Esercizio 1. Errori di cancellazione

Si risponda alle seguente domande:

1. Si calcoli analiticamente $\lim_{x\to 0} \frac{x}{e^x-1}$
2. Si calcoli la quantità $y = \frac{x}{e^x-1}$ per $x = 10^{-15}$ usando MATLAB. Quale è l'errore commesso? (Si usi opportunamente il comando format). Perchè l'errore è grande?
3. Si sostituisca a e^x la sua espansione in serie Taylor intorno a $x = 0$ arrestata al termine di secondo ordine. Si calcoli quindi un'espressione approssimata di $\frac{x}{e^x-1}$.
4. Si calcoli la nuova quantità con MATLAB e si verifichi l'errore commesso in questo caso.

Soluzione

1. Usando il teorema di de l'Hôpital troviamo

$$\lim_{x\to 0} \frac{x}{e^x-1} = \lim_{x\to 0} \frac{x'}{(e^x-1)'} = \lim_{x\to 0} \frac{1}{e^x} = 1$$

.

2. Calcoliamo in MATLAB $y = \frac{x}{e^x-1}$ per $x = 10^{-15}$:

```
>> x=1e-15
>> format long
>> y1=x/(exp(x)-1)
   y1=0.90071992547410
```

L'errore percentuale relativo è

```
>> (1-y1)/1
   9.92800745259007
```

3. L'espansione in serie Taylor intorno a $x = 0$ arrestata al termine di secondo ordine è: $e^x \simeq 1 + x + \frac{x^2}{2}$. Quindi espressione approssimata di $\frac{x}{e^x - 1}$ è:

$$y(x) \simeq \frac{x}{1 + x^2 + \frac{x^2}{2} - 1} = \frac{1}{1 + \frac{x}{2}}.$$

Per calcolare questa espressione con MATLAB si scrive il commando

```
>> y2=1/(1+x/2)
>> (1-y2)*100 %errore percentuale
   4.441e-14
```

Esercizio 2. Metodo di bisezione (I)

Si consideri il problema della ricerca degli zeri dell'equazione non lineare

$$f(x) = e^x - x^2 - sin(x) - 1 = 0, \quad -2 \leq x \leq 2 \qquad (1)$$

Si risponda alle seguente domande:

1. Si tracci un grafico della funzione nell'intervallo considerato. Si localizzino le radici con l'aiuto del comando zoom .
2. Il metodo di bisezione è applicabile per calcolare entrambe le radici? Motivare la risposta.
3. Applicare il metodo quando possibile con *toll* $:= 10^{-12}$ e verificare che il numero di iterazioni effetuate è in accordo con quanto previsto dalla teoria.

Soluzione

Per capire se possiamo utilizzare il metodo di bisezione disegniamo la funzione:

```
>> x=[-2:0.01:2];
>> fun='exp(x)-x.^2-sin(x)-1';
>> yf=eval(fun);
>> plot(x,yf,'r');
>> grid
```

