Idee su errore
e precisione arbitraria

`github.com/nrizzo`

14 settembre 2018

1 Introduzione

Nello studio del calcolo della variante di Ackermann su insiemi ereditariamente finiti definita come

$$\mathbb{R}_A(x) = \sum_{y \in x} 2^{-\mathbb{R}_A(y)},$$

sorgono problemi di analisi numerica, tra i quali come modellare l'incertezza su numeri reali e la rappresentazione di un calcolatore di numeri razionali a precisione arbitraria.

2 Considerazioni su errore e precisione

Definizione 1 (Approssimazione per difetto e per eccesso)**.** Dati $x, \tilde{x} \in \mathbb{R}$, diciamo che $\tilde{x}$ approssima per difetto x se e solo se, per definizione,

$$\tilde{x} = x - \varepsilon_x, \quad \text{con} \quad \varepsilon_x \geq 0;$$

allo stesso modo diciamo che $\tilde{x}$ approssima per eccesso x se e solo se, per definizione,

$$\tilde{x} = x + \varepsilon_x, \quad \text{con} \quad \varepsilon_x \geq 0.$$

Inoltre ci riferiamo a $\pm\varepsilon_x$ come l'errore compiuto nell'approssimazione di x.

Esempio. Dati $u = (u_{n-1} \dots u_1 u_0)_b$ e $v = (v_{m-1} \dots v_1 v_0)_b$, supponiamo di voler trovare un'approssimazione del risultato dell'operazione aritmetica generica op su u e v, in simboli

$$w = (w_{q-1} \dots w_1 w_0, w_{-1} \dots)_b = u \operatorname{op} v.$$

Fissata la precisione desiderata $p \in \mathbb{N}$, ci viene data la procedura `op_trunc` che restituisce

$$\tilde{w} = (w_{q-1} \dots w_1 w_0, w_{-1} \dots w_{-p})_b = w - \underbrace{\sum_{i<-p} \left(w_i b^i\right)}_{=\varepsilon_w}.$$

cioè un'approssimazione per difetto di w per cui vale che $0 \leq \varepsilon_w < 1 \cdot b^{-p}$. In modo simmetrico la procedura `op_round` approssima w per eccesso in modo da commettere un errore minore di b^{-p}.

Definizione 2 (Funzioni approssimate)**.** Data $f\colon \mathbb{Q} \to \mathbb{R}$, $p \in \mathbb{N}$, $\tilde{f}\colon \mathbb{Q} \to \mathbb{Q}$, diciamo che $\tilde{f}$ approssima per difetto f con precisione p se e solo se, per definizione,

$$\tilde{f}(x) = f(x) - \varepsilon_f, \qquad 0 \leq \varepsilon_f < b^{-p},$$

mentre diciamo che $\tilde{f}$ approssima per eccesso f con precisione p se e solo se, per definizione,

$$\tilde{f}(x) = f(x) + \varepsilon_f, \qquad 0 \leq \varepsilon_f < b^{-p}.$$

Poiché i nostri scopi sono pratici, di ogni funzione aritmetica possiamo avere o costruire la funzione approssimata. Alcune implementazioni utili sono:

- `trunc` e `round`, che dati un razionale $u = (u_{n-1} \dots u_1 u_0)_b$ e $p \in \mathbb{N}$, approssimano u (o la funzione di uguaglianza $f(x) = x$) per eccesso o per difetto alla precisione p;
- `div_trunc` e `div_round`, dati $u = (u_{n-1} \dots u_1 u_0)_b$, $v = (v_0)_b$ e la precisione $p \in \mathbb{N}$, restituiscono l'approssimazione di u/v rispettivamente per difetto e per eccesso, con precisione p.

Definizione 3. Diciamo che x è approssimato dall'intervallo (topologicamente) chiuso $X = [a, b]$, con $a, b \in \mathbb{Q}$, se e solo se, per definizione, $x \in X$.

Nella teoria di Hansen [1], X è chiamato un numero intervallare, i valori precisi sono intervalli degeneri e viene notato che gli estremi possono non essere rappresentabili da un numero di macchina: in tal caso si effettua *outward rounding*, cioè l'approssimazione per difetto dell'estremo sinistro con il più grande numero di macchina più piccolo dell'estremo stesso, e l'approssimazione per eccesso dell'estremo destro con il più piccolo numero di macchina più grande di esso.

La definizione di intervalli che contengono i valori esatti semplifica il calcolo e lo studio dell'errore, poiché calcolando operazioni aritmetiche sugli estremi si maneggiano dati non affetti da errore.

