

Serie di Fourier, integratori, derivatori e gnuplot

fuso@df.unipi.it; <http://www.df.unipi.it/~fuso/dida>

(Dated: version 1 - FF, 24 novembre 2013)

Questa nota intende discutere l'uso di gnuplot per simulare funzioni periodiche tramite sviluppo in serie di Fourier e per simulare il funzionamento di circuiti integratori (o derivatori), come quelli realizzati nelle esercitazioni 5 e 7.

I. SVILUPPO IN SERIE DI FOURIER

Come si dimostra (in matematica), una funzione periodica $g(t)$, con periodo $T = 1/f = 2\pi/\omega$, può essere espressa come sovrapposizione lineare di una costante e di funzioni armoniche di pulsazione ω e $\omega_n = n\omega$ (dette rispettivamente armonica fondamentale e armoniche superiori di ordine n). Esistono vari modi per esprimere l'operazione, che dipendono fondamentalmente dall'impiego di grandezze reali o complesse. Per i nostri scopi il modo migliore consiste nell'usare una sovrapposizione (serie) di funzioni seno e coseno:

$$g(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} b_n \cos(\omega_n t) + \sum_{n=1}^{\infty} c_n \sin(\omega_n t), \quad (1)$$

dove il coefficiente a_0 tiene conto del valore medio della funzione (è nullo per funzioni "alternate") e i coefficienti di Fourier b_n e c_n si ricavano dalle espressioni

$$b_n = \frac{2}{T} \int_0^T g(t) \cos(\omega_n t) dt \quad (2)$$

$$c_n = \frac{2}{T} \int_0^T g(t) \sin(\omega_n t) dt. \quad (3)$$

Non è di certo questa la sede per entrare nel merito della tanta matematica che è coinvolta nell'argomento (definizioni, dimostrazioni, applicazioni, alcune strepitosamente importanti); mi piace però ricordare che, in qualche modo, l'approccio utilizzato, che serve a sviluppare su una base (volendo, ortonormale) una funzione *periodica*, può essere considerato un caso limite di un approccio più generale (analisi di Fourier) che permette di scrivere in forma integrale funzioni dipendenti in qualsiasi modo dal tempo. Tutto questo, se ancora non lo avete fatto, lo studierete in altri corsi (metodi, meccanica quantistica, campi, etc.).

A. Onda quadra e onda triangolare

I coefficienti di Fourier dipendono ovviamente dalla forma della funzione $g(t)$ che si sta considerando. Essi possono essere calcolati attraverso le relazioni date sopra, che richiedono di svolgere integrali sul tempo. Questi integrali possono essere facili o difficili a seconda della funzione considerata.

Ci sono due esempi molto significativi dal punto di vista pratico. Essi riguardano lo sviluppo in serie di Fourier dell'onda quadra e dell'onda triangolare.

Prendiamo come $g(t)$ un'onda quadra di ampiezza $1/2$ (niente fisica, per il momento...) e per comodità facciamo in modo che essa sia dispari; questo vuol dire per esempio che essa vale $-1/2$ per $(-T/2, 0)$ e $+1/2$ per $(0, T/2)$. In altre parole scegliamo per comodità l'origine dei tempi in modo tale che l'onda esca fuori proprio in questo modo. Calcolando i coefficienti di Fourier si vede che quelli di tipo b_n sono tutti nulli (per questioni di parità) e anche quelli di tipo c_n sono nulli per n pari. Dato che la funzione prescelta è alternata (a media nulla), è anche $a_0 = 0$, per cui i soli coefficienti di Fourier, come è facilissimo calcolare (provate!), sono solo

$$c_n = \frac{2}{n\pi}, \quad n \text{ dispari}. \quad (4)$$

Si può dimostrare con un po' di lavoro che, comunque si scelga l'origine dei tempi, lo sviluppo di un'onda quadra contiene *solo armoniche dispari*.

