

Filtri RC e Python

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

(Dated: version 3 - FF, 22 novembre 2014)

Questa nota riporta istruzioni e commenti sulla costruzione del diagramma di Bode usando Python e fa riferimento all'esperienza pratica di progettazione e costruzione di filtri RC "a un polo" (passa-basso o passa-alto, con un solo elemento resistivo e un solo elemento capacitivo).

I. INTRODUZIONE

Un filtro RC è un dispositivo passivo; infatti i suoi componenti non sono in grado di produrre amplificazione di segnali, per cui, in ogni caso, il filtro si comporta come un elemento che attenua, e sfasa, il segnale in uscita V_{out} rispetto a quello in ingresso V_{in} . La relazione tra uscita e ingresso è contenuta nella cosiddetta *funzione di trasferimento* o *funzione di risposta* del circuito, $T(\omega)$, generalmente complessa.

Nell'ambito del metodo simbolico, cioè supponendo segnali sinusoidali in uscita e in ingresso ed usando la forma fasoriale per esprimerli, si ha: $V_{\omega,out} = T(\omega)V_{\omega,in}$. Le ampiezze (ovvero i valori picco-picco, il comportamento è lo stesso per ampiezza e picco-picco) sono determinate dai moduli dei fasori considerati: $|V_{\omega,out}| = |T(\omega)V_{\omega,in}| = A(\omega)|V_{\omega,in}|$, dove abbiamo introdotto la funzione (reale) $A(\omega) = |T(\omega)|$ che, essendo pari al rapporto tra ampiezza in uscita e ampiezza in ingresso, mostra proprio di quanto l'uscita è attenuata rispetto all'ingresso. Infatti, per i filtri passivi, è sempre $A(\omega) \leq 1$.

Alla funzione $A(\omega)$ si dà spesso il nome di *attenuazione* del circuito. Dal punto di vista linguistico, ciò non è molto corretto, dato che, se fosse per ipotesi $A(\omega) > 1$, si dovrebbe parlare di fattore di amplificazione, o *guadagno* del circuito. Inoltre l'attenuazione è evidentemente maggiore per valori sempre più piccoli di $A(\omega)$, cosa che sicuramente non aiuta a fare chiarezza. Tuttavia useremo, qui e altrove, la denominazione di attenuazione per questa funzione.

A. decibel

Molto spesso si ha a che fare con attenuazioni che hanno valori numerici molto piccoli (ovvero amplificazioni con valori numerici molto alti). Un'unità di misura opportuna in tali situazioni è il *decibel* (dB). L'attenuazione dell'ampiezza del segnale in dB è definita nel seguente modo:

$$A(\omega) \text{ [dB]} = 20 \log_{10} \left(\frac{V_{out}}{V_{in}} \right), \quad (1)$$

dove V_{out} e V_{in} rappresentano le *ampiezze* dei segnali in uscita e in ingresso.[1]

II. PASSA-BASSO

Un filtro passa-basso di quelli considerati qui è composto dalla serie di un resistore e di un condensatore. L'ingresso è collegato al generatore di forme d'onda, l'uscita misura la d.d.p. ai capi del condensatore.

La funzione di trasferimento complessa per il circuito considerato (stiamo trascurando la resistenza interna del generatore e gli effetti di quella dello strumento di misura) è del tipo:

$$T(\omega) = \frac{1}{1 + j\omega RC} = \frac{1}{1 + j\omega/\omega_T}, \quad (2)$$

dove abbiamo definito la frequenza angolare di taglio $\omega_T = 1/(RC)$. Ricordando il legame tra frequenza angolare e frequenza, $\omega = 2\pi f$, e definendo la *frequenza di taglio* $f_T = \omega_T/(2\pi) = 1/(2\pi RC)$, si ha anche

$$T(f) = \frac{1}{1 + jf/f_T}. \quad (3)$$

Lo sfasamento $\Delta\phi$ tra i fasori $V_{\omega,out}$ e $V_{\omega,in}$ è dato dal rapporto tra parte immaginaria e parte reale della funzione di trasferimento, cioè

$$\tan(\Delta\phi) = \frac{\text{Im}\{T(f)\}}{\text{Re}\{T(f)\}} = -\frac{f}{f_T}, \quad (4)$$

Si ha $\Delta\phi \rightarrow 0$ per $f \rightarrow 0$, $\Delta\phi \rightarrow -\pi/2$ per $f \rightarrow \infty$; inoltre è $\Delta\phi = -\pi/4$ per $f = f_T$. Notate che i segni negativi che compaiono negli sfasamenti richiedono attenzione per essere determinati dal punto di vista sperimentale, per cui di fatto è spesso sufficiente, e generalmente più semplice, misurare tali sfasamenti in valore assoluto.

