

Esercizi per l'esame

Aritmetica Finita - 6

Come richiesto implementiamo il calcolo del seno iperbolico tramite definizione

```
1 naif_sinh = @(x) (exp(x)-exp(-x))./2;
```

Adesso calcoliamo l'errore assoluto e l'errore relativo, prendendo come valore di riferimento l'implementazione interna di MATLAB del seno iperbolico. Valutiamo gli errori per valori di x compresi tra 10^{-12} e 10^0 . In particolare evidenzieremo tali errori nei punti $x = 10^{-12}, 10^{-11}, \dots, 10^{-1}, 10^0$

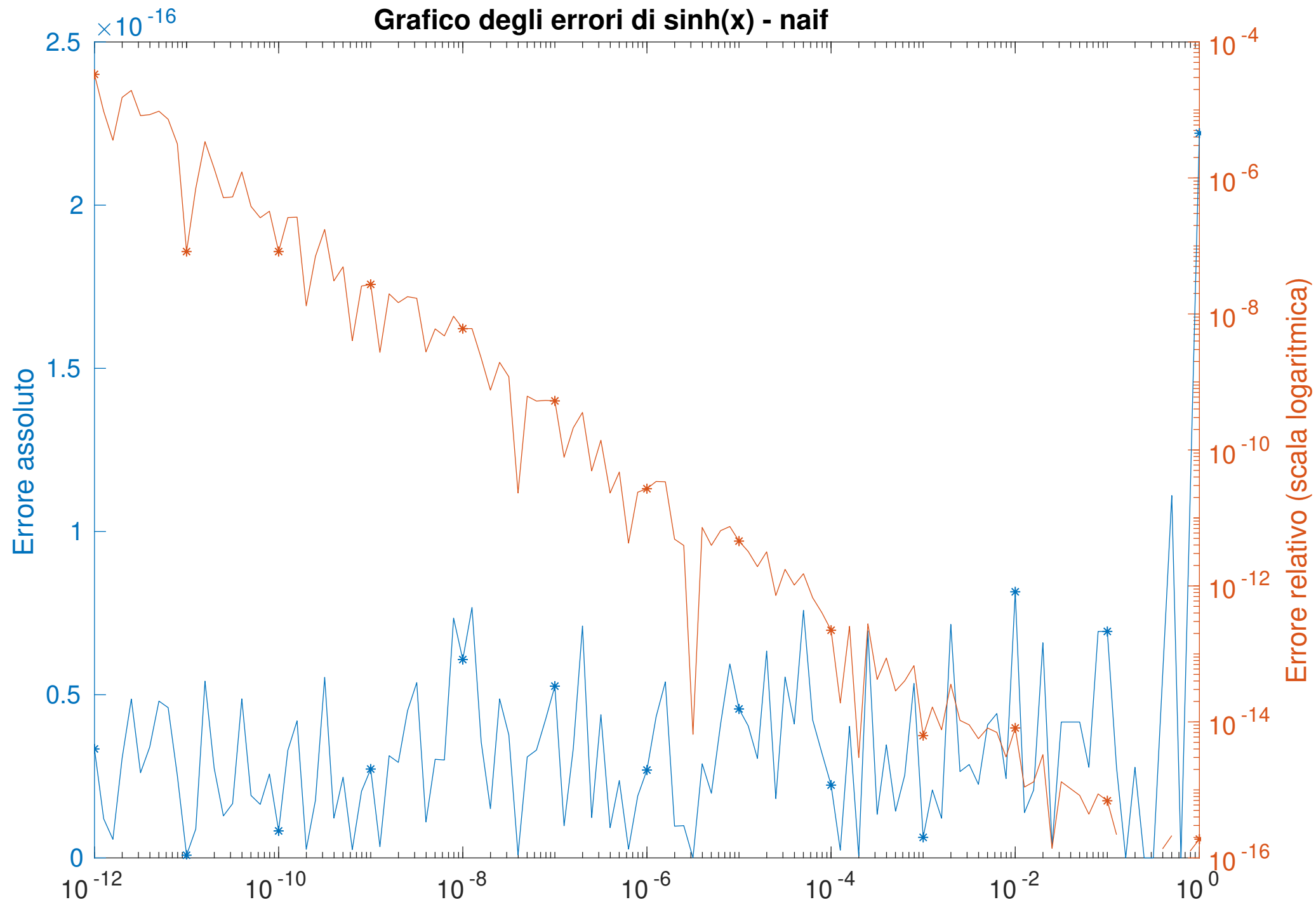
```
1 asse_x = logspace(-12,0,10*(13-1)+1);  
2 fine = 1:10:130;  
3 err_assoluto = abs(sinh(asse_x)-naif_sinh(asse_x));  
4 err_relativo = err_assoluto./sinh(asse_x);
```