Per l'onda triangolare, che per semplicità supponiamo pari, a media nulla, e sempre di ampiezza $1/2$, i conti sono un po' più complicati. Il modo più semplice di ricordare i coefficienti di Fourier che valgono in questo caso è di immaginare l'onda triangolare come integrale (nel tempo) dell'onda quadra, che è proprio uno degli argomenti su cui verte questa nota. Di conseguenza i termini dello sviluppo che contengono seni diventano coseni e, integrando, si trova che per un'onda triangolare gli unici coefficienti non nulli sono i seguenti:

$$b_n = \frac{4}{(n\pi)^2}, \quad n \text{ dispari}. \quad (5)$$

II. GRAFICI DI SOMME DI FUNZIONI IN GNPLOT

Gnuplot non è propriamente lo strumento ideale per fare "simulazioni", cioè per prevedere in modo numerico grandezze che possono essere registrate sperimentalmente. In questo caso, in particolare, vogliamo verificare che onda quadra e onda triangolare possono essere ben approssimate dallo sviluppo di Fourier.

Scartabellando tra l'help di gnuplot ho trovato che il software è in grado di fare somme tramite il comando `sum[i=i0:ifin]`, dove i è un indice che viene incrementato di una unità a partire da $i0$ fino a $ifin$ (in sostanza la somma è fatta su un loop che parte da $i0$ e si interrompe in $ifin$). Questo comando può essere usato assieme a

quello di plot, per cui di fatto si possono graficare somme di funzioni (ganzo).

Due precisazioni importanti: come retaggio di quando i computer avevano poca memoria da allocare, di default gnuplot grafica le funzioni con un numero molto piccolo di punti. Poiché in ogni caso la serie di Fourier è un'approssimazione, è meglio non renderla ancor più approssimativa usando un numero di punti troppo piccolo! Pertanto è opportuno anteporre alle altre istruzioni di creazione del grafico il comando `set samples 10000`, o qualcosa del genere (10000 punti sono più che sufficienti!). Altra precisazione, o premessa, è che il senso di fare un'approssimazione è tanto più critico quanto più rognosa è la funzione da approssimare: l'onda quadra, che passa in modo estremamente brusco da un valore al suo opposto, è rognosa, per cui non aspettatevi risultati eclatanti.

Allora, per l'onda quadra ho scritto il comando `plot sum[i=0:50] sin((2*i+1)*f*2*pi*x)*(2/((2*i+1)*pi))`, il quale dice a gnuplot di graficare la *somma* di funzioni del tipo $\sin(n2\pi ft)$ con n dispari (infatti è posto pari a $(2i+1)$, con l'indice i che parte da zero) e f frequenza dell'onda (ovviamente c'è da dire prima a gnuplot quanto vale f , per esempio scrivendo `f=1e3` che significa $f = 1$ kHz), pesate per l'opportuno coefficiente di Fourier. Il risultato per la somma dei primi 50 termini (n va da 1 a 101, dispari) è mostrato in Fig. 1, dove si è supposto $f = 1$ kHz. Si vede che l'onda quadra è effettivamente abbastanza ben riprodotta, a parte un po' di problemini numerici proprio in prossimità dei punti a derivata tendente a infinito.

Simile operazione si può fare per graficare lo sviluppo di Fourier di un'onda triangolare. Il comando in questo caso è `plot sum[i=0:50] cos((2*i+1)*f*2*pi*x)*(4/((2*i+1)*pi)**2)` e il risultato, mostrato in Fig. 2, appare davvero ottimo: infatti bastano poche componenti per approssimare bene l'onda triangolare.

Anche voi dovete cercare di costruire onde di questo genere. Può essere molto istruttivo!

III. INTEGRATORI E DERIVATORI

Un filtro RC passa-basso (o passa-alto) può comportarsi da integratore (o derivatore) se usato nella banda di frequenze attenuata, cioè a frequenze molto maggiori (o molto minori) della frequenza di taglio f_T .

