Laboratorio di Telecomunicazioni - a.a. 2010/2011
Lezione n. 4

Analisi dei sistemi nel dominio del tempo

L.Verdoliva

In questa terza lezione ci occuperemo dell'elaborazione di segnali tempo discreto mediante sistemi LTI (convoluzione) e della realizzazione in Matlab di sistemi descritti da equazioni alle differenze a coefficienti costanti (ARMA).

1 Sistemi LTI

I sistemi lineari tempo invarianti (LTI) costituiscono la più importante classe di sistemi, sia perché modellano adeguatamente molte situazioni reali, sia perché risultano completamente caratterizzati dalla conoscenza della risposta impulsiva, $h(n) = \mathcal{T}[\delta(n)]$; in particolare l'uscita di un sistema LTI è data dalla somma di convoluzione tra l'ingresso $x(n)$ e $h(n)$:

$$y(n) = x(n) * h(n) = \sum_{k=-\infty}^{+\infty} x(k)h(n-k)$$

Facendo un cambio di variabili, si può facilmente verificare che la convoluzione è commutativa, cioè $x(n)*h(n) = h(n)*x(n)$. In Matlab esiste già un comando per il calcolo della convoluzione, la cui sintassi è la seguente:

```
>> y=conv(x,h)
```

Calcolate la convoluzione tra i seguenti segnali: $x(n) = \delta(n+1) + \delta(n) + \delta(n-1) + \delta(n-2)$ e $h(n) = \frac{1}{2}\delta(n) + \frac{1}{2}\delta(n-1)$ e verificate che scambiando i ruoli di $x(n)$ e $h(n)$ il risultato non cambia. Anche in questo caso (come per la correlazione) l'uscita non dà indicazioni sull'istante iniziale e finale della sequenza. Provate allora a scrivere una funzione con il seguente prototipo:

```
function [y,ny]=convoluzione(x,nx,h,nh);
```

tenendo presente che il risultato della convoluzione è compreso nell'intervallo $[n_{xi}+n_{hi}, n_{xf}+n_{hf}]$, dove n_{xi}, n_{hi}, n_{xf} e n_{hf} sono gli istanti iniziali e finali di $x(n)$ e $h(n)$.

1.1 Esempi ed esperimenti proposti

a) *Convoluzione tra due finestre rettangolari*

Siano $x(n)$ e $h(n)$ due finestre rettangolari di lunghezza N e M, rispettivamente, cioè $x(n) = \mathcal{R}_N(n)$ e $h(n) = \mathcal{R}_M(n)$. Per calcolare l'uscita $y(n) = x(n) * h(n)$, per $N = 4$ e $M = 4$.

Ripetete l'esperimento aumentando la lunghezza del vettore di ingresso, e verificate che la convoluzione di due sequenze rettangolari di lunghezza N e M è una sequenza trapezoidale di durata $L = M + N - 1$, cioè a causa della memoria del filtro la lunghezza del segnale in ingresso si allunga di $M - 1$ campioni. Questo risultato è valido per due sequenze generiche purchè abbiano entrambe durata finita.

b) *Convoluzione tra un gradino e una sequenza esponenziale*

Si consideri adesso il calcolo della convoluzione tra $x(n) = u(n)$ e $h(n) = a^n\, u(n)$, con $0 < a < 1$. Analiticamente risulta:

$$y(n) = \sum_{k=-\infty}^{+\infty} h(k)x(n-k) = \sum_{k=-\infty}^{+\infty} a^k\, u(k)\, u(n-k) = \sum_{k=0}^{n} a^k = \frac{1 - a^{n+1}}{1 - a} \qquad n \geq 0$$

Per $a = 1/2$ verificate col Matlab che l'uscita tende a $\frac{1}{1-a} = 2$ all'aumentare di n.

c) *Convoluzione tra sequenze di durata finita*

Si consideri adesso il calcolo della convoluzione tra $x(n) = \mathcal{R}_5(n)$ e $h(n) = 2^n\, \mathcal{R}_7(n)$. Determinare la soluzione analitica e verificarne la correttezza con il Matlab.

I sistemi LTI sono generalmente classificati in sistemi con risposta impulsiva a durata finita (FIR) e sistemi con risposta impulsiva a durata infinita (IIR).

