

Lezione 9

Nella prima parte di questa lezione vedremo alcune osservazioni ed esempi conclusivi sul modello del calcolatore. Nella seconda parte proseguiamo l'analisi del metodo di bisezione iniziata la lezione scorsa.

Prima parte

- Le definizioni di algoritmo *accurato* e di algoritmo *stabile* quando utilizzato per approssimare il valore di f in x sono state date per $f(x) \neq 0$.

– Se $f(x) = 0$ e $\phi(x) = 0$ la relazione:

$$\phi(x) = (1 + e) f(x)$$

è verificata per $e = 0$, cioè: $\phi(x)$ è una “piccola” perturbazione moltiplicativa di $f(x)$ e quindi ϕ è un algoritmo accurato quando utilizzato per approssimare f in x .

– Se $x = 0$, $f(0) = 0$ e $\phi(0) = 0$ la relazione:

$$\phi(0) = (1 + e_2) f((1 + e_1)0)$$

è verificata per $e_1 = e_2 = 0$, cioè: $\phi(0)$ è una “piccola” perturbazione moltiplicativa del valore di f in un punto “vicino” a 0 e quindi ϕ è un algoritmo stabile quando utilizzato per approssimare f in $x = 0$.

- Siano $f : \Omega \rightarrow \mathbb{R}$, $\Omega \subset \mathbb{R}$ una funzione elementare e $\phi : \Omega \cap M \rightarrow M$ la funzione predefinita corrispondente ad f (ovvero tale che: per ogni $\xi \in \Omega \cap M$, $\phi(\xi) = \text{rd}(f(\xi))$).¹

L'algoritmo $\Phi : \Omega \rightarrow M$ definito da $\Phi(x) = \phi(\text{rd}(x))$ è stabile quando utilizzato per approssimare f per ogni $x \in \Omega$.

(Infatti: per ogni x esistono e_1 ed e_2 tali che:

$$\Phi(x) = (1 + e_2) f((1 + e_1)x) \quad \text{e} \quad |e_1| \leq u, |e_2| \leq u$$

Come già osservato, le limitazioni su e_1 ed e_2 consentono di ritenerli in ogni caso “piccoli.” Dunque l'algoritmo è stabile per ogni x .)

- Siano $*$ un'operazione aritmetica, $f : \Omega \rightarrow \mathbb{R}$ la funzione definita da $f(x_1, x_2) = x_1 * x_2$ e \otimes la pseudo-operazione aritmetica corrispondente a $*$.

L'algoritmo $\Phi : \Omega \rightarrow M$ definito da $\Phi(x) = \text{rd}(x_1) \otimes \text{rd}(x_2)$ è stabile quando utilizzato per approssimare f per ogni $(x_1, x_2) \in \Omega$.

(Infatti: per ogni x_1, x_2 esistono e_1, e_2 ed e_3 tali che:

$$\Phi(x_1, x_2) = (1 + e_3)((1 + e_1)x_1 * (1 + e_2)x_2) = (1 + e_3)f((1 + e_1)x_1, (1 + e_2)x_2)$$

e:

$$|e_1| \leq u, \quad |e_2| \leq u, \quad |e_3| \leq u$$

Dunque: $\Phi(x)$ è una “piccola” perturbazione moltiplicativa del valore di f in un punto “vicino” ad x_1, x_2 . Quanto scritto costituisce precisamente l'estensione della definizione di *stabilità* di un algoritmo al caso di funzioni di più variabili.)

Salvo casi particolarmente semplici, un algoritmo è definito *componendo* più funzioni predefinite. Le due osservazioni precedenti mostrano che gli “algoritmi elementari” utilizzati per costruire algoritmi più complessi *sono stabili*. La prossima osservazione mostra che la composizione di algoritmi stabili *non sempre* genera algoritmi stabili e chiarisce perchè ciò accade.

