UNIVERSITÀ DEGLI STUDI DI TRIESTE

Analisi Numerica

Aritmetica di macchina e analisi degli errori

Ángeles Martínez Calomardo

amartinez@units.it

Laurea Triennale in Intelligenza Artificiale e Data Analytics

Argomenti

Introduzione

La soluzione al calcolatore di un problema matematico è affetta da errori di vario tipo:

1. Errori dovuti alla modellazione matematica del problema reale ed errori presenti nei dati sperimentali.
2. Errori di troncamento commessi nella trasformazione di un problema matematico (dimensione infinita) in uno di dimensione finita.
3. Errori di arrotondamento dovuti al fatto che sul calcolatore si può rappresentare soltanto un sottoinsieme finito dei numeri reali.

I punti 2 e 3 sono oggetto di studio dell'Analisi Numerica.

Effetti disastrosi degli errori numerici

Fallimento del missile Patriot: il giorno 25 Febbraio 1991, durante la prima guerra del golfo, un missile Patriot fallì l'intercettazione di un missile Scud iracheno che centrò il suo obbiettivo causando la morte di 28 soldati americani e un centinaio di feriti.

- Causa: L'orologio interno del sistema misurava il tempo in decimi di secondo, poi questo numero intero veniva moltiplicato per 0.1 per ottenere il tempo in secondi e memorizzato usando soli 24 bit.
- 0.1 non ha un'espansione binaria finita $\longrightarrow$ ad ogni decimo di secondo l'errore che si commette è (circa) 0.95×10^{-7} secondi.
- Il computer che regolava i lanci dei Patriot rimase in funzione per 100 ore il che produsse un errore pari a $0.95 \times 10^{-7} \times 10 \times 3600 \times 100 \approx 0.34$ secondi.
- Lo Scud viaggiava a Mach 5 (1700 m/sec) con un conseguente errore nella traiettoria di circa 600 metri.

Effetti disastrosi degli errori numerici

Esplosione del razzo Ariane 5: il 4 giugno 1996, avvenuta a soli 40 secondi dal lancio dovuta a un overflow per una conversione di un numero reale memorizzato usando 64 bit in un numero intero con soli 16 bit. Il numero che rappresentava la velocità orizzontale del razzo era più grande del massimo numero rappresentabile con soli 16 bit. La missione era costata 7.5 miliardi di dollari.

Altri esempi:

1. Crollo di una piattaforma petrolifera nel mare del Nord (Norvegia) nel 1991.
2. Distruzione del veicolo spaziale Mars Climate orbiter nel 1999.

Altre informazioni alla pagina del professore D. Arnold

`http://www.ima.umn.edu/~arnold/disasters/`

e su

`http://marsprogram.jpl.nasa.gov/msp98/orbiter`

Sistema posizionale

- Fissato un numero naturale $B > 1$ che chiameremo base,e un numero $x \in \mathbb{R}$ con un numero finito di cifre d_k, $k = -m, -m+1, \ldots, -1, 0, 1, \ldots, , n-1, n$, si definisce x_B la rappresentazione posizionale di x in base B :

$$\boxed{x_B = (-1)^s \left(d_n d_{n-1} \ldots d_1 d_0 . d_{-1} \ldots d_{-m}\right) = (-1)^s \left(\sum_{k=-m}^{n} d_k B^k\right) \qquad d_n \neq 0}$$

con $s = 0$ se il numero è positivo, $s = 1$ se è negativo e $d_k \in \{0, 1, 2, \ldots, B-1\}$.

- Esempi:

$$(867.0985)_{10} = (-1)^0 \cdot \left(8 \cdot 10^2 + 6 \cdot 10^1 + 7 \cdot 10^0 + 9 \cdot 10^{-2} + 8 \cdot 10^{-3} + 5 \cdot 10^{-4}\right)$$

$$(-10110.0001)_2 = (-1)^1 \cdot \left(1 \cdot 2^4 + 1 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^{-4}\right)$$

- In generale ogni $x \in \mathbb{R}$ si scrive, fissata la base B, come:

$$x_B = (-1)^s \underbrace{\left(\sum_{k=0}^{n} d_k B^k\right)}_{\text{Parte intera}} + \underbrace{\left(\sum_{k=1}^{\infty} d_{-k} B^{-k}\right)}_{\text{Parte frazionaria}} \qquad d_n \neq 0$$

Domanda: perché la serie che rappresenta la parte frazionaria converge?

Sistema posizionale

- Risposta: si utilizza il confronto con la serie geometrica di ragione B^{-1} (criterio di confronto tra serie a termini non negativi.)
  Poiché $d_{-k} \leq B-1, \qquad k=1,2,\ldots,\infty$:

$$\sum_{k=1}^{\infty}(B-1)B^{-k} = (B-1)\sum_{k=1}^{\infty}B^{-k}$$

  e la serie geometrica di ragione B^{-1} è convergente:

$$\sum_{k=1}^{\infty}B^{-k} = \frac{B^{-1}}{1-B^{-1}}.$$

  Nota: $(0.99999\ldots)_{10} = 1$ e $(0.111111\ldots)_2 = 1$.

- La parte frazionaria di un numero **irrazionale** è infinita (perché?).

- Un numero **razionale** può avere rappresentazione data da un numero finito di cifre in una base e infinito in un'altra:

$$x = \frac{1}{3} \qquad x_{10} = 0.\overline{3} \qquad x_3 = 0.1$$

$$x = 0.1 \qquad x_{10} = 0.1 \qquad x_2 = 0.0\overline{0011}$$

Conversione di base

Base 2 $\longrightarrow$ Base 10

Per trasformare un numero da base 2 a base 10 è sufficiente esprimerlo con la sua notazione posizionale.

Esempio: convertiamo in base 10 il numero binario $x = 10001000.01$

$$
\begin{aligned}
x &= 1\cdot 2^7 + 0\cdot 2^6 + 0\cdot 2^5 + 0\cdot 2^4 + 1\cdot 2^3 + 0\cdot 2^2 + 0\cdot 2^1 + 0\cdot 2^0 \\
&\quad + 0\cdot 2^{-1} + 1\cdot 2^{-2} \\
&= 2^7 + 2^3 + 2^{-2} \\
&= 128 + 8 + 0.25 \\
&= 136.25
\end{aligned}
$$

Conversione di base

Base 10 $\longrightarrow$ Base 2

La conversione si effettua in due passi:

1. Parte intera:
   - Si divide per due la parte intera del numero, si prende poi il quoziente e lo si divide per due ... e così via finchè il quoziente risulta 0.
   - I resti che si ottengono da queste divisioni, scritti in ordine inverso rispetto a quello in cui sono stati ottenuti, formano le cifre binarie della parte intera.

2. Parte frazionaria:
   - Si moltiplica per due la parte frazionaria del numero decimale.
   - La parte intera del risultato rappresenta la corrispondente cifra del numero in base 2.
   - Si prende poi la parte frazionaria del risultato, la si moltiplica per 2 e si riapplica il procedimento.
   - Il procedimento si arresta o quando la parte frazionaria vale 0 oppure ci si accorge che le cifre binarie si ripetono periodicamente.

