UNIVERSITÀ DI PADOVA FACOLTÀ DI INGEGNERIA

TESI DI LAUREA

Realizzazione di un algoritmo real time per il rendering spaziale binaurale del suono, basato su modelli antropometrici

Laureando: Stefano Mezzalira

Relatore: Ch.mo Prof. Federico Avanzini

Corso di Laurea Magistrale in Ingegneria Informatica

Data Laurea : 18 Aprile 2011

Anno Accademico 2010/2011

Nothing as it seems ...

The LaTeX template for this thesis is available at
`www.dei.unipd.it/~avanzini`

Prefazione

La creazione di nuovi modelli matematici, per la manipolazione del suono è un compito tanto arduo quanto stimolante. L'analisi e la sintesi della dimensione uditiva umana, apre le porte ad un nuovo modo di intendere e vivere il suono. Le applicazioni del suono digitale nei computer, e più in generale la manipolazione del suono in un ambiente virtuale, sono molte: dall'industria del divertimento (videogame, film, musica ...) a quella dell'aviazione [7].

Sebbene il sistema uditivo ricopra un ruolo molto importante nella vita umana, storicamente nel mondo dei computer si è dato sempre più importanza all'apparato visivo. Attraverso la crescente disponibilità di strumenti hardware per la manipolazione del suono, la disparità di trattamento tra audio e video si sta assottigliando. La realtà virtuale è un obbiettivo che più campi della scienze e della tecnica cercano di raggiungere. Negli ultimi tempi la visione stereoscopica si è sviluppata velocemente consenstendo la realizzazione di apparecchi 3D fruibili al grande pubblico.

La stessa espansione ha ricevuto anche l'audio con l'avvento, qualche anno prima, dei sistemi surround con altoparlanti. L'effetto che si può ottenere, però, con delle semplici cuffie stereo, è sorprendente [1]. La sensazione di immersione nell'ambiente virtuale si ottiene, nella maggior parte dei casi, con delle particolari registrazioni chiamate binaurali. Questa tecnica prevede l'utilizzo di un soggetto (o di un manichino) e di due microfoni posizionati all'interno delle orecchie. É infatti l'interazione del suono con il corpo uno dei fattori che ci permette di localizzare un suono nello spazio. Per ottenere l'effetto tridimensionale è sufficiente, quindi, riprodurre i segnali registrati, attraverso delle cuffie.

La pressione acustica prodotta da una sorgente sonora sul timpano è modificata da più fattori: frequenza del segnale, posizione della fonte sonora, anatomia del soggetto. Se si considera l'insieme uomo−ambiente come un sistema LTI è possibile determinare univocamente la risposta all'impulso, e quindi il comportamento del sistema stesso. La risposta all'impulso è detta HRIR *Head Related Impulse Response*, e la relativa trasformata in frequenza è detta HRTF *Head Related Transfer Function*. Effettuando una convoluzione nel tempo tra HRIR e un qualunque segnale è quindi possibile ottenere l'effetto 3D.

Questo approccio è molto efficace ma altrettanto costoso: le HRIR sono individualizzate, cioè variano da soggetto a soggetto. Per ottenere dei risultati di spazializzazione consistenti si deve essere a conoscenza delle HRIR dell'ascoltarore. Poichè questo non è

[1] http://www.youtube.com/watch?v=8IXm6SuUigI

sempre possibile, a causa della difficoltà della misurazione, riuscire a sintetizzare le HRIR utilizzando caratteristiche antropometriche del soggetto è molto importante. Uno degli obbiettivi è di riuscire a realizzare un modello che date alcune influenti caratteristiche dell'ascoltatore (come la forma della pinna, la dimensione della testa, etc...) riesca a ricostruire le HRTF individualizzate. Questa approssimazione deve essere tale che, scelte le coordinate spaziali della fonte sonora, la sensazione spaziale percepita dall'individuo sia realistica. Ingannare l'ascoltatore non è semplice perché oltre alla ricostruzione delle HRTF, bisogna ricreare anche l'effetto che produce l'ambiente circostante (i.e. riverberi, riflessioni, etc...).

Il lavoro svolto in questa tesi mira a sviluppare un algoritmo di sintesi delle HRTF, presentato in [4], nell'ambiente real time Pure Data. Tutte le manipolazioni a cui il suono è sottoposto devono poter essere gestite da questo sistema real time: è richiesta, quindi, velocità e leggerezza.

Il capitolo 1 descrive le caratteristiche del sistema uditivo umano e più dettagliatamente le HRTF, ponendo l'accento su una tecnica per la sintesi di queste funzioni: i modelli strutturali. Infine è presente un piccolo paragrafo sulla psicoacustica.

Il capitolo 2 analizza l'algoritmo presentato in [3]. In particolare questo algoritmo scompone le HRTF in due sezioni: risonante e riflettente. La parte risonante a sua volta è composta da due picchi mentre quella riflettente da tre notch. Ogni elemento è determinato da una frequenza centrale, una frequenza di banda e un guadagno. Avendo a disposizione queste informazioni è possibile ricostruire le HRTF, dalla composizione di queste due sezioni.

Il capitolo 3 cerca di legare alcune misurazioni antropometriche ai parametri del modello introdotto nel capitolo 2.

Il capitolo 4 presenta in dettaglio il programma usato per la realizzazione dell'algoritmo in real time: Pure Data. É un progetto open source che permette di gestire e manipolare segnali audio e video.

Il capitolo 5 discute i dettagli implementativi dell'algoritmo: la scelta e la realizzazione dei filtri necessari per la composizione della sezione risonante e riflettente. Inoltre si studiano i risultati ottenuti analizzando le HRTF di quattro ascoltatori.

Il capitolo 6 conclude il lavoro di tesi, commentando i risultati ottenuti e indicando possibili sviluppi di questo lavoro. L'esito, da un punto di vista analitico, è soddisfacente: le HRTF vengono ricostruite abbastanza fedelmente. L'implementazione real time garantisce un buon trade off tra accuratezza e velocità. Lo spostamento del suono nel piano verticale non è, da un punto di vista percettivo, molto accurato ma accettabile. Per una validazione psicoacustica del modello è necessario effettuare una verifica di ascolto con più soggetti. Queste prove potranno essere effettuate con l'ausilio di un sistema di *motion capture* per sfruttare anche l'aspetto interattivo del programma.

Indice

Elenco delle figure

Elenco delle tabelle

Capitolo 1

Suono nello spazio

> ...Il suono è vita: ci dà la misura, le coordinate, le distanze, le presenze dentro e fuori...
>
> Roberto Vecchioni, *Le parole non le portano le cicogne*

In questo primo capitolo si descrivono le principali nozioni per comprendere come l'uomo percepisce e localizza un suono. Nel paragrafo 1.1 viene introdotto il sistema uditivo umano. Nel paragrafo 1.2 si definisce il suono e vengono introdotte alcune tecniche per aumentare la sensazione di spazializzazione. Vengono poi descritte le HRTF (paragrafo 1.3) ed un modello utilizzato per sintetizzare queste funzioni 1.4. Infine è presente un piccolo paragrafo sulla psicoacustica 1.5.

1.1 Apparato Uditivo

L'uomo risponde, per la maggior parte delle volte, razionalmente a stimoli prodotti dall'ambiente circostante. Cinque sono i sensi attraverso i quali può analizzare il mondo in cui vive. L'udito è uno di questi. L'organo fondamentale nella percezione del suono è l'orecchio. Esso è composto da tre parti. L'orecchio esterno, formato dalla pinna, ovvero il padiglione auricolare, e dal condotto uditivo esterno. Il suo compito fondamentale è di convogliare il suono verso la seconda sezione dell'orecchio. La pinna è indispensabile nella localizzazione del suono nello spazio. In particolare essa apporta informazioni relative all'elevazione della sorgente sonora. Il condotto uditivo esterno si comporta, invece, da risonatore bidimensionale. L'orecchio medio agisce come un trasformatore di energia meccanica. Esso contiene i tre ossicini più piccoli del corpo umano: martello, incudine e staffa. Il martello è collegato con il timpano, posto alla fine del condotto uditivo. Il movimento della membrana provoca attraverso un complesso gioco di leve tra questi ossicini, lo spostamento della staffa collegato direttamente con l'ultima sezione dell'orecchio: quello interno. Con questo sistema di leve l'energia trasmessa dal timpano alla staffa è raddoppiata ed inoltre la grande differenza di superficie tra il timpano e la finestra ovale (membrana a cui è collegata la staffa) fornisce un cospicuo guadagno di ampiezza del modulo della pressione acustica. Il compito fondamentale dell'orecchio medio è di fornire

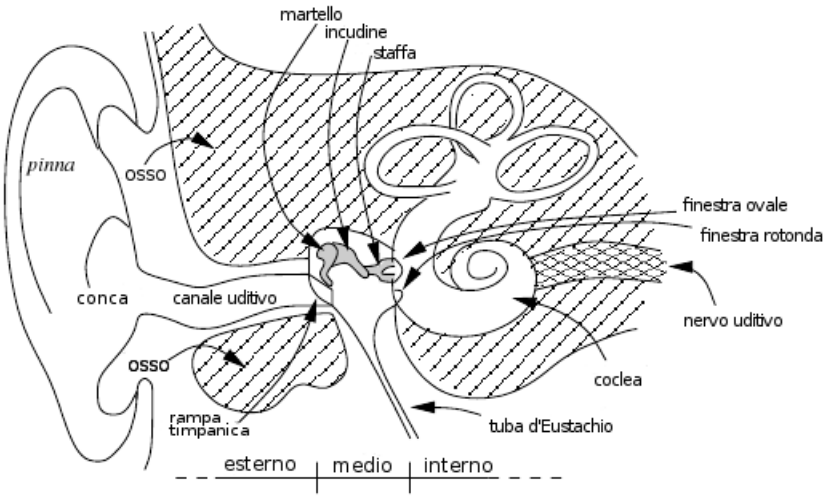


Figura 1.1: Sistema uditivo

una certa impedenza tra l'aria e il fluido cocleare: senza questo accorgimento si avrebbe una perdita di energia di circa 30 dB. Un'altra funzione molto importante riguarda la protezione del sistema uditivo nel caso in cui si presentino al timpano suoni molto forti. Se la pressione acustica supera un certo livello i muscoli dell'orecchio si contraggono (involontariamente) per diminuire l'energia trasmessa all'orecchio. L'ultima sezione da analizzare è l'orecchio interno. L'organo più importante per la percezione del suono in questa sezione è la coclea; inserita all'interno dell'osso temporale. La staffa è in contatto con l'interno della coclea attraverso una membrana detta finestra ovale. La coclea è divisa in tre condotti (o rampe), separati da due membrane. La più spessa è detta membrana basilare, mentre la più sottile è la membrana di Reissner. I due condotti (rampa vestibolare e rampa timpanica), sono riempiti con un fluido chiamato perilinfa. La perilinfa è in contatto diretto con il liquido cefalorachidiano contenuta nella cavità del cervello. Il terzo condotto è la rampa media che è riempita con un liquido, l'endolinfa. L'endolinfa è in contatto con il sistema vestibolare. Le oscillazioni sono quindi trasmesse dalla staffa alla perilinfa ed infine alla membrana basilare. Poiché questi fluidi e le pareti in cui sono posti sono incomprimibili, il volume del liquido presentato alla finestra ovale deve essere equalizzato. Questo processo avviene nella finestra rotonda, una seconda membrana posta in vicinanza alla rampa timpanica alla base della coclea. Adagiato sulla membrana basilare si trova l'organo di Corti: questo ha una struttura composta da un doppio ordine di cellule acustiche ciliate, interne ed esterne (circa 20.000). Le cellule acustiche sono in contatto con le cellule nervose che fanno parte del nervo vestibolo cocleare. Le vibrazioni prodotte dalla membrana basilare vengono convertite in impulsi elettrici da queste cellule. Il nuovo segnale giunge, quindi, nell'aerea acustica della corteccia celebrale, e successivamente al lobo temporale del cervello: qui avviene la decodifica dell'impulso elettrico in percezione del suono.