¹L'insieme di definizione di ϕ è, più correttamente: $\{x \in \Omega \text{ t.c. } \text{rd}(x) \in \Omega\}$

- Siano $f, g : \mathbb{R} \rightarrow \mathbb{R}$ due funzioni e ϕ, γ gli algoritmi, *stabili* quando utilizzati per approssimare, rispettivamente, i valori di f e g , definiti da:

$$\phi(x) = \text{rd}\left(f(\text{rd}(x))\right) \quad , \quad \gamma(x) = \text{rd}\left(g(\text{rd}(x))\right)$$

Si vuole studiare la stabilità dell'algoritmo $\Phi(x) = \phi(\gamma(x))$ quando utilizzato per approssimare i valori di $F(x) = f(g(x))$.

Tenuto conto della definizione di γ , per ogni x esistono $e_1, e_2 \in \mathbb{R}$ tali che:

$$\Phi(x) = \phi\left((1 + e_2)g((1 + e_1)x)\right) \quad \text{con} \quad |e_1| \leq u, |e_2| \leq u$$

Tenuto conto della definizione di ϕ e del fatto che $(1 + e_2)g((1 + e_1)x) = \gamma(x)$ è un numero di macchina, per ogni x esiste $e_3 \in \mathbb{R}$ tale che:

$$\Phi(x) = (1 + e_3)f\left((1 + e_2)g((1 + e_1)x)\right) \quad \text{con} \quad |e_3| \leq u$$

Per leggere $\Phi(x)$ come perturbazione moltiplicativa del valore di F in un opportuno punto, determiniamo $t \in \mathbb{R}$ tale che:

$$f\left((1 + e_2)g((1 + e_1)x)\right) = (1 + t)f\left(g((1 + e_1)x)\right)$$

Ponendo $(1 + e_3)(1 + t) = 1 + \tau$ si ottiene:

$$\Phi(x) = (1 + e_3)(1 + t)f\left(g((1 + e_1)x)\right) = (1 + \tau)F((1 + e_1)x)$$

con:

$$\tau = e_3 + t + t e_3$$

Per giudicare la stabilità di Φ occorre decidere se τ , ovvero t , sia “piccolo.” In altri termini occorre indagare il *condizionamento* del calcolo di f in $g((1 + e_1)x)$:

- Se il calcolo di f in $g((1 + e_1)x)$ è *ben condizionato* allora t risulta “piccolo.” Dunque anche τ lo è e l'algoritmo Φ risulta *stabile*.
- Se il calcolo di f in $g((1 + e_1)x)$ *non* è ben condizionato allora l'algoritmo Φ *può* risultare *non stabile*.

- *Esempio 1*

Si consideri la funzione f definita, per ogni $x \geq 0$ da: $f(x) = x - \sqrt{x}$ e l'algoritmo ϕ definito, anch'esso per ogni $x \geq 0$, da: $\phi(x) = \text{rd}(x) \ominus \text{SQRT}(\text{rd}(x))$. Si vuole studiare la stabilità di ϕ quando utilizzato per approssimare i valori di f per $x > 0$.

Per ogni x esistono e_1, e_2 ed e_3 tali che:

$$\phi(x) = (1 + e_3)\left((1 + e_1)x - (1 + e_2)\sqrt{(1 + e_1)x}\right)$$

con $|e_1| \leq u, |e_2| \leq u$ ed $|e_3| \leq u$.

Posto $\xi = \text{rd}(x) = (1 + e_1)x$, per ogni $\xi \neq 0, 1$ si ha:

$$\xi - (1 + e_2)\sqrt{\xi} = (1 + t)(\xi - \sqrt{\xi})$$

con:

$$t = -\frac{\sqrt{\xi}}{\xi - \sqrt{\xi}} e_2$$

e quindi:

$$\phi(x) = (1 + e_3)(1 + t)\left((1 + e_1)x - \sqrt{(1 + e_1)x}\right) = (1 + e_3)(1 + t)f((1 + e_1)x)$$

Per decidere la stabilità di ϕ occorre indagare la grandezza di t . Tenuto conto della limitazione per e_2 si ottiene:

$$|t| \leq \left| \frac{\sqrt{\xi}}{\xi - \sqrt{\xi}} \right| u$$

Se il coefficiente:

$$K(\xi) = \left| \frac{\sqrt{\xi}}{\xi - \sqrt{\xi}} \right|$$

ha valori non troppo grandi l'algoritmo risulta stabile, altrimenti non potremo concludere nulla.