Per determinare l'attenuazione $A(f)$ delle ampiezze occorre considerare il *modulo* dei fasori (appunto le ampiezze delle forme d'onda), che, come già affermato, è determinato dal *modulo* della funzione di trasferimento. Si ha per un filtro passa-basso:

$$A(f) = |T(f)| = \frac{1}{\sqrt{1 + (f/f_T)^2}}. \quad (5)$$

Si vede facilmente che $A(f) \rightarrow 1$ per $f \rightarrow 0$, $A(f) \rightarrow 0$ per $f \rightarrow \infty$; inoltre è $A(f) = 1/\sqrt{2}$ per $f = f_T$. Spesso si definisce un'ulteriore frequenza caratteristica dei filtri, $f_{1/2}$, che equivale alla frequenza a cui l'attenuazione vale $A(f) = 1/2$. Si vede facilmente che $f_{1/2} = \sqrt{3}f_T$ (e inoltre a tale frequenza si ha $\Delta\phi = \pi/3$).

A. Misure sul filtro passa-basso

Ho costruito e fatto misure su un filtro passa-basso realizzato con $R = (3.27 \pm 0.03)$ kohm (misurata con multimetro digitale) e $C = 0.1 \mu\text{F}$ con tolleranza 10%. Per questo filtro ci si aspetta $f_{T,att} = (487 \pm 53)$ Hz, essendo l'incertezza dovuta principalmente alla tolleranza sulla capacità. Notate che l'uso di un valore di resistenza ben più alto della resistenza interna del generatore ($r = 50$ ohm) ci permette di considerare *approssimativamente* trascurabile quest'ultima, almeno fino a frequenze non “troppo” alte, e dunque di considerare ideale il generatore. La validità di questa approssimazione può essere verificata svolgendo i calcoli oppure, visto che siamo sperimentali, verificando che V_{in} non cambi “troppo”, né in ampiezza né in forma, quando la frequenza viene aumentata; nel mio caso non ho riscontrato distorsioni evidenti e l'ampiezza è rimasta costante entro qualche punto percentuale, cioè entro l'incertezza della misura dell'ampiezza stessa. Dunque d'ora in avanti considererò ideale il generatore.

In primo luogo ho determinato sperimentalmente la frequenza di taglio f_T : a questo scopo ho misurato l'ampiezza picco-picco del segnale in ingresso $V_{in} = (9.9 \pm 0.5)$ V, dove l'incertezza tiene conto dell'errore di calibrazione dato dal costruttore e della difficoltà di posizionare correttamente i cursori dell'oscilloscopio sui picchi. Quindi ho cercato al variare della frequenza (girando lentamente la manopola del generatore di forme d'onda) il valore per cui questa ampiezza si riduceva di un fattore $1/\sqrt{2}$, che è la frequenza di taglio. Ho ottenuto $f_T = (480 \pm 10)$ Hz, valore in accordo con le attese. Per determinare l'incertezza, che è dovuta non solo alla precisione con cui si determina la frequenza (il frequenzimetro integrato nell'oscilloscopio ha una precisione nominale dello 0.01%, quella del frequenzimetro integrato nel generatore di forme d'onda, debitamente termalizzato, è 20 ppm $\equiv 0.002\%$) o alla stabilità del generatore (che fa “ballare” qualche cifra significativa sui display dei frequenzimetri), ma soprattutto all'errore che si compie nella misura dell'ampiezza e nell'accuratezza con cui può essere determinata la frequenza generata (basta sfiorare la manopola per vedere sensibili cambiamenti di frequenza), ho ripetuto varie volte la misura e stimato di conseguenza l'incertezza. Ho anche misurato lo sfasamento a questa frequenza, ottenendo $\Delta\phi_T = (0.26 \pm 0.03) \pi$ rad, dove l'incertezza è dovuta principalmente alla difficoltà di individuare l'istante in cui le tracce passano per il livello di zero (le tracce sono spesse). Le incertezze determinate sono in entrambi i casi superiori alla precisione nominale dell'oscilloscopio (3%, nelle scale usate), che rappresenta in sostanza un errore sistematico, inadeguato a descrivere tutte le sorgenti di errore che hanno praticamente luogo.