Conversione di base

Base 10 $\longrightarrow$ Base 2

Esempio: convertiamo il numero 389.1 da base 10 a base 2

Parte intera

	quoziente	resto
$389 \div 2$	194	1
$194 \div 2$	97	0
$97 \div 2$	48	1
$48 \div 2$	24	0
$24 \div 2$	12	0
$12 \div 2$	6	0
$6 \div 2$	3	0
$3 \div 2$	1	1
$1 \div 2$	0	1

$\uparrow$

Parte frazionaria

$.1 \times 2$	$=$	0.2	$\rightarrow 0$
$.2 \times 2$	$=$	0.4	$\rightarrow 0$
$.4 \times 2$	$=$	0.8	$\rightarrow 0$
$.8 \times 2$	$=$	1.6	$\rightarrow 1$
$.6 \times 2$	$=$	1.2	$\rightarrow 1$
$.2 \times 2$	$=$	0.4	$\rightarrow 0$
$.4 \times 2$	$=$	0.8	$\rightarrow 0$
$.8 \times 2$	$=$	1.6	$\rightarrow 1$
$.6 \times 2$	$=$	1.2	$\rightarrow 1$

$\downarrow$

Osserviamo che nella parte decimale la sequenza 0011 si ripete ciclicamente.

Il numero 389.1 in base 10 corrisponde pertanto al numero $110000101.\overline{00011}$.

Rappresentazione in virgola mobile normalizzata

- Fissata una base B, ogni numero (intero o reale) $x \neq 0$ si può scrivere in virgola mobile normalizzata come:

$$x = (-1)^s B^e \left(\sum_{k=1}^{\infty} d_k B^{-k}\right) \text{ con } \begin{cases} d_1 > 0 \\ 0 \leq d_k \leq B-1 \\ e \in \mathbb{Z} \end{cases}$$

- L'espressione precedente si può scrivere in modo più compatto come:

  $x = \pm p B^e, \qquad \text{dove} \qquad B^{-1} \leq p < 1,$

  dove il numero reale p è detto **mantissa** e il numero intero e è detto **esponente**.
- Esempi: In base $B = 10$:

$$x = 0.00745 \quad \Longrightarrow \quad 0.745 \cdot 10^{-2}$$
$$x = 70408.102 \quad \Longrightarrow \quad 0.70408102 \cdot 10^{5}$$

In base $B = 2$:

$$x = 11001.111 \quad \Longrightarrow \quad 0.11001111 \cdot 2^{5}$$

Rappresentazione in virgola mobile normalizzata

- La rappresentazione di un numero in virgola mobile non è unica, ma quella normalizzata sì ($d_1 \neq 0$ **garantisce l'unicità della rappresentazione**).
- Esempio: $x = 43.75$ si può scrivere in virgola mobile come

$$0.4375 \cdot 10^2 \qquad 4.375 \cdot 10^1 \qquad 43.75 \cdot 10^0 \qquad 0.04375 \cdot 10^3 \qquad \cdots$$

$x = 43.75$ in virgola mobile normalizzata è $0.4375 \cdot 10^2$, con $d_1 = 4 \neq 0$

- Altri esempi:
  - $x = 453.25 = 0.45325 \cdot 10^3 = (-1)^0 \left(4 \cdot 10^{-1} + 5 \cdot 10^{-2} + 3 \cdot 10^{-3} + 2 \cdot 10^{-4} + 5 \cdot 10^{-5}\right) \cdot 10^3$
  - $x = -0.0026 = -0.26 \cdot 10^{-2} = (-1)^1 \left(2 \cdot 10^{-1} + 6 \cdot 10^{-2}\right) \cdot 10^{-2}$
  - Il numero irrazionale π in virgola mobile normalizzata si scrive $\pi = (0.314157\ldots) \cdot 10^1 = (-1)^0 \left(3 \cdot 10^{-1} + 1 \cdot 10^{-2} + 4 \cdot 10^{-3} + 1 \cdot 10^{-4} + 5 \cdot 10^{-5} + 7 \cdot 10^{-6} \cdots\right) \cdot 10^1$

Numeri macchina

- All'interno del calcolatore i numeri vengono immagazzinati in virgola mobile normalizzata con un numero **finito** t di cifre per la mantissa e un numero **finito** di cifre per codificare l'esponente $(L \leq e \leq U)$.
- Fissata una base B (generalmente $B = 2$), fissato t e fissati $L < 0$ e $U > 0$, si definisce insieme di numeri macchina $\mathbb{F}(B,t,L,U)$:

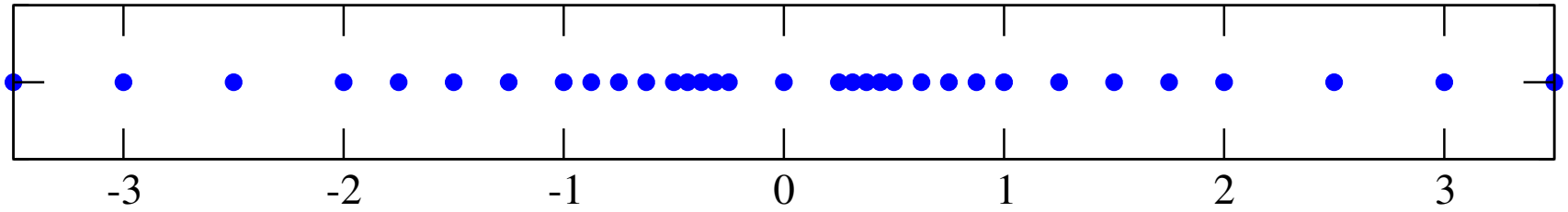
$$\mathbb{F}(B,t,L,U) = \left\{ x \quad | \quad x = (-1)^s B^e \left(\sum_{k=1}^{t} d_k B^{-k} \right) \right\} \cup \{0\},$$
$$\text{con} \quad d_1 > 0, \quad 0 \leq d_k \leq B-1, \quad L \leq e \leq U$$

- Lo zero si codifica con mantissa nulla ed esponente nullo.
- Un numero x in $\mathbb{F}(B,t,L,U)$ lo scriveremo d'ora in poi come

$$x = (-1)^s \cdot (0.d_1 d_2 \ldots d_t) \cdot B^e$$

Esempio: $F(2,3,-1,2)$

- Base 2 con 3 cifre per la mantissa (normalizzata)
- Mantisse possibili: $0.100,\ 0.101,\ 0.110,\ 0.111.$
- Ad ogni mantissa si abbina uno degli $U-L+1=2-(-1)+1=4$ esponenti possibili: $2^{-1},2^{0},2^{1},2^{2}$.
- I numeri macchina in $F(2,3,-1,2)$ sono dunque:



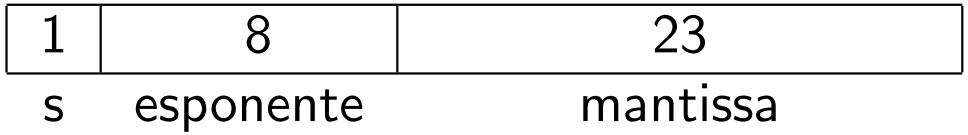
- I numeri macchina sono più addensati quanto più piccoli sono e la loro separazione aumenta man mano che aumenta il loro valore assoluto.

Nel calcolatore

- La base è 2; le cifre sono 0 o 1 (bit).
- Per codificare un numero macchina $x = (-1)^s \cdot (0.d_1 d_2 \ldots d_t) \cdot B^e$ è sufficiente memorizzare:
  - il segno (un bit)
  - le cifre della mantissa (t bit)
  - l'esponente.
- Ogni numero macchina occupa una parola di memoria di 32 bit in singola precisione o 2 parole consecutive di 32 bit (64 bit) in doppia precisione.