Disegniamo infine un unico grafico (con due assi verticali) dove rappresentare gli errori relativi e assoluti.

```
1 figure();  
2 title('Grafico degli errori di sinh(x) - naif');  
3 yyaxis left;  
4 semilogx(asse_x,err_assoluto,'-',10.^(-12:0),err_assoluto(fine),'*');  
5 ylabel('Errore assoluto');  
6 yyaxis right;  
7 loglog(asse_x,err_relativo,'-',10.^(-12:0),err_relativo(fine),'*');  
8 ylabel('Errore relativo (scala logaritmica)');
```

Come si può notare nel grafico seguente, l'errore relativo aumenta per piccoli valori di x . Questo è giustificato poiché il numeratore della definizione del seno iperbolico presenta una differenza tra quantità che diventano tanto più prossime quanto più x è piccolo, portando quindi ad un fenomeno di cancellazione numerica.

Grafico degli errori di $\sinh(x)$ - naif



Aritmetica Finita - 7

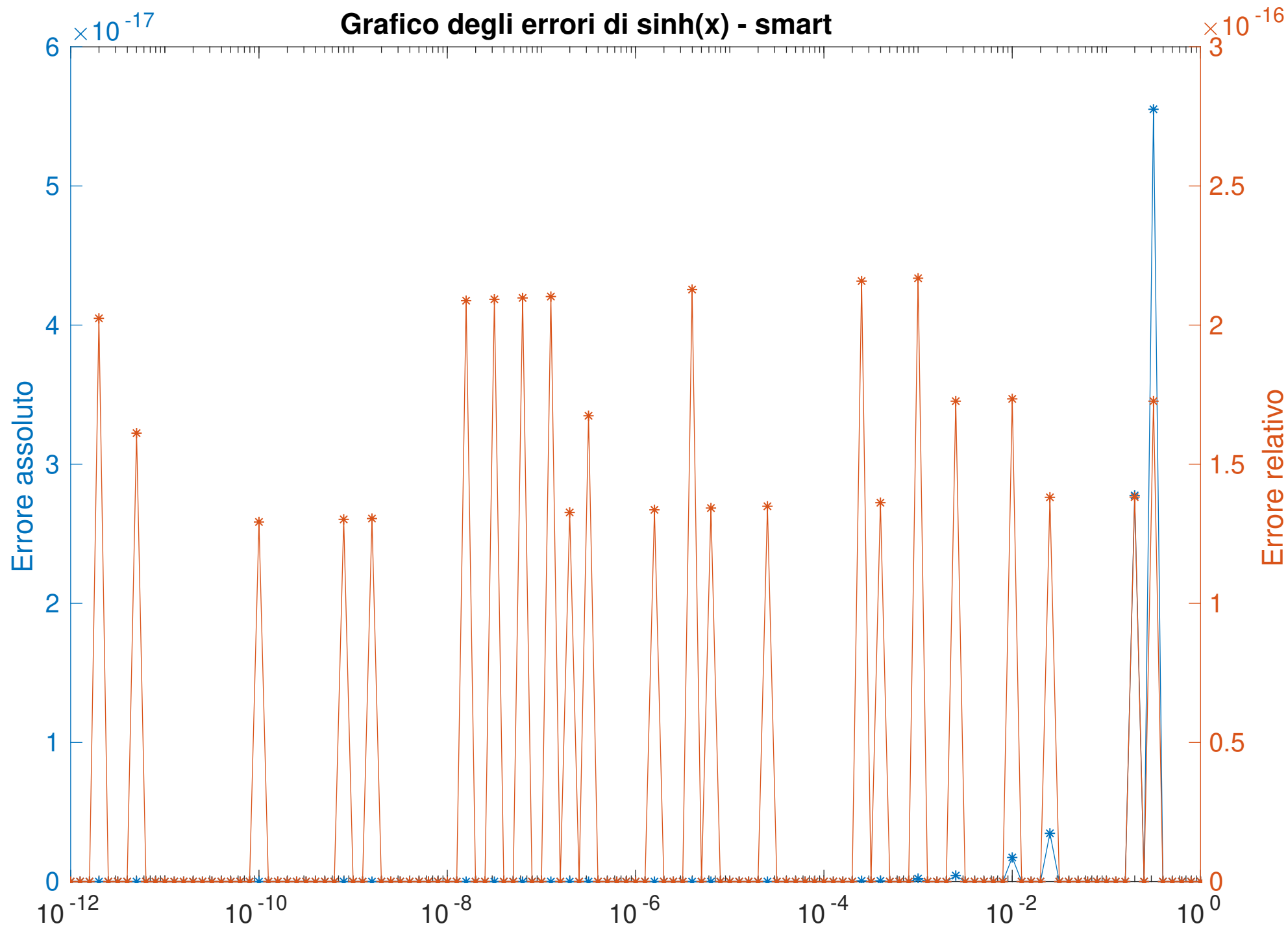
Continuando con i dati dell'esercizio precedente, implementiamo il calcolo del seno iperbolico in modo da evitare fenomeni di cancellazione numerica.

```
1 mio_sinh = @(x) (expm1(x)+expm1(x)./(expm1(x)+1))/2;
```

