1 Lezione del 28-02-25

1.1 Operazioni sui numeri macchina

Abbiamo introdotto l'insieme dei numeri macchina. Vediamo adesso come eseguire **operazioni** fra elementi di questi insiemi.

Notiamo che, di base, dati $x, y \in F(\beta, m, L, U)$, non necessariamente $x \circ y \in F(\beta, m, L, U)$ per le comuni operazioni aritmetiche $+, -, \times, \div$.

Quello che faremo è quindi approssimare tali operazioni, cioè dire:

- $x \oplus y = Rn(x + y)$
- $x \ominus y = Rn(x - y)$
- $x \otimes y = Rn(x \times y)$
- $x \oslash y = Rn(x \div y)$

Effetto di questa approsimazione è negare proprietà note dei reali, come ad esempio l'associativa:

$$(x \oplus y) \oplus z \neq x \oplus (y \oplus z)$$

$$(x \oplus y) \otimes z \neq x \oplus (y \otimes z)$$

Cioè, la valutazione di una formula con ordini diversi ma equivalenti in aritmetica esatta può portare a risultati differenti nell'insieme dei numeri di mmacchina.

1.1.1 Errore nel calcolo di funzione

Sia $f : \mathbb{R}^m \to \mathbb{R}$, e si voglia calcolare $f(P_0)$, con $P_0 = \left(x_1^{(0)}, x_2^{(0)}, ...x_m^{(0)}\right) \in \mathbb{R}^m$. Le operazioni aritmetiche $+, -, \times, \div$ possono essere viste come funzioni di questo tipo. Ci interroghiamo quindi sulla fonte dell'errore nella loro valutazione:

1. Nel caso contenga funzioni irrazionali o trascendenti, f verrà approssimata con una funzione $\overline{f}$ che coinvolge solo operazioni aritmetiche di base $+, -, \times, \div$;
2. $\overline{f}$ viene tradotta in un *algoritmo* $\overline{f}_a$, ovvero in una formula che coinvolge $\oplus, \ominus, \otimes, \oslash \leftarrow +, -, \times, \div$. La formula ottenuta finora viene detta **algoritmo**;
3. Potrebbe essere che $P_0 \neq F(\beta, m, L, U)$, e quindi viene approssimato a $P_1 = Rn(P_0)$.

Quindi, vogliamo $f(P_0)$ ma possiamo solo approssimarla come $\overline{f}_a(P_1)$.

Ad esempio, poniamo di voler calcolare e^π. I passaggi nell'ordine appena visto saranno:

1. Approssimamo l'esponenziale al secondo grado dello svillupppo di Taylor:

$$e^x \approx 1 + x + \frac{x^2}{2} = \overline{f}(x)$$

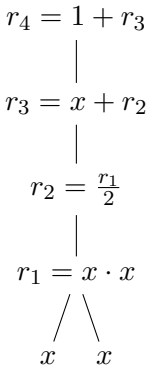
2. Si riporta la $\overline{f}(x)$ a $\overline{f}_a(x)$:

$$1 \oplus (x \oplus ((x \otimes x) \oslash 2))$$

Indicheremo algoritmi di questo tipo anche attraverso **risultati intermedi**:

- $r_1 = x \cdot x$
- $r_2 = \frac{r_1}{2}$
- $r_3 = x + r_2$
- $r_4 = 1 + r_3$

con il risultato finale inteso come l'ultimo risultato intermedio. Un modo di visualizzare i risultati intermedi di un algoritmo può essere un albero:



dove la *radice* rappresenta il **risultato** e le *foglie* rappresentano le variabili della funzione.

3. Si approssima π al numero macchina più vicino:

$$Rn(\pi) = 3.1415 = P_1$$

Avremo quindi la formula finale:

$$1 \oplus (P_1 \oplus ((P_1 \otimes P_1) \oslash 2))$$

Diamo quindi la definizione di **errore assoluto**:

Definizione 1.1: Errore assoluto

Data $f : \mathbb{R}^m \to \mathbb{R}$, un punto $P_0 \in \mathbb{R}^m$ ed un algoritmo $\overline{f}_a$, l'errore assoluto è dato da:

$$\sigma_f = \overline{f}_a(P_1) - f(P_0), \quad P_1 = Rn(P_0)$$

e di errore relativo:

Definizione 1.2: Errore relativo

Date le ipotesi della definizione 2.1, l'errore relativo è dato da:

$$\epsilon_f = \frac{\overline{f}_a(P_1) - f(P_0)}{f(P_0)} = \frac{\sigma_f}{f(P_0)}$$