Le condizioni "limite" sono facilmente dimostrabili usando le espressioni delle funzioni (complesse) di trasferimento $T(f)$. Vi ricordo che queste funzioni di trasferimento erano state ottenute usando il metodo simbolico, cioè immaginando di poter esprimere le grandezze dipendenti dal tempo come fasori. Questo significa che esse valgono se le funzioni periodiche considerate sono di tipo seno o coseno, cioè per onde sinusoidali. Inoltre, per sem-

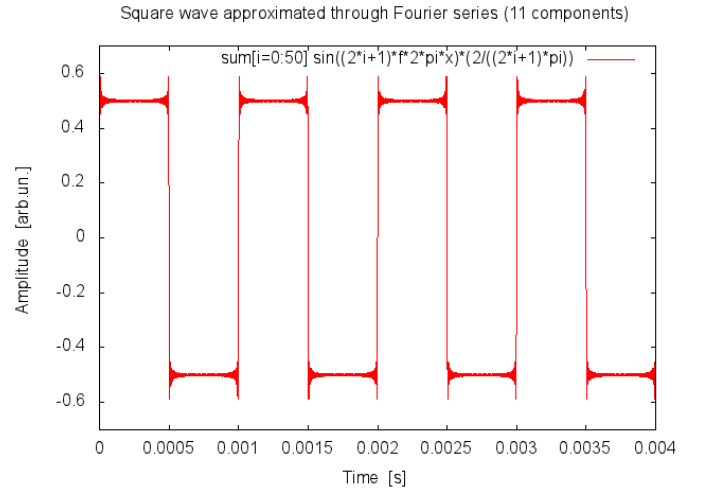


Figura 1. Sviluppo in serie di Fourier per un'onda quadra con $f = 1$ kHz ottenuto con gnuplot.

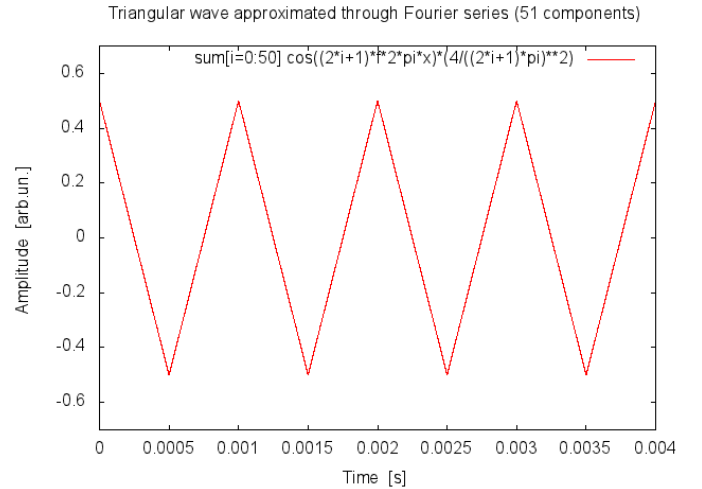


Figura 2. Sviluppo in serie di Fourier per un'onda triangolare con $f = 1$ kHz ottenuto con gnuplot.

plicità, considero il generatore come ideale, cioè trascuro la sua resistenza interna.

Per un passa-basso si ha $T(f) = 1/(1 + jf/f_T)$. Per $f \gg f_T$ si ha $T(f) \rightarrow 1/j(f/f_T)$, cioè la funzione di trasferimento tende a divenire puramente immaginaria. Nel mondo dei fasori si sa che la moltiplicazione per un termine immaginario del tipo $1/jf$, che se volete può essere scritto come $-j(1/f)$, equivale a *integrare* nel tempo il fasore di ingresso $V_{\omega, in}$. Se preferite, lo sfasamento corrispondente è $\Delta\phi = -\arctan(f/f_T) \rightarrow -\pi/2$, e sfasare di $-\pi/2$ una funzione seno o coseno significa ottenere un coseno o seno, cioè eseguire un'integrazione.[1]