Inoltre ho anche individuato la frequenza $f_{1/2}$ alla quale l'ampiezza in uscita è la metà di quella in ingresso, trovando $f_{1/2} = (876 \pm 12)$ Hz (sfasamento corrispondente $\Delta\phi_{1/2} = (0.32 \pm 0.03)\pi$ rad). I valori di f_T e $f_{1/2}$ sono in accordo con le attese ($f_T = 1/(2\pi RC)$ e $f_{1/2} = \sqrt{3}f_T$)

basate sui valori nominali di progetto. Inoltre gli sfasamenti misurati sono anche in accordo con le aspettative (rispettivamente $\pi/4$ e $\pi/3$).

Ho quindi misurato l'ampiezza picco-picco V_{out} del segnale in uscita dal filtro a diverse frequenze: avendo in mente di dover spazzare un range piuttosto ampio e di dover produrre un grafico in carta bilogarithmica, ho deciso di acquisire i dati a frequenze via via crescenti, l'una circa pari al doppio della precedente (cioè ho variato le frequenze di circa un'ottava alla volta). I dati sono mostrati in Fig. 1. Per le incertezze ho cercato di tenere conto sia dell'errore sistematico dovuto alla calibrazione dell'oscilloscopio, per altro usato su diverse scale, che dell'incertezza di misura dovuta ad esempio allo spessore delle tracce, alla presenza di rumore sovrapposto al segnale, alla difficoltà di individuare i livelli, etc.. Per la misura delle frequenze ho considerato un'incertezza pressoché trascurabile (inferiore all'1%), stimata osservando la cifra ballerina dei display.

Infine faccio notare un aspetto fastidioso dal punto di vista sperimentale, legato alla progressiva riduzione del valore di ampiezza dell'onda prodotta dal generatore (a prescindere dal carico, sembra essere un problema di mancata stabilizzazione in temperatura) nel corso delle misure. Per minimizzarne gli effetti ho cercato di eseguire le misure rapidamente.

B. Best-fit

Ho quindi eseguito un best-fit dei dati col metodo del minimo χ^2 usando la funzione Eq. 5 moltiplicata per V_{in} , cioè la

$$V(f) = \frac{V_{in}}{\sqrt{1 + (f/f_T)^2}}, \quad (6)$$

dove ho imposto V_{in} pari alla misura eseguita, senza incertezza, e lasciato come unico parametro libero f_T .

Ovviamente mi sono servito di Python, in particolare della routine di minimizzazione `curve_fit` che fa parte del pacchetto `scipy.optimize`. Ricordo che la routine restituisce nell'ordine un array unidimensionale (vettore) che contiene il valore dei parametri ottenuti dalla procedura di best-fit e un array bidimensionale (matrice) di covarianza, che ovviamente non è di interesse alcuno nel caso di fit a un solo parametro. Ricordo anche che gli argomenti utili per la chiamata `curve_fit` sono, nell'ordine: la funzione di fit, definita opportunamente nello script, l'array di valori della “variabile indipendente”, quello dei dati misurati, l'array dei valori iniziali dei parametri (costituito da un solo elemento nel caso di fit a un solo parametro), l'array delle σ_j , cioè dei valori da impiegare come peso per la minimizzazione del χ^2 . Visto che nel mio set di dati l'incertezza sulla misura di frequenza f_j (la “variabile indipendente”) è trascurabile rispetto a quella sulla misura della $V_{out,j}$, ho posto $\sigma_j = \Delta V_{out,j}$. Notate che la chiamata `curve_fit` ha un ulteriore argomento opzionale, che non serve per i nostri scopi, e quindi non viene