1.2 Il suono nella fisica

Il suono per definizione è una perturbazione di carattere oscillatorio che si propaga con una data frequenza in un mezzo elastico [2]. Esso è nella maggior parte dei casi l'aria. L'equazione che regola la propagazione del suono è quella d'Alembert:

$$\nabla^2 p(\mathbf{x}, t) = \frac{1}{c^2} \frac{\partial^2 p}{\partial t^2}(\mathbf{x}, t) \tag{1.1}$$

dove x rappresenta le coordinate Euclidee nello spazio e p è la pressione acustica. $\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}$ indica l'operatore Laplaciano 3-dimensionale. c è la costante che indica la velocità di propagazione del suono. L'ambiente nel quale un suono si propaga produce, delle versioni ritardate e attenuate del suono stesso. Questo fenomeno è conosciuto con il nome di riverberazione. L'effetto principale è quello di modificare le caratteristiche del suono ed in particolare quelle spaziali. Il corretto uso dell'ambiente gioca un ruolo fondamentale nella composizione musicale: si possono migliorare alcuni parametri del suono come il timbro, l'altezza e l'intensità. Si pensi ad esempio ai canti gregoriani; se non fossero eseguiti in chiese o cattedrali, in cui la riverberazione è importante, il risultato finale non sarebbe lo stesso. Uno degli obiettivi dello studio della spazializzazione, è quello di riuscire a riprodurre la pressione acustica prodotta dal suono e presente nei due timpani. L'evoluzione delle tecnologie riguardanti la manipolazione dei segnali digitali (DSP) e l'aumento della capacità di calcolo dei processori hanno facilitato l'espansione e l'applicazione del suono Virtuale 3D in molte aree. Con suono Virtuale 3D si intende l'abilità di ricreare nell'ascoltatore l'illusione che un determinato suono (generato da un sistema DSP) sia posizionato in una locazione virtuale nelle vicinanze dell'ascoltatore. La posizione della sorgente è normalmente specificata in termini di azimut θ, elevazione ϕ e distanza r.

Riuscire a virtualizzare un suono significa presentare ai due timpani delle orecchie gli stessi segnali che produrrebbe un suono reale posizionato in quel particolare punto nello spazio. Le modalità di simulazione variano in base al sistema di output che si intende utilizzare; ad esempio auricolari o altoparlanti. Il Dolby 5.1 è un esempio di virtualizzazione del suono con altoparlanti. Utilizzando più sorgenti sonore, posizionate strategicamente tutt'intorno all'ascoltatore si riescono a ricreare i segnali acustici binaurali desiderati. In questo modo il suono viene generato dagli altoparlanti posti in vicinanza alla zona in cui è presente la sorgente sonora virtuale. Questo tipo di soluzione raggiunge un buon grado di fedeltà riproduttiva. Sfortunatamente, questo sistema è fortemente influenzato dalla posizione degli altoparlanti; inoltre solo una piccola zona spaziale ed una unica orientazione dell'ascoltatore permettono la creazione di un corretto segnale binaurale ai timpani. Altri fattori da tenere in considerazione sono la riverberazione che la stanza di ascolto produce ed il fenomeno del cross-talk: il suono emesso da un altoparlante sarà sempre udito da entrambe le orecchie.

Il secondo approccio prevede l'utilizzo degli auricolari. In questo caso bisogna simulare tutti gli elementi che il suono naturale incontra nel suo tragitto dalla sorgente ai nostri ricevitori; la forma della stanza ed anche il corpo dell'ascoltatore (testa, torso etc...)

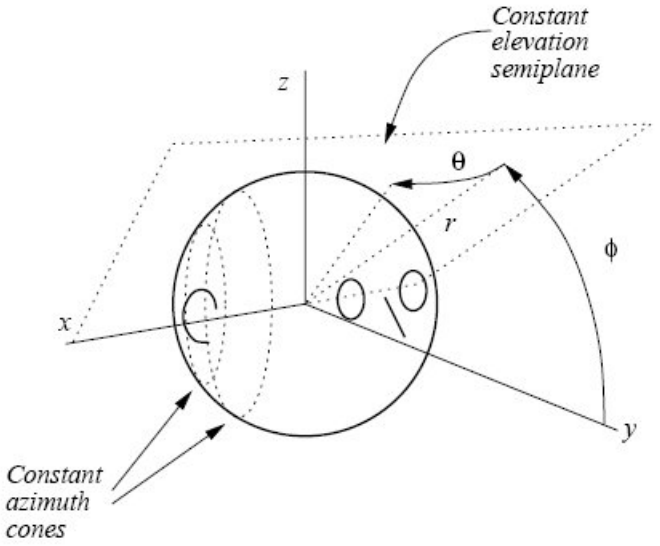


Figura 1.2: Sistema di riferimento

influiscono nella percezione spaziale del suono. Tutti questi ostacoli devono essere modellati opportunamente. In questo caso si parla di modello strutturale, concetto sviluppato originariamente in [11]. Poiché tutte le manipolazioni effettuate al suono possono essere eseguite da un computer, quest'approccio è completamente portabile. Inoltre, non si verifica il fenomeno del cross-talk. Le cuffie, tuttavia, non garantiscono una risposta in frequenza piatta, tendono a conferire alla sorgente un'impressione di vicinanza eccessiva e sono scomode da indossare per un lungo periodo di tempo.
Uno dei problemi principali dell'ascolto in cuffia è l'internalizzazione. L'ascoltatore ha l'impressione che la fonte sonora sia posizionata all'interno della sua testa. Sembra che la causa di questo problema sia la tendenza dell'uomo ad internalizzare gli stimoli totali percepiti, quando essi sono chiaramente prodotti artificialmente e posti a grande distanza. Una tecnica che è ritenuta valida nell'esternalizzare i suoni quando essi sono presentati in cuffia è la decorrelazione [16]. Con questo termine si indica un processo in cui un segnale audio in ingresso è trasformato in più segnali di output le cui forme d'onda, differiscono le une dalle altre, sebbene il suono e la sorgente siano le stesse. Questo è proprio il fenomeno che si verifica nella localizzazione del suono, causato principalmente dal riverbero della stanza. La misura di correlazione tra due segnali periodici $y_R(n)$ e $y_L(n)$ può essere determinata dalla cross-correlazione $\Omega(l)$, con l differenza temporale tra i due canali:

$$\Omega(l) = \lim_{K\to\infty} \frac{1}{2K+1} \sum_{n=-K}^{K} y_R[n] y_L[n-l] \tag{1.2}$$

Se un canale è una copia ritardata dell'altro allora la correlazione avrà un picco per un valore pari alla differenza temporale. Si sceglie, inoltre, un fattore di normalizzazione

per produrre un valore di correlazione tra 1 (pure shift) e -1 (aperiodici). Il grado di correlazione è il valore assoluto massimo che la funzione assume. Due segnali con un grado uno sono chiamati coerenti. Quando ad un ascoltatore vengono presentati alle due orecchie segnali rumorosi, il grado relativo di correlazione produce un'impressione spaziale in cui la sorgente sonora viene posta al centro della testa, equidistante dall'orecchio destro e sinistro. Segnali parzialmente coerenti producono un impressione spaziale che è meno definita nello spazio rispetto al caso di perfetta coerenza. Il grado di correlazione è ridotto in presenza di riverberazione. Le tecniche che sono utilizzate per ricreare il riverbero, quindi, possono essere utilizzate per decorrelare due segnali. Se lo scopo principale è solo quello di esternalizzare un suono bisogna evitare artifici sonori che producano un eccessiva colorazione del suono o code riverberanti non naturali. Il grado di correlazione è utile inoltre per costruire una trama spaziale per più sorgenti sonore. É stato mostrato che l'ascoltatore tende ad assegnare un'impressione spaziale unica a suoni che hanno un grado di correlazione simile. Ad esempio un paio di segnali vocali coerenti a cui è aggiunto un segnale rumoroso, producono un'impressione sonora posta al centro della testa. Un aumento drastico dell'esternalizzazione si ottiene se lo stimolo binaurale varia dinamicamente in funzione dei movimenti della testa (*head tracking*). Sebbene questa tecnica sia molto costosa dal punto di vista computazionale e la principale situazione di utilizzo di un sistema audio 3d sia davanti ad uno schermo di un computer (quindi con testa immobile) questo sistema negli ultimi anni si sta sviluppando velocemente.
Ottimi risultati si sono, comunque, ottenuti nella localizzazione delle fonti sonore ed esternalizzazione, in condizione di assenza di riverbero ed assoluta immobilità della testa [18]. Sembra che se la riproduzione preservi le condizioni di un naturale ascolto, il fenomeno dell'internalizzazione non si verifichi. Questa condizione non è riproducibile con gli auricolari in commercio (risposta in frequenza non piatta) ed il controllo sul grado di correlazione sembra essere l'unica procedura utilizzabile per esternalizzare un suono. L'effetto di esternalizzaizone è stato studiato approfonditamente in [14], dove si mostra che una buona soluzione si ottiene con l'indipendenza del ITD dalla frequenza, mentre l'effetto del ILD è accumulato lungo tutto lo spettro. L'utilizzo di sistemi DSP per la creazione dei segnali binaurali, coinvolgono speciali filtri digitali che possiedono una risposta all'impulso specifica HRIR: *Head Related Impulse Response*.