Singola precisione

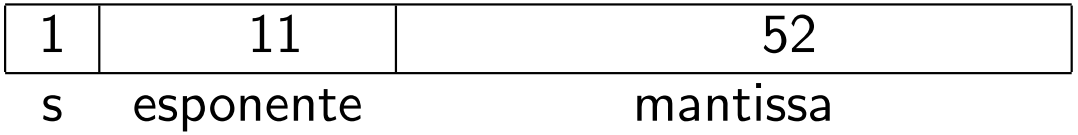
- Ogni numero macchina occupa 32 bit, distribuiti nei tre campi segno, esponente e mantissa come segue:



- Attualmente, l'insieme di numeri macchina in singola precisione è: $F(2,24,-126,127)$.
- Con 23 bit si codificano 24 cifre della mantissa (1 bit nascosto).
- Dei $2^8 = 256$ esponenti possibili, 2 si riservano per usi speciali.
- I numeri rappresentabili in singola precisione sono $2 \cdot (U - L + 1) \cdot (B - 1) \cdot B^{t-1} + 1 = 2 \cdot 254 \cdot 1 \cdot 2^{23} + 1 \approx 4.2789 \cdot 10^9$.

Doppia precisione

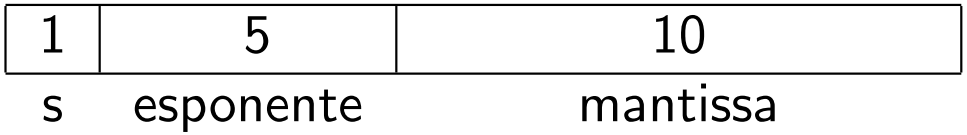
- Ogni numero macchina occupa 64 bit, distribuiti nei tre campi segno, esponente e mantissa come segue:



- L'insieme di numeri macchina in doppia precisione è: $F(2,53,-1022,1023)$.
- Con 52 bit si codificano 53 cifre della mantissa (1 bit nascosto).
- Dei $2^{11} = 2048$ esponenti possibili, 2 si riservano per usi speciali.
- I numeri rappresentabili in doppia precisione sono $2 \cdot (U - L + 1) \cdot (B - 1) \cdot B^{t-1} + 1 = 2 \cdot 2046 \cdot 1 \cdot 2^{52} + 1 \approx 1.8438 \cdot 10^{19}$.

Half precision

- Con l'avvento delle GPU (Graphics Processing Units) è diventata fondamentale la rappresentazione dei numeri con soli 16 bits.
  Tale formato permette di eseguire i calcoli più velocemente e risparmiare memoria.
- In esso, ogni numero macchina occupa 16 bit, distribuiti nei tre campi segno, esponente e mantissa come segue:



- Attualmente, l'insieme di numeri macchina in precisione dimezzata è: $F(2,11,L,U)$. **Esercizio:** Quanto vale L? Quanto vale U?
- Con 10 bit si codificano 11 cifre della mantissa (1 bit nascosto).
- Dei $2^5 = 32$ esponenti possibili, 2 si riservano per usi speciali.
- **Esercizio:** Quanti numeri possono essere rappresentati in precisione dimezzata?

Approssimazione di un numero reale

- In $F(B,t,L,U)$ quando un numero $x = pB^e \in \mathbb{R}$ ha più di t cifre nella mantissa (con esponente $L \leq e \leq U$) può venire approssimato con un numero macchina, che chiameremo $\text{fl}(x)$, in due modi possibili:

  troncamento *Nella mantissa p si cancella la parte che eccede la t-esima cifra.*

  arrotondamento *Alla mantissa p si aggiunge $\frac{B}{2}B^{-(t+1)}$ e poi si tronca alla t-esima cifra.*

- Esempio: Con $t = 6$, per troncamento:

$$x = 0.745645897 \quad \text{fl}(x) = tr(0.745645897) = 0.745645$$

  mentre per arrotondamento:

$$\text{fl}(x) = tr(0.745645897 + 0.0000005) = tr(0.745646397) = 0.745646$$

- Si noti che arrotondare equivale a sommare 1 alla t-esima cifra della mantissa, d_t, se la successiva cifra (d_{t+1}) è $\geq \frac{B}{2}$ altrimenti la cifra t-esima rimane invariata.

Underflow e Overflow

In $F(B,t,L,U)$, nell'approssimare x con $\mathrm{fl}(x)$

- se l'esponente $e > U$, si produce un Overflow . Il numero x viene rappresentato come *Inf*.
- se l'esponente $e < L$, si produce un Underflow. Il numero x viene rappresentato come 0.

Errori assoluti e relativi

- Se x è un numero reale e x^* una sua approssimazione definiamo

  errore assoluto : la quantità $|x - x^*|$,

  errore relativo : la quantità $\dfrac{|x - x^*|}{|x|}$ (se $x \neq 0$).

- Nel caso si abbia a che fare con enti diversi da numeri reali (funzioni, vettori, matrici) le definizioni sono le stesse a patto di sostituire il valore assoluto con un'opportuna norma.
- L'errore relativo fornisce una indicazione più precisa della distanza fra x e x^*.
- Esempio. Siano $x = 0.456789 \cdot 10^{-30}$ e $x^* = 0.6 \cdot 10^{-30}$.

  errore assoluto $|x - x^*| = 0.143211 \cdot 10^{-30}$

  errore relativo $\dfrac{|x - x^*|}{|x|} = 0.313517$
- L'errore relativo è maggiore del 31%!!

Errori di rappresentazione

Maggiorazione dell'errore assoluto

- Errore assoluto per troncamento:

$$|x - \mathrm{fl}(x)| = |pB^e - \bar{p}B^e| = |p - \bar{p}|B^e \leq B^{-t}B^e$$

Esempio:

$$x = 0.745645897 \quad \mathrm{fl}(x) = 0.745645,$$
$$|x - \mathrm{fl}(x)| = 0.000000897 = 0.897 \cdot 10^{-6} < 10^{-6}.$$

- Errore assoluto per arrotondamento:

$$|x - \mathrm{fl}(x)| = |pB^e - \bar{p}B^e| = |p - \bar{p}|B^e \leq \frac{B}{2}B^{-(t+1)}B^e$$

Esempio:

$$x = 0.745645897 \quad \mathrm{fl}(x) = 0.745646,$$
$$|x - \mathrm{fl}(x)| = 0.000000103 = 1.03 \cdot 10^{-7} < 5 \cdot 10^{-7}$$

Errori di rappresentazione

Maggiorazione dell'errore relativo

- Errore relativo per troncamento:

$$\frac{|x - \mathrm{fl}(x)|}{|x|} = \frac{|pB^e - \bar{p}B^e|}{pB^e} = \frac{|p - \bar{p}|}{p} \leq \frac{B^{-t}}{B^{-1}} = B^{1-t}$$

- Errore relativo per arrotondamento:

$$\frac{|x - \mathrm{fl}(x)|}{|x|} = \frac{|pB^e - \bar{p}B^e|}{pB^e} = \frac{|p - \bar{p}|}{p} \leq \frac{\frac{B}{2}B^{-(t+1)}}{B^{-1}} = \frac{1}{2}B^{1-t}$$

- Attualmente, la maggior parte dei sistemi implementa la tecnica di arrotondamento perché produce mediamente errori più piccoli.

Precisione di macchina

Definizione

Si chiama precisione di macchina, e si denota con il simbolo $\mathbf{u}$ *(unit roundoff):*

$$\mathbf{u} = \frac{1}{2} \cdot B^{1-t}$$

- La precisione di macchina rappresenta il massimo errore relativo che si commette nell'approssimare il numero reale x con il suo corrispondente numero macchina $\mathrm{fl}(x)$ per arrotondamento.
- In $F(2,24,-126,127)$ (singola precisione) $\mathbf{u} = 2^{-24} \approx 5.96 \cdot 10^{-8}$
- In $F(2,53,-1022,1023)$ (doppia precisione) $\mathbf{u} = 2^{-53} \approx 1.11 \cdot 10^{-16}$

Lo standard ANSI IEEE-754r

- Scritto nel 1985 e modificato nel 1989 e, più recentemente, nel 2008 costituisce lo standard ufficiale per la rappresentazione binaria dei numeri all'interno del calcolatore e l'aritmetica di macchina (il nome dello standard in inglese "Binary floating point arithmetic for microprocessor systems").
- Secondo lo standard un numero non nullo normalizzato si scrive come

$$x = (-1)^s \cdot (1+f) \cdot 2^{e^* - bias}.$$

- La mantissa si rappresenta dunque come $1.d_1 d_2 \ldots d_\tau$ essendo $f = 0.d_1 d_2 \ldots d_\tau$.
- τ identifica il numero di bit usato per codificare la parte frazionaria della mantissa. Il numero di cifre totali per la mantissa è $t = \tau + 1$.
- Il vero esponente del numero e si immagazzina in traslazione come $e* = e + bias$.
- In questa maniera non serve un bit di segno per l'esponente.
- Il *bias* in singola precisione vale 127 mentre in doppia 1023.