1.1.2 Errore di funzioni razionali

Assumiamo per adesso f **funzione razionale**, ovvero f si può definire con un numero di operazioni in $+, -, \times, \div$. Assumere funzioni razionali ci permette di prendere $f = \overline{f}$ e $f_a = \overline{f}_a$ (non ci sono irrazionali da riportare ai razionali). Potremo allora dire:

$$\sigma_f = f_a(P_1) - f(P_0) = f_a(P_1) - f(P_1) + f(P1) - f(P_0)$$

che possiamo dividere in:

$$\sigma_f = \sigma_a + \sigma_d, \quad \sigma_a = f_a(P_1) - f(P_1), \quad \sigma_d = f(P_1) - f(P_0)$$

che chiamiamo rispettivamente **errore assoluto algoritmico** e **errore assoluto inerente**.

Allo stesso modo possiamo definire l'**errore relativo**:

$$\epsilon_f = \frac{\sigma_f}{f(P_0)} = \frac{f_a(P_1) - f(P_0)}{f(P_0)} = \frac{f_a(P_1) - f(P_1)}{f(P_0)} + \frac{f(P_1) - f(P_0)}{f(P_0)}$$

$$= \frac{f_a(P_1) - f(P_1)}{f(P_1)} \cdot \frac{f(P_1)}{f(P_0)} + \frac{f(P_1) - f(P_0)}{f(P_0)}$$

che si divide nuovamente in :

$$\epsilon_f = \epsilon_a + \epsilon_d, \quad \epsilon_a = \frac{f_a(P_1) - f(P_1)}{f(P_1)}, \quad \epsilon_d = \frac{f(P_1) - f(P_0)}{f(P_0)}$$

che chiamiamo rispetivamente **errore relativo algoritmico** e **errore relativo inerente**. Questo viene da:

$$\epsilon_f = [...] = \frac{f_a(P_1) - f(P_1)}{f(P_1)} \cdot \frac{f(P_1)}{f(P_0)} + \frac{f(P_1) - f(P_0)}{f(P_0)}$$

si nota che $\frac{f(P_1)}{f(P_0)} = 1 + \frac{f(P_1) - f(P_0)}{f(P_0)}$, e quindi:

$$= \epsilon_a \cdot \left(1 + \frac{f(P_1) - f(P_0)}{f(P_0)}\right) + \epsilon_d = \epsilon_a(1 + \epsilon_d) + \epsilon_d = \epsilon_a + \epsilon_d + \epsilon_a\epsilon_d \approx \epsilon_a + \epsilon_d$$

assumendo $\epsilon_a\epsilon_d \approx 0$.

Ci interessa dare stime superiori per i valori assoluti di errori assoluti e relativi, come avevamo fatto per gli errori delle funzioni di arrotondamento da reali a numeri macchina. In generale, quindi, per limitare $|\sigma_f|$ cercheremo disuguaglianze $|\sigma_a| < \tau_1$, $|\sigma_d| < \tau_2$, da cui:

$$|\sigma_f| < \tau_1 + \tau_2$$

1.1.3 Stima dell'errore inerente

Avevamo quindi definito l'errore assoluto inerente:

$$\sigma_d = f(P_1) - f(P_0)$$

Sotto l'ipotesi $f \in C^1(D)$ per $D \subset \mathbb{R}^m$ che contiene P_0, si può usare lo sviluppo di Taylor di f in P_0, troncato al primo ordine:

$$f(P_1) - f(P_0) = f(P_0) + \nabla f(\overline{P})^T(P_1 - P_0) - f(P_0) = \nabla f(\overline{P})^T(P_1 - P_0)$$

$$= \sum_{j=1}^{m} \frac{\partial f}{\partial x_j}(\overline{P}) \cdot \left(x_j^{(1)} - x_j^{(0)}\right) \approx \sum_{j=1}^{m} \frac{\partial f}{\partial x_j} P_0 \cdot \left(x_j^{(1)} - x_j^{(0)}\right)$$

dove $\overline{P}$ è un punto che sta sul segmento $\overline{P_1 P_0}$. Da questo potremo dire:

$$\sigma_d = \sum_{j=1}^{m} \frac{\partial f}{\partial x_j} P_0 \cdot \sigma_j$$

dove $\sigma_j = \left(x_j^{(1)} - x_j^{(0)}\right)$ è l'**errore di arrotondemento** nella componente j di P_0, e $\frac{\partial f}{\partial x_j} P_0$ viene detto **coefficiente di amplificazione**.