1.3 Head Related Transfer Function

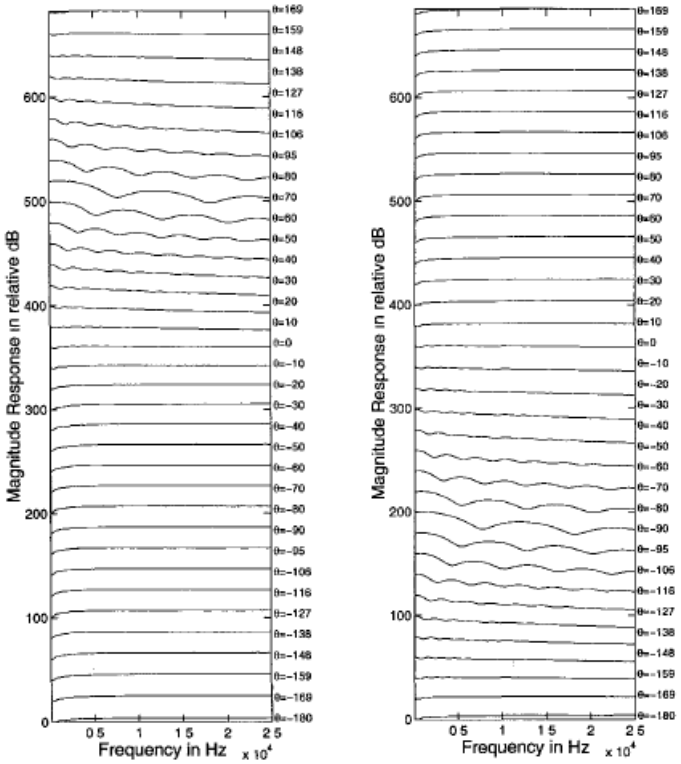
L'abilità dell'essere umano nel localizzare un suono risiede nella sua capacità d'analizzare lo spettro del segnale audio. Questi "indizi"spettrali sono chiamati *Head-Related Transfer Function* (HRTF). Questa funzione è la trasformata di Furier del HRIR. Essa è una funzione molto complessa che dipende oltre alla posizione della sorgente sonora individuata dalla tripletta (θ, ϕ, r) anche dalla frequenza.

Un semplice modello introdotto da Lord Rayleight [23], dal quale si sono sviluppati molte altre ricerche che hanno portato alla definizione di HRTF, è la *Duplex Theory*. Questo modello spiega in che modo l'uomo riesce a discriminare suoni posizionati nel

piano dell'azimut. Vengono introdotte due grandezze: *interaural time difference* ITD e *interaural level difference* ILD. ITD è definita come la differenza temporale di percezione di un suono tra l'orecchio destro e quello sinistro. Analogamente ILD è definita come la differenza di ampiezza presente alle due orecchie. In generale, un suono è percettivamente più vicino ad un orecchio quando questo è investito precedentemente dal fronte d'onda. Se il suono in questione è una sinusoide lo scostamento laterale percepito sarà proporzionale alla differenza di fase del suono presente alle due orecchie. Per frequenze della sinusoide che sono comparabili con il diametro della testa le informazioni apportate dal ITD diventano ambigue. A queste frequenze (circa 1500Hz) infatti ITD può corrispondere a distanze che sono più lunghe di una sola lunghezza d'onda. Dopo una certa frequenza quindi la differenza di fase (ITD) non aiuta più nella localizzazione del suono. Per frequenze superiori a 1500Hz la testa inizia un'azione di schermatura: meno energia arriva all'orecchio più distante dalla fonte sonora (ILD). La relazione tra la locazione percepita non varia in funzione della sola ILD, in questo caso bisogna tenere in considerazione anche la frequenza. Per una data frequenza, infatti, l'azimut percepito varia linearmente con il logaritmo del ILD [8]. La *Duplex theory* non spiega in alcun modo come si può determinare la posizione del suono al di fuori del piano dell'azimut. Esistono, infatti, diversi punti nello spazio, con svariati valori per l'elevazione e distanza che producono valori di ITD e ILD molto simili. L'insieme di tutti questi punti è chiamato cono di confusione. Il problema sembra essere più accentuato nel piano che separa le due orecchie e passa verticalmente lungo la testa. Per un qualsiasi suono posizionato lungo tale piano i valori di ITD e ILD sono più o meno zero. L'informazione apportata da queste due grandezze è quindi nulla. Poiché l'uomo è in grado di discriminare sorgenti sonore posizionate lungo tale piano, devono entrare in gioco altre caratteristiche utili alla distinzione. In questo caso l'informazione utile non è derivata dal fatto che possediamo due orecchie. Da un punto di vista fisico il segnale che arriva alle due orecchie è uguale (segnale monoaurale). In questa situazione, le informazioni necessarie a discriminare la posizione della fonte sonora, vanno ricercate nello spettro del segnale. Purtroppo, la relazione tra lo spettro del segnale e la locazione spaziale del suono non è semplice come la relazione con ITD/ILD. Per analizzare questo problema molti ricercatori hanno effettuato delle misurazioni dei segnali presenti alle orecchie degli ascoltatori per diverse posizioni spaziali di un dato suono. Queste misurazioni sono chiamate HRTF e riassumono il filtraggio acustico causato dalla testa, torso e pinna, dipendete dalla posizione spaziale della sorgente sonora. Più formalmente HRTF ad un orecchio è definita come il rapporto dipendente dalla frequenza tra il livello di pressione sonora (SPL) $\Phi^{(l),(r)}(\theta, \phi, \omega)$ presente al corrispondente timpano e il SPL nel cosiddetto *free-field* al centro della testa $\Phi_f(\omega)$ come se l'ascoltatore fosse assente:

$$H^{(l),(r)} = \frac{\Phi^{(l),(r)}(\theta, \phi, \omega)}{\Phi_f(\omega)} \tag{1.3}$$

Attraverso queste misurazioni si è scoperto che le HRTF possono essere approssimati con filtri a fase minima [19]. Grazie a questa proprietà si riesce a specificare univocamente

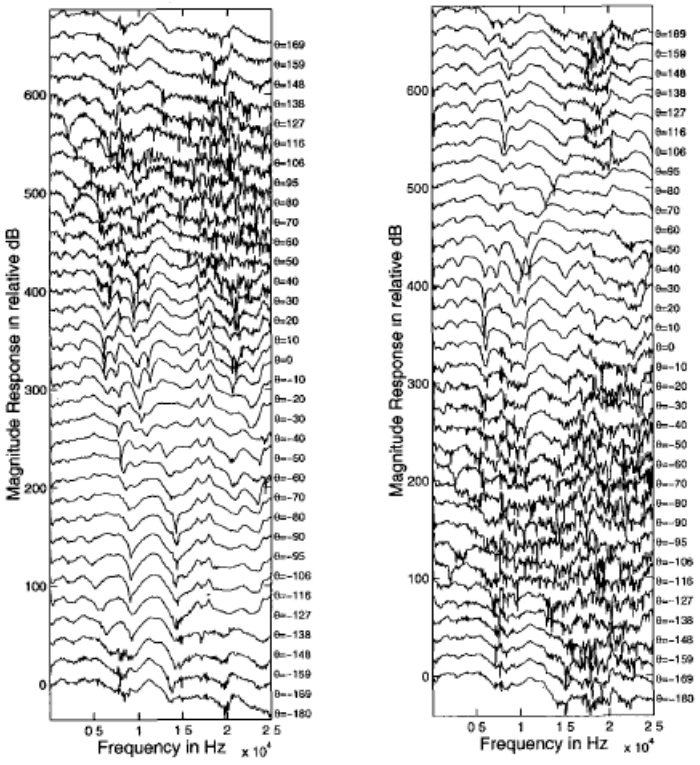


(a) Orecchio sinistro (b) Orecchio destro

Figura 1.3: HRTF teoriche in funzione dell'azimut (elevazione 0).

la fase del HRTF grazie al solo modulo. Inoltre questa assunzione permette di separare l'informazione apportata dall'ITD dalle specifiche dei filtri che approssimano le HRTF.

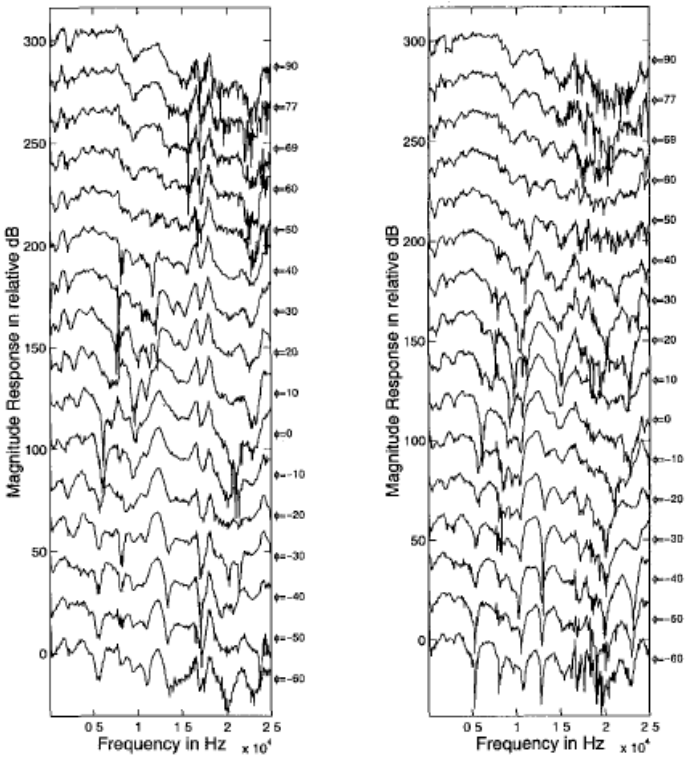
Molti sono stati i tentativi atti a comprendere la struttura del HRTF, visualizzando ed analizzando queste complicate funzioni, nel dominio del tempo, della frequenza e dello spazio [9]. Per confrontare queste complicate funzioni è utile gestire e studiare due differenti insiemi di HRTF: quelle misurate empiricamente su soggetti umani e quelle determinate da modelli matematici. Per trovare qualche sorta di struttura nei dati delle HRTF, si studiano due fenomeni correlati con le HRTF, i cui effetti sono ben noti: la diffrazione della testa e l'influenza che questa ha con l'elevazione. La misurazione empirica viene effettuata installando dei microfoni molto piccoli nei condotti uditivi del soggetto e successivamente vengono proposti all'ascoltatore diverse tipologie di suoni per differenti locazioni spaziali. Questi stimoli sono prodotti da altoparlanti posti solitamente a 1,5 m di distanza dal soggetto in una stanza anecoica. In tal modo è possibile registrare esattamente le HRTF per ogni soggetto. I modelli analitici più semplici sono derivati dalle soluzioni di equazioni acustiche delle onde che investono una sfera rigida (la testa). In particolare si determina la pressione presente in due punti della sfera che rappresentano le due orecchie. Calcolando queste pressioni per differenti frequenze ed angoli di incidenza del piano, si possono computare sistematicamente le HRTF destre e sinistre.