1.2 Sistemi FIR

Consideriamo per semplicità un filtro FIR causale e supponiamo che $h(n)$ abbia durata M (filtro di ordine M), allora risulta:

$$y(n) = \sum_{m=0}^{M} h(m)x(n-m) = \sum_{m=0}^{M} b_m x(n-m)$$

dove si è posto $h(m) = b_m$ per $m = 0, \cdots, M$. L'uscita del sistema è semplicemente una combinazione lineare pesata dei campioni di ingresso del segnale; il sistema opera quindi come una "finestra" che vede solo i più recenti M campioni per determinare l'uscita. Per questo motivo i sistemi FIR sono spesso chiamati filtri a media mobile (Moving-Average, MA). Essendo finita la memoria del sistema, sono facilmente realizzabili con un numero finito di locazioni di memoria, moltiplicatori e addizionatori.

Realizzate in matlab la convoluzione tra il segnale $x(n) = 4\,\mathcal{R}_{11}(n) + 12\,\mathcal{R}_5(n)$ e il filtro con coefficienti $b_0 = 1/2$ e $b_1 = 1/2$, l'uscita risulta quindi in ogni istante la media aritmetica degli ultimi due valori della sequenza di ingresso $x(n)$. Ripetete l'operazione di filtraggio nel caso in cui $b_0 = 1/2$ e $b_1 = -1/2$. Come cambia l'uscita? Che comportamento hanno questi due filtri?

1.3 Sistemi IIR

I filtri IIR hanno risposta impulsiva a durata infinita pertanto non sono fisicamente realizzabili, tuttavia esiste una sottoclasse di tali sistemi (che include anche i filtri FIR) che ammette pratica realizzazione grazie all'uso della ricorsione. Questi sistemi possono essere descritti da un'*equazione alle differenze* a coefficienti costanti, del tipo:

$$\sum_{k=0}^{N} a_k\, y(n-k) = \sum_{m=0}^{M} b_m\, x(n-m), \qquad a_0 \neq 0 \tag{1}$$

Supporremo che il sistema evolva da una condizione iniziale di riposo (condizioni iniziali omogenee). Si può dimostrare allora che in questo caso l'equazione alle differenze definisce un sistema lineare e tempo invariante (LTI), denominato *sistema ARMA* (Auto-Regressive Moving-Average, cioè auto-regressivo a media mobile). L'equazione (1), assumendo per semplicità $a_0 = 1$, può essere riscritta nella forma:

$$y(n) = \underbrace{-\sum_{k=1}^{N} a_k\, y(n-k)}_{\text{AR}} + \underbrace{\sum_{m=0}^{M} b_m\, x(n-m)}_{\text{MA}}, \tag{2}$$

nella quale abbiamo evidenziato la parte ricorsiva o *autoregressiva* (AR) e la parte non ricorsiva o a *media mobile* (MA). Tale forma consente di calcolare l'uscita $y(n)$ in maniera esplicita, a partire dai valori dell'ingresso (parte MA) e dai valori precedentemente calcolati dell'uscita stessa (parte AR).
Si possono considerare i seguenti casi particolari di sistema ARMA:

$$\begin{array}{llll} M = 1, b_0 \neq 0 & y(n) = -\sum_{k=1}^{N} a_k\, y(n-k) + b_0\, x(n) & \text{sistema AR} \\ N = 0, a_0 \neq 0 & y(n) = \sum_{m=0}^{M} b_m\, x(n-m) & \text{sistema MA} \end{array}$$

In Matlab, per risolvere equazioni alle differenze del tipo (1), è disponibile il comando `filter`, che può essere utilizzato per implementare filtri ARMA, AR o MA. La sintassi di tale comando è la seguente:

```
>> y=filter(b,a,x);
```

dove il vettore `b` contiene i coefficienti $b_0, b_1, \ldots, b_M$, il vettore `a` contiene, invece, i coefficienti $a_0, a_1, \ldots, a_N$, ed il segnale in ingresso è contenuto nel vettore `x`. Si noti che in questo modo si assume implicitamente che le condizioni iniziali da cui evolve il sistema siano nulle; in ogni caso, il comando `filter` prevede anche la possibilità di specificare le condizioni iniziali (ma noi non faremo uso di questa possibilità, si veda l'help in linea per ulteriori dettagli). Richiamiamo l'attenzione sul fatto che bisogna porre il filtro nella forma (1) per identificare correttamente i coefficienti `b` ed `a`; per intenderci, al primo membro della (1) compare solo l'uscita, mentre al secondo membro compare solo l'ingresso.