Per l'errore relativo inerente, che avevamo definito come:

$$\epsilon_d = \frac{f(P_1) - f(P_0)}{f(P_0)}$$

potremo fare considerazioni simili:

$$\epsilon_d = \frac{\sum_{j=1}^{m} \frac{\partial f}{\partial x_j}(P_0) \cdot \sigma_j}{f(P_0)} = \sum_{j=1}^{m} \frac{x_j^{(1)} - x_j^{(0)}}{x_j^{(0)}} \cdot \frac{\partial f}{\partial x_j}(P_0) \cdot \frac{x_j^{(0)}}{f(P_0)}$$

dove $\epsilon_j = \frac{x_j^{(1)} - x_j^{(0)}}{x_j^{(0)}}$ sarà l'**errore di arrotondamento relativo** nella componente j di P_0 e $P_j = \frac{\partial f}{\partial x_j}(P_0) \cdot \frac{x_j^{(0)}}{f(P_0)}$ viene detto **coefficiente di amplificazione dell'errore relativo**.

La formula finale sarà quindi:

$$\epsilon_d = \sum_{j=1}^{m} \epsilon_j P_j$$

I problemi in cui si devono calcolare f i cui coefficienti di amplificazione degli errori relativi sono grandi in modulo (o ce n'è almeno uno sufficientemente grande) si dicono **malcondizionati**. Viceversa, se $|P_j|$ è vicino a 1 per ogni j il problema si dice **ben condizionato**, cioè che ϵ_d è di un ordine di grandezza comparabile a $\max(\epsilon_i)$

Notiamo inoltre che il condizionamento di un problema dipende solamente dalla sua struttura matematica.

1.1.4 Errori inerenti delle operazioni aritmetiche

Vediamo gli errori inerenti associati alle 4 operazioni aritmetiche $+, -, \times, \div$:

Operazione	σ_d	ϵ_d
$x \oplus y$	$\sigma_x + \sigma_y$	$\frac{x}{x+y}\epsilon_x + \frac{y}{x+y}\epsilon_y$
$x \ominus y$	$\sigma_x - \sigma_y$	$\frac{x}{x-y}\epsilon_x - \frac{y}{x-y}\epsilon_y$
$x \otimes y$	$y\sigma_x + x\sigma_y$	$\epsilon_x + \epsilon_y$
$x \oslash y$	$\frac{1}{y}\sigma_x - \frac{x}{y^2}\sigma_y$	$\epsilon_x - \epsilon_y$

Notiamo come somme e sottrazioni non amplificano gli errori totali, mentre prodotti e divisioni non amplificano gli errori relativi (riguardo agli errori inerenti). Questo significa che somme e sottrazioni possono avere errori relativi grandi quando $|x + y| << \min\{|x|, |y|\}$. Questo effetto viene detto **cancellazione numerica**.

1.1.5 Stima dell'errore algoritmico

Avevamo definito un algoritmo $f_a(x)$ di cui vogliamo stimare l'errore algoritmico assoluto $\sigma_a = f_a(P_1) - f(P_1)$. Assumiamo $P_1 = Rn(P_0) \in F(\beta, m, L, U)$, cioè gli operandi come privi di errori iniziali di arrotondamento.

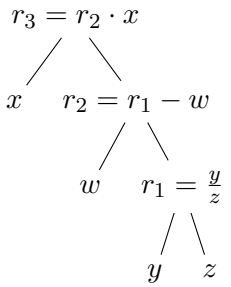
L'idea è di seguire l'errore generato dall'algoritmo sul grafo (o albero) che lo rappresenta sfruttando le relazioni per l'errore inerente nelle 4 operazioni aritmetiche. Prendiamo ad esempio la funzione:

$$f(x, y, z, w) = x \cdot \left(\frac{y}{z} - w \right)$$

Avremo i risultati intermedi:

- $r_1 = \frac{y}{z}$
- $r_2 = r_1 - w$
- $r_3 = r_2 \cdot x$

Di cui riportiamo il grafo:



dove ϵ_3, ϵ_2 e ϵ_1 saranno termini associati ad ogni risultato intermedio che rappresenteranno gli errori di troncamento dei risultati intermedi stessi, e ϵ_{r3}, ϵ_{r2} e ϵ_{r1} rappresenteranno gli errori inerenti delle singole operazioni per il calcolo dei risultati intermedi.