(a) Orecchio sinistro (b) Orecchio destro

Figura 1.4: HRTF misurate in funzione dell'azimut (elevazione 0).

Una derivazione matematica di questo tipo può essere trovata in [8]. Il modello sferico viene utilizzato per predire gli effetti di diffrazione. Per alcune frequenze ed angoli di incidenza, la sfera ha un effetto amplificativo. Sorprendentemente, questo effetto (*bright spot*) si verifica per alcune locazioni controlaterali (l'orecchio più distante dalla fonte sonora e quindi schermato dalla testa). Le caratteristiche spettrali che corrispondono all'elevazione sono influenzate dalla pinna. L'intervallo di frequenze nella quale opera è $6 \div 8$ kHz. In queste frequenze si possono ritrovare dei pattern nelle HRTF riconducibili psicoacusticamente alla percezione dell'elevazione. Le HRTF sono state studiate ampiamente nel dominio della frequenza. Si è cercato di parametrizzare queste funzioni utilizzando per esempio la tecnica delle *Principal Component Analysis* (PCA) [17]. Il dominio della frequenza ha mostrato alcune importanti differenze tra le HRTF teoriche e quelle misurate. Mentre le HRTF teoriche sono private di rumore, il *signal to noise ratio* (SNR) di quelle misurate sembra essere funzione della locazione spaziale. In particolare HRTF controlaterali hanno un andamento meno armonioso rispetto a quelle ispilaterali (l'orecchio è posto davanti alla fonte sonora, la testa quindi non scherma il suono). Lo SNR delle sorgenti ipsilaterali è generalmente più elevato rispetto alle sorgenti controlaterali. Questo fenomeno può essere spiegato dal fatto che l'orecchio controlaterale riceve meno energia rispetto a quello ipsilaterale. Inoltre le HRTF misurate risultano più



(a) Orecchio sinistro (b) Orecchio destro

Figura 1.5: HRTF misurate in funzione dell'elevazione (azimut 0).

complesse: sono presenti più picchi ed avvallamenti nel modulo rispetto a quello delle HRTF teoriche. Questi andamenti sono causati dall'interazione del suono con il torso e la pinna, fenomeni non considerati nel modello sferico. Anche gli effetti dell'elevazione possono essere osservati nel dominio della frequenza. Si notano degli avvallamenti che si muovono lungo l'asse della frequenza quando l'elevazione aumenta. Inoltre sono presenti alcuni picchi che si appiattiscono al diminuire dell'elevazione, figura 1.5. Effetti di diffrazione si possono osservare maggiormente nelle HRTF teoriche. Delle leggere oscillazioni si ritrovano nelle HRTF controlaterali corrispondenti agli azimut $+90$, -90 per l'orecchio sinistro e destro rispettivamente. Un effetto di amplificazione è rappresentato da un lobo a basse frequenze a partire dagli angoli di 127 fino a 60 con picco a 90. Quest'amplificazione è causata dall'effetto di diffrazione che la testa produce a basse frequenze nei lati controlaterali. Ulteriori effetti di amplificazione sono presenti ad alte frequenze per sorgenti ispilaterali, causate dalla riflessione e dalla vicinanza dell'orecchio alla testa (intesa come superficie rigida). Nel dominio del tempo le HRTF sono chiamate HRIR, *Head-Related Impulse Response* figura 1.6. Poiché la complessità di un sistema di spazializzazione dipende largamente dalla lunghezza di queste risposte all'impulso, si cerca di minimizzare la lunghezza delle HRIR preservando quanto più possibile le caratteristiche che influenzano la percezione spaziale. In generale si può affermare che una fonte sonora

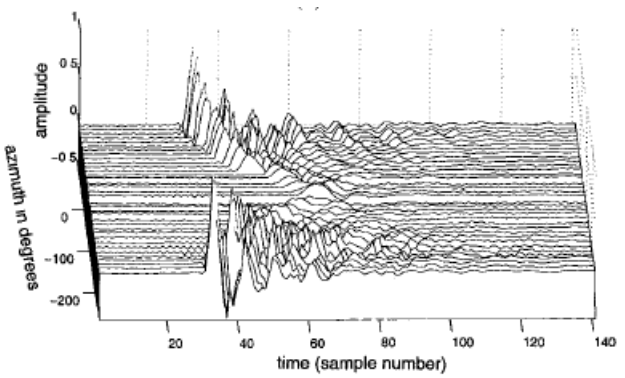


Figura 1.6: HRIR misurata in funzione dell'elevazione (azimut 0).

molto distante dall'orecchio ipsilaterale, sia per l'azimut sia per l'elevazione, ha un HRIR caratterizzata da una bassa ampiezza iniziale. Questo è in accordo con quanto afferma la *Duplex Theory*: ITD ed ILD sono più grandi per sorgenti sonore con un grande valore assoluto per l'azimut. Le HRIR che corrispondono a fonti controlaterali posizionate nell'area di schermatura della testa, hanno un valore di ampiezza iniziale relativamente elevato (fenomeno riscontrato anche con le HRTF).
Un confronto con le HRIR teoriche e misurate rivela una più ampia ricchezza di quest'ultime. Le HRIR misurate contengono molti picchi secondari (oltre a quelli principali) che quelle teoriche non possiedono. Sebbene questi effetti possono essere la conseguenza di una misurazione rumorosa, essi sono in gran parte relazionati con l'interazione del fronte d'onda con la pinna; fenomeno che nel modello teorico non viene valutato. Esistono ulteriori metodi di visualizzazione incentrati sull'energia che le due orecchie ricevono in funzione della posizione spaziale mantenendo costante la frequenza. Queste rappresentazioni in [9] sono chiamate *spatial frequency response surface* SFRS (figura 1.7). In questi grafici l'elevazione è legata ai massimi locali a specifiche frequenze. Un SFRS con un unico massimo suggerisce che il sistema uditivo favorisce l'individuazione di un punto quando è presentato come rumore a banda ristretta a quella frequenza. Questo è in accordo con quanto affermato dalla teoria delle bande direzionali. Questa teoria asserisce che alcuni segnali con una piccola banda in frequenza sono correlati con alcune direzioni spaziali. Ad esempio, alcuni test psicoacustici hanno mostrato che per certe frequenze molti soggetti tendono a localizzare, in termini di elevazione, suoni compresi tra i 6 e gli 8 kHz rispetto a locazioni sonore nel free field. L'effetto di diffrazione è facilmente osservabile negli SFRS sia per le HRTF teoriche sia per quelle misurate. Si presentano dei punti di massimo nell'orecchio controlaterale per frequenze comprese tra 2 e 4 kHz (*brigth spot*).

1.4 Modello Strutturale

Nel proprio lavoro di tesi, Genuit propose un metodo per approssimare le HRTF attraverso l'uso di filtri digitali. Il principio base sta nel segmentare il corpo umano in

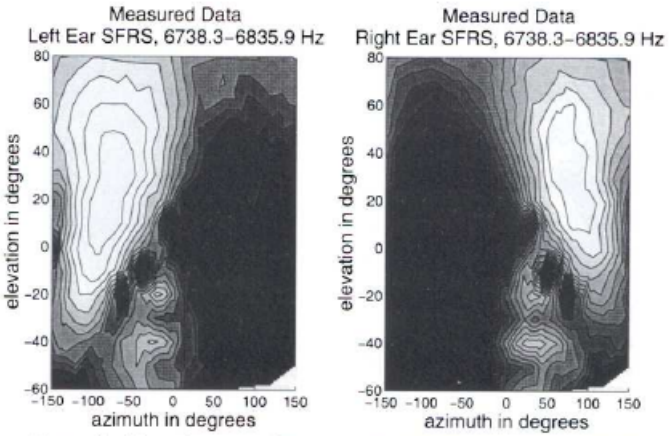


Figura 1.7: Esempio di SFRS. Particolare del bright spot.

più componenti, caratterizzandole in base alle misure antropometriche dell'ascoltatore. La giustificazione di tale modello discende dal fatto che le onde sonore arrivano al timpano seguendo diversi percorsi.

1.4.1 Testa

La testa introduce due effetti principali: ITD e ILD. Un'approssimazione abbastanza accurata per ITD prevede che il suono sia abbastanza distante e la testa sia pressoché sferica. La prima semplificazione (fonte sonora distante) implica che le onde sonore che colpiscono la testa (di raggio a) siano poste su un piano. La distanza supplementare che un raggio acustico deve compiere per raggiungere l'orecchio più distante può essere determinata considerando l'angolo d'incidenza θ e utilizzando la seguente formula:

$$\frac{a}{c}(\theta + \sin(\theta)) \tag{1.4}$$

Come si può notare ITD è indipendente dalla frequenza. Questa grandezza può essere approssimata utilizzando un filtro delay frazionario $F_{ITD}(\theta, z)$. La realizzazione del ILD risulta un po' più complicata. In questo caso è necessario considerare la dipendenza dalla frequenza: a valori bassi la differenza di percezione tra le due orecchie non è molto elevata, essa cresce per valori più alti. Anche in questa situazione possiamo considerare la testa come una sfera con un raggio pari ad a. Si normalizza la frequenza $\mu = \omega a/c$ e la distanza $\rho = r/a$. Considerato un punto sopra la sfera, la diffrazione di un'onda acustica può essere espressa dalla seguente funzione di trasferimento:

$$H(\rho, \mu, \theta) = -\frac{\rho}{\mu} e^{-i\mu\rho} \sum_{m=0}^{\infty} (2m+1) P_m(\cos\theta) \frac{h_m(\mu\rho)}{h'_m(\mu)} \tag{1.5}$$

P_m ed h_m rappresentano rispettivamente il Polinomio di Legendre di ordine m, e la derivata rispetto al suo argomento della funzione sferica di Hankel. θ_{inc} è l'angolo sotteso dal

raggio immaginaria che unisce il centro della sfera con la sorgente sonora e il raggio che congiunge il punto di misurazione sulla sfera alla sorgente. Questa funzione ha un comportamento asintotico ben definito per grandi valori della distanza. É sufficiente studiare questa funzione per distanze che tendono ad infinito quando essa diviene arbitrariamente grande. Quindi $H(\rho, \mu, \theta)$ può essere approssimata da un filtro parametrico $\widetilde{H}(\theta_{inc}, \mu)$ dipendente solo da θ_{inc}. Alcuni autori [11] hanno proposto la seguente soluzione:

$$\widetilde{H}(\theta_{inc}, \mu) = \frac{1 + \frac{j}{2}\mu \cdot \alpha(\theta_{inc})}{1 + \frac{j}{2}\mu}, 0 \leq \alpha(\theta_{inc}) \leq 2 \tag{1.6}$$

$\alpha(\theta_{inc})$ determina la posizione degli zeri. Poiché tale risposta in frequenza sia simile a $H(\rho, \mu, \theta)$, α può essere scelto in questo modo:

$$\alpha(\theta_{inc}) = (1 + \frac{\alpha_{min}}{2}) + (1 - \frac{\alpha_{min}}{2}) \cos(\frac{\theta_{inc}}{\theta_{min}}) \tag{1.7}$$

dove i parametri ausiliari α_{min} e θ_{min} sono scelti per regolare la dipendenza di α su θ_{min}.