Teorema 1. Dato $x \in X = [a, b]$, con $a, b \in \mathbb{R}$, il prefisso in comune delle rappresentazioni binaria dei due estremi (se esiste) è anche prefisso di x.

Teorema 2. Dato $x \in X$ e dato $Y \supseteq X$, il prefisso in comune tra gli estremi di Y (se esiste) è anche prefisso di x e degli estremi di X.

3 Principi di aritmerica intervallare

Dati $X = [a, b]$ e $Y = [c, d]$, se $\bullet$ denota una delle operazioni di addizione, sottrazione, moltiplicazione e divisione, allora il risultato dell'operazione corrispondente applicata a X e Y è

$$X \bullet Y = \{x \bullet y \mid x \in X, y \in Y\}.$$

3.1 Addizione e sottrazione

$$X + Y = [a + c, b + d]$$
$$X - Y = [a - d, b - c]$$

3.2 Immagine di funzione continua monotona

$$f(X) = \begin{cases} [f(a), f(b)] & \text{se } f \text{ è non decrescente} \\ [f(b), f(a)] & \text{se } f \text{ è non crescente} \end{cases}$$

4 Approssimazione di 2^{-X}

In questa sezione studiamo lo schema per un possibile algoritmo di approssimazione, da parte di un calcolatore, del calcolo della potenza di due con un esponente reale negativo approssimato da un intervallo di razionali. Supponiamo inoltre di lavorare con numeri razionali in base 2 o una sua potenza. Lo scopo finale è trovare un'approssimazione di 2^{-X} che sia ottenibile con semplici operazioni aritmetiche e di approssimazione da parte di un calcolatore.

4.1 Parte intera e MacLaurin per la parte frazionaria

Per i principi di aritmetica intervallare mostrati, e perché $f(x) = 2^{-x}$ è una funzione monotona decrescente,

$$2^{-X} = 2^{-[x_{\min}, x_{\max}]} = [2^{-x_{\max}}, 2^{-x_{\min}}], \qquad x_{\min}, x_{\max} \geq 0,$$

quindi un algoritmo che approssimi 2^{-X} deve approssimare $2^{-x_{\max}}$ e $2^{-x_{\min}}$ rispettivamente per difetto e per eccesso (*outward rounding*). D'ora in poi ci riferiremo a $x_{\max}$ e a $x_{\min}$ con x, sapendo di dover approssimare per difetto in un caso e per difetto nell'altro.

Avendo a disposizione funzioni aritmetiche elementari, si è scelto di approssimare l'elevamento a potenza con il suo polinomio di MacLaurin di grado arbitrario (si veda [2, p. 227]): dati $f(z) = 2^{-z}$ e $n \in \mathbb{N}$, allora $f(z) = T_n(z) + R_n(z)$ con

$$T_n(z) = \sum_{i=0}^{n} \left((-1)^i \cdot \frac{(z \log 2)^i}{(i)!} \right), \qquad R_n(z) = \sum_{i>n} \left((-1)^i \cdot \frac{(z \log 2)^i}{(i)!} \right);$$

poiché come centro del polinomio si è scelto lo 0, è utile che x sia piccolo e ciò si può ottenere separando prima l'estremo x nella sua parte intera $\lfloor x \rfloor$ e parte frazionaria $\{x\}$. Allora

$$\begin{aligned} 2^{-x} &= 2^{-\lfloor x \rfloor} \cdot 2^{-\{x\}} \\ &= 2^{-\lfloor x \rfloor} \cdot \big(T_n(\{x\}) + R_n(\{x\})\big) \\ &= 2^{-\lfloor x \rfloor} \cdot T_n(\{x\}) + 2^{-\lfloor x \rfloor} \cdot R_n(\{x\}); \end{aligned} \tag{1}$$

così facendo il problema non si complica, perché l'elevamento a potenza di 2 con esponente un intero è un'operazione semplice da eseguire con numeri in base $b = 2^k$ per qualche intero k.

L'equazione (1) è equivalente a

$$2^{-\lfloor x \rfloor} \cdot T_n(\{x\}) = 2^{-x} - 2^{-\lfloor x \rfloor} \cdot R_n(\{x\}), \tag{2}$$

cioè $2^{-\lfloor x \rfloor} \cdot T_n(\{x\})$ è un'approssimazione di 2^{-x}. Il segno del resto del polinomio, poiché la serie di Taylor converge alternatamente, dipende dalla parità di n: se

n è pari, $R_n(\{x\})$ è negativo ed è un'approssimazione per eccesso, se n è dispari allora è un'approssimazione per difetto.