– Si ha:

$$K(x) = \left| \frac{\sqrt{x}}{x - \sqrt{x}} \right| = \left| \frac{1}{\sqrt{x} - 1} \right|$$

e:

$$\lim_{x \rightarrow 0^+} K(x) = 1 \quad , \quad \lim_{x \rightarrow 1} K(x) = +\infty$$

Il coefficiente $K(\xi)$ assume quindi valori molto grandi per $\xi \approx 1$.

– *Esercizio*

Utilizzare *Scilab* per disegnare il grafico della funzione $K(x)$ per $x \in (0, 1) \cup (1, 10)$ e dedurre dal grafico ottenuto informazioni sulla stabilità dell'algoritmo.

Si osservi che quello che è stato fatto dall'introduzione di ξ in poi è studiare il *condizionamento* del calcolo di $x_1 - x_2$ in $x_1 = \xi, x_2 = \sqrt{\xi}$.

• *Esempio 2*

Si consideri la funzione f definita, per ogni $x \geq 0$ da: $f(x) = x + \sqrt{x}$ e l'algoritmo ϕ definito, anch'esso per ogni $x \geq 0$, da: $\phi(x) = \text{rd}(x) \oplus \text{SQRT}(\text{rd}(x))$. Si vuole studiare la stabilità di ϕ quando utilizzato per approssimare i valori di f per $x > 0$.

Per ogni x esistono e_1, e_2 ed e_3 tali che:

$$\phi(x) = (1 + e_3) \left((1 + e_1)x + (1 + e_2)\sqrt{(1 + e_1)x} \right)$$

con $|e_1| \leq u, |e_2| \leq u$ ed $|e_3| \leq u$.

Posto $\xi = \text{rd}(x) = (1 + e_1)x$, poichè $x > 0$ implica $\xi > 0$ si ha:

$$\xi + (1 + e_2)\sqrt{\xi} = (1 + t)(\xi + \sqrt{\xi})$$

con:

$$t = \frac{\sqrt{\xi}}{\xi + \sqrt{\xi}} e_2$$

e quindi:

$$\phi(x) = (1 + e_3)(1 + t) \left((1 + e_1)x + \sqrt{(1 + e_1)x} \right) = (1 + e_3)(1 + t)f((1 + e_1)x)$$

Per decidere la stabilità di ϕ occorre indagare la grandezza di t . Tenuto conto della limitazione per e_2 e che per ogni $\xi > 0$ si ha:

$$\left| \frac{\sqrt{\xi}}{\xi + \sqrt{\xi}} \right| < 1$$

si ottiene:

$$|t| \leq u$$

e l'algoritmo risulta *stabile* per ogni $x > 0$.

Si osservi che quello che è stato fatto dall'introduzione di ξ in poi è studiare il *condizionamento* del calcolo di $x_1 + x_2$ in $x_1 = \xi, x_2 = \sqrt{\xi}$.

• *Esempio 3*

Si consideri la funzione f definita, per ogni $x \geq 0$ da: $f(x) = x - \sqrt{x}$. Per ogni $x > 0$ si ha:

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

Sia ϕ l'algoritmo, definito anch'esso per ogni $x \geq 0$, da:

$$\phi(0) = 0 \quad , \quad \phi(x) = \left(\text{rd}(x) \otimes (\text{rd}(x) \ominus 1) \right) \oslash \left(\text{rd}(x) \oplus \text{SQRT}(\text{rd}(x)) \right) \quad \text{per } x > 0$$

Si vuole studiare la stabilità di ϕ quando utilizzato per approssimare i valori di f per $x > 0$.

Per ogni $x > 0$ poniamo: $\xi = \text{rd}(x)$, $f_1(x) = x(x - 1)$, $\phi_1(x) = \xi \otimes (\xi \ominus 1)$, $f_2(x) = x + \sqrt{x}$ e $\phi_2(x) = \xi \oplus \text{SQRT}(\xi)$. Si ha:

- Esistono numeri reali e_1, e_2 ed e_3 tali che $|e_1| \leq u$, $|e_2| \leq u$, $|e_3| \leq u$ e:

$$\phi_1(x) = (1 + e_3)(1 + e_2)\xi(\xi - 1) = (1 + e_3)(1 + e_2)f_1((1 + e_1)x)$$

Posto $(1 + e_3)(1 + e_2) = 1 + t_{23}$ si ha:

$$\phi_1(x) = (1 + t_{23})f_1((1 + e_1)x)$$

e, tenuto conto delle limitazioni su e_2 ed e_3 :

$$|t_{23}| \leq 2u + u^2 \approx 2u$$

Se ne deduce che ϕ_1 è un algoritmo *stabile* quando utilizzato per approssimare f_1 per $x > 0$.