Lo standard ANSI IEEE-754r

Lo standard riserva due dei possibili valori per l'esponente per codificare due situazioni speciali:

- $e^* = 0$, viene riservato per la codifica dello zero ed eventuali numeri denormalizzati.
- $e^* = 255$ o $e^* = 2047$, che corrisponde a un esponente vero pari a 128 (singola) o 1024 (doppia), viene riservato per la codifica di
  - `Inf` (Overflow)
  - `NaN` (Not a Number), ovvero operazioni del tipo $\frac{0}{0}$, $\texttt{Inf} - \texttt{Inf}$, $\frac{\texttt{Inf}}{\texttt{Inf}}$.
- `Inf` viene codificato con mantissa nulla, mentre `NaN` con mantissa $\neq 0$.
- Più informazione sullo standard è reperibile al sito `http://it.wikipedia.org/wiki/IEEE_754`

Esempio di codifica

Vediamo come si rappresenta il numero $x = 126$ secondo lo standard IEEE in singola precisione.

- $(126)_{10} = (1111110)_2$
- Il numero $(1111110)_2$ si scrive, in virgola mobile normalizzata secondo lo standard IEEE come $1.111110 \cdot 2^6$.
- Della mantissa si immagazzinano solo le cifre dopo la virgola.
- $e^* = 6 + 127 = 133$, che in binario corrisponde a 10000101.
- Il numero 126 viene codificato con la stringa

  0 10000101 11111000000000000000000

Esempio di codifica

Vediamo come si rappresenta il numero $x = -8.265625$ secondo lo standard IEEE in doppia precisione.

- $(-8.265625)_{10} = (-1000.010001)_2$
- Il numero $(-1000.010001)_2$ si scrive, in virgola mobile normalizzata secondo lo standard IEEE come $-1.000010001 \cdot 2^3$.
- Della mantissa si immagazzinano solo le cifre dopo la virgola.
- $e^* = 3 + 1023 = 1026$, che in binario corrisponde a 10000000010.
- Il numero x viene codificato con la stringa

$$1 \quad 10000000010 \quad 0000100010\underbrace{00\cdots00}_{43 \text{ zeri}}$$

Massimo e minimo numero rappresentabile

L'insieme F di numeri macchina è limitato sia inferiormente che superiormente.

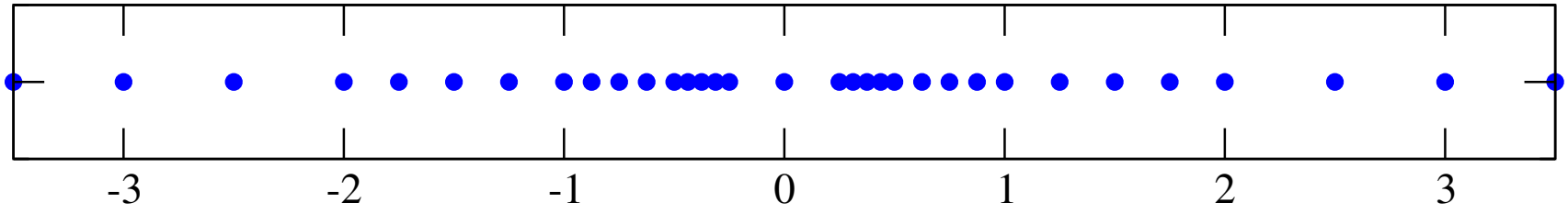
- Il numero più grande rappresentabile in aritmetica di macchina ha tutte le cifre della mantissa uguali a $B-1$ ed esponente U.
  Esempio:
  In $F(10,6,L,U)$ sarebbe $0.999999 \cdot 10^U = (1-10^{-6}) \cdot 10^U$
- Nello standard IEEE-754r in doppia precisione $(F(2,53,-1022,1023))$ il più grande numero rappresentabile è $(2-2^{-52}) \cdot 2^{1023} = 1.7977 \cdot 10^{308}$
- Il più piccolo numero rappresentabile in aritmetica di macchina ha tutte le cifre della mantissa uguali a 0 tranne la prima (virgola mobile normalizzata) ed esponente L.
  Esempio:
  In $F(10,6,L,U)$ sarebbe $0.100000 \cdot 10^L = 10^{-1} \cdot 10^U$
- Nello standard IEEE-754r in doppia precisione $(F(2,53,-1022,1023))$ il più piccolo numero rappresentabile è $1 \cdot 2^{-1022} = 2.2251 \cdot 10^{-308}$

Distanza assoluta tra due numeri macchina consecutivi

- A differenza di quello che occorre in $\mathbb{R}$, in $\mathbb{F}$ la distanza tra x e il suo elemento successivo x_+ non è infinitamente piccola, ma un valore ben determinato. Essendo $x = (-1)^s \cdot (1 + 0.d_1 d_2 d_3 \cdots d_\tau) \cdot B^e$ e $x_+ = (-1)^s \cdot (1 + 0.d_1 d_2 d_3 \cdots d_\tau + 1) \cdot B^e$, la **distanza assoluta** tra x e x_+ è:

$$\Delta x = |x - x_+| = B^{-\tau} \cdot B^e = B^{e-\tau}.$$

- Questa distanza è uguale per tutti i numeri macchina aventi lo stesso esponente.
- L'incremento/decremento di una unità dell'esponente comporta un incremento/decremento di in fattore pari alla base della distanza assoluta tra due numeri macchina consecutivi.
- Come visto in precedenza: i numeri macchina sono più addensati quanto più piccoli sono e la loro separazione aumenta man mano che aumenta il loro valore assoluto.



Distanza relativa

- La **distanza relativa** tra x e il suo elemento successivo x_+ si ottiene dividendo quella assoluta per il numero x:

$$\frac{|x - x_+|}{|x|} = \frac{B^{e-\tau}}{p \cdot B^e} = \frac{B^{-\tau}}{p}.$$

- Si può vedere che la distanza relativa tra due numeri macchina consecutivi ha un andamento periodico.
- La massima distanza relativa tra due numeri macchina consecutivi è:

$$\varepsilon_M = B^{-\tau}$$

  che si ottiene quando la mantissa p è uguale a 1.
- Nello standard IEEE-754r in doppia precisione $\varepsilon_M = 2^{-52}$.

Precisione di macchina (2)

- La precisione di macchina definita anteriormente come il massimo errore relativo di arrotondamento, coincide anche con

$$\mathbf{u} = \frac{\varepsilon_M}{2}.$$

- In un computer che usa doppia precisione secondo lo standard IEEE-754r il valore della precisione di macchina è pari a $2^{-53} \approx 1.11 \cdot 10^{-16}$.