Per un passa-alto si ha $T(f) = 1/(1 - jf/f_T)$. Per $f \ll f_T$ si ha $T(f) \rightarrow -1/j(f_T/f) = jf/f_T$, cioè la funzione di trasferimento tende a divenire anche in questo caso puramente immaginaria. Nel mondo dei fasori si sa

che la moltiplicazione per un termine immaginario del tipo jf equivale a *derivare* nel tempo il fasore di ingresso $V_{\omega, in}$. Se preferite, lo sfasamento corrispondente è $\Delta\phi = \arctan(f_T/f) \rightarrow \pi/2$, e sfasare di $\pi/2$ una funzione seno o coseno significa ottenere un coseno o seno, cioè eseguire una derivazione.

Naturalmente queste operazioni di integrazione (o derivazione) hanno un “costo”: infatti l’attenuazione del segnale è grande, essendo data da $A(f) \sim f_T/f$ (o $A(f) \sim f/f_T$), per cui l’ampiezza del segnale integrato (o derivato) tende a zero. Tuttavia, benché attenuato, il segnale subisce un’integrazione (o derivazione) temporale che può essere sfruttata praticamente, come dimostrato nell’esercitazione 7, dove un’onda quadra veniva inizialmente integrata assumendo una forma triangolare, e quindi di nuovo derivata ritornando alla forma quadra, con ampiezza ovviamente diminuita in maniera grosso modo proporzionale a $(f_{T, int}/f)(f/f_{T, der}) = f_{T, int}/f_{T, der}$, cioè al rapporto tra le frequenze di taglio del passa-alto (derivatore) e del passa-basso (integratore).

A. Regime “intermedio”

L’aspetto più interessante è probabilmente quello che coinvolge l’impiego di questi circuiti in condizioni di frequenza che non soddisfano pienamente le situazioni limite sopra descritte. Nel caso dell’integratore, che è il circuito che tratterò in dettaglio, voi avete già sperimentato cosa succede nel corso dell’esercitazione 5. Inviando un’onda quadra all’ingresso di un passa-basso, avete visto che, in funzione della frequenza, la forma d’onda in uscita passava da quadra (a basse frequenze) a triangolare (ad alte frequenze), con un’ampia regione di transizione nella quale la forma assomigliava a una pinna di squalo più o meno distorta, come da voi notato in modo colorito.

Mettiamo insieme quello che sappiamo: conosciamo le funzioni di trasferimento, cioè lo sfasamento e l’attenuazione, per funzioni armoniche. Sappiamo inoltre scomporre una funzione periodica di forma arbitraria, per esempio un’onda quadra, in armoniche. Allora, se applichiamo la funzione di trasferimento alle varie armoniche di cui è (approssimativamente) costituita l’onda in ingresso e poi ricomponiamo l’onda in uscita sovrapponendo le armoniche opportunamente trasferite (attenuate e sfasate), dovremmo essere in grado di approssimare la forma d’onda in uscita dall’integratore a qualsiasi frequenza.

Farlo a mano mi pare assai complicato. Farlo con gnuplot, pur se non agevolissimo, è molto efficace e divertente. Allora ho preso un’onda quadra scritta secondo lo sviluppo di Fourier, esattamente come specificato sopra. Ho simulato il passaggio di quest’onda quadra in un filtro passa basso, per il quale ho scelto la frequenza di taglio $f_T = 4.8$ kHz, corrispondente a quella del circuito montato in esercitazione 5, attenuando e sfasando opportunamente le componenti. Ricordo che l’attenuazione è $A(f) = f_T/f$, dunque le varie armoniche di frequenza $f_n = \omega_n/(2\pi) = nf$, con f frequenza fondamentale (è