$T_n(\{x\})$ purtroppo non può essere esattamente calcolato per due principali motivi:

- contiene potenze di $\{x\}\log 2$, valore che deve essere approssimato se $\{x\}$ è maggiore di 0, poiché il logaritmo naturale di 2 è trascendente;
- contiene potenze e divisioni per intero e, anche se le prime potrebbero essere rappresentate non affette da errore a discapito delle prestazioni, il risultato delle seconde può non essere rappresentabile con un numero finito di cifre in base b.

Allora è utile definire il polinomio di MacLaurin $T'(z)$ per $f'(z) = e^{-z}$ ed effettuare quello che sostanzialmente è un cambio di base, cioè

$$f'(z) = T'(z) + R'(z), \qquad T'(z) = \sum_{i=1}^{n}\left(\frac{z^i}{(i)!}\right), \quad R'(z) = \sum_{i>n}\left(\frac{z^i}{(i)!}\right).$$

Sia $y = \{x\}\log 2$ e sia $\tilde{y}$ la sua approssimazione con precisione p_y. Allora, riscrivendo la (1),

$$\begin{aligned} 2^{-x} &= 2^{-\lfloor x\rfloor}\cdot 2^{-\{x\}} \\ &= 2^{-\lfloor x\rfloor}\cdot e^{-\{x\}\log 2} = 2^{-\lfloor x\rfloor}\cdot e^{-y} \\ &= 2^{-\lfloor x\rfloor}\cdot e^{\varepsilon_y}\cdot e^{-\tilde{y}} \\ &= 2^{-\lfloor x\rfloor}\cdot e^{\varepsilon_y}\cdot \left(T'_n(\tilde{y}) + R'_n(\tilde{y})\right) \\ &= 2^{-\lfloor x\rfloor}\cdot e^{\varepsilon_y}\cdot T'_n(\tilde{y}) + 2^{-\lfloor x\rfloor}\cdot e^{\varepsilon_y}\cdot R'_n(\tilde{y}), \end{aligned} \tag{3}$$

in cui ε_y è positivo se $\tilde{y}$ è un'approssimazione per eccesso, negativo altrimenti, ma vale che $|\varepsilon_y| < b^{-p}$. La (3) è equivalente a

$$2^{-\lfloor x\rfloor}\cdot T'_n(\tilde{y}) = 2^{-x} + \left(\frac{2^{-x}}{e^{\varepsilon_y}} - 2^{-x}\right) - 2^{-\lfloor x\rfloor}\cdot R'_n(\tilde{y}). \tag{4}$$

4.2 Approssimare il polinomio di MacLaurin

Come abbiamo notato, anche $T'_n(\tilde{y})$ deve essere approssimato. Il seguente algoritmo, in pseudocodice, dati $\tilde{y} \in \mathbb{Q}$, $n \in \mathbb{N}$ e $q \in \mathbb{N}$, descrive un possibile calcolo di $\tilde{T}'_n(\tilde{y})$, in cui `approx` e `div_approx` corrispondono a `trunc` o `round` e a `div_trunc` e `div_round`, a seconda che si approssimi per eccesso o per difetto con precisione q.

```
0  rec_exp_e_frac(ty, n, q)
1  {
2          res = 0;
3          for (i = n; i > 0; i--) {
4                  if (i%2 == 0)
5                          res += +1;
6                  else
7                          res += -1;
8
9                  res *= ty;
10                 res = approx(res, q);
11
12                 res = div_approx(res, i, q);
13         }
14         res += 1;
15
16         return res;
17 }
```

Supponiamo di voler calcolare $T'_n(\tilde{y})$ per difetto: allora sia n dispari. Lo schema in figura 4.2 tiene conto del valore di `res` durante l'esecuzione di `rec_exp_e_frac` e determina che

$$\tilde{T}'_n(\tilde{y}) = T'_n(\tilde{y}) - \sum_{i=2}^{n} \left(\frac{\tilde{y}^{i-1}}{(i)!} \alpha_i \right) - \sum_{i=2}^{n} \left(\frac{\tilde{y}^{i-2}}{(i-1)!} \beta_i \right),$$

cioè $\tilde{T}'_n(\tilde{y})$ è un'approssimazione per difetto di $T'_n(\tilde{y})$. In modo simmetrico, se volessimo calcolare $T'_n(\tilde{y})$ per eccesso, avremmo che

$$\tilde{T}'_n(\tilde{y}) = T'_n(\tilde{y}) + \underbrace{\sum_{i=2}^{n} \left(\frac{\tilde{y}^{i-1}}{(i)!} \alpha_i \right) + \sum_{i=2}^{n} \left(\frac{\tilde{y}^{i-2}}{(i-1)!} \beta_i \right)}_{=\varepsilon_a}.$$