Aritmetica di macchina

- Abbiamo visto che il numero x non si rappresenta esattamente nel calcolatore ma si immagazzina come $\mathrm{fl}(x)$ con un errore relativo massimo pari a $u = \frac{1}{2}B^{1-t}$.
- **Domanda:** Come si propagano questi errori di rappresentazione quando si effettuano delle operazioni aritmetiche con i numeri macchina?
- Per distinguere le operazioni aritmetiche in $\mathbb{F}$ eseguite al calcolatore da quelle definite in $\mathbb{R}$ useremo la seguente notazione.
  Dati x, y reali,

$$\begin{aligned}
\text{somma } \oplus \qquad & x \oplus y = \mathrm{fl}\left(\mathrm{fl}(x) + \mathrm{fl}(y)\right) \\
\text{sottrazione } \ominus \qquad & x \ominus y = \mathrm{fl}\left(\mathrm{fl}(x) - \mathrm{fl}(y)\right) \\
\text{prodotto } \otimes \qquad & x \otimes y = \mathrm{fl}\left(\mathrm{fl}(x) \cdot \mathrm{fl}(y)\right) \\
\text{divisione } \oslash \qquad & x \oslash y = \mathrm{fl}\left(\mathrm{fl}(x) / \mathrm{fl}(y)\right)
\end{aligned}$$

Errori nelle operazioni macchina

- Ricordiamo l'errore relativo di rappresentazione: $\varepsilon_x = \dfrac{|x - \mathrm{fl}(x)|}{|x|} \leq \mathbf{u}$.
- Definiamo ora l'errore relativo introdotto dalle operazioni macchina.

$$\varepsilon_{x,y}^{\oplus} = \frac{|(x+y) - (x \oplus y)|}{|x+y|}$$

Analogamente per le altre operazioni.

- Si può dimostrare che

$$\begin{aligned} \varepsilon_{x,y}^{\oplus} &\leq \left|\frac{x}{x+y}\right| \varepsilon_x + \left|\frac{y}{x+y}\right| \varepsilon_y \\ \varepsilon_{x,y}^{\otimes} &\lessapprox \varepsilon_x + \varepsilon_y \\ \varepsilon_{x,y}^{\oslash} &\leq |\varepsilon_x - \varepsilon_y| \end{aligned}$$

dove ε_x e ε_y sono tali che $\varepsilon_x, \varepsilon_y \leq \mathbf{u}$

Errori nelle operazioni macchina: addizione

- Dimostrazione di $\varepsilon_{x,y}^{\oplus} \leq \left|\frac{x}{x+y}\right| \varepsilon_x + \left|\frac{y}{x+y}\right| \varepsilon_y$
- Assumiamo $x \neq 0,\ y \neq 0,\ x+y \neq 0$ e $\mathrm{fl}\left(\mathrm{fl}(x)+\mathrm{fl}(y)\right) = \mathrm{fl}(x)+\mathrm{fl}(y)$ cioè la somma di $\mathrm{fl}(x)$ e $\mathrm{fl}(y)$ sia un numero macchina.
- In caso contrario la dimostrazione è più complicata ma l'enunciato rimane vero.

$$
\begin{aligned}
\varepsilon_{x,y}^{\oplus} &= \frac{|(x+y)-(x\oplus y)|}{|x+y|} = \frac{|(x+y)-\mathrm{fl}\left(\mathrm{fl}(x)+\mathrm{fl}(y)\right)|}{|x+y|} \\
&\leq \frac{|(x-\mathrm{fl}(x))|}{|x+y|} + \frac{|(y-\mathrm{fl}(y))|}{|x+y|} \\
&= \frac{|(x-\mathrm{fl}(x))|\ |x|}{|x+y||x|} + \frac{|(y-\mathrm{fl}(y))|\ |y|}{|x+y||y|} \\
&= \frac{|(x-\mathrm{fl}(x))|}{|x|}\frac{|x|}{|x+y|} + \frac{|(y-\mathrm{fl}(y))|}{|y|}\frac{|y|}{|x+y|} = \varepsilon_x \frac{|x|}{|x+y|} + \varepsilon_y \frac{|y|}{|x+y|}
\end{aligned}
$$

Errori nelle operazioni macchina: prodotto

- Dimostrazione di $\varepsilon_{x,y}^{\otimes} \overset{\leq}{\approx} \varepsilon_x + \varepsilon_y$.
- Assumiamo $x \neq 0,\ y \neq 0,\ x \cdot y \neq 0$ e $\mathsf{fl}\left(\mathsf{fl}(x) \cdot \mathsf{fl}(y)\right) = \mathsf{fl}(x) \cdot \mathsf{fl}(y)$

$$
\begin{aligned}
\varepsilon_{x,y}^{\otimes} &= \frac{|(x \cdot y) - (x \otimes y)|}{|x \cdot y|} = \frac{|(x \cdot y) - \mathsf{fl}\left(\mathsf{fl}(x) \cdot \mathsf{fl}(y)\right)|}{|x \cdot y|} = \frac{|(x \cdot y) - \mathsf{fl}(x) \cdot \mathsf{fl}(y)|}{|x \cdot y|} \\
&= \frac{|x \cdot y - x \cdot \mathsf{fl}(y) + x \cdot \mathsf{fl}(y) - \mathsf{fl}(x)\mathsf{fl}(y)|}{|x \cdot y|} = \\
&= \frac{|x \cdot (y - \mathsf{fl}(y)) + (x - \mathsf{fl}(x)) \cdot \mathsf{fl}(y)|}{|x \cdot y|} = \\
&\leq \frac{|x \cdot (y - \mathsf{fl}(y))|}{|x \cdot y|} + \frac{|(x - \mathsf{fl}(x)) \cdot \mathsf{fl}(y)|}{|x \cdot y|} \ \left(\text{assumendo } \frac{\mathsf{fl}(y)}{y} \approx 1\right) \\
&= \frac{|y - \mathsf{fl}(y)|}{|y|} + \frac{|x - \mathsf{fl}(x)|}{|x|} = \varepsilon_y + \varepsilon_x
\end{aligned}
$$

Errori nelle operazioni macchina

$$
\begin{aligned}
\varepsilon_{x,y}^{\oplus} &\leq \left|\frac{x}{x+y}\right|\varepsilon_x + \left|\frac{y}{x+y}\right|\varepsilon_y \\
\varepsilon_{x,y}^{\otimes} &\lessapprox \varepsilon_x + \varepsilon_y \\
\varepsilon_{x,y}^{\oslash} &\leq |\varepsilon_x - \varepsilon_y|
\end{aligned}
$$

- **Osservazioni:**
  1. Le operazioni macchina prodotto e divisione introducono un errore dell'ordine della precisione di macchina.
  2. Con la somma (sottrazione) non si può garantire che il risultato dell'operazione sia affetto da un errore relativo piccolo. In particolare l'errore per la somma è grande quando $x \approx -y$. Questo fenomeno si chiama **Cancellazione numerica**.

Cancellazione numerica

- È la perdita di cifre significative nel risultato dovuto alla sottrazione di due numeri quasi uguali.