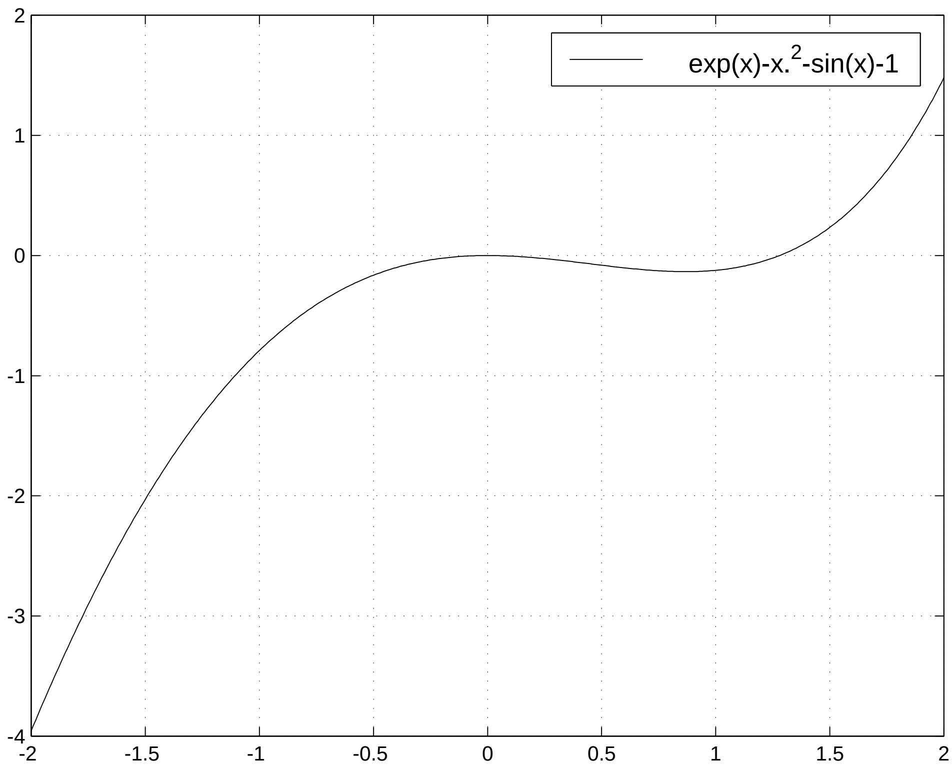


Figura 1:

Dalla figura 1 si può notare che l'equazione presenta 2 radici:una in 0 e l'altra (che nel seguito verrà indicata

con $\overline{\alpha}$) in un intorno di 1.3.
Per calcolare la radice $x = 0$ non è possibile trovare un intervallo $[a, b]$ contenente la sola radice $x = 0$ per il quale $f(a)f(b) < 0$.

Per la seconda radice, prendendo per esempio l'intervallo $[1, 1.5]$ sono soddisfatte le ipotesi per l'applicabilità del metodo di bisezione. Stimiamo il numero di iterazioni necessarie:

```
>> a=1; b=1.5; toll=1e-12;
>>m=log((b-a)/toll)/log(2)-1
m=
    37.8658
```

Quindi saranno necessarie almeno 38 iterazioni. Verifichiamolo utilizzando il programma `qssbisez`

```
>> nmax=100;
>> [xvect,xdif,fx,it]=qssbisez(a,b,nmax,toll,fun);

Numero di iterazioni : 38
Radice calcolata : 1.27970133
```

Per ottenere un errore assoluto al passo k minore di ϵ dobbiamo fare $k \geq log_2 \frac{b-a}{\epsilon} - 1$ iterazioni. Per trovare la soluzione α con un'errore $\epsilon/100$ dobbiamo fare $l \geq log_2 \frac{b-a}{\epsilon/100} - 1$. Quindi per ridurre l'errore ϵ con un fattore 100 sono necessarie $l - k \simeq log_2 100$ iterazioni supplementari.

Esercizio 3. Metodo di bisezione (II)

Si vuole determinare l'ascissa α del punto di minimo relativo della funzione

$$f = \frac{\sin(x)}{x} \qquad 0 < x < 2\pi.$$

1. A tale fine si utilizzi *opportunamente* il metodo di bisezione, dopo avere localizzato graficamente α. Si usi a tale scopo il comando Matlab `plot`.
2. Si giustifichi l'applicabilità del metodo di bisezione per la ricerca di α.
3. Si stimi teoricamente quante iterazioni sono necessarie per trovare α con un errore inferiore a $\varepsilon = 10^{-10}$
4. Si verifichi tale risultato utilizzando il programma `qssbisezione`.
5. Si stimi teoricamente quante iterazioni supplementari sono necessarie per ridurre l'errore precedentemente trovato di un fattore 100. Si verifichi tale risultato utilizzando il programma `qssbisez`

Soluzione

Rappresentiamo graficamente la funzione $f(x)$:

```
>> x=[0.01:0.01:2*pi];
>> fun='sin(x)./x';
>> yfun=eval(fun);
>> plot(x,yfun,'r');
```

```
>> grid
>> hold on
```

Osserviamo che il punto di minimo relativo della funzione $f(x)$ nell'intervallo $(0, 2\pi)$ corrisponde al punto in cui si annulla la sua derivata prima $f'(x)$. L'espressione della derivata prima è:

$$f'(x) = \frac{x\cos(x) - \sin(x)}{x^2} \tag{2}$$

Rappresentiamo sul grafico precedente anche la derivata:

```
>> dfun='(x.*cos(x)-sin(x))./(x.^2)';
>> ydfun=eval(dfun);
>> plot(x,ydfun,'b');
>> zoom on
```

Con l'aiuto del comando `zoom` stimiamo il valore $\alpha \simeq 4.5$.