Chiamando l'errore ε_a ($\varepsilon_a^{\max}$ nell'approssimazione per difetto, $\varepsilon_a^{\min}$ in quella per eccesso) e riprendendo la (4), abbiamo che

$$\begin{aligned} (4) &\iff 2^{-\lfloor x \rfloor} \cdot \left(\tilde{T}'_n(\tilde{y}) - \varepsilon_a \right) = 2^{-x} + \left(\frac{2^{-x}}{e^{\varepsilon_y}} - 2^{-x} \right) - 2^{-\lfloor x \rfloor} \cdot R'_n(\tilde{y}) \\ &\iff 2^{-\lfloor x \rfloor} \cdot \tilde{T}'_n(\tilde{y}) = 2^{-x} + \left(\frac{2^{-x}}{e^{\varepsilon_y}} - 2^{-x} \right) - 2^{-\lfloor x \rfloor} \cdot R'_n(\tilde{y}) + 2^{-\lfloor x \rfloor} \cdot \varepsilon_a, \end{aligned} \tag{5}$$

cioè $2^{-\lfloor x \rfloor} \cdot \tilde{T}'_n(\tilde{y})$ è un'approssimazione di 2^{-x}. Sapendo che $\tilde{y}$ è l'approssimazione di $\{x\} \log 2$, notando che le sommatorie sono parte della serie di MacLaurin

per $f(z) = e^z$ e data la precisione q di ogni operazione, allora

$$|\varepsilon_a| < e^{\log 2} \cdot b^{-q} + e^{\log 2} \cdot b^{-q} = 4 \cdot b^{-q}. \tag{6}$$

`res`	`i`	Prossima operazione
-1	n	`res *= ty`
$-\tilde{y}$	n	`res = trunc(res, q)`
$-\tilde{y} - \alpha_n$	n	`res = div_trunc(res, i, q)`
$-\frac{\tilde{y}}{n} - \frac{\alpha_n}{n} - \beta_n$	$n-1$	`res += 1`
$-\frac{\tilde{y}}{n} + 1 - \frac{\alpha_n}{n} - \beta_n$	$n-1$	`res *= ty`
$-\frac{\tilde{y}^2}{n} + \tilde{y} - \frac{\tilde{y}}{n}\alpha_n - \tilde{y} \cdot \beta_n$	$n-1$	`res = trunc(res, q)`
$-\frac{\tilde{y}^2}{n} + \tilde{y} - \frac{\tilde{y}}{n}\alpha_n - \tilde{y} \cdot \beta_n - \alpha_{n-1}$	$n-1$	`res = div_trunc(res, i, q)`
$-\frac{\tilde{y}^2}{n(n-1)} + \frac{\tilde{y}}{n-1} - \frac{\tilde{y}}{n(n-1)}\alpha_n - \frac{\tilde{y}}{n-1}\beta_n - \frac{\alpha_{n-1}}{n-1} - \beta_{n-1}$	$n-1$	`res = div_trunc(res, i, q)`
$\vdots$		
$\underbrace{\sum_{i=0}^{n}\left((-1)^i\frac{\tilde{y}^i}{(i)!}\right)}_{=T(\tilde{y})'} - \sum_{i=2}^{n}\left(\frac{\tilde{y}^{i-1}}{(i)!}\alpha_i\right) - \sum_{i=2}^{n}\left(\frac{\tilde{y}^{i-2}}{(i-1)!}\beta_i\right)$		

Figura 1: Calcolo dell'approssimazione per difetto $\tilde{T}_n'(\tilde{y})$; α_i e β_i sono gli errori commessi dall'approssimazione e dall'approssimazione della divisione alla $(i - n + 1)$-esima iterazione.

4.3 Conclusioni

Ricapitolando, l'approssimazione di 2^{-X} si conduce al calcolo di un'approssimazione di $2^{-x_{\max}}$ per difetto e di $2^{-x_{\min}}$ per eccesso in cui l'algoritmo immaginato:

1. divide $x_{\max}$ o $x_{\min}$ in parte intera e frazionaria, spezzando la potenza in due fattori;
2. della parte frazionaria calcola l'approssimazione (di $x_{\max}$ per eccesso, di $x_{\min}$ per difetto) della moltiplicazione per $\log 2$, sostanzialmente cambiando la base della potenza problematica da 2 ad e;
3. calcola un'approssimazione del polinomio di MacLaurin (di grado pari per approssimare per difetto, dispari altrimenti) per $f(z) = e^{-z}$ con il metodo di Horner, approssimando alla precisione q ogni moltiplicazione e divisione;
4. infine moltiplica questo risultato per l'elevamento a potenza di due della parte intera trovato al punto (1), ottenendo l'estremo cercato.