Ora calcoliamo e disegniamo il grafico dell'errore relativo di quest'ultima implementazione calcolato rispetto a quella interna a MATLAB

```
1 figure();
2 mio_err_assoluto = abs(sinh(asse_x)-mio_sinh(asse_x));
3 mio_err_relativo = mio_err_assoluto./sinh(asse_x);
4 title('Grafico degli errori di sinh(x) - smart');
5 yyaxis left;
6 semilogx(asse_x,mio_err_assoluto,'-*');
7 ylabel('Errore assoluto');
8 yyaxis right;
9 semilogx(asse_x,mio_err_relativo,'-*');
10 ylabel('Errore relativo');
```

Come si può notare nel grafico seguente, l'errore assoluto (e quindi quello relativo) sono drasticamente diminuiti, in alcuni punti sono addirittura nulli. Questo indica che l'implementazione alternativa è preferibile per valori di x prossimi a zero.



Aritmetica Finita - 8

Definiamo, come richiesto, un'espressione matematicamente equivalente alla differenza divisa di ordine uno di $f(x) = x^3$ ma priva di cancellazione numerica:

$$f[x_0, x_0 + h] = \frac{(x_0 + h)^3 - x_0^3}{h} = \frac{x_0^3 + 3x_0^2h + 3x_0h^2 + h^3 - x_0^3}{h} = 3x_0^2 + 3x_0h + h^2$$

Si può notare come l'ultima espressione sia esente da cancellazione numerica in quanto è la somma di sole quantità strettamente positive. Calcoliamo ora i valori delle due differenze divise per specifici valori di x e di h

