Esame di Programmazione Informatica

16 febbraio 2022

Esercizio 1 (14/30)

Si consideri una generica funzione del tempo $f(t)$ da campionare agli istanti di tempo t_i ad intervalli regolari Δt, cioè $t_i = i\Delta t$. Gli istanti di tempo coprono l'intervallo $[0, T]$, cioè vanno da $t_0 = 0$ a $t_N = N\Delta t$, con t_N l'ultimo istante di tempo minore o uguale a T.

Come visto a lezione, il campionamento si effettua quindi valutando la funzione $f(t)$ agli istanti di tempo considerati, cioè $f_i = f(t_i)$ con $i = 0, \ldots, N$.

Si può definire l'energia E del precedente segnale campionato come:

$$E = \sum_{i=0}^{N} f_i^2$$

Scrivere una funzione MATLAB che prenda in ingresso:

- il function handle della funzione $f(t)$
- l'intervallo di campionamento Δt
- il tempo finale T

e restituisca in uscita:

- il vettore $\mathbf{f} = [f_0, f_1, \ldots, f_N]$ dei valori della funzione campionata agli istanti t_i
- l'energia E del segnale campionato

Si mostri poi come applicare la precedente funzione nel caso di $f(t) = \sin(100\pi t)$ (si può utilizzare sia una funzione anonima che una funzione esterna, ma è consigliata la prima), $\Delta t = 1/100$ e $T = 1$. Che energia E si ottiene?

Soluzione: come spesso accade, conviene utilizzare le operazioni vettoriali. Il vettore degli istanti t_i può essere generato come al solito con l'istruzione `0:dt:T` e le altre due operazioni, ossia il campionamento ed il calcolo di E, possono quindi essere fatte direttamente in maniera vettoriale:

campionamento_E.m

```
function [fi, E] = campionamento_E(f, dt, T)
   ti = 0:dt:T ;
   fi = f(ti) ;
   E  = sum(fi .^ 2) ;
end
```

Si poteva equivalentemente scrivere un ciclo `for` che scorre su tutti gli istanti t_i, calcolando f_i e la somma richiesta E un termine per volta.

Utilizzo della precedente funzione nel caso di $f(t) = \sin(100\pi t)$, implementata tramite funzione anonima, $\Delta t = 1/100$ e $T = 1$.

```
f  = @(t) sin(100*pi*t) ;
dt = 1/100 ;
T  = 1 ;
[f_vettore, E] = campionamento_E(f, dt, T) ;
```

ottenendo $f_i = E = 0$ (questo succede perchè la frequenza di campionamento $1/\Delta t = 100$ non è sufficiente a campionare un segnale armonico di frequenza 50: si hanno infatti campioni tutti nulli, $f_i = \sin(i\pi) = 0$).

Esercizio 2 (14/30)

Dato un generico poligono con i vertici ordinati in senso antiorario, la seguente funzione MATLAB calcola l'angolo interno α dati tre vertici consecutivi A, B e C, come da seguente figura:

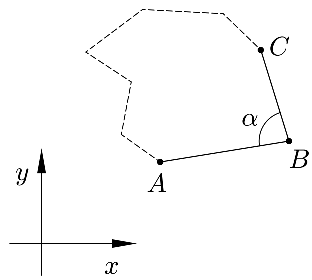

Figura 1: Angolo interno α in un generico poligono.

```
function alfa = angolo_interno(A, B, C)
   M = B - A ;
   N = C - B ;
   g = acos( dot(M,N) / ( norm(M)*norm(N) ) ) ;
   alfa = pi + g*sign(M(2)*N(1)-M(1)*N(2)) ;
end
```

dove, come al solito, A, B e C sono tutti vettori di lunghezza 2, cioè le 2 coordinate cartesiane del vertice considerato.

Utilizzare la precedente funzione per calcolare la somma degli angoli interni del pentagono di vertici $V_1 = (0, 0)$, $V_2 = (2, 0)$, $V_3 = (2, 2)$, $V_4 = (1, 3)$, $V_5 = (0, 2)$. Non è necessario scrivere una funzione. L'utilizzo di un ciclo `for` varrà valutato con più punti: in tal caso conviene utilizzare un vettore **x** delle coordinate x di tutti i vertici ed un altro vettore **y** delle coordinate y di tutti i vertici, oppure, ancora meglio, una singola matrice **xy** contenente entrambe le coordinate di tutti i vertici.

In un poligono di n vertici la somma degli angoli interni è $(n-2)\pi$. La somma calcolata coincide con la teoria?

Soluzione: volendo operare attraverso un ciclo `for`, conviene quindi inserire le coordinate dei vertici in una matrice `xy`. Aggiungiamo poi per comodità in testa ed in coda alla matrice le coordinate dell'ultimo e del primo vertice, rispettivamente:

```
xy = [ 0 2 2 1 0 ;
       0 0 2 3 2 ] ;
xy = [ xy(:,end) xy xy(:,1) ] ;
```

Calcoliamo con un ciclo `for` gli angoli interni uno a uno, scorrendo quindi i 5 angoli interni relativi a ciascun vertice del pentagono con un indice i che va da 1 a 5. Di volta in volta i vertici consecutivi A, B e C saranno estratti dalla matrice `xy` attraverso indici consecutivi, cioè $i, i+1$ e $i+2$:

```
somma_alfa = 0 ;
for i = 1 : 5
   A = xy(:,i) ;
   B = xy(:,i+1) ;
   C = xy(:,i+2) ;
   somma_alfa = somma_alfa + angolo_interno(A, B, C) ;
end
```

Alternativamente, si poteva procedere in maniera "manuale" calcolando esplicitamente gli angoli uno a uno, partendo dai singoli vertici memorizzati separatamente:

```
V1 = [0 0] ;
...
V5 = [0 2] ;

alfa(1) = angolo_interno(V5, V1, V2) ;
alfa(2) = angolo_interno(V1, V2, V3) ;
...
alfa(5) = angolo_interno(V4, V5, V1) ;
somma_alfa = sum(alfa)
```

ottenendo esattamente una somma di $3\pi = (n-2)\pi$, coincidente con la teoria.

Domande a risposta multipla (5/30)

Domanda 1 (2/30)

In un calcolatore, la rappresentazione di un numero in virgola mobile (*floating point*):

- Viene utilizzata perchè permette di rappresentare esattamente qualsiasi numero
- Viene utilizzata perchè nel calcolatore è possibile solo utilizzare quantità binarie (bit)
- Non può essere utilizzata nelle CPU e quindi si utilizza solo nei calcoli a mano
- Comporta quasi sempre un certo errore

Soluzione: Comporta quasi sempre un certo errore.

Domanda 2 (3/30)

Data la seguente funzione MATLAB, dove $\mathbf{x} = [x_1, \ldots, x_{20}]$ è un vettore di 20 elementi:

```
function y = f(x)
   y = sin(x) ;
   if y(1) > 0
      x = abs(y) ;
   end
   y = sum(y) ;
end
```

che funzione matematica essa implementa?

- $\sum_{i=1}^{20} \sin x_i$
- $\sum_{i=1}^{20} |x_i|$
- $|\sin x|$
- $\sum_{i=1}^{20} \sin(|\sin x_i|)$ se $\sin x_1 > 0$, $\sum_{i=1}^{20} \sin x_i$ altrimenti

Soluzione: $\sum_{i=1}^{20} \sin x_i$