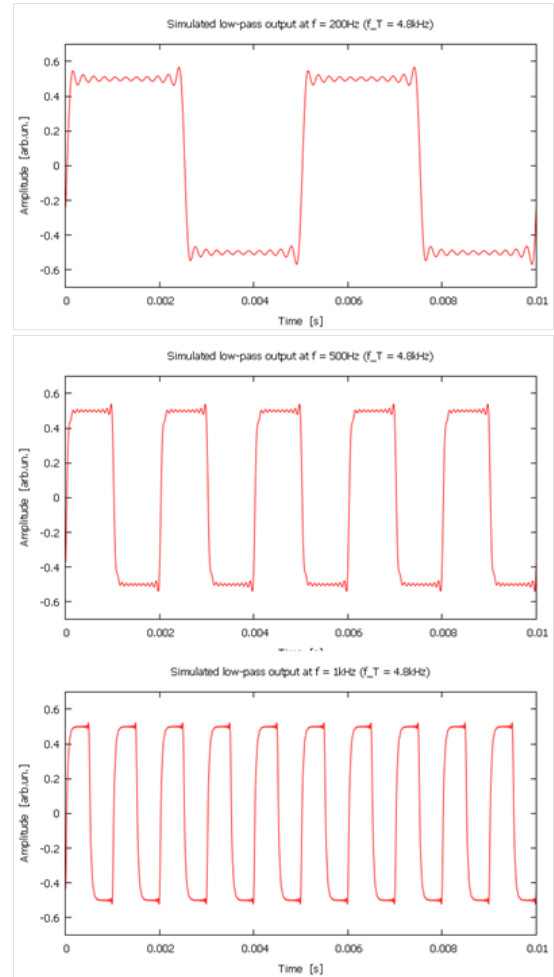


Figura 3. Risultato della simulazione descritta nel testo, dove si suppone che un’onda quadra, di frequenza come indicato nei grafici, venga inviata in ingresso a un integratore RC con $f_T = 4.8$ kHz.

quella dell’onda che in esercitazione 5 era prodotta dal generatore, che poi farò variare per simulare il comportamento del circuito a varie frequenze proprio come voi l’avete variata girando la manopola del generatore) vengono attenuate di $f_T/f_n = (f_T/f)(1/n)$. Il fatto che l’uscita dipenda in maniera inversamente proporzionale dal numero di armonica fa pensare che poche componenti siano sufficienti per approssimare bene l’onda in uscita, per cui nella simulazione mi limiterò a considerare le prime 21 armoniche.

Inoltre occorre tenere conto dello sfasamento $\Delta\phi = -\arctan(f/f_T)$, che, scritto per le varie armoniche, diventa $\Delta\phi_n = -\arctan(f_n/f_T) = -\arctan(nf/f_T)$. L’argomento dell’arcotangente cresce con n , dunque da una certa armonica in poi lo sfasamento tenderà a $-\pi/2$, cioè l’onda sarà effettivamente integrata.

Il comando di gnuplot è un po’ lungo da scrivere, ma credo facilmente comprensibile: `plot sum[i=0:10] sin((2*i+1)*2*pi*f*x-atan((2*i+1)*f/ft))* (2/((2*i+1)*pi))*(1/sqrt(1+((2*i+1)*f/ft)**2)).`