1.4.2 Orecchio esterno

Per quanto riguarda l'orecchio esterno bisogna modellare la pinna e il condotto uditivo. Dai molti studi condotti in proposito [8] si è visto che la pinna apporta informazioni sull'elevazione della sorgente sonora. Batteau [6] ha proposto un modello con la conca (concha) e l'elice (rim) che fungono da agenti riflettori.

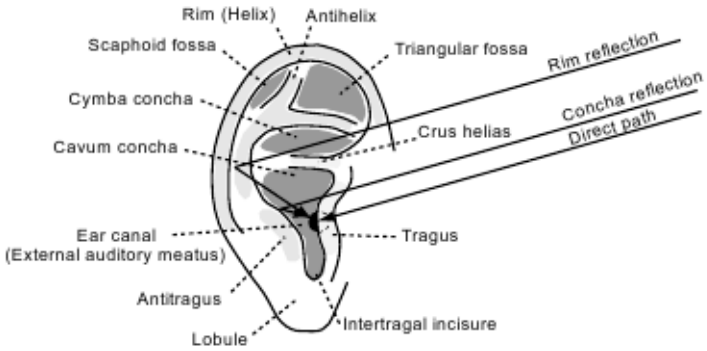


Figura 1.8: Anatomia della pinna

Queste componenti interagiscono con il suono che segue il percorso diretto causando dei notch molto evidenti nello spettro. Se il tempo di ritardo è T, la frequenza che separa i notch spettrali è di $1/T$, e poiché il tempo di ritardo varia in funzione dell'elevazione anche la frequenza dei notch deve cambiare di conseguenza. La figura 1.9 mostra una formalizzazione del concetto espresso da Batteau. ρ_A e ρ_V sono dei coefficienti costanti

di riflessione, τ_A è un tempo di delay e τ_V un tempo di ritardo dipendente dall'elevazione. Questo modello garantisce un buon grado di approssimazione per la maggior parte degli

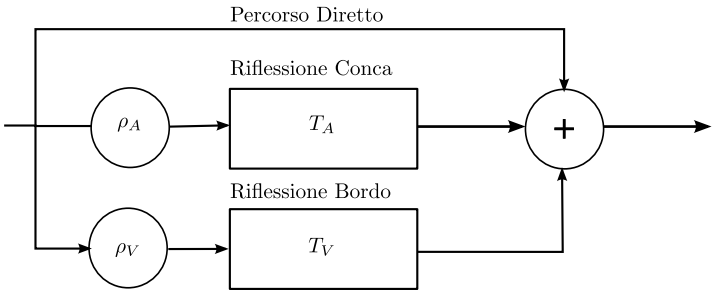


Figura 1.9: Modello della pinna

angoli di elevazione. Purtroppo i notch prodotti sono più larghi rispetto ai notch reali. I coefficienti di riflessione dovrebbero, quindi, dipendere dalla frequenza. Una possibile realizzazione digitale prevede l'utilizzo di filtri FIR a pettine (*comb*). Un analisi nel tempo risulta molto complicata, per questo si preferisce passare nel dominio della frequenza. Considerando lo spettro delle HRTF è possibile identificare i notch prodotti dalla pinna. Nel capitolo 2 verrà presentata una possibile soluzione introdotta da [4]. Per quanto riguarda il condotto uditivo, esso si comporta come un risonatore ad una dimensione e può essere approssimato come un cilindro con pareti ad alta impedenza acustica e larghezza costante. Un contributo importante nello studio della Pinna è stato apportato da Shaw [27]. Un sistema acustico leggermente smorzato (Pinna) è espresso in termini di modi.Shaw individuò sei modi principali. Il primo modo si verifica indipendentemtne dalla direzione del suono a 4,2 kHz e può essere approssimato come un modo risonante con profondità pari ad un quarto della lunghezza d'onda. Il secondo modo si manifesta a 7,1 kHz, dove si possono osservare due zone di pressione distinte in vicinanza della cimba. Questo modo è più accentuato quando l'angolo di elevazione è di 68°. Il modo tre compare a 9,6 kHz ed ha due zone di pressione distinte. Ne è presente una positiva in vicinanza della fossa che si trasforma in una zona negativa in prossimità della cimba. É presente un'ulteriore zona di pressione positiva nella conca. Il modo tre presenta la massima eccitazione quando l'onda sonora ha un angolo di incidenza di circa 73°. Il modo quattro avviene a 12,1 kHz. Nuovamente si ossevano delle zone con pressione negativa vicino alla fossa, le quali diventano positive quando ci si avvicina alla cimba. Il modo quattro è eccitato a una elevazione di $-60°$. I modi cinque e sei si manifestano in corrispondenza di 14,4 kHz e 16,7 kHz rispettivamente. Presentano più o meno la stessa mappatura delle zone di pressione e sono eccitate con un angolo di elevazione pari a 7°. Il primo modo è quello dominante e dipende principalmente dalla profondità della conca. I modi due e tre sono chiamate risonanze trasversalisono determinanti nel piano verticali in quanto si ativanto con angoli di elevazione elevati. Similmente i modi quattro, cinque e sei invece apportano più informazioni nel piano orizzontale, poichè sono eccitati con angoli di elevazioni che si avicinano allo zero. Partendo da queste constatazioni Shaw ha creato dei modelli (conca cilindrica, conca rettangolare) che sono in grado di approssimare

la pinna umana. Il modello finale di Shaw, chiamato modello E, è capace di produrre le risonanze alle corrette frequenze, moduli per più angoli di incidenza.

1.4.3 Torso e spalle

Il terzo elemento da considerare è il torso. Innanzitutto esso fornisce una riflessione addizionale da sommare al suono diretto, inoltre crea un effetto di schermatura per segnali provenienti dal basso. Per semplificare i calcoli si può approssimare il torso con un ellissoide. La sagoma finale dell'uomo modellato (torso e testa) assomiglia ad un pupazzo di neve ed è per questo che il modello viene chiamato *snowman*. Il ritardo tra il raggio acustico diretto e quello riflesso non cambia così tanto se la sorgente sonora si muove lungo il piano orizzontale. D'altro canto questo ritardo varia consistentemente se la sorgente si muove in verticale: l'effetto è maggiore se la fonte è proprio di fronte all'ascoltatore ($\sim$1ms). Anche in questo caso il torso produce dei notch nello spettro. In particolare essi dovranno essere posti ad una frequenza che è inversamente proporzionale al ritardo e quindi all'elevazione. Raggiunto un particolare valore per l'elevazione il torso non si comporta più come un corpo riflettore, anzi provoca un effetto di schermatura. Sebbene gli effetti introdotti da questo elemento non siano molto marcati, essi sono importanti perché si verificano soprattutto a basse frequenze. Si può dire che le frequenze modificate dal torso sono complementari rispetto a quelle a cui opera la pinna. Per una realizzazione digitale si può pensare di analizzare le HRIR. In questo modo è possibile stimare il ritardo dipendente da θ e ϕ introdotto dalle riflessioni. Questo nuovo parametro verrà introdotto in un filtro FIR a linea di ritardo con un coefficiente di ampiezza ben calibrato.

1.4.4 Considerazioni

Questo modello sebbene abbia una buona approssimazione non tiene in considerazione le interazioni presenti tra i suoi componenti. Oltre all'analisi in tempo o frequenza delle HRTF è possibile collegare i parametri dei filtri a misure antropometriche. Questo è stato il lavoro compiuto da Genuit che da 34 misure del corpo ha ricavato questo modello. Non esiste tuttavia alcuna garanzia che queste misurazioni siano necessarie e sufficienti per determinare le HRTF. Oltre ad essere funzioni molto complicate le HRTF variano da persona a persona.Può infatti accadere che per una stessa HRTF, le sensazioni spaziali prodotte da queste funzioni siano buone per una persona mentre non lo siano per altre. Si sono studiati metodi per riuscire ad aggirare o risolvere questo problema. In [5] si è cercato di valutare l'accuratezza di HRTF non individualizzate. Si è concluso che per l'azimut i risultati sono buoni e robusti, mentre per l'elevazione gli esiti non sono molto soddisfacenti: i problemi di *front-back* confusion sono maggiori rispetto ad HRTF individualizzate. Rimane quindi più efficacie la tecnica che utilizza HRTF individualizzate ma ovviamente è più difficile da applicare su larga scala.