- Per quanto mostrato nell'Esempio 2, ϕ_2 è un algoritmo *stabile* quando utilizzato per approssimare f_2 : per ogni x esiste un numero reale e_4 "piccolo" tale che:

$$\phi_2(x) = (1 + e_4)f_2((1 + e_1)x)$$

- Esiste un numero reale e_5 tale che $|e_5| \leq u$ e:

$$\phi(x) = \phi_1(x) \oslash \phi_2(x) = (1 + e_5) \frac{\phi_1(x)}{\phi_2(x)} = \frac{(1 + e_5)(1 + t_{23})}{1 + e_4} \frac{f_1((1 + e_1)x)}{f_2((1 + e_1)x)}$$

Posto:

$$1 + \theta = \frac{(1 + e_5)(1 + t_{23})}{1 + e_4}$$

si ha:

$$\phi(x) = (1 + \theta) f((1 + e_1)x)$$

e, tenuto conto delle limitazioni su e_4, e_5 e t_{23} :

$$\theta = \frac{t_{23} - e_4 + e_5 + e_5 t_{23}}{1 + e_4} \quad |\theta| \leq \frac{5u + 4u^2 + u^3}{1 - u} \approx 5u$$

Se ne deduce che ϕ è un algoritmo *stabile* quando utilizzato per approssimare f per ogni $x > 0$.

Seconda parte

Riprendiamo adesso l'analisi del *metodo di bisezione*. Sia $f : [a, b] \rightarrow \mathbb{R}$ una funzione continua tale che $f(a)f(b) < 0$. Consideriamo la prima procedura che realizza il metodo:

$$z = \text{Bisezione}(f, a, b)$$

Si è visto che, se la procedura non determina uno zero di f dopo un numero finito di iterazioni, costruisce una successione di *intervalli* $I_k = [a_k, b_k]$ tale che:

$$\lim \text{mis } I_k = 0$$

e la successione dei *punti medi* x_k degli intervalli. Per tali successioni si ha:

- Le successioni a_k e b_k sono convergenti ed hanno lo stesso limite. Detto α il limite comune delle successioni si ha poi:

$$\lim_{k \rightarrow \infty} x_k = \alpha \quad \text{e} \quad \alpha \text{ è zero di } f$$

Infatti:

- Per costruzione la successione a_k risulta *monotona non decrescente e superiormente limitata* (da b), dunque convergente: $\lim a_k = A$.
- Analogamente: la successione b_k risulta *monotona non crescente e inferiormente limitata* (da a), dunque convergente: $\lim b_k = B$.
- La successione $\text{mis } I_k = b_k - a_k$ è differenza di successioni convergenti e quindi:

$$0 = \lim \text{mis } I_k = \lim (b_k - a_k) = B - A \quad \text{dunque} \quad A = B$$

- Posto $\alpha = A$, poiché $a_k < x_k < b_k$ si ha:

$$\lim x_k = \alpha$$

- Sia, ad esempio: $f(a) < 0$ e $f(b) > 0$. Per ogni k si ha: $f(a_k) < 0$ e $f(b_k) > 0$. Dunque, per la continuità di f :

$$\lim f(a_k) = f(\alpha) \geq 0 \quad \text{e} \quad \lim f(b_k) = f(\alpha) \leq 0$$

e quindi $f(\alpha) = 0$.

Nella seconda procedura che realizza il metodo:

$$z = \text{Bisezione}(f, a, b, \delta)$$

si è introdotto il *criterio d'arresto*:

dato $\delta > 0$: *Se* $\text{mis } I_k < \delta$ *allora* arresta la costruzione

allo scopo di rendere *praticamente utilizzabile* la procedura (ovvero di assicurare che in ogni caso la procedura *termina l'esecuzione*).