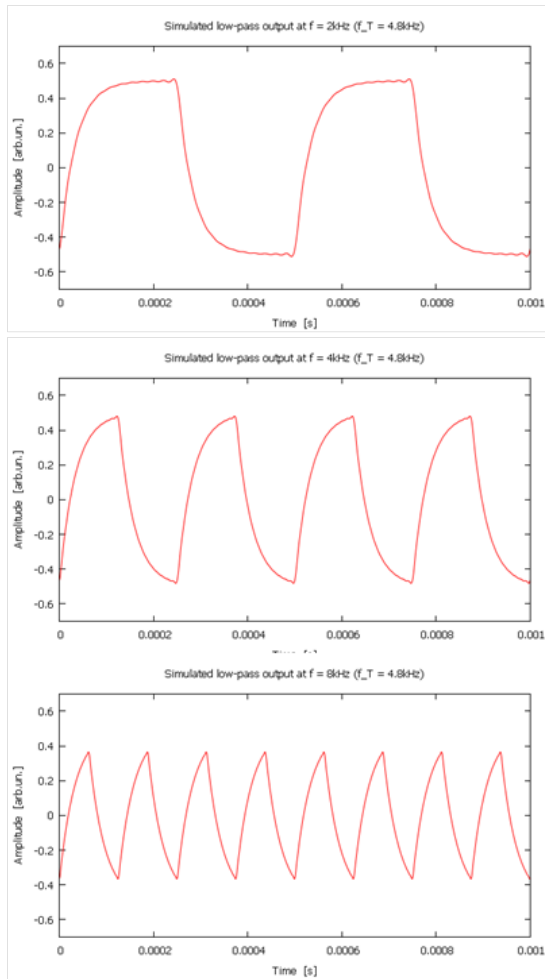


Figura 4. Analoga a Fig. 3, ma per frequenze maggiori (si notino le diverse scale degli assi).

I risultati per diversi valori della frequenza f sono riportati nelle Figs. 3 e 4 (notate le diverse scale orizzontali, praticamente ho simulato anche l'azione dell'aggiustare la base dei tempi dell'oscilloscopio). A me le forme d'onda ottenute paiono in buon accordo (qualitativo) con quanto avete osservato nell'esercitazione 5. In particolare la famosa pinna di squalo si vede molto bene.

Naturalmente andando ancora avanti con la frequenza la forma d'onda tende a diventare sempre più triangolare. Per esempio questo è evidente in Fig. 5, dove ho simulato il comportamento del circuito per $f = 10f_T$ (cioè 48 kHz): in queste condizioni l'onda in uscita è praticamente integrata “in modo completo”, cioè l'onda quadra diventa un'onda triangolare senza apprezzabili distorsioni. Allo stesso tempo si vede come l'ampiezza in uscita sia notevolmente attenuata rispetto all'ingresso (l'onda quadra in ingresso ha ampiezza pari a 1/2). Se si trattasse di onde sinusoidali, l'attenuazione dovrebbe essere pari a circa $f_T/f = 1/10$, mentre invece si vede che, usando un'onda quadra e dunque sovrapponendo le varie componenti attenuate di quanto devono, l'attenuazione è un po' minore

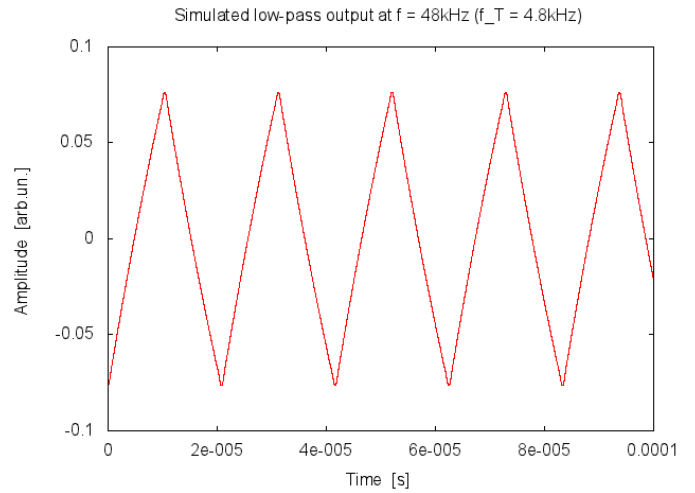


Figura 5. Risultato della simulazione descritta nel testo supponendo che l'onda quadra in ingresso abbia frequenza $f = 10f_T$.

(vale circa 1/7). Tuttavia come ordine di grandezza questo valore è simile a quanto predetto basandosi su forme d'onda sinusoidali.

Concludo con una raccomandazione: provate anche voi, magari a simulare il derivatore. Nella realtà degli esperimenti è sempre utilissimo poter contare su delle “simulazioni”, e spesso è più facile (e più divertente) di quanto si creda poterle eseguire in modo sufficientemente corretto!

[1] Lo stesso sfasamento, segno a parte, si ha derivando. Per distinguere tra integrale e derivata guardate dove finisce il

termine f , che qui è al denominatore per cui si tratta di integrazione.