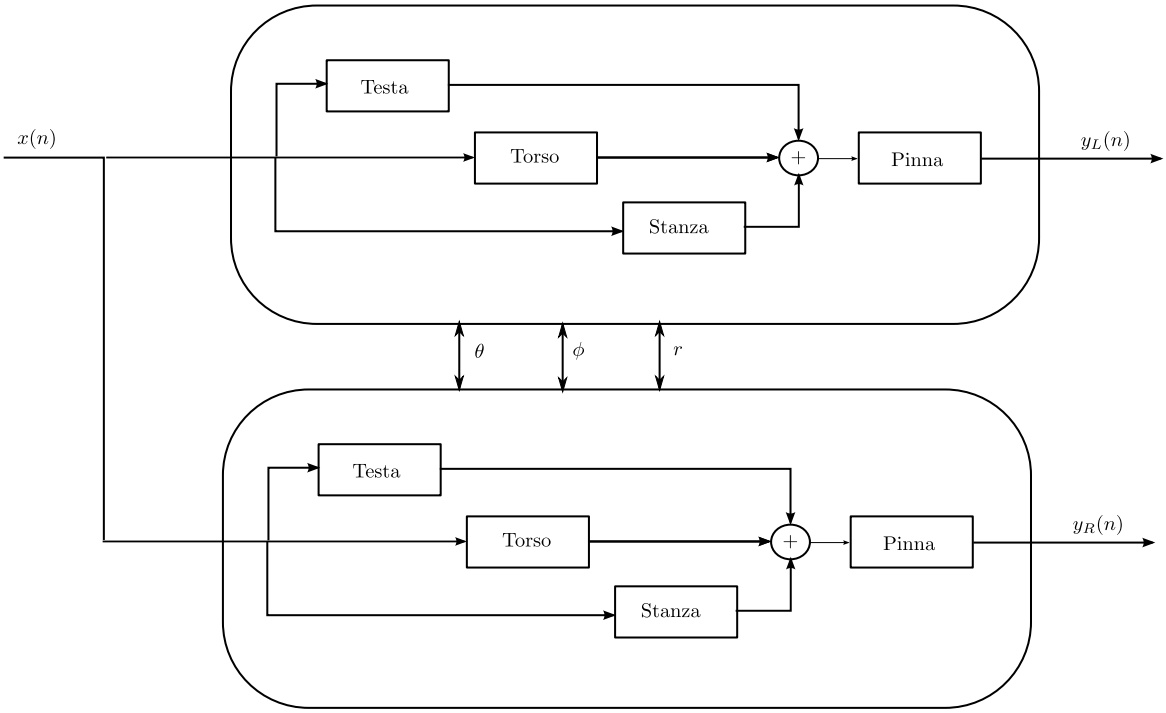


Figura 1.10: Modello Strutturale

1.5 Psicoacustica

Misurare da un punto di vista percettivo l'esattezza delle HRTF non è un compito facile. Cambiamenti nella sorgente o nel mezzo di comunicazione possono portare a diverse percezioni spaziali. É necessario costruire un linguaggio che permetta all'ascoltatore di descrivere le diverse posizioni in cui sono poste le fonti sonore. Mentre le grandezze fisiologiche, come la pressione del sangue, possono essere misurate direttamente, quelle percettive coinvolgono un processo mentale e devono essere quindi indicate come probabilità di risposta ad uno stimolo esterno. A causa della non consistenza di ogni essere umano (delle volte anche tra lo stesso soggetto) devono essere usate delle tecniche statistiche: bisogna poter quindi creare esperimenti ripetibili per riuscire a descrivere la natura della percezione uditiva spaziale. Uno dei problemi principali consiste nel trovare un linguaggio che ha la più alta consistenza tra i soggetti. Gli esperimenti di psicoacustica si svolgono in situazioni che non sono del tutto simili alla realtà; questa è una necessità. Eliminare variabili non controllabili è fondamentale per riuscire a ripetere un esperimento. Purtroppo più è il controllo che si vuole ottenere meno la situazione ricreata è fedele alla realtà. Come per tutte le applicazioni ingegneristiche è necessario trovare un buon *trade off*.

La maggioranza degli studi sulla spazialità utilizza stanze anecoiche. Si eliminano così, le riflessioni, uno degli aspetti fondamentali per la percezione spaziale. Il tipo di segnale utilizzato per lo stimolo, inoltre, è in molti casi una sinusoide, rumore bianco (o filtrato) o degli impulsi. Non molti di questi suoni sono presenti in natura. Per ricreare

un suono complesso si utilizza il rumore, mentre gli impulsi sono utilizzati per simulare una transizione tra più suoni complessi. Quindi devono essere previste delle limitazioni nelle applicazioni derivanti dagli esperimenti di psicoacustica. La ricerca in questo campo rimane comunque molto importante perché permette di delineare i meccanismi per la localizzazione del suono in situazioni controllate.

Le misure fisiche per descrivere un suono, frequenza, intensità, contenuto spettrale e durata, hanno una controparte psicoacustica. Per la frequenza si utilizza il termine pitch. Per una funzione sinusoidale, l'orecchio interno converte le vibrazioni del timpano in un punto preciso nella membrana basilare, in maniera tale che ad un raddoppio della frequenza corrisponda un raddoppio della distanza. Per suoni reali la relazione tra frequenza e pitch diventa più complicata. La tecnica del vibrato modifica la frequenza anche di un semitono ad un rate di 5 Hz, ma è percepito un singolo pitch. Per di più il pitch prodotto da un basso e riprodotto da un altoparlante che non è in grado di generare tali frequenze viene comunque percepito da un ascoltatore. In molti casi l'orecchio determina il pitch alla frequenza fondamentale corretta basata sulla relazione armonica della più alta armonica che l'altoparlante è in grado di riprodurre.

Loudness è una grandezza psicoacustica associata all'intensità. Loudness di una forma d'onda è funzione della frequenza. Per un onda sinusoidale il contorno equivalente di loudness può essere determinato osservando il grafo di Fletcher-Munson. Una curva in questo grafico indica che, per esempio, una sinusoide a 60dB SPL non è intensa (Loud) a 200Hz come lo è a 4000Hz. Non sorprende che l'intervallo delle frequenze entro il quale risiede il parlato ha la massima sensibilità. La funzione bass boost presente in molti prodotti di acustica di consumo compensa la decrescente sensitività alle basse frequenze.

Lo spettro di una sorgente sonora è il maggior responsabile per la qualità percettiva del timbro. Molte volte è chiamato colore del tono. Una definizione negativa legata alla spazializzazione del suono è la seguente: due suoni con lo stesso pitch, timbro e loudness, posizionati in locazioni virtuali distinte, vengono percepiti in maniera diversa. In [25] si individuano cinque parametri per distinguere due suoni che altrimenti sarebbero identici:

- l'intervallo tra un tono ed una porzione rumorosa
- l'inviluppo spettrale
- Attack Decade Sustain Release
- piccoli cambiamenti di pitch nella frequenza fondamentale e nell'inviluppo
- l'informazione contenuta nell'attacco del suono comparata con la porzione successiva.

La percezione del timbro può essere spiegata utilizzando le cosiddette bande critiche. Esse sono degli intervalli di frequenze relazionate con le regioni della membrana basilare. Per semplificare si può pensare al sistema uditivo umano come ad un banco di 24 filtri, ognuno realizzato con successive frequenza centrali e larghezza di banda in modo da

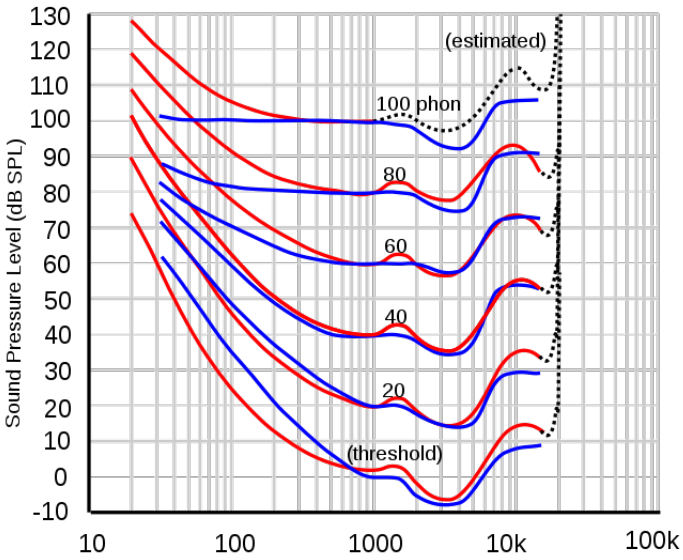


Figura 1.11: Grafico di Fletcher- Munson

coprire tutto lo spetto dell'udibile. La taglia di una banda critica è approssimata ad 1/3 dell'ampiezza di banda di un ottava. In questo modo armoniche posizionate all'interno di una banda critica saranno mascherate dall'armonica più forte (nella stessa banda), rispetto ad armoniche poste in altre bande. Questo suggerisce che il sistema uditivo analizza una forma d'onda entro ogni banda critica indipendentemente. L'energia totale entro ogni banda critica per ogni orecchio è utile per cercare, per esempio, delle prove che coinvolgono aspetti legati alla localizzazione spaziale.

Capitolo 2

Modello per le PRTF

É ben noto che l'orecchio esterno, o pinna, fornisce le maggiori informazioni per determinare l'elevazione di una fonte sonora. Poiché la grandezza e la forma delle pinne variano notevolmente da persona a persona questi stimoli spettrali sono influenzati enormemente dall'ascoltatore. Questa vasta personalizzazione ha portato a sviluppare tecniche per derivare le PRTF da parametri antropometrici e non da difficili e dispendiose misurazioni in laboratorio. In questo capitolo si descrive l'algoritmo sviluppato in [4]. Nel paragrafo 2.1 è presente una dissertazione sullo stato dell'arte. Successivamente nel paragrafo 2.2 si descrive l'algoritmo che analizza le PRTF e le scompone in più sezioni. Dopo aver analizzato i risultati dell'algoritmo (paragrafo 2.3) seguono tre paragrafi che espongono le metodologie per la sintesi delle PRTF.

2.1 Stato dell'arte

In lavori precedenti sulla composizione strutturale delle HRTF, si è visto che un modello che combini indipendentemente i contributi della pinna e della testa produce risultati soddisfacenti [10]. Per semplificare le operazioni di analisi e sintesi, quindi, si può pensare alla pinna come ad un oggetto indipendente: svincolata dalla testa. Ciò comporta una significante semplificazione della risposta della pinna: le PRTF (*Pinna Related Transfer Function*). Partendo da questa ipotesi molti autori hanno presentato i loro lavori sulla sintesi delle PRTF.
Come affermato in [6], le componenti del suono ad alta frequenza, che investono l'orecchio dell'ascoltatore, nel caso in cui la lunghezza d'onda sia inferiore rispetto alla dimensione della pinna, sono solitamente riflessi dalla conca e dall'elice. A causa delle interferenze tra l'onda diretta e quelle riflesse, si possono notare degli avvallamenti nello spettro del segnale: i notch. Essi sono posti con un periodo di $1/\tau_i$ dove τ_i è il ritardo del' $i-$esima riflessione.
Queste osservazioni permettono di realizzare un modello della pinna che simuli i percorsi che l'onda sonora è costretta a percorrere: quello diretto e quelli riflessi (vedi 1.4.2). Questo modello fornisce dei risultati soddisfacenti, ma i coefficienti di riflessione, poiché non variano in funzione della frequenza, tendono a sovrastimare il numero di notch nello

spettro.
Un approccio introdotto in [5], prevede l'introduzione di quattro percorsi paralleli che rappresentano diversi cammini che l'onda sonora è costretta a seguire a seguito di rimbalzi causati dalla pinna. Ognuno di questi tragitti è caratterizzato da un tempo di ritardo e da un fattore d'ampiezza che indica la perdita di energia. Inoltre, poiché le cavità della pinna agiscono come dei risonatori è necessario introdurre nel modello un blocco che approssimi questo comportamento. I parametri con cui costruire questo modello sono determinati decomponendo le HRIR in quattro costituenti sinusoidali, opportunamente scalati e ritardati. A ciascun parametro è associato il corrispondente percorso simulato dal modello.
Si è cercato, successivamente, di associare i parametri del modello con otto caratteristiche antropometriche di ogni soggetto. Purtroppo quest'approccio è complicato e richiede l'utilizzo di uno scanner 3D. Essendo uno strumento molto costoso, l'applicabilità si ristringe ai soli laboratori di ricerca più avanzati.