La presenza di un criterio d'arresto è *obbligatoria* in ogni procedura iterativa per renderla praticamente utilizzabile. Inoltre:

- Ciascun criterio d'arresto deve avere le proprietà seguenti:
 - * Deve essere *calcolabile*: ad ogni iterazione la procedura deve *essere in grado di verificare* se il criterio è soddisfatto.
 - * Deve essere *efficace*: in ogni caso il criterio deve *arrestare la costruzione* della successione dopo un numero *finito* di iterazioni.
 - * L'ultimo elemento costruito della successione deve *approssimare l'oggetto cercato con l'accuratezza richiesta* dall'utilizzatore.

Il criterio d'arresto proposto *ha* le tre proprietà: è *calcolabile*, infatti a ciascuna iterazione la procedura conosce a_k e b_k , può calcolare $\text{mis } I_k = b_k - a_k$ e verificare se è minore del parametro δ fornito dall'utilizzatore; è *efficace*, infatti si ha $\lim \text{mis } I_k = 0$ e per ogni $\delta > 0$ la disuguaglianza $\text{mis } I_k < \delta$ è *certamente* verificata dopo un numero *finito* di iterazioni. Infine, quando il criterio di arresto è verificato la procedura restituisce x_k , punto medio dell'ultimo intervallo calcolato I_k . L'intervallo I_k , per costruzione, contiene almeno uno zero α di f . Si ha allora:

$$|x_k - \alpha| \leq \frac{\text{mis } I_k}{2} < \frac{1}{2} \delta$$

ovvero la procedura restituisce un valore che approssima uno zero di f con *errore assoluto* inferiore a $\frac{1}{2} \delta$. Il criterio verifica dunque la terza proprietà a patto che l'utilizzatore misuri l'accuratezza con l'errore assoluto. Per questo motivo il criterio d'arresto proposto è classificato *di tipo assoluto*.

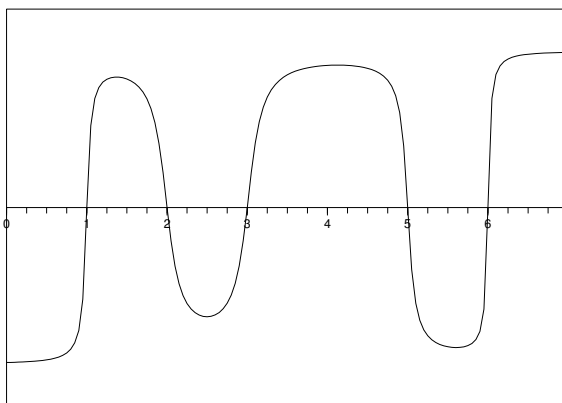
Un esempio di criterio d'arresto che è efficace e restituisce un valore che approssima uno zero di f con l'accuratezza richiesta *ma non è calcolabile*, è il seguente. Sia α uno zero di f in $[a, b]$:

dato $\delta > 0$: *Se* $|x_k - \alpha| < \delta$ *allora* arresta la costruzione

La procedura non conosce α e quindi non può verificare se il criterio è soddisfatto.

Esercizi

1. Siano f la funzione definita, per $x \in \mathbb{R}$, da $f(x) = e^{-x}$ e **EXP** la funzione predefinita corrispondente alla funzione elementare e^x . Sia infine $\phi(x) = 1 \oslash \mathbf{EXP}(\text{rd}(x))$ l'algoritmo utilizzato per approssimare i valori di f . Discutere la stabilità di ϕ .
2. Sia $f(x) = x^2 - x$. Detta $C(x, e)$ la funzione di condizionamento del calcolo di f in x , determinare:
 - L'insieme di definizione di $C(x, e)$;
 - $C(\frac{1}{2}, e)$;
 - $\lim_{x \rightarrow 0} C(x, e)$.
3. Il grafico della funzione $f : [0, 7] \rightarrow \mathbb{R}$ è rappresentato nella figura seguente.



Sia x_k la successione costruita dalla procedura *Bisezione*($f, 0, 7$). Determinare $\lim_{k \rightarrow \infty} x_k$.