Esempio: In $\mathbb{F}(10,5,*,*)$ consideriamo i numeri $a = 0.73415507$ e $b = 0.73415448$. Naturalmente:

$$\mathrm{fl}(a) = 0.73416, \qquad \mathrm{fl}(b) = 0.73415.$$

- Calcoliamo $a - b = 0.59 \cdot 10^{-6}$ (valore esatto).
- Nell'aritmetica di macchina $\mathrm{fl}(\mathrm{fl}(a) - \mathrm{fl}(b)) = 0.00001 = 10^{-5}$.
- L'errore relativo.
$$\frac{|(a-b) - (a \ominus b)|}{|a-b|} = \frac{0.59 \cdot 10^{-6} - 10^{-5}}{0.59 \cdot 10^{-6}} = \frac{0.941}{0.059} = 15.949 = 1595\% \text{ !!!}$$
- La precisione di macchina in questa aritmetica sarebbe:

$$\mathbf{u} = \frac{1}{2} B^{1-t} = \frac{1}{2} 10^{-4} = 5 \cdot 10^{-5}$$

Cancellazione numerica: esempio 2

- Anche in casi non clamorosi come il precedente, l'errore può comunque essere molto maggiore della precisione di macchina.
- Sia l'aritmetica di macchina $\mathbb{F}(10,6,*,*)$. Consideriamo ora $a = 0.147554326$, $b = 0.147251742$. Naturalmente:

$$\mathrm{fl}(a) = 0.147554, \qquad \mathrm{fl}(b) = 0.147252.$$

- Calcoliamo la differenza $a - b = 0.000302584$ (valore esatto).
- Nell'aritmetica di macchina $a \ominus b = 0.000302$.
- L'errore relativo.

$$\frac{|(a-b)-(a\ominus b)|}{|a-b|} = \frac{0.000302584 - 0.000302}{0.000302584} \approx 1.9 \cdot 10^{-3} \approx 0.19\%.$$

- La precisione di macchina per $\mathbb{F}$: $\mathbf{u} = \frac{1}{2}B^{1-t} = \frac{1}{2}10^{-5} = 5 \cdot 10^{-6}$

Proprietà delle operazioni con numeri reali non valide in $\mathbb{F}$.

Proprietà associativa

In $\mathbb{F}$ $(a \oplus b) \oplus c \neq a \oplus (b \oplus c)$

- Importante: Il risultato di una operazione dipende dall'ordine in cui vengono effettuate le operazioni intermedie!
- Esempio:

$$(1 \oplus 10^{-15}) \ominus 1 = 1.11 \cdot 10^{-15}$$

$$(1 \ominus 1) \oplus 10^{-15} = 10^{-15}$$

- Inoltre l'associatività può essere violata per problemi di overflow o underflow. Esempio: $a = 1.0 \cdot 10^{308} \quad b = 1.1 \cdot 10^{308} \quad c = -1.001 \cdot 10^{308}$

$$a \oplus (b \oplus c) = 1.0 \cdot 10^{308} \oplus (0.99 \cdot 10^{307}) = 1.099 \cdot 10^{308}$$

$$(a \oplus b) \oplus c = \text{Inf} \oplus c = \text{Inf}$$

Cancellazione numerica e stabilità di un algoritmo

Definizione

Un metodo numerico (formula, algoritmo) si dice **stabile** *se non propaga gli errori (inevitabili) dovuti alla rappresentazione dei numeri nel calcolatore. Altrimenti si dice* **instabile**.

- La cancellazione numerica genera delle formule instabili.
- Per evitare i problemi legati alla cancellazione numerica occorre trasformare le formule in altre numericamente più stabili.
- La stabilità è un concetto legato all'algoritmo usato per risolvere un determinato problema.

Cancellazione numerica e stabilità di un algoritmo

Esempio

- La formula

$$\sqrt{x+\delta}-\sqrt{x} \qquad \text{per} \quad \delta \to 0$$

è soggetta a cancellazione numerica. Si può risolvere il problema razionalizzando:

$$\sqrt{x+\delta}-\sqrt{x} \quad \frac{\sqrt{x+\delta}+\sqrt{x}}{\sqrt{x+\delta}+\sqrt{x}} = \frac{\delta}{\sqrt{x+\delta}+\sqrt{x}}$$

- Esempio:

In doppia precisione $\sqrt{1+10^{-14}}-\sqrt{1} = 4.884981308350 69 \cdot 10^{-15}$

con un errore relativo pari a $0.023 = 2.3\%$.
Utilizzando la formula stabile si ottiene

$$\frac{10^{-14}}{\sqrt{1+10^{-14}}+\sqrt{1}} = 4.999999999999 99 \cdot 10^{-15}$$

con un errore relativo pari a $1.58 \cdot 10^{-16}$.

Esempio di algoritmo instabile

Formula risolutiva dell'equazione di secondo grado

Si vuole risolvere l'equazione $ax^2 + bx + c = 0$ con $a \neq 0$.
Dividendo per a:

$$x^2 + \frac{b}{a}x + \frac{c}{a} = 0$$

e ponendo $\frac{b}{a} = 2p$ e $\frac{c}{a} = -q$, possiamo scrivere:

$$x^2 + 2px - q = 0$$

La formula risolutiva dell'equazione di secondo grado calcola le due radice come:

$$x_{1,2} = \frac{-2p \pm \sqrt{4p^2 + 4q}}{2}$$

ovvero

$$x_{1,2} = -p \pm \sqrt{p^2 + q}$$

Esempio di algoritmo instabile

Formula risolutiva dell'equazione di secondo grado

Dato $x^2 + 2px - q$, con $p^2 + q \geq 0$ consideriamo la formula che valuta la radice via:

$$x_1 = -p + \sqrt{p^2 + q}. \tag{1}$$

- $p^2 + q \geq 0$ implica radici reali.
- Questa formula è potenzialmente instabile per $p \gg q > 0$ a causa della sottrazione tra p e $\sqrt{p^2 + q}$ (cancellazione).

Soluzione: Calcolare la radice con un secondo algoritmo stabile via razionalizzazione di (1):

$$\begin{aligned} x_1 &= -p + \sqrt{p^2 + q} = \frac{(-p + \sqrt{p^2 + q})(p + \sqrt{p^2 + q})}{(p + \sqrt{p^2 + q})} \\ &= \frac{q}{(p + \sqrt{p^2 + q})} \end{aligned} \tag{2}$$

Esempio

- Si vuole risolvere l'equazione $x^2 - \mathbf{2p}x + 10^{-2} = 0$ per $\mathbf{p} = 10^4, 10^5, 10^6, 10^7, 10^8$.

- Usando la formula risolutiva dell'equazione di secondo grado si ottiene (in doppia precisione usando Matlab):

$\mathbf{p}$	x_1	x_1 vera	ERR. REL
$1.0e+04$	$5.0000016927e-07$	$4.9999999999e-07$	$3.4e-07$
$1.0e+05$	$5.0000380725e-08$	$5.0000000000e-08$	$7.6e-06$
$1.0e+06$	$5.0058588386e-09$	$5.0000000000e-09$	$1.2e-03$
$1.0e+07$	$0.0000000000e+00$	$5.0000000000e-10$	$1.0e+00$
$1.0e+08$	$0.0000000000e+00$	$5.0000000000e-11$	$1.0e+00$

- Le soluzioni corrette alla precisione di macchina sono state calcolate con il secondo algoritmo (2).

Esempio di algoritmo instabile

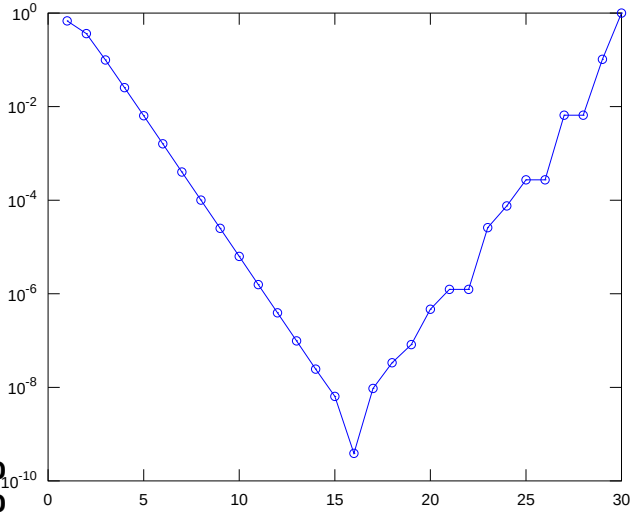
Successione approssimante π

Se si vuole approssimare il valore di π con la seguente formula ricorsiva

$$\begin{aligned} z_2 &= 2 \\ z_{n+1} &= 2^{n-0.5}\sqrt{1-\sqrt{1-4^{1-n}z_n^2}}, \quad n = 2,3,\ldots, \end{aligned}$$

si ottiene la seguente successione di valori (dove si è posto $r = 4^{1-n}z_n^2$).