Un'altra soluzione [22], per sintetizzare le riflessioni, opera sia nel dominio del tempo sia nella frequenza. Raykar, ha osservato che le riflessioni della testa e del torso, gli effetti della pinna e le riflessioni delle ginocchia possono essere osservate sia nel tempo sia nella frequenza. In particolare, nel dominio del tempo, tre diverse creste sono causate dall'interazione del suono con tutti gli ostacoli che durante il suo tragitto incontra: rispettivamente la diffrazione della testa e gli effetti della pinna, la riflessione del torso e per ultimo la riflessione delle ginocchia. Per estrarre le frequenze dei notch causati esclu-

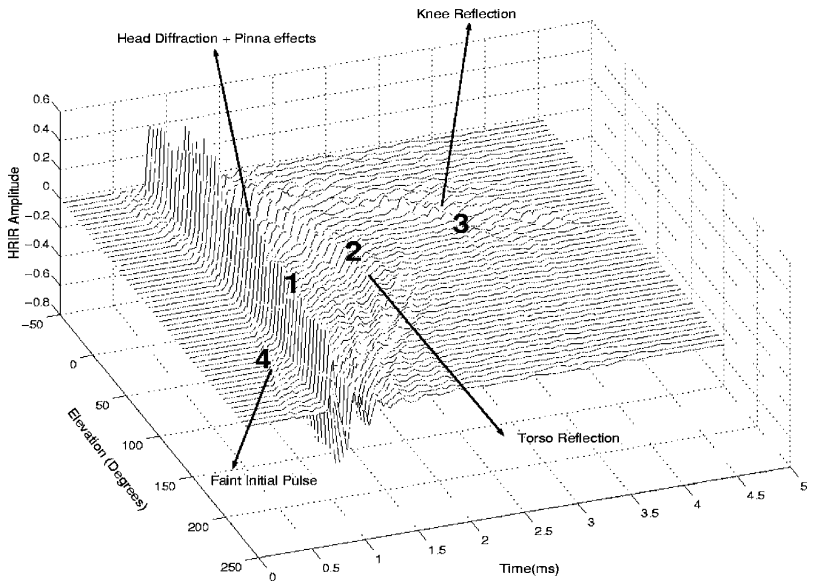


Figura 2.1: Individuazione nelle HRIR dell'interazione tra suono e ostacoli

sivamente dalla pinna gli autori hanno utilizzato delle tecniche basate sui residui delle predizioni lineari delle HRIR. I notch determinati con questa tecnica sono relazionati con la forma della pinna: per ogni notch estratto, la corrispondente distanza è disegnata sulla fotografia della pinna. I picchi spettrali sono determinati in parallelo utilizzando anche

in questo caso una predizione lineare. Il risultato ottenuto è consistente con i risultai presenti in letteratura. In particolare per osservare ciò, si può considerare un sistema con un singolo sommatore e un unico ritardo. Nel dominio del tempo si ha che:

$$y(t) = x(t) + \alpha x(t - t_d) \tag{2.1}$$

con α l'attenuazione della riflessione e t_d il ritardo.
Nel dominio della frequenza:

$$Y(s) = X(s) + \alpha X(s) e^{-st_d} \tag{2.2}$$

$$H(s) = \frac{Y(s)}{X(s)} = 1 + \alpha e^{-st_d} \tag{2.3}$$

Con $H(s)$ filtro *comb*. Se assumiamo che non ci sia attenuazione tra segnale diretto e riflesso gli zeri (o notch) nello spettro sono posizionati:

$$f_n = \frac{n}{2t_d}, \;\; n = 1,\; 3,\; 5,\; ... \tag{2.4}$$

Da questa ultima formula si può notare il legame tra il ritardo dell'onda riflessa e diretta e le frequenze dei notch. Da questa considerazione molti autori hanno cercato di determinare le frequenze dei notch utilizzando solamente delle immagini della pinna.
Un contributo importante nella costruzione di un modello per le PRTF a basso costo è stato introdotto da Satarzadeh [24]. Si è visto che considerando una sola riflessione e utilizzando una struttura di notch non adeguata, le PRTF non vengono correttamente riprodotte. Da queste considerazioni, le PRTF con elevazioni intorno allo zero sono state sintetizzate attraverso un modello composto da filtri passa banda e filtri a pettine. Essi rappresentano rispettivamente le risonanze e i notch presenti nello spettro. I due filtri passa banda sono sommati tra di loro e il risultato è introdotto nel filtro *comb*. É stato dimostrato che questo modello offre una buona approssimazione sia alle PRTF composte da molti notch sia con PRTF che non presentato molti avvallamenti.

2.2 Algoritmo

Partendo dal lavoro di Satarzadeh e dal modello "risonanze più ritardo", in [4] si è costruito un algoritmo che permette di estrarre dalle PRTF un filtro multinotch adattabile a misure antropometriche. Come bacino di dati si è utilizzato il database CIPIC [1], di pubblico dominio che contiene HRIR misurate per 1250 direzioni spaziali per 45 diversi soggetti. Poiché le PRTF non sono sensibili all'azimut, si considerano solo HRIR nel piano mediano, con elevazione che varia da -45 a 90 gradi. Considerato che il modulo della risposta in frequenza di una testa sferica privata delle orecchie è piatta, l'unico accorgimento che bisogna contemplare per ottenere una PRTF è di finestrare la corrispondente HRIR. Non considerando il segnale dopo 1.0ms si eliminano tutti gli effetti causati dalla riflessione delle spalle, torso e ginocchia.

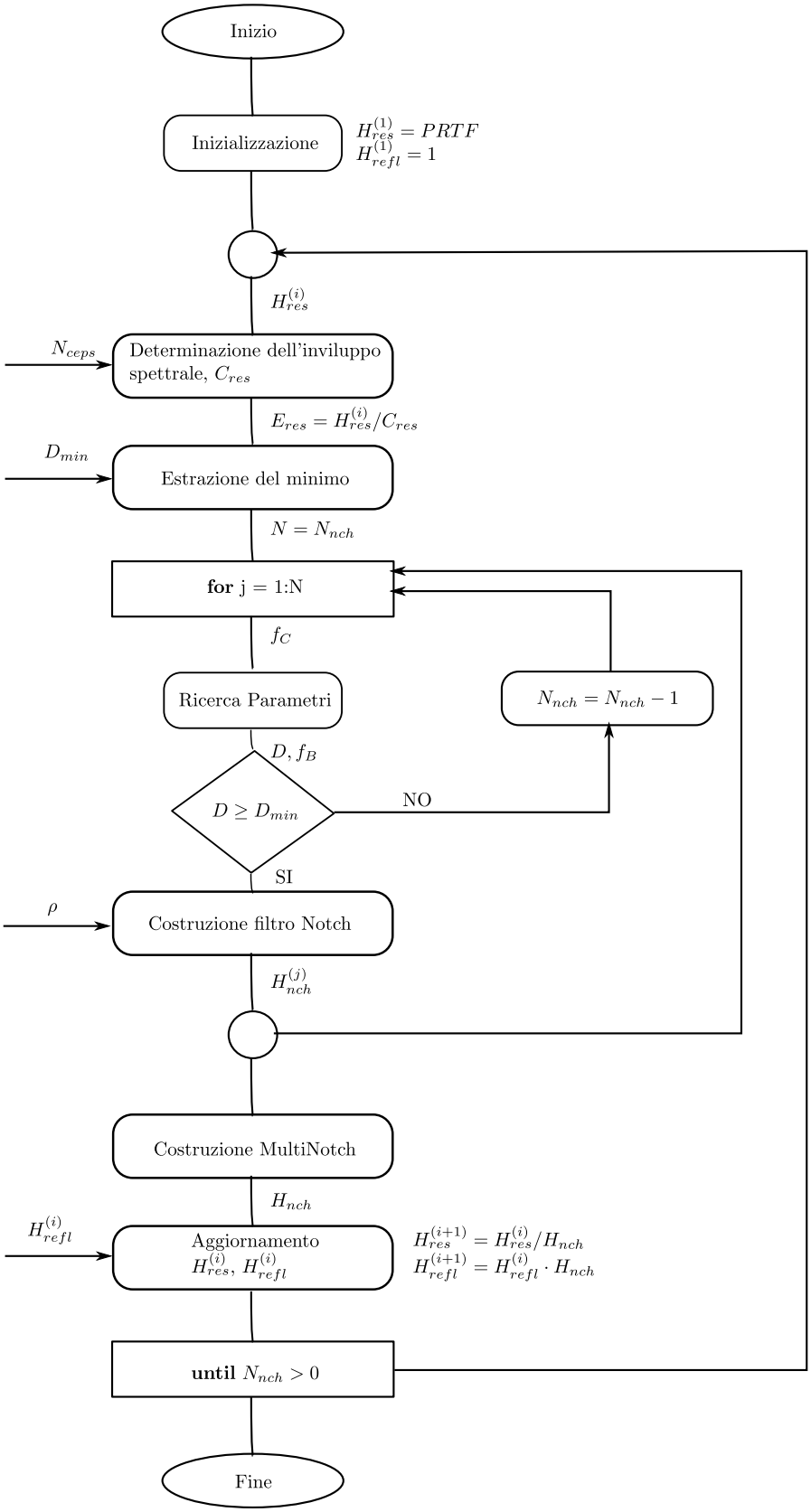


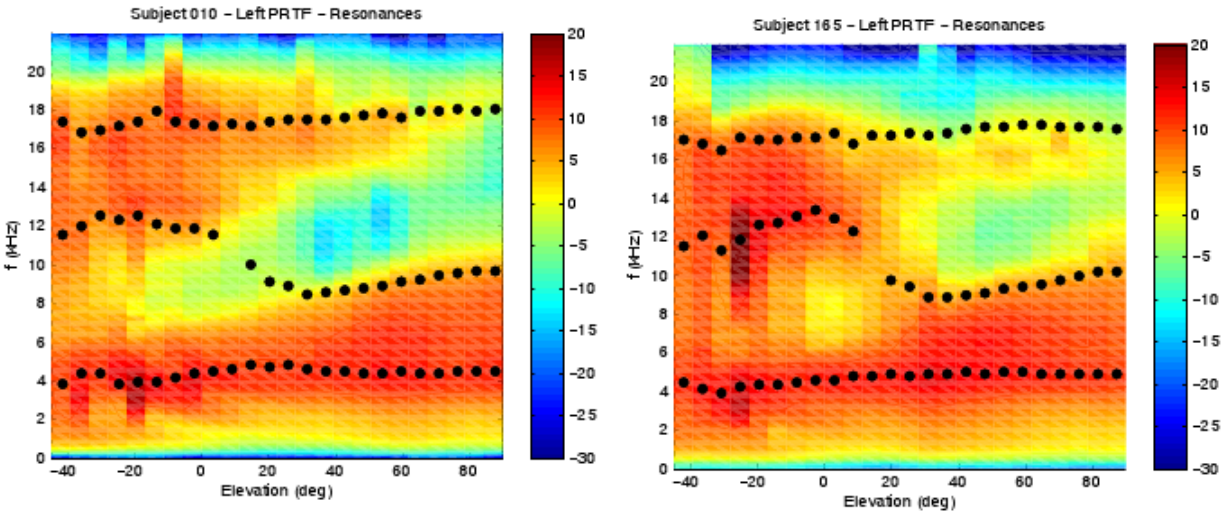
Figura 2.2: Diagramma di flusso dell'algoritmo.