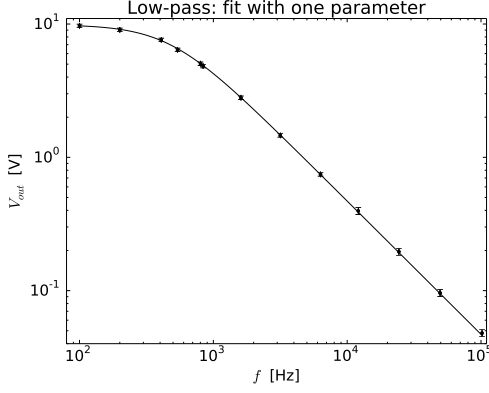


Figura 1. Dati sperimentali e risultato del best-fit a un parametro per le misure sul filtro passa-basso.

messo (cercate in rete il significato di questo ulteriore parametro per capire perché esso non serve). Per vostra comodità, lo script impiegato si trova nella mia pagina web sotto il nome di `low_pass_fit_one_parameters.py`: esso è pressoché analogo, dal punto di vista concettuale, ad altri script realizzati per eseguire fit non lineari a un singolo parametro.

La curva di best-fit ottenuta è sovrapposta ai dati di Fig. 1 come una linea continua: si nota un accordo decisamente buono. Il risultato della procedura è

$$f_T = (474.9 \pm 2.1) \text{ Hz} \quad (7)$$

$$\chi^2 = 2.9 \quad (8)$$

$$\text{ndof} = 12, \quad (9)$$

che mostra un valore del χ^2 decisamente basso, probabilmente a causa di una qualche sovrastima delle incertezze, e un parametro di fit f_T in accordo con le attese.

Come da prassi consigliata, anche se l'esito del best-fit è ragionevole, ho cercato di "migliorarlo". Per esempio, tenendo conto del fatto che impiegando un solo parametro il valore di V_{in} viene imposto pari a quello misurato, senza incertezza, ho provato a lasciare anche V_{in} come parametro libero del best-fit, realizzando dunque un fit a due parametri (lo script corrispondente si chiama, con grande fantasia, `low_pass_fit_two_parameters.py`). Usando due parametri, è *necessario* estrarre dalla procedura di best-fit anche le informazioni relative alla "correlazione" (ovvero covarianza normalizzata) tra i parametri di fit.

I risultati ottenuti (non riporto l'uscita grafica, che è praticamente indistinguibile rispetto a Fig. 1) sono

$$f_T = (478.8 \pm 4.8) \text{ Hz} \quad (10)$$

$$V_{in} = (9.83 \pm 0.07) \text{ V} \quad (11)$$

$$\chi^2 = 2.7 \quad (12)$$

$$\text{ndof} = 11 \quad (13)$$

$$\text{"Correl."} = -0.89. \quad (14)$$

Il valore dei parametri è ancora in accordo con le attese, il χ^2 (ridotto) non cambia in modo apprezzabile, ma la

correlazione, negativa a causa del fatto che i due parametri compaiono al numeratore e denominatore della stessa funzione, è molto alta (un valore unitario indica completa correlazione), cosa che comporta, tra l'altro, un sensibile aumento dell'incertezza sul parametro f_T . La mia conclusione è che il fit a due parametri non è "migliorativo", almeno non in modo sostanziale.

C. Misure sul filtro passa-alto

Ho fatto le stesse operazioni (misura, grafico, best-fit) anche per un filtro passa-alto, costruito con $R = (672 \pm 5) \text{ ohm}$ e $C = 0.1 \text{ } \mu\text{F}$ (con tolleranza del 10%). Per questo filtro mi aspetto $f_T = (2370 \pm 256) \text{ Hz}$. La scelta della frequenza di taglio un po' più alta che non per il passa-basso aiuta a fare misure (qui sono interessanti i dati presi a $f < f_T$ e in questo modo si riesce a esplorare facilmente un buon intervallo di frequenze). Purtroppo, però, con la disponibilità di condensatori del laboratorio questo comporta di usare una resistenza prossima alla resistenza interna del generatore. Anche in questo caso ho controllato che l'ampiezza V_{in} (misurata qui come $V_{in} = (9.0 \pm 0.4) \text{ V}$) rimanesse "abbastanza" costante al variare della frequenza nel range di mio interesse, trovando che essa diminuisce del 4% a $f \sim 10 \text{ Hz}$. Dunque anche qui l'approssimazione di generatore ideale, benché non completamente soddisfatta, può essere ragionevolmente assunta, visto che la variazione riscontrata è appena superiore all'incertezza di calibrazione dello strumento impiegato.

