\leq |M_3(x)| = \sqrt{x}\left(\sqrt{x+1}+\sqrt{x}\right) \leq 2(x+1), \end{aligned}$$

l'algoritmo risulta stabile se e solo se x non è grande.

Per l'Algoritmo 2 dell'Esempio A si ha:

1 $a = x+1$ $\quad \beta_a \doteq \gamma_1$

2 $b = \sqrt{a}$ $\quad \beta_b \doteq \frac{1}{2}\beta_a + \gamma_2 \doteq \frac{1}{2}\gamma_1 + \gamma_2$

3 $c = \sqrt{x}$ $\quad \beta_c \doteq \gamma_3$

4 $d = b+c$ $\quad \beta_d \doteq \frac{b}{b+c}\beta_b + \frac{c}{b+c}\beta_c + \gamma_4 \doteq \frac{1}{2}\,\frac{b}{b+c}\gamma_1 + \frac{b}{b+c}\gamma_2 + \frac{c}{b+c}\gamma_3 + \gamma_4$

5 $y = 1/d$ $\quad \varepsilon_{\text{alg}} \doteq -\beta_d + \gamma_5 \doteq -\frac{1}{2}\,\frac{b}{b+c}\gamma_1 - \frac{b}{b+c}\gamma_2 - \frac{c}{b+c}\gamma_3 - \gamma_4 + \gamma_5.$

Per cui, per $x > 0$,

$$M_1(x) = -\frac{1}{2}\cdot\frac{\sqrt{x+1}}{\sqrt{x+1}+\sqrt{x}},\ M_2(x) = -\frac{\sqrt{x+1}}{\sqrt{x+1}+\sqrt{x}},$$
$$M_3(x) = -\frac{\sqrt{x}}{\sqrt{x+1}+\sqrt{x}},\ M_4(x) = -1.$$

Poichè $|M_j(x)| \leq 1$, $j \in \{1,2,3,4\}$, l'algoritmo è stabile per ogni dato.

Consideriamo ora l'Esempio B.

Per l'Algoritmo 1 si ha

1 $a = x_1 \cdot x_2$ $\quad \beta_a \doteq \gamma_1$

2 $y = a + x_1$ $\quad \varepsilon_{\text{alg}} \doteq \frac{a}{a+x_1}\beta_a + \gamma_2 \doteq \frac{a}{a+x_1}\gamma_1 + \gamma_2.$

Quindi, per $x \in \mathbb{R}^2$, si ha

$$M_1(x) = \frac{x_1x_2}{x_1x_2+x_1} = \frac{x_2}{x_2+1}$$

e l'algoritmo è stabile se e solo se x_2 non è a vicino a -1.

Per l'Algoritmo 2 si ha

1 $a = x_2 + 1$ $\quad \beta_a \doteq \gamma_1$

2 $y = x_1 \cdot a$ $\quad \varepsilon_{\text{alg}} \doteq \beta_a + \gamma_2 \doteq \gamma_1 + \gamma_2.$

Quindi, per $x \in \mathbb{R}^2$,

$$M_1(x) = 1$$

e l'algoritmo è stabile per ogni dato.

5.5 Studio dell'instabilità

Si supponga di voler determinare su quali dati x un algoritmo per calcolare i valori della funzione f è instabile.

Come fatto per gli indici di condizionamento, nel caso $n = 1$, si può studiare un indice di stabilità $M_j(x)$ come funzione della variabile reale $x \in F$, dove

$$F = \{x \in D : x \neq 0 \text{ e } f(x) \neq 0\}$$

cercando punti a sulla frontiera di F tali che

$$\lim_{x \to a} M_j(x) = \infty.$$

Si può ritenere che l'algoritmo è instabile su x se e solo se x è vicino a uno di questi punti a, per almeno uno degli indici di stabilità M_j.

Esempio 15 *Consideriamo la funzione*

$$f(x) = x^{\frac{1}{x}} = e^{\frac{\log x}{x}},\ x > 0.$$

Si ha $F = (0, +\infty)$ con punti di frontiera 0 e $+\infty$. Nella teoria del condizionamento si è visto che f è mal condizionata su x se e solo se x è piccolo.