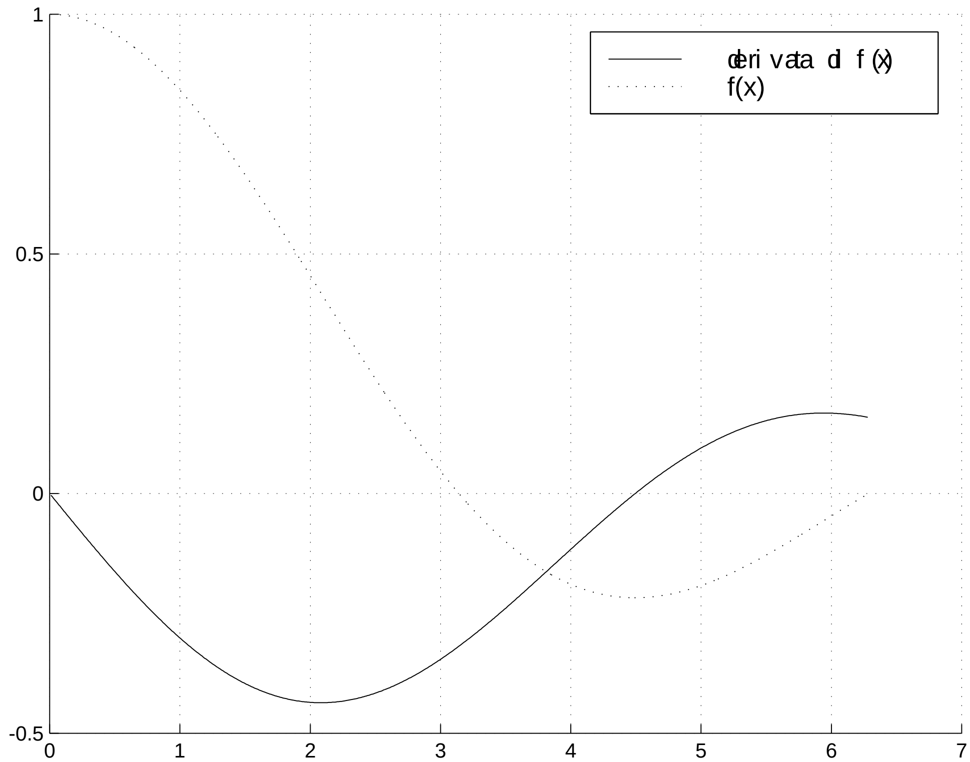
Dal grafico della derivata prima della funzione osserviamo che $f'(0) = 0$, mentre $f'(2\pi) > 0$. Per applicare il metodo di bisezione è necessario che la funzione di cui si vuole trovare lo zero assuma valori di segno diverso agli estremi. In questo caso possiamo utilizzare

$$a = 0 + \delta, \quad b = 2\pi,$$

con $\delta << 1$ (ad esempio $\delta = 0.0001$), poiché osserviamo che $f'(x) < 0$ nell'intervallo aperto $(0, \alpha)$. Con questa scelta possiamo applicare il metodo di bisezione per la ricerca di α.

Scriviamo l'espressione dell'errore commesso al passo k:

$$|e^{(k)}| < (b - a)/2^{k+1} - 1,$$



dove $e^{(k)} = x^{(k)} - \alpha$ è *l'errore assoluto* al passo k. Quindi per avere $|x^{(k)} - \alpha| \leq toll$ deve essere:

$$k > \log_2(b-a) - \log_2(toll) - 1 = \frac{\log((b-a)/toll)}{\log(2)} - 1 \simeq$$
$$\simeq \frac{\log((b-a)/toll)}{0.6931} - 1. \tag{3}$$

Nel nostro caso $(b-a) \simeq 2\pi$ e $toll = 10^{-10}$, quindi:

```
>> k=(log((2*pi)/(1e-10)))/0.6931-1

k =

   34.8732
```

quindi saranno necessarie almeno 35 iterazioni perché l'errore commesso sia minore di 10^{-10}.
Verifichiamo la stima ottenuta utilizzando il programma `bisezione`:

```
>> [xvect,xdif,fx,it]=qssbisezione(0.0001,2*pi,50,...
                           1e-10,dfun);
    Numero di Iterazioni : 35
    Radice calcolata     : 4.49340946
```