I valori delle frequenze caratteristiche ottenuti dalle misure, che sono tutti in buon accordo con le attese, sono: $f_T = (2290 \pm 10) \text{ Hz}$ [$\Delta\phi_T = (0.22 \pm 0.04)\pi \text{ rad}$], $f_{1/2} = (1280 \pm 12) \text{ Hz}$ [$\Delta\phi_{1/2} = (0.35 \pm 0.03)\pi \text{ rad}$]; ricordate che, per un passa-alto, deve essere $f_{1/2} = f_T/\sqrt{3}$. Quindi ho riportato i dati in un grafico bilogarithmico (Fig. 2) e fatto il best-fit secondo la funzione

$$V(f) = \frac{V_{in}}{\sqrt{1 + (f_T/f)^2}} \quad (15)$$

che, come si può facilmente dimostrare con un semplice modello (che trascura gli effetti delle resistenze interne), è attesa descrivere l'ampiezza del segnale in uscita per questo circuito. Lo script corrispondente si chiama, sempre con fantasia, `high_pass_fit_two_parameters.py`

I risultati del best-fit sono

$$f_T = (2288.8 \pm 9.1) \text{ Hz} \quad (16)$$

$$\chi^2 = 2.1 \quad (17)$$

$$\text{ndof} = 13, \quad (18)$$

che sono ancora in accordo con le attese (il χ^2 è ancora molto basso a causa della possibile sovrastima delle incertezze).

Non riporto i risultati del fit a due parametri fatto in questo caso, che fornisce ancora valori in linea con le attese e una correlazione molto elevata (stavolta di segno positivo).

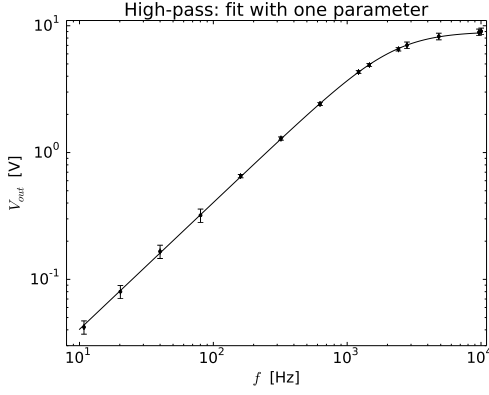


Figura 2. Dati sperimentali e risultato del best-fit a un parametro per le misure sul filtro passa-alto.

III. ATTENUAZIONE DI -20 DB/DECADE

Nel caso di $f \gg f_T$ l'attenuazione del filtro passa-basso si può approssimare con $A(f) \sim f_T/f$. Analogamente per il passa-alto, nel caso di $f \ll f_T$, l'attenuazione si può approssimare con $A(f) \sim f/f_T$. Note che questi andamenti sono “universali”, nel senso che, una volta stabilita la frequenza di taglio, essi valgono per qualsiasi filtro RC “a un polo”.

Prendiamo un passa-basso e misuriamone l'attenuazione a due diverse frequenze (tutte e due *superiori* a f_T), f_1 e $f_2 = 10f_1$: si avrà $A_2 = A_1/10$, cioè il segnale in uscita alla frequenza f_2 è $(f_2/f_1) = 10$ volte più piccolo rispetto a quello a frequenza f_1 . Che questo sia confermato dalle misure si vede benissimo in Fig. 1. Analogamente nel caso del filtro passa-alto si vede come, per frequenze *inferiori* a f_T , la frequenza più bassa (f_1) è attenuata di un fattore (f_2/f_1) rispetto alla frequenza più alta (f_2).

Quando la frequenza viene moltiplicata per un fattore 10 si dice che essa è variata di una *decade*. Se si misura l'attenuazione in dB, a frequenze rispettivamente più alte o più basse di f_T (per passa-basso o passa-alto), si vede che, quando si varia la frequenza di una decade, l'attenuazione in dB è $20 \log_{10}(1/10) = -20$ dB. In altre parole questi filtri *attenuano di -20 dB per decade*. Poiché una decade è composta da un po' più di tre ottave, si dice anche che questi filtri *attenuano di -6 dB per ottava*.

Sempre ragionando in termini di dB, notate che alla frequenza di taglio l'attenuazione vale $20 \log_{10}(1/\sqrt{2}) \approx -3$ dB, e alla frequenza $f_{1/2}$ essa vale $20 \log_{10}(1/2) \approx -6$ dB.