Analizziamo la stabilità dell'algoritmo basato sull'espressione

$$f(x) = e^{\frac{\log x}{x}}.$$

L'algoritmo è

$$\begin{aligned} a &= \log x \\ b &= \tfrac{a}{x} \\ y &= e^b. \end{aligned}$$

L'analisi dell'errore algoritmico è

1 $a = \log x$ $\quad \beta_a \doteq \frac{1}{x}\beta_x + \gamma_1 = \gamma_1$

2 $b = \frac{a}{x}$ $\quad \beta_b \doteq \beta_a - \beta_x + \gamma_2 \doteq \gamma_1 + \gamma_2.$

3 $y = e^b$ $\quad \varepsilon_{\text{alg}} \doteq b\beta_b + \gamma_3 \doteq b(\gamma_1 + \gamma_2) + \gamma_3 = b\gamma_1 + b\gamma_2 + \gamma_3$

Si ha

$$M_1(x) = M_2(x) = b = \frac{a}{x} = \frac{\log x}{x}.$$

Avendosi

$$\lim_{x\to 0} \frac{\log x}{x} = -\infty$$
$$\lim_{x\to +\infty} \frac{\log x}{x} = 0,$$

si conclude che l'algoritmo è instabile se e solo se x è piccolo, esattamente dove la funzione è mal condizionata.

Nel caso $n > 1$, come fatto per lo studio del mal condizionamento, si può cercare di esprimere un indice di stabilità $M_j(x)$ in termine di un parametro reale t e, quindi, far diventare $M_j(x)$ funzione della sola variabile reale t e in questo modo ricondursi al caso $n = 1$. Ad esempio, nel caso dell'esempio B Algoritmo 1, dove $n = 2$, questo parametro t è x_2.

Esercizio. Si studi la stabilità dell'algoritmo per calcolare i valori di

$$f(x) = x^x = e^{x \log x},\ x > 0,$$

basato sull'espressione

$$f(x) = e^{x \log x}.$$

Il condizionamento di tale funzione è stato studiato in un esercizio della teoria del condizionamento.

Esercizio. Si studi il condizionamento di

$$f(x) = \log \frac{x+1}{x},\ x > 0,$$

e la stabilità dei tre algoritmi basati sulle tre diverse espressioni

$$f(x) = \log \frac{x+1}{x} = \log\left(1 + \frac{1}{x}\right) = \log(x+1) - \log x.$$

Esercizio. Si studi il condizionamento di

$$f(x) = \sqrt{x_1 + x_2} - \sqrt{x_1},\ x \in \mathbb{R}^2 \text{ con } x_1, x_2 > 0,$$

e la stabilità dei due algoritmi basati sulle due diverse espressioni

$$f(x) = \sqrt{x_1 + x_2} - \sqrt{x_1} = \frac{x_2}{\sqrt{x_1 + x_2} + \sqrt{x_1}}.$$

Esercizio. Si studi il condizionamento di

$$f(x) = x_1 x_2 + x_1 x_3,\ x \in \mathbb{R}^3,$$

e la stabilità dei due algoritmi basati sulle due diverse espressioni

$$f(x) = x_1 x_2 + x_1 x_3 = x_1 (x_2 + x_3).$$

5.6 Alcune considerazioni sugli errori

Fissato un dato x, nel processo di calcolare $f(x)$ vengono commessi i seguenti errori:

- errori di misurazione: $\widehat{\varepsilon}_i$, $i \in \{1, \ldots, n\}$;
- errori di rappresentazione in macchina dei dati misurati: δ_i, $i \in \{1, \ldots, n\}$;
- errori nell'esecuzione delle operazioni aritmetiche e delle funzioni matematiche elementari: γ_j, $j \in \{1, \ldots, m\}$.

Gli errori δ_i e gli errori γ_j hanno ordine di grandezza non superiore a (quello di) eps. Assumiamo poi che gli errori $\widehat{\varepsilon}_i$ abbiamo ordine di grandezza non superiore a (quello di) una tolleranza prefissata TOL.

Supponiamo TOL $\gg$ eps: in genere l'ordine di grandezza di TOL va da 10^{-2} a 10^{-5}, mentre eps ha ordine di grandezza 10^{-16} nello standard IEEE in doppia precisione.

Si possono presentare nel calcolare $f(x)$ le seguenti quattro situazioni che ora elenchiamo.

Situazione 1. f è ben condizionata sul dato x e l'algoritmo usato per calcolare i valori di f è stabile sul dato x.

Nella Situazione 1:

- ε_{in} ha ordine di grandezza non superiore a TOL;
- ε_{mac} ha ordine di grandezza non superiore a eps;
- ε_{alg} ha ordine di grandezza non superiore a eps.

Quindi, $\varepsilon_y \doteq \varepsilon_{\text{in}} + \varepsilon_{\text{mac}} + \varepsilon_{\text{alg}}$ ha ordine di grandezza non superiore a TOL.

La sottostante tabella dice per gli Esempi A e B e gli Algoritmi 1 e 2 quando si è nella Situazione 1.