Si noti che l'uscita `y` calcolata dal comando `filter` è un vettore della stessa lunghezza di `x` (a rigore dovrebbe essere più lunga, per effetto della dispersione temporale introdotta dal sistema, cioè dalla convoluzione tra ingresso e risposta impulsiva). Come esempio provate a determinare la convoluzione tra due sequenze rettangolari di durata 4 con il comando `filter`, osserverete che l'uscita è stata troncata in modo da avere la stessa durata dell'ingresso. Di conseguenza, se si vuole osservare l'uscita anche per istanti di tempo successivi, bisogna allungare artificialmente l'ingresso aggiungendo degli zeri in coda.

Per filtri MA, per osservare correttamente tutto il segnale di uscita basta allungare il segnale d'ingresso con $M-1$ zeri, pari alla lunghezza del filtro MA meno 1. Supponiamo, per esempio, che il legame ingresso/uscita sia del tipo:

$$y(n) = x(n) + 2x(n-1) + x(n-2)$$

e di porre in ingresso la sequenza $[2, 1, -1, 0, 1, 4, 3]$, i comandi

```
>> a = [1]; b = [1 2 1]; y = filter(b,a,x);
```

restituiscono in y solo 7 dei 9 valori dell'uscita, mentre con la modifica

```
>> x = [x zeros(1,length(b)-1)]; y = filter(b,a,x);
```

viene generata l'uscita completa. Chiaramente quest'ultimo comando dà esattamente lo stesso risultato che si otterrebbe con il seguente codice:

```
>> h = b; y = conv(x,h);
```

Osserviamo inoltre che il comando `filter` non consente di specificare l'asse dei tempi su cui è assegnato l'ingresso: tuttavia, si può osservare che, data la natura *tempo-invariante* del filtro, l'uscita è calcolata esattamente *negli stessi istanti* in cui è definito (implicitamente o esplicitamente) l'ingresso.
Consideriamo adesso un filtro AR in cui il legame ingresso/uscita è:

$$y(n) - y(n-1) + 0.9\, y(n-2) = x(n)$$

In questo caso, è semplice ricavare l'uscita con i seguenti comandi:

```
>> a = [1, -1, 0.9]; b = [1]; y = filter(b,a,x);
```

Se volessimo determinare la risposta impulsiva (teoricamente di durata infinita) nell'intervallo $[-20 : 120]$ dovremmo digitare il seguente codice:

```
>> n = [-20:120]; x = [(n == 0)]; h = filter(b,a,x);
>> subplot(211); stem(n,x); title('Ingresso x(n)');
>> subplot(212); stem(n,h); title('Risposta impulsiva');
```

E' chiaro che se aumentiamo la durata del vettore dei tempi ci avviciniamo all'espressione corretta della risposta impulsiva. Provate adesso a calcolare e diagrammare la risposta al gradino.

1.4 Esempi ed esperimenti proposti

a) *Sistema AR.* Sia $y(n) = \frac{2}{3}\, y(n-1) + x(n)$. Calcolate e diagrammate la risposta impulsiva $h(n)$ per $n = -5, \ldots, 20$ (ricordate di porre prima l'equazione nella forma (1) per identificare correttamente i coefficienti da passare alla funzione `filter`); verificate che la risposta impulsiva si può esprimere come $h(n) = \left(\frac{2}{3}\right)^n\, u(n)$.

b) *Sistema ARMA.* Sia Sia $y(n) = \frac{2}{3}\, y(n-1) + x(n) + \frac{1}{2}x(n-1)$. Calcolate e diagrammate la risposta impulsiva $h(n)$ per $0 \leq n \leq 50$; verificate che la risposta impulsiva si può esprimere come $h(n) = \left(\frac{2}{3}\right)^n\, u(n) + \frac{1}{2}\left(\frac{2}{3}\right)^{n-1}\, u(n-1)$.

c) *Filtraggio di un'interferenza sinusoidale.* Si consideri il seguente segnale per $0 \leq n \leq 40$:

$$y(n) = x(n) + i(n)$$

dove $x(n)$ è il segnale utile e $i(n) = A\cos(2\pi\nu_0 n + \theta)$ è un segnale interferente sinusoidale.

- Se $x(n) = V$, $A = 1$, $\nu_0 = 1/2$ e $\theta = 0$, verificate che il filtro con risposta impulsiva $h(n) = \frac{1}{2}\delta(n) + \frac{1}{2}\delta(n-1)$ è in grado di rimuovere perfettamente l'interferenza per ogni valore della costante V;
- Se $x(n) = (1.02)^n$, $A = 1/2$, $\nu_0 = 1/8$ e $\theta = \pi/4$, valutate l'effetto di un filtro con risposta impulsiva $h(n) = \frac{1}{M}\sum_{k=1}^{M}\delta(n-k)$ all'aumentare di M e commentate i risultati.