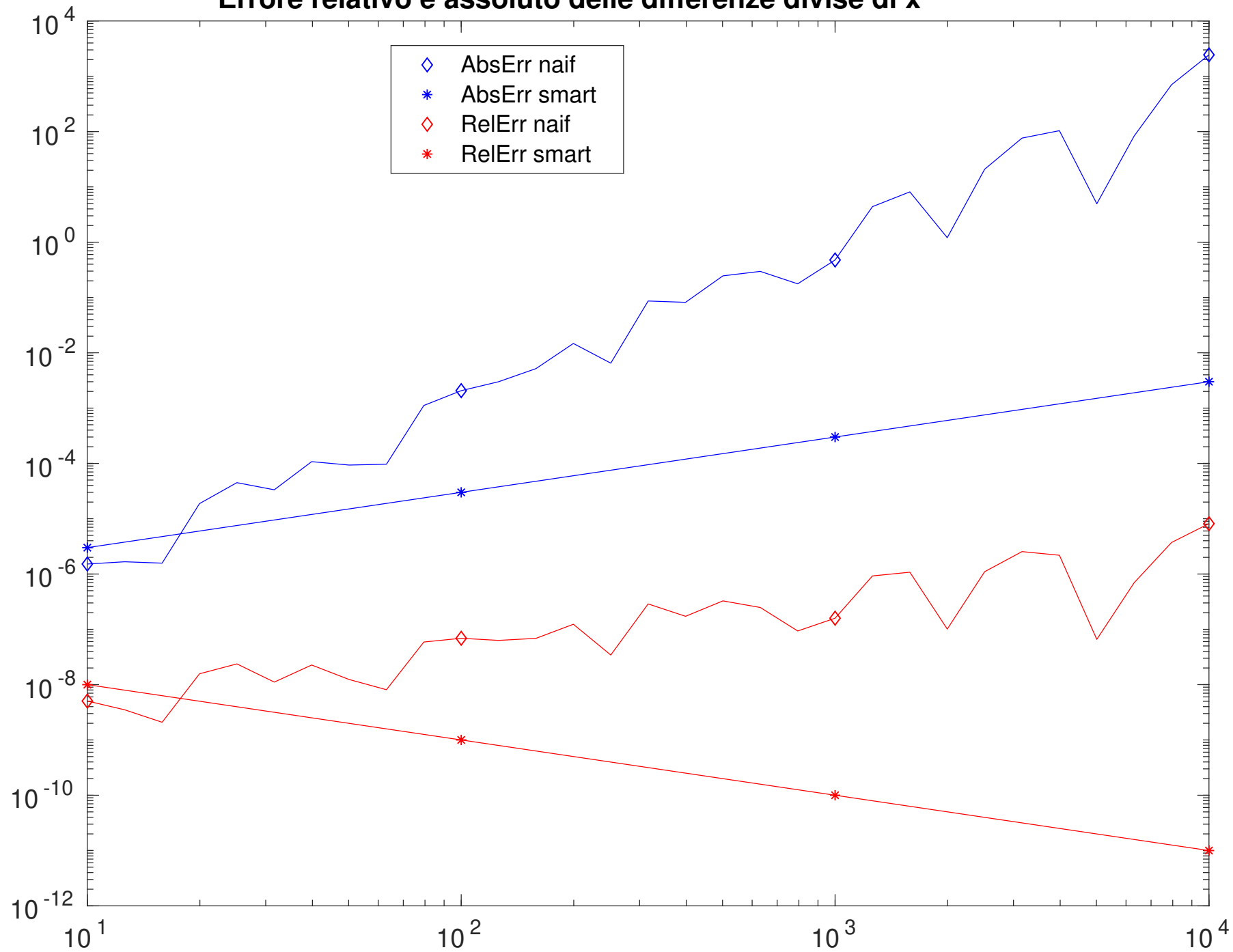
```
1 asse_x = logspace(1,4,10*(4-1)+1);
2 fine = 1:10:40;
3 h = 10^-7;
4 differenza_divisa = ((asse_x+h).^3-(asse_x).^3)/h;
5 differenza_divisa_no_err = 3*(asse_x).^2 + 3*asse_x*h + h^2;
```

Calcoliamo gli errori ed infine disegniamo i grafici richiesti

```
1 err_assoluto = abs(differenza_divisa - 3*asse_x.^2);
2 err_assoluto_smart = abs(differenza_divisa_no_err - 3*asse_x.^2);
3 err_relativo = err_assoluto./abs(3*asse_x.^2);
4 err_relativo_smart = err_assoluto_smart./abs(3*asse_x.^2);
5 figure();
6 loglog( 10.^(1:4),err_assoluto(fine),'bd', ...
7         10.^(1:4),err_assoluto_smart(fine),'b*',...
8         10.^(1:4),err_relativo(fine),'rd',...
9         10.^(1:4),err_relativo_smart(fine),'r*',...
10        asse_x,err_assoluto,'-b',...
11        asse_x,err_assoluto_smart,'-b',...
12        asse_x,err_relativo,'-r',...
13        asse_x,err_relativo_smart,'-r');
14 legend({'AbsErr naif','AbsErr smart','RelErr naif','RelErr smart'},'Location','best');
15 title('Errore relativo e assoluto delle differenze divise di x^3');
```

Errore relativo e assoluto delle differenze divise di x

3



Ripetendo lo stesso esercizio con $f(x) = x^5$ otteniamo

$$f[x_0, x_0 + h] = \frac{f(x_0 + h) - f(x_0)}{h} = \frac{(x_0 + h)^5 - x_0^5}{h} = \dots = 5x_0^4 + 10x_0^3h + 10x_0^2h^2 + 5x_0h^3 + h^4$$

