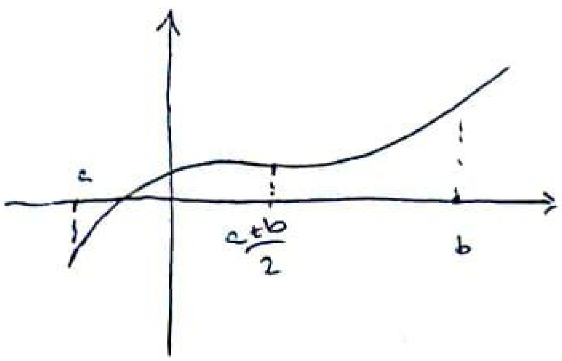
METODO DI BISEZIONE



Consiste nello sfrutta il Th. degli zeri, in effetti il metodo di bisezione è una dimostrazione costruttiva del th.

Th. degli zeri

Sia $f \in C[a,b]$ se $f(a) \cdot f(b) < 0$, $\exists c \in [a,b]$ t.c. $f(c) = 0$, ovvero c è radice di f.

Il metodo è estremamente semplice, calcoliamo il p. medio $x_m = \frac{a+b}{2}$ (per irrobustire il calcolo su un calcolatore, ovvero evitare errori di overflow/underflow possiamo anche calcolare il p. medio in questo modo $x_m = a + \frac{b-a}{2}$), valutiamo $f(x_m)$ se è 0 abbiamo finito altrimenti ripetiamo il procedimento su uno dei 2 intervalli $[a, x_m]$ oppure $[x_m, b]$, basta vedere in quale dei 2 vale che $f(a) \cdot f(b) < 0$.

Errore del metodo di bisezione (vedi Bur. - F. pg. 51)

Cercare di stimare l'errore in metodo numerico è un po' come dimostrarne la correttezza, infatti se un metodo ha un errore molto grande, o che addirittura non riusciamo a stimare, è piuttosto inutile dato che i risultati "non sono attendibili"

Ora con il metodo di bisezione vogliamo mostrare che viene generata una sequenza $\{p_n\}_{n=1}^{\infty}$ che approssima lo zero di una funzione f.

$$|p_n - p| \leq \frac{b-a}{2^n} \quad \text{quando } n \geq 1$$

Dim: $\forall n,\ n \geq 1$ abbiamo

$$b_n - a_n = \frac{1}{2^{n+1}}(b-a) \quad \text{e} \quad p \in (a_n, b_n)$$

Dato che $p_n = \frac{1}{2}(a_n + b_n)$ $\forall n \geq 1$, ne segue che

$$|p_n - p| \leq \frac{1}{2}(b_n - a_n) = \frac{b-a}{2^n}$$

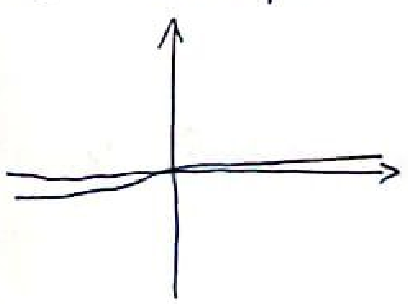
Quindi $\{p_n\}_{n=1}^{\infty}$ converge a p con un tasso di convergenza $O\left(\frac{1}{2^n}\right)$

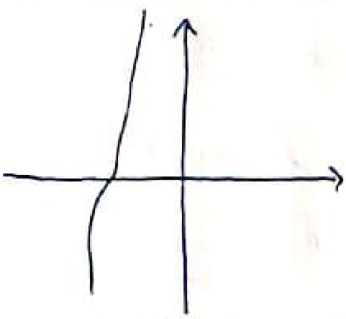
$$p_n = p + \underbrace{O\left(\frac{1}{2^n}\right)}_{\text{tende a } 0}$$

∎

Ciò ci assicura della correttezza ~~del~~ del metodo, purtroppo però i calcolatori non possono eseguire passaggi al limite quindi questa successione verrà stoppata dopo un certo numero di iterazioni producendo un errore.

Un altro problema, quello ~~di~~ enunciato sopra non è un "problema" ma più una caratteristica dei metodi numerici, riguarda la velocità di convergenza del metodo di bisezione, il quale oltre a non essere estremamente veloce per sua natura, può incontrare anche funzioni particolari che rallentano ulteriormente il calcolo, sotto riporto alcuni esempi





Sono un po' due casi ~~pratici~~ speculari. In quello a sx il metodo raggiunge subito una buona approx, ma "non se ne rende conto" e continua.

Viste queste caratteristiche dell'alg. di bisezione, ~~d~~ possiamo dire che come metodo funziona ma vi sono altri strumenti più efficaci.

bisection_method.m

home > antonio > elaborati_matlab > metodo bisezione > bisection_method.m > ...

```matlab
function [root, ecc_iter] = bisection_method(fun, a, b, tol, max_iter)
    ecc_iter = 0;
    root = a + (b-a)/2;
    fa = fun(a);
    fp = fun(root);
    while abs(fp) > tol && ecc_iter < max_iter
        if(fp*fa < 0)
            b = root;
        else
            a = root;
            fa = fun(a);
        end
        ecc_iter = ecc_iter + 1;
        root = a + (b-a)/2;
        fp = fun(root);
    end
    if(ecc_iter >= max_iter)
        ecc_iter = 1;
    else
        ecc_iter = 0;
    end
end

% durante ogni iterazione al massimo valutiamo la funzione due volte
% ecc_iter assume il valore 1 quando si supera
% il numero massimo di iterazioni 0 altrimenti
% inoltre non occorre valutare la fuzione in b!

```

0 0 Ln 27, Col 48 Spaces: 4 UTF-8 LF MATLAB 1