$n+1$	r	$1-\sqrt{1-r}$	z_{n+1}	$\frac{\lvert z_{n+1}-\pi\rvert}{\pi}$
...	...	...	...	...
10	1.505e-04	7.529e-05	3.14157294036	6.27e-06
11	3.764e-05	1.882e-05	3.14158772527	1.57e-06
12	9.412e-06	4.706e-06	3.14159142150	3.92e-07
13	2.353e-06	1.176e-06	3.14159234561	9.80e-08
14	5.882e-07	2.941e-07	3.14159257654	2.45e-08
15	1.470e-07	7.353e-08	3.14159263346	6.41e-09
16	3.676e-08	1.838e-08	3.14159265480	3.88e-10
17	9.191e-09	4.595e-09	3.14159264532	2.63e-09
18	2.297e-09	1.148e-09	3.14159260737	1.47e-08
19	5.744e-10	2.872e-10	3.14159291093	8.19e-08
...	...	...	...	...
28	2.220e-15	1.110e-15	3.16227766016	6.58e-03
29	5.551e-16	3.330e-16	3.46410161513	1.03e-01
30	1.665e-16	1.110e-16	4.00000000000	2.73e-01
31	5.551e-17	0.000e+00	0.00000000000	1.00e+00
32	0.000e+00	0.000e+00	0.00000000000	1.00e+00



Esempio di algoritmo instabile

Successione ricorrente

Si vuole calcolare la seguente successione di integrali definiti:

$$I_n = \frac{1}{e}\int_0^1 x^n e^x\,dx \qquad (n \geq 0)$$

$$\text{dove} \qquad I_0 = \frac{1}{e}\int_0^1 e^x\,dx = 1 - \frac{1}{e} \approx 0.632121.$$

Per $n \geq 1$, integrando per parti, si ottiene

$$I_n = \frac{1}{e}\left\{[x^n e^x]_0^1 - n\int_0^1 x^{n-1} e^x\,dx\right\} = 1 - nI_{n-1},$$

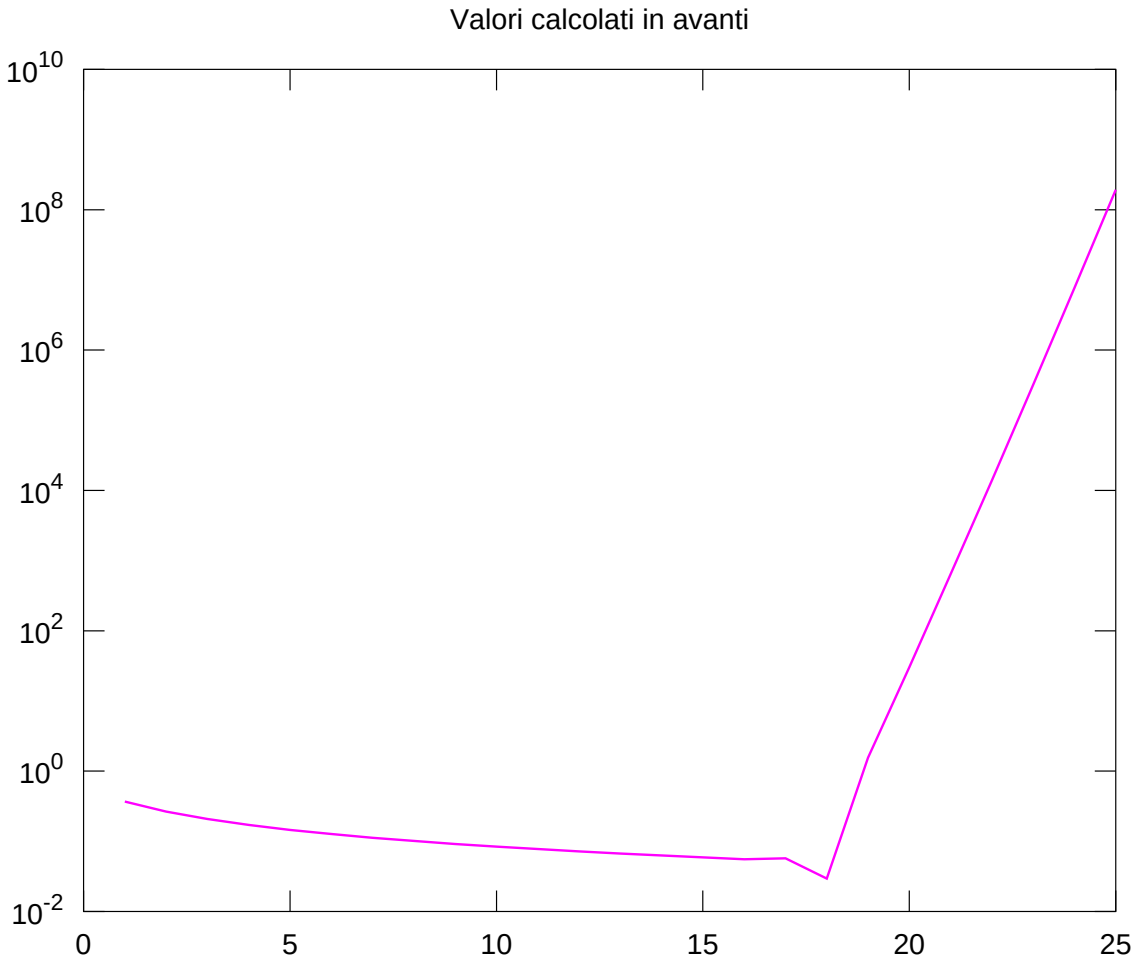
e quindi la formula ricorsiva:

$$\text{fissato } I_0, \qquad I_n = 1 - n\,I_{n-1} \qquad n \geq 1.$$

Si noti che $0 < I_n < 1$.

Instabilità della formula ricorsiva

Implementando tale formula per il calcolo di I_n, per $n = 2, \cdots, 25$, si ottengono i valori plottati in modulo nel seguente grafico:



Instabilità della formula ricorsiva

La formula $I_n = 1 - n\, I_{n-1}$ è instabile, quindi amplifica l'errore ad ogni passo.

Infatti, nel calcolatore

$$(I_n + \varepsilon_n) = 1 - n(I_{n-1} + \varepsilon_{n-1}).$$

Sottraendo dalla precedente equazione la relazione $I_n = 1 - nI_{n-1}$ si può quantificare l'errore:

$$\varepsilon_n = -n\, \varepsilon_{n-1}, \quad \text{e per induzione} \quad |\varepsilon_n| = n!\, |\varepsilon_{0.}|$$

Il fattore $n!$ amplifica l'errore di rappresentazione iniziale (su I_0), ε_0.

- Esempio. Nel calcolo di I_{20} l'errore è $\varepsilon_{20} = 20!\, \varepsilon_0 \approx 2.7 \cdot 10^2$.

Formula alternativa stabile

Successione ricorrente all'indietro

Il valore di I_{m-k} si può calcolare a partire da una approssimazione di I_m mediante questa formula all'indietro

$$I_{n-1} = \frac{1}{n}(1 - I_n), \qquad n = m, m-1, \ldots, m-k+1 \ .$$

che si ottiene dalla precedente ricavando I_{n-1} in funzione di I_n.

Questa formula è stabile perché l'errore ad ogni passo diminuisce anziché aumentare.