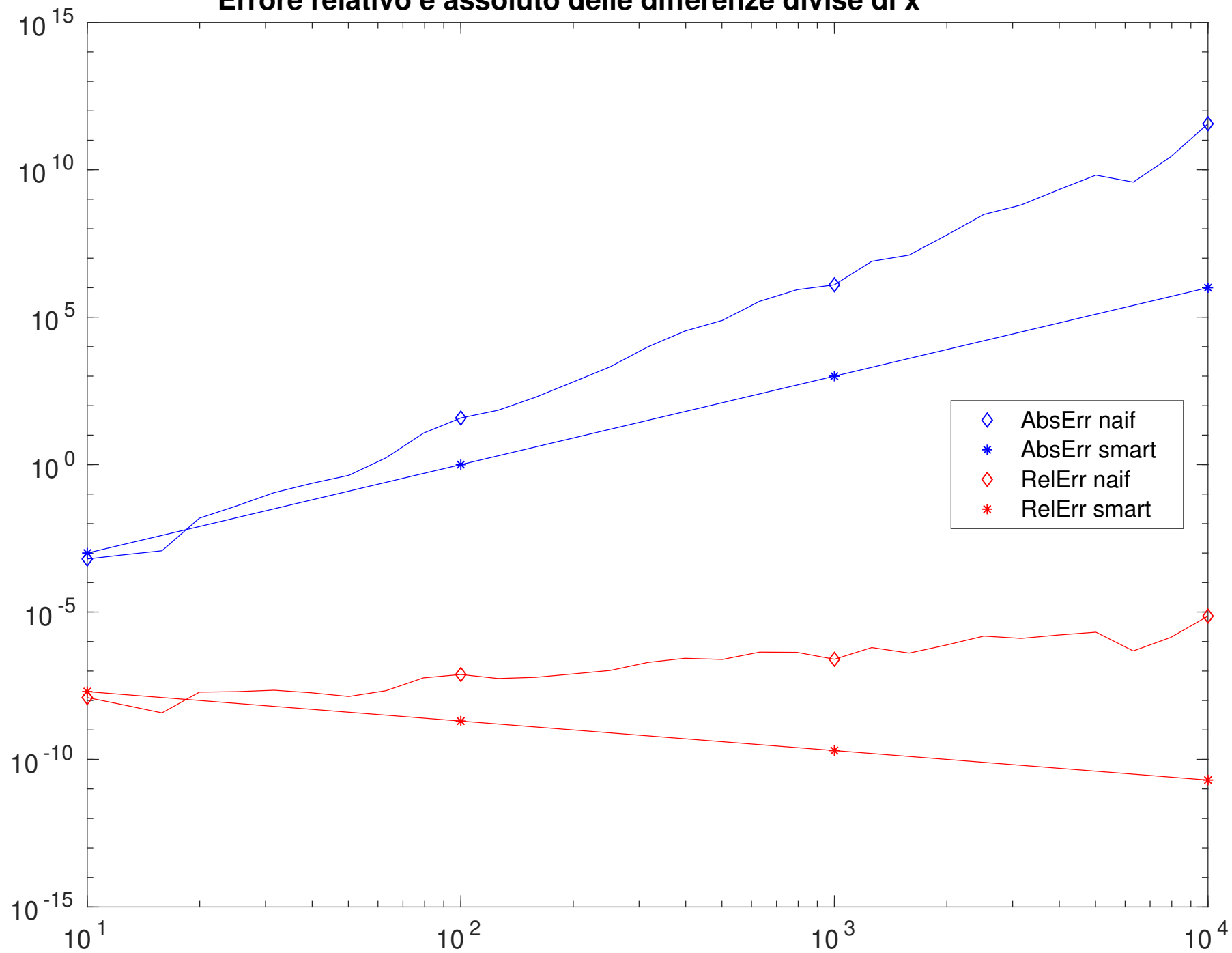
```

1 differenza_divisa      = ((asse_x+h).^5-(asse_x).^5)/h;
2 differenza_divisa_no_err = 5*(asse_x).^4 + 10*asse_x.^3*h + 10*asse_x.^2*h^2 + h^4;
3 err_assoluto           = abs(differenza_divisa - 5*asse_x.^4);
4 err_assoluto_smart     = abs(differenza_divisa_no_err - 5*asse_x.^4);
5 err_relativo           = err_assoluto./abs(5*asse_x.^4);
6 err_relativo_smart     = err_assoluto_smart./abs(5*asse_x.^4);
7 figure();
8 loglog(10.^(1:4),err_assoluto(fine),'bd', ...
9      10.^(1:4),err_assoluto_smart(fine),'b*',...
10     10.^(1:4),err_relativo(fine),'rd',...
11     10.^(1:4),err_relativo_smart(fine),'r*',...
12     asse_x,err_assoluto,'-b',...
13     asse_x,err_assoluto_smart,'-b',...
14     asse_x,err_relativo,'-r',...
15     asse_x,err_relativo_smart,'-r');
16 legend({'AbsErr naif','AbsErr smart','RelErr naif','RelErr smart'},'Location','best');
17 title('Errore relativo e assoluto delle differenze divise di x^5');