Partiamo dalla radice per valutare gli errori:

$$\epsilon_d = \epsilon_{r_3} = \epsilon_3 + \epsilon_x + \epsilon_{r_2} = \epsilon_3 + \epsilon_{r_2}$$

$$= \epsilon_3 + \epsilon_2 + \frac{-zw}{y - zw} \cdot \epsilon_w + \frac{y}{y - zw} \cdot \epsilon_{r_1} = \epsilon_3 + \epsilon_2 + \frac{y}{y - zw} \cdot \epsilon_{r_1}$$

$$= \epsilon_3 + \epsilon_2 + \frac{y}{y - zw} (\epsilon_1 + \epsilon_y - \epsilon_z) = \epsilon_3 + \epsilon_2 + \frac{y}{y - zw} \cdot \epsilon_1 = \epsilon_a$$

Dove abbiamo ignorato i termini di errore relativo ϵ_i legati ad ogni variabile (come avevamo detto sopra, gli operandi sono considerati come privi di errore di arrotondamento). Per la stima di ϵ_3, ϵ_2 e ϵ_1, vale $\epsilon_i \leq U$ precisione macchina. Nel caso di errori assoluti vale $|\sigma_i| \leq U \cdot \max(x_i)$ considerata ogni variabile x_i.

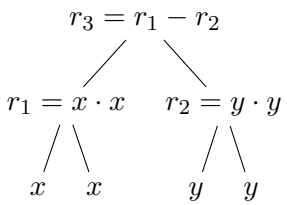
Chiaramente, diversi algoritmi equivalenti in aritmetica esatta potranno avere errori algoritmici diversi fatte tutte le approssimazioni.

Prendiamo ad esempio la funzione $f(x, y) = x^2 - y^2$. Potremmo adottare due algoritmi:

1. Si prendono i risultati intermedi:

$$r_1 = x \cdot x$$
$$r_2 = y \cdot y$$
$$r_3 = x - y$$

Da cui il grafo:



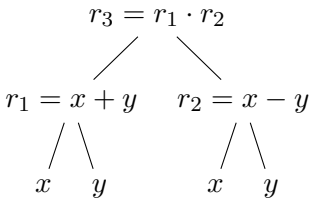
Questo approccio potrebbe risultare il più intuitivo: dalla stima dell'errore si ha:

$$\epsilon_{r_3} = \epsilon_1 + \frac{x^2}{x^2-y^2}\epsilon_{r_1} - \frac{y^2}{x^2-y^2}\epsilon_{r_2} = \epsilon_1 + \frac{x^2}{x^2-y^2}(\epsilon_1+\epsilon_x+\epsilon_x) - \frac{y^2}{x^2-y^2}(\epsilon_2+\epsilon_y+\epsilon_y)$$
$$= \epsilon_1 + \frac{x^2}{x^2-y^2}\epsilon_1 - \frac{y^2}{x^2-y^2}\epsilon_2$$

2. Un altro approccio è quello di prendere i risultati intermedi:

$$r_1 = x + y$$
$$r_2 = x - y$$
$$r_3 = r_1 \cdot r_2$$

Da cui il grafo:



Vediamo la stima dell'errore:

$$\epsilon_{r_3} = \epsilon_3 + \epsilon_{r_1} + \epsilon_{r_2} = \epsilon_3 + \epsilon_1 + \frac{x}{x+y}\epsilon_x + \frac{y}{x+y}\epsilon_y + \epsilon_2 + \frac{x}{x-y}\epsilon_x - \frac{y}{x-y}\epsilon_y$$
$$= \epsilon_3 + \epsilon_1 + \epsilon_2$$

Notiamo che la stima dell'errore del secondo approccio è più conveniente, in quanto più strettamente limitata al di sotto di un valore fisso: $\epsilon_1 + \epsilon_2 + \epsilon_3 = 3U$, contro il $\left(1 + \frac{|x^2+y^2|}{|x^2-y^2|}\right) U$ del primo approccio.

Abbiamo visto quindi tenciche per la stima di ϵ_a e ϵ_d (σ_a e σ_d), che ci permettono di calcolare $|\epsilon_f| \leq |\epsilon_a| + |\epsilon_d|$ ($|\sigma_f| \leq |\sigma_a| + |\sigma_d|$).

Un problema classico sarà quello di, data f, un algoritmo risolutivo f_a e una stima degli errori d_{x_i}, di stimare σ_f per $P_0 \in D \subset \mathbb{R}^m$.

Il problema inverso potrebbe essere quello di, dato $\tau > 0$, f e un punto $P_0 \in \mathbb{R}^n$, determinare un algoritmo ed un valore di precisione macchina U tali per cui $|s_f| < \tau$.