Formula alternativa stabile

La formula all'indietro smorza l'errore

Per l'errore al passo $n-1$ si trova

$$\varepsilon_{n-1} = \frac{-1}{n}\,\varepsilon_n.$$

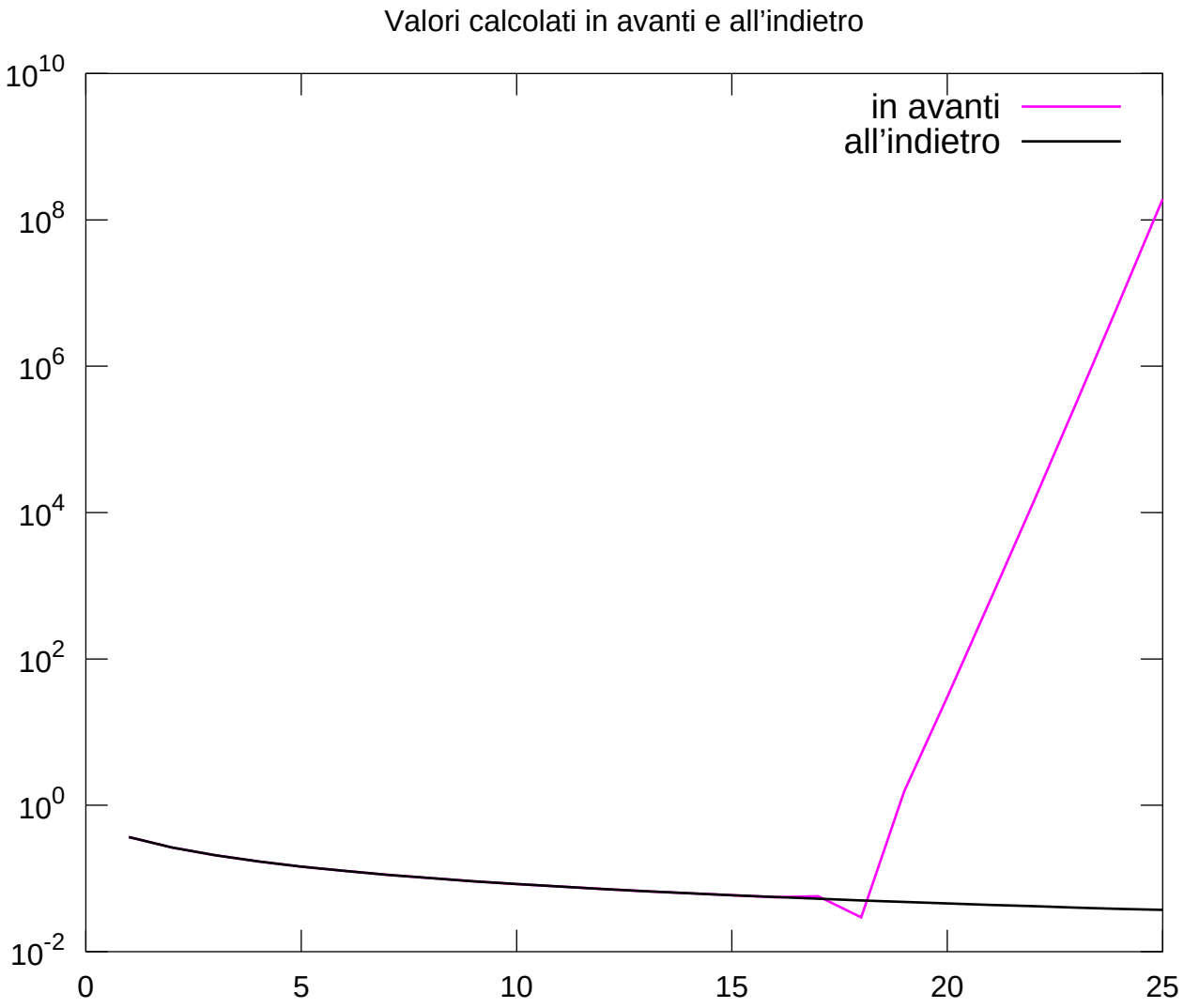
Partendo da m

$$|\varepsilon_{m-1}| = \frac{|\varepsilon_m|}{m}, \quad |\varepsilon_{m-2}| = \frac{|\varepsilon_m|}{m(m-1)}, \ \ldots, \ |\varepsilon_{m-k}| = \frac{|\varepsilon_m|}{m(m-1)\cdots(m-k+1)}.$$

- La produttoria al denominatore abbatte rapidamente l'errore iniziale!
- Per esempio, per calcolare I_{25} partendo da $I_{40} = 0.5$, l'errore iniziale $|\varepsilon_{40}| < 0.5$ verrebbe abbattuto di un fattore

$$40 \cdot 39 \cdots 27 \cdot 26 = 5.2602 \cdot 10^{22}.$$

Confronto tra le formule



Condizionamento di un problema

Definizione

Un problema si dice mal condizionato se a piccole variazione nei dati corrispondono grandi variazioni nei risultati.

- In caso contrario il problema si dice ben condizionato.
- Il malcondizionamento è indipendente dall'algoritmo scelto per risolvere il problema: se il problema è mal condizionato nessun algoritmo, per quanto stabile, potrà dare una soluzione corretta al problema stesso.

Condizionamento di un problema

Esempio

Consideriamo il problema di risolvere il sistema lineare

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

che ha come soluzione $x=1,\ y=1$.

Se perturbiamo il coefficiente della x nella prima equazione di 0.01 si ottiene

$$\begin{cases} 1.01x+y & = & 2 \\ 1001x+1000y & = & 2001 \end{cases}$$

Il nuovo sistema ha come soluzione $x=-0.11111111,\ y=2.11222222$ con un errore relativo su x e su y pari a

$$err_x = \frac{1+0.11111111}{1} = 1.11111111, \qquad err_y = \frac{|1-2.11222222|}{1} = 1.1122222222$$

entrambi maggiori del 100%.

Numero di condizionamento di un problema

- La quantità che misura il grado di sensibilità di un problema rispetto a piccole variazioni nei dati si dice Numero di condizionamento.
- Consideriamo il problema di valutare una funzione di una variabile f in un punto: $y = f(x)$.
- Se il dato di ingresso x è perturbato di una quantità Δx, assumendo f derivabile, per il teorema di Lagrange

$$f(x+\Delta x) - f(x) = \Delta x f'(\xi).$$

Detto $\Delta y = f(x+\Delta x) - f(x)$ l'errore assoluto sul valore della funzione, quello relativo è

$$\left|\frac{\Delta y}{y}\right| = \left|\frac{\Delta x f'(\xi)}{y}\right| = \frac{|\Delta x|}{|x|}\frac{|x \cdot f'(\xi)|}{|y|}$$

Al limite, per $\Delta x \to 0$, $\xi \to x$.

$$\left|\frac{\Delta y}{y}\right| = K(f,x)\frac{|\Delta x|}{|x|}$$

dove $K(f,x) = \dfrac{|x \cdot f'(x)|}{|y|}$ è detto Numero di condizionamento.

Numero di condizionamento di un problema

Esempio

Data la funzione

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

calcoliamo analiticamente il numero di condizionamento

$$K(f,x) = \frac{|x \cdot f'(x)|}{|y|} = \frac{\left| x \cdot \frac{-2x}{2\sqrt{1-x^2}} \right|}{|\sqrt{1-x^2}|} = \frac{x^2}{1-x^2}.$$

Il problema sarà peggio condizionato quanto più x si approssima a 1:

x	$1-10^{-6}$	$1-10^{-12}$	$1-10^{-15}$
$K(f,x)$	$4.99999 \cdot 10^{+05}$	$5.00011 \cdot 10^{+11}$	$5.0039996 \cdot 10^{+14}$