```

Errore relativo e assoluto delle differenze divise di x

5



Vettori, matrici e sistemi lineari - 8

Viene data una matrice definita così:

$$A = \begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix} \quad \text{dove} \quad A_{11} \in \mathbb{R}^{p \times p}, A_{12} \in \mathbb{R}^{p \times q}, A_{22} \in \mathbb{R}^{q \times q}$$

Viene chiesto di calcolare in funzione di q il numero di moltiplicazioni totali necessarie per la risoluzione del sistema lineare (eliminazione gaussiana più sostituzione all'indietro).

Considerando la matrice A come una semplice matrice di ordine $n = p + q$, è ben noto che sono necessarie circa $\frac{1}{3}n^3 + \frac{1}{2}n^2$ moltiplicazioni per la completa risoluzione di un sistema lineare. Ovvero, in funzione di q si ha $\frac{1}{3}(q+p)^3 + \frac{1}{2}(p+q)^2$.

Invece, utilizzando la decomposizione a blocchi ed immaginando anche \vec{x} e \vec{b} a blocchi, si nota che:

$$\begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \implies \begin{bmatrix} A_{11}x_1 + A_{12}x_2 \\ A_{22}x_2 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$$

Dalla equazione inferiore notiamo che può essere calcolato x_2 tramite un sistema lineare di ordine q , ovvero con $\frac{1}{3}q^3 + \frac{1}{2}q^2$ moltiplicazioni. Poi, sostituendo nell'equazione superiore, si ricava $A_{11} x_1 = b_1 - A_{12} x_2$. Tale operazione richiede il prodotto matrice vettore, ovvero $p * q$ moltiplicazioni. Anche questo sistema è lineare e si risolve con $\frac{1}{3}p^3 + \frac{1}{2}p^2$ moltiplicazioni. Ovvero utilizzando la forma a blocchi della matrice A , si risolve il sistema lineare in $\frac{1}{3}q^3 + \frac{1}{2}q^2 + \frac{1}{3}p^3 + \frac{1}{2}p^2 + p * q$ moltiplicazioni.