Quindi, dato $X = [x_{\min}, x_{\max}]$ (o $X = (x_{\min}, x_{\max})$), dati $n, m \in \mathbb{N}$ tali che n è dispari e m è pari e dati $p, q \in \mathbb{N}$, l'algoritmo calcola

$$\left(2^{-\lfloor x_{\max} \rfloor} \cdot \tilde{T}'_n(\tilde{y}_{\max}),\ 2^{-\lfloor x_{\min} \rfloor} \cdot \tilde{T}'_m(\tilde{y}_{\min})\right) \supseteq \left(2^{-x_{\max}}, 2^{-x_{\min}}\right). \tag{7}$$

Per l'estremo inferiore vale

$$2^{-\lfloor x_{\max} \rfloor} \cdot \tilde{T}'_n(\tilde{y}_{\max}) = 2^{-x_{\max}} + \underbrace{\left(\frac{2^{-x_{\max}}}{e^{\varepsilon_{y_{\max}}}} - 2^{-x_{\max}}\right)}_{\text{cambio di base}} - \underbrace{\frac{R'_n(\tilde{y}_{\max})}{2^{\lfloor x_{\max} \rfloor}}}_{\text{MacLaurin}} + \underbrace{\frac{\varepsilon_a^{\max}}{2^{\lfloor x_{\max} \rfloor}}}_{\text{appr. polin.}},$$

con $\tilde{y}_{\max}$ l'approssimazione per eccesso di $\{x_{\max}\} \log 2$ e $\varepsilon_{y_{\max}}$ l'errore (positivo) commesso, $R'_n(\tilde{y})$ il resto del polinomio di MacLaurin per $f(z) = e^{-z}$ e $\varepsilon_a^{\max}$ l'errore (negativo) commesso nel calcolo di $\tilde{T}'_n(\tilde{y})$. In modo simmetrico, per l'estremo superiore vale che

$$2^{-\lfloor x_{\min} \rfloor} \cdot \tilde{T}'_m(\tilde{y}_{\min}) = 2^{-x_{\min}} + \underbrace{\left(\frac{2^{-x_{\min}}}{e^{\varepsilon_{y_{\max}}}} - 2^{-x_{\min}}\right)}_{\text{cambio di base}} - \underbrace{\frac{R'_m(\tilde{y}_{\min})}{2^{\lfloor x_{\min} \rfloor}}}_{\text{MacLaurin}} + \underbrace{\frac{\varepsilon_a^{\min}}{2^{\lfloor x_{\min} \rfloor}}}_{\text{appr. polin.}},$$

con $\tilde{y}_{\min}$ l'approssimazione per difetto di $\{x_{\min}\} \log 2$ e $\varepsilon_{y_{\min}}$ l'errore (negativo) commesso, $R'_m(\tilde{y})$ il resto del polinomio di MacLaurin per $f(z) = e^{-z}$ e $\varepsilon_a^{\min}$ l'errore (positivo) commesso nel calcolo di $\tilde{T}'_m(\tilde{y})$.

Se si volesse stimare l'errore compiuto, delle maggiorazioni utili dell'errore commesso introducendo il polinomio di MacLaurin e nel suo calcolo sono:

- $|R'_n(\tilde{y})| < \dfrac{\tilde{y}^n}{(n)!} < \dfrac{(\log 2)^n}{(n)!}$ e $|R'_m(\tilde{y})| < \dfrac{\tilde{y}^m}{(m)!} < \dfrac{(\log 2)^m}{(m)!}$, perché la serie di Taylor converge alternatamente e $\tilde{y}$ è l'approssimazione di un valore tra 0 e 1 moltiplicato per $\log 2$;
- $|\varepsilon_a^{\max}| < 4 \cdot b^{-p}$ e $|\varepsilon_a^{\min}| < 4 \cdot b^{-q}$ per la disequazione (6), in cui b è la potenza di due che viene utilizzata come base per rappresentare i numeri di macchina a precisione arbitraria.

References

[1] *Global Optimization Using Interval Analysis*, Second Edition, Revised and Expanded, E. Hansen, G. W. Walster, Marcel Dekker Inc., 2004.

[2] *Analisi matematica*, Seconda Edizione, Bertsch, Dal Passo, Giacomelli, McGraw-Hill, 2011.