A. Bode per il passa-basso

Finalmente arriviamo al diagramma di Bode. Esso è una rappresentazione dell'attenuazione in dB di un filtro in funzione della frequenza (o, qualche volta, del rapporto f/f_T), cioè è un grafico, in scala *semilogaritmica* (l'asse

logaritmico è l'*orizzontale*, quello delle frequenze), della grandezza $20 \log_{10}(A(f))$.

Utilizzando i potenti mezzi di Python, la costruzione del diagramma di Bode dai dati sperimentali è piuttosto semplice. Innanzitutto occorre costruire un array contenente i valori dell'attenuazione $A_j = V_{out,j}/V_{in}$. Inoltre, poiché V_{in} è affetto da incertezza ΔV_{in} , occorre usare la propagazione dell'errore per costruire l'array delle incertezze $\Delta A_j = (\Delta V_{out,j}/V_{out,j} + \Delta V_{in}/V_{in})A_j$. Quindi occorre creare l'array dell'attenuazione in dB, $Adb_j = 20 \log_{10}(A_j)$, e la sua incertezza, di nuovo usando per bene la propagazione dell'errore: $\Delta Adb_j = (20/\ln(10))(\Delta A_j/A_j)$.

Nello script `low_pass_bode.py`, disponibile nella mia pagina web, ciò è fatto con le istruzioni:

```
# create the array Aj and DeltaAj
A = V/V0
dA = (dV/V+dV0/V0)*A

# create the corresponding arrays in dB
Adb = 20*numpy.log10(A)
dAdb = (20/numpy.log(10))*(dA/A)
```

A questo punto l'array Adb_j può essere graficato in funzione di f , facendo attenzione al fatto che, ora, essendo i dati da graficare già di tipo logaritmico, la rappresentazione deve essere *semilogaritmica*. Il risultato è mostrato in Fig. 3.

Per aggiungere ancora un po' di pepe (o sale), ho deciso di eseguire un best-fit dei dati nella *regione di transizione*, quella in cui l'attenuazione segue l'andamento $A(f) \sim f_T/f$. Questa regione è quella che nel Bode plot ha un andamento lineare, così come era lineare l'andamento osservato in Fig. 1. Ho deciso arbitrariamente di considerare nel best-fit solo i dati acquisiti a $f > 1$ kHz, che corrisponde grosso modo al doppio della frequenza di taglio f_T determinata prima.

Per selezionare i punti da usare nel best-fit (solo nel best-fit, per la rappresentazione li voglio graficare tutti) ho usato lo stesso approccio non particolarmente elegante impiegato per l'analisi lineare dei dati sulla legge di Ohm, e vi rimando alla nota corrispondente per i dettagli.

Visto che l'andamento è lineare, il fit dovrebbe essere eseguito usando le relazioni analitiche, come abbiamo fatto con la legge di Ohm (la funzione è la stessa). Però, per risparmiare tempo e perché confido in Python, stavolta ho deciso di fare un best-fit numerico, cioè usando la routine `curve_fit`. I dati che intendo fittare sono espressi in dB, cioè essi sono già stati modificati per l'estrazione del logaritmo a base 10 (e poi moltiplicati per 20, secondo la definizione). Dunque la funzione di fit, che chiamerò qui $g(f)$, deve essere espressa in maniera congrua. Essa può infatti essere determinata calcolando il $20 \log_{10}(A(f))$, con $A(f) = f_T/f$, cioè

$$g(f) = 20 \log_{10}\left(\frac{f_T}{f}\right) = \quad (19)$$

$$= \kappa - 20 \log_{10} f, \quad (20)$$

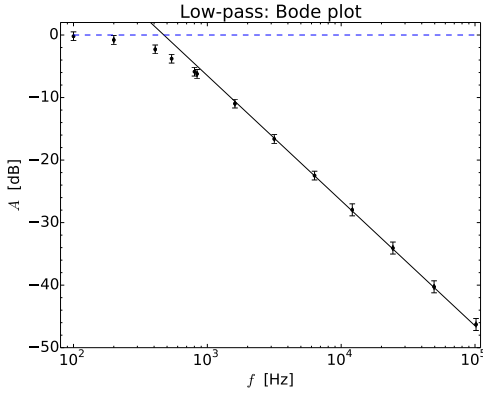


Figura 3. Diagramma di Bode e best-fit della regione di transizione per il filtro passa-basso, eseguito come descritto nel testo.

con $\kappa = 20 \log_{10} f_T$ parametro libero di fit.