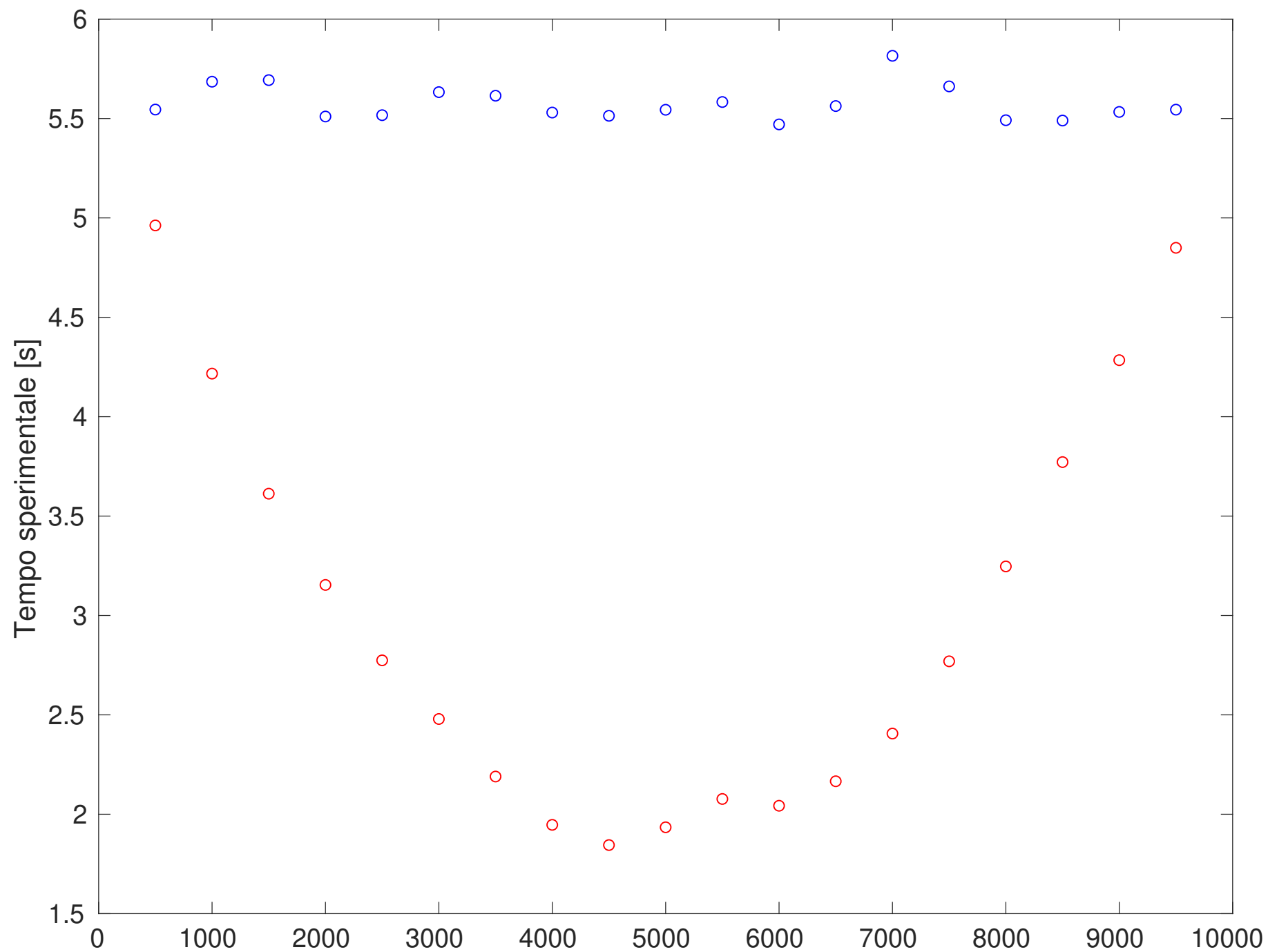
```
1 tempo_teorico_mod01 = @(q,n) ones(size(q)).*(n.^3/3 + n.^2/2);  
2 tempo_teorico_mod02 = @(q,n) q.^3/3 + q.^2/2 + (n-q).^3/3 + (n-q).^2/2 + (n-q).*q;
```

Vettori, matrici e sistemi lineari - 9

Come richiesto, effettuiamo la misurazione dei tempi di esecuzione dei due metodi di risoluzione. Riportiamo quindi il grafico dei tempi ottenuti

```
1  n = 10000;
2  s = 500;
3  asse_x = s:s:(n-s);
4  tempi_mod01 = zeros(n/s-1,1);
5  tempi_mod02 = zeros(n/s-1,1);
6  for q = asse_x
7      p = n - q;
8      A11 = rand(p,p);
9      A12 = rand(p,q);
10     A22 = rand(q,q);
11     b1 = ones(p,1);
12     b2 = ones(q,1);
13     A = [A11,A12;zeros(q,p),A22];
14     b = [b1;b2];
15
16     tic;
17     x = A\b;
18     tempi_mod01(q/s) = toc();
19
20     tic;
21     x2 = A22\b2;
22     x1 = A11\b1-A12*x2;
23     tempi_mod02(q/s) = toc();
24 end
25 figure();
26 plot(asse_x,tempi_mod01,'ob',asse_x,tempi_mod02,'or');
27 ylabel('Tempo sperimentale [s]');
```

Come si può notare dal grafico seguente, il valore di q per il quale si ottiene il maggior vantaggio è $q \equiv 5000$, ovvero quando $q \approx \frac{n}{2}$.



Vettori, matrici e sistemi lineari - 10

Con riferimento al punto precedente, aggiungiamo al grafico le curve teoriche ricavate al punto 8, opportunamente riscalate.

```
1 yyaxis right;  
2 plot( asse_x,tempo_teorico_modo1(asse_x,n), '--b',asse_x,tempo_teorico_modo2(asse_x,n), '--r');  
3 ylabel('Tempo teorico --');
```