	Algoritmo 1	Algoritmo 2
Esempio A	x non grande	per ogni x
Esempio B	x_2 non vicino a -1	x_2 non vicino a -1

Situazione 2. f è ben condizionata sul dato x e l'algoritmo usato per calcolare i valori di f è instabile sul dato x.

Nella Situazione 2:

- ε_{in} ha ordine di grandezza non superiore a TOL;
- ε_{mac} ha ordine di grandezza non superiore a eps;
- ε_{alg} ha "in generale" ordine di grandezza superiore a eps.

Ricordando che gli indici di stabilità vanno a moltiplicare gli errori γ_j che hanno ordine di grandezza non superiore a eps, si può dire che, solo per una instabilità dell'algoritmo caratterizzata da indici di stabilità con ordine di grandezza superiore a $\frac{\text{TOL}}{\text{eps}}$, ε_{alg} ha "in generale" ordine di grandezza superiore a TOL e, quindi, $\varepsilon_y \doteq \varepsilon_{\text{in}} + \varepsilon_{\text{mac}} + \varepsilon_{\text{alg}} \approx \varepsilon_{\text{alg}}$ ha ordine di grandezza superiore a TOL.

Qualora si sia in questa situazione sfavorevole, l'algoritmo instabile va sostituito con un algoritmo stabile per finire nella Situazione 1.

La sottostante tabella dice per gli Esempi A e B e gli Algoritmi 1 e 2 quando si è nella Situazione 2.

	Algoritmo 1	Algoritmo 2
Esempio A	x grande	per nessun x
Esempio B	per nessun x	per nessun x

Situazione 3. f è mal condizionata sul dato x e l'algoritmo usato per calcolare i valori di f è stabile sul dato x.

Nella Situazione 3:

- ε_{in} ha "in generale" ordine di grandezza superiore a TOL;
- ε_{mac} ha "in generale" ordine di grandezza superiore a eps;
- ε_{alg} ha ordine di grandezza non superiore a eps.

Quindi, $\varepsilon_y \doteq \varepsilon_{\text{in}} + \varepsilon_{\text{mac}} + \varepsilon_{\text{alg}}$ ha "in generale" ordine di grandezza superiore a TOL.

La Situazione 3 è una non favorevole a causa del mal condizionamento di f. La stabilità dell'algoritmo non è di alcuna utilità.

La sottostante tabella dice per gli Esempi A e B e gli Algoritmi 1 e 2 quando si è nella Situazione 3.

	Algoritmo 1	Algoritmo 2
Esempio A	per nessun x	per nessun x
Esempio B	per nessun x	x_2 vicino a -1

Situazione 4. f è mal condizionata sul dato x e l'algoritmo usato per calcolare i valori di f è instabile sul dato x.

Nella Situazione 4:

- ε_{in} ha "in generale" ordine di grandezza superiore a TOL;
- ε_{mac} ha "in generale" ordine di grandezza superiore a eps;
- ε_{alg} ha "in generale" ordine di grandezza superiore a eps.

Quindi, come nella Situazione 3, $\varepsilon_y \doteq \varepsilon_{\text{in}} + \varepsilon_{\text{mac}} + \varepsilon_{\text{alg}}$ ha "in generale" ordine di grandezza superiore a TOL.

La Situazione 4 è una non favorevole a causa del mal condizionamento di f e dell'instabilità dell'algoritmo. A differenza della Situazione 2, l'instabilità dell'algoritmo non è il fattore determinante la situazione sfavorevole: rimpiazzando l'algoritmo instabile con uno stabile si finisce nella Situazione 3, dove, comunque, ε_y ha "in generale" ordine di grandezza superiore a TOL.

La sottostante tabella dice per gli Esempi A e B e gli Algoritmi 1 e 2 quando si è nella Situazione 4.

	Algoritmo 1	Algoritmo 2
Esempio A	per nessun x	per nessun x
Esempio B	x_2 vicino a -1	per nessun x

Si osservi che quando il dato x è tale che

$$x = \widehat{x} = \text{fl}\,(\widehat{x})$$

cioè x è un dato che non si ottiene da una misurazione e le componenti $x_1, \ldots, x_n$ di x sono numeri di macchina, allora

$$\varepsilon_{\text{in}} = 0 \quad \text{e} \quad \varepsilon_{\text{mac}} = 0$$

e quindi

$$\varepsilon_y = \varepsilon_{\text{alg}}.$$

In questo caso, indipendentemente dal buon o mal condizionamento di f su x, ha senso sostituire un algoritmo instabile con uno stabile.