Ovviamente, con questa scelta della funzione di best-fit, occorre fornire un valore iniziale adeguato alla routine di minimizzazione e inoltre, a best-fit eseguito, occorre riconvertire il valore del parametro di best-fit e della sua incertezza ai valori del “parametro fisico” $f_T = 10^{\kappa/20}$ e della sua incertezza $\Delta f_T = \Delta \kappa (\kappa/20) 10^{\kappa/20-1}$, dove $\Delta \kappa$ è l’incertezza sul parametro di fit. Tutto questo è effettivamente implementato nello script `low_pass_bode.py`.

La curva di best-fit è sovrapposta ai dati sperimentali (matematicamente trattati!) in Fig. 3 e si nota un buon accordo (per la sola regione di transizione, ovviamente). Il risultato del best-fit è

$$f_T = (473.4 \pm 11.4) \text{ Hz} \quad (21)$$

$$\chi^2 = 1.1 \quad (22)$$

$$\text{ndof} = 6, \quad (23)$$

che è ancora in accordo con le attese (ma notate come sia l’incertezza nella valutazione di f_T che il χ^2 siano aumentati).

Come ultimo aspetto, il grafico di Fig. 3 mostra anche la cosiddetta *corner frequency* f_C , che corrisponde all’intercetta fra la curva che descrive l’attenuazione del filtro nella regione di transizione (la retta continua) e la retta $A [\text{dB}] = 0$ (linea tratteggiata in blu). Avendo eseguito il best-fit della regione di transizione, è facile indivi-

duare analiticamente tale intercetta: deve infatti essere $f_C = 10^{\kappa/20}$. Osservate che questa frequenza *corrisponde* di fatto alla frequenza di taglio f_T così come è stata determinata sopra. Dunque $f_C = f_T$, come atteso per un filtro ben funzionante e ben descritto da un modello adeguato.

B. Bode per il passa-alto

Naturalmente il diagramma di Bode può essere costruito anche per il filtro passa-alto, usando lo stesso approc-

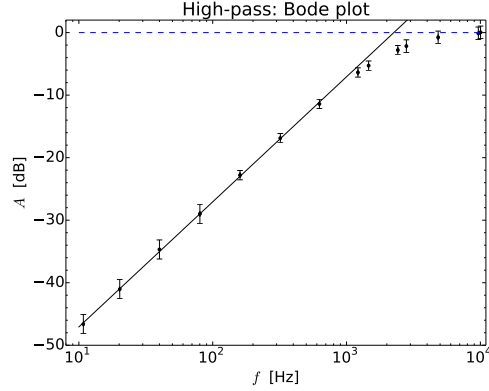


Figura 4. Diagramma di Bode e best-fit della regione di transizione per il filtro passa-alto, eseguito come descritto nel testo.

cio messo a punto sopra, con la sola avvertenza di considerare, per il fit della regione di transizione, i dati corrispondenti a frequenze inferiori alla f_T (nell’esempio qui riportato, i dati sono quelli acquisiti per $f < 1.5$ kHz).

Il risultato, ottenuto con lo script `high_pass_bode.py` disponibile nella mia pagina web, è mostrato in Fig. 4. Il best-fit restituisce i seguenti dati:

$$f_T = (2255 \pm 63) \text{ Hz} \quad (24)$$

$$\chi^2 = 0.6 \quad (25)$$

$$\text{ndof} = 6 \quad (26)$$

essi sono ancora in ragionevole accordo con le attese, e la corner frequency f_C risulta ancora compatibile con la f_T determinata prima (il χ^2 è decisamente basso, probabilmente sempre per una possibile sovrastima delle incertezze).

[1] Occhio: esiste anche un’altra definizione di attenuazione in dB, che si riferisce però alle attenuazioni di potenze, o comunque di grandezze che dipendono quadraticamente dall’ampiezza. Proprio per dirla tutta, esistono ancora altre definizioni di dB (più propriamente il simbolo da usare

sarebbe diverso) che servono per valutare il livello *assoluto* di un segnale o di una potenza, o intensità, come ad esempio in acustica. Attenti a non fare confusione!