L'idea che sta alla base dell'algoritmo prevede di compensare iterativamente il modulo della PRTF con un filtro multinotch appropriato in modo che vengano considerati tutti i notch più importanti. Quando si è raggiunta l'approssimazione desiderata, per esempio all'iterazione $n-$esima, lo spettro della PRTF $H_{res}^{(n)}$ conterrà la componente risonante, mentre la combinazione dei filtri multinotch fornirà la componente riflessiva $H_{refl}^{(n)}$. Le tre condizioni iniziali, da scegliere, caratterizzano fortemente il risultato finale:

- N_{ceps} il numero di coefficienti di *cepstral* usati per stimare l'inviluppo delle PRTF ad ogni iterazione.
- D_{min} la minima soglia di profondità in dB dei notch da considerare
- ρ il fattore di riduzione per ogni banda passante dei filtri notch (vedi 2.5)

Alla partenza dell'algoritmo vengono impostati i valori per le risonanze $H_{res}^1 = PRTF$, e le riflessioni $H_{refl}^1 = 1$. Queste due risposte in frequenza verranno ad ogni iterazione aggiornate fino a contenere, al termine dell'algoritmo, la componente risonante e riflessiva delle PRTF. Se N_{nch} è il numero di notch identificati al termine di ogni iterazione, l'algoritmo terminerà quando questo valore sarà pari a zero. Il numero di iterazioni e la qualità della decomposizione sono legati con un doppio filo alla scelta dei parametri precedenti. Come si può facilmente intuire impostare quasi a zero il valore di D_{min} può portare ad un numero eccessivo di iterazioni, mentre un valore troppo elevato può non considerare importati riflessioni.
Per estrarre correttamente i minimi locali nelle PRTF causati dai notch della pinna è necessario compensare la componente risonante. A questo scopo, si calcola la parte reale del cepstrum di H_{res}. Da questi è possibile ottenere l'inviluppo spettrale di H_{res}, effettuando una FFT, chiamata C_{res}. N_{ceps} deve essere scelto adeguatamente, in quanto determina il grado di dettaglio dell'inviluppo stesso. All'aumentare di N_{ceps}, il contributo dei notch in modulo e banda diminuisce a discapito delle risonanze che diventano più dettagliate. Sperimentalmente si è trovato che un buon compromesso si ottiene con $N_{ceps} = 4$. Una volta che l'inviluppo C_{res} è stato determinato lo si sottrae da H_{res} ottenendo il residuo E_{res}.
E_{res} avrà un modulo pressoché piatto interrotto da qualche notch. Il parametro N_{notch} è impostato al numero di minimi locali presenti in E_{res} più profondi di D_{min}. Lo scopo è di creare per ogni notch, un filtro notch del secondo ordine, definito da tre parametri: la frequenza centrale f_C, la profondità del notch D e la banda passante f_B. Per ogni minimo locale la frequenza centrale è banalmente determinata, mentre D è pari al valore del modulo di E_{res} calcolato in f_C. f_B è invece pari alla banda a 3dB, ovvero alla differenza tra i valori del modulo di E_{res} a sinistra (f_l) e destra (f_r) di 3dB rispetto a f_C. Questa definizione di f_B non è più vera, in queste situazioni :

- se $D <$ 3dB la banda a 3dB non è definita ed f_r e f_l sono poste ad un valore intermedio tra 0 e $-D$.



(a) Soggetto 010 (b) Soggetto 165

Figura 2.3: Risonanze per due soggetti del Database CIPIC

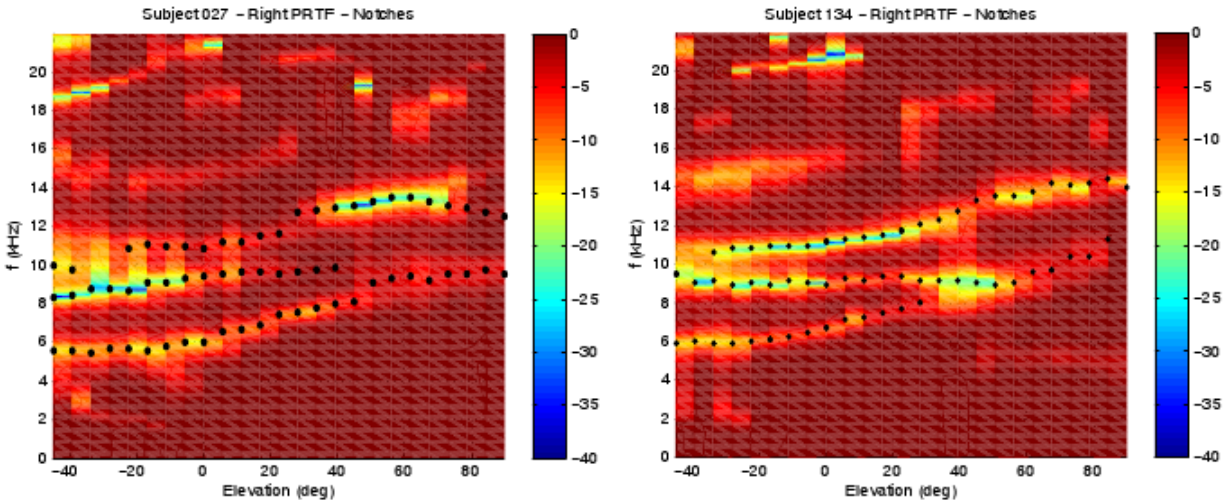
- se il minimo locale di E_{res} immediatamente precedente f_C non supera i 0 dB mentre il massimo locale immediatamente seguente lo supera, f_B è posto uguale al doppio di metà banda tra f_C e f_r

- se il massimo locale non supera i 0dB si sposta verticalmente E_{res} finché un minimo o massimo locale incontra i 0dB. f_B è calcolato come prima tranne nel caso in cui la profondità del notch è minore di D_{min} (modificato). In questo caso il notch corrente non viene considerato e N_{nch} viene decrementato di uno.

Quando i tre parametri f_C, D, f_B sono determinati si procede alla costruzione del filtro multinotch (vedi 2.5). Esso sarà identificato univocamente dai coefficienti del numeratore b e del numeratore a della risposta in frequenza H_{nch}. Si calcolano questi coefficienti per ogni N_{nch} notch. Effettuando di volta in volta una convoluzione tra tutti i coefficienti computati durante un'iterazione, si ottengono i coefficienti finali del filtro multinotch. Come ultima operazione dell'iterazione bisogna aggiornare i valori di $H_{refl}^{(i+1)} = H_{refl}^{(i)} \cdot H_{nch}$ e di $H_{res}^{(i+1)} = H_{res}^{(i)} / H_{nch}$. Questo cilco termina quando N_{ntc} è pari a zero.

2.3 Risultati

Nell'intervallo di frequenza entro il quale gli effetti della pinna sono rilevanti (3÷18 kHz) l'algoritmo produce una decomposizione realistica. Il guadagno della componente riflessiva è unitario al di fuori di queste frequenze. Le risonanze hanno un buon comportamento e simile ai risultati ottenuti in letteratura entro le frequenze rilevanti. In figura 2.3 sono rappresentate le risonanze di alcuni soggetti per tutti gli angoli di elevazione considerati. Ogni frequenza centrale delle risonanze è stata determinata da un sistema



(a) Soggetto 027 (b) Soggetto 134

Figura 2.4: Notch per due soggetti del Database

di identificazione basato su un modello ARMA del sesto ordine. Si possono notare due aree con forte guadagno (colori caldi). La prima, intorno a 4 kHz, rimane costante per tutte le elevazioni e per tutti i soggetti del database. La banda passante cresce all'aumentare dell'elevazione. La seconda area di interesse differisce da soggetto a soggetto sia per quanto riguarda il modulo sia per la forma. Si riesce, comunque, ad individuare una tendenza comune ai soggetti: per basse elevazioni le risonanze si posizionano tra i 12kHz e i 18kHz. Per elevazioni maggiori intorno ai 12kHz il modulo assume valori negativi. Questo può essere causato da limitazioni di computabilità. Le risonanze a più alta frequenza sono, da un punto di vista percettivo, irrilevanti poiché sono al limite dell'intervallo di frequenze udibile. Da queste osservazioni si è pensato di realizzare solo due filtri risonanti. Come per le risonanze, la figura 2.4 rappresenta le frequenze di notch in base all'elevazione. Questi notch dipendono fortemente dall'angolazione verticale e dalla forma della pinna. Quando la sorgente è posizionata sopra la testa dell'ascoltatore nelle PRTF non sono presenti dei notch evidenti. Non appena l'elevazione diminuisce il numero e la profondità dei notch aumenta e varia tra i soggetti. Nonostante ciò si possono notare alcune analogie. Si è utilizzato l'algoritmo di tracciamento delle parziali noto con il nome di McAulay-Quartieri. Quest'algoritmo veniva usato per raggruppare le parziali di sinusoidi, tra più finestre temporali contigue, in funzione della loro posizione spaziale. Modificando l'algoritmo in modo che l'elevazione venga considerata come il tempo e i notch siano le parziali, si può realizzare l'algoritmo che traccia i notch. Poiché è importante focalizzarsi sulle frequenze in cui opera la pinna si è deciso di eliminare le traccie al di fuori dell'intervallo $4 \div 14$ kHz, e quelle che non presentano un avvallamento superiore a 5dB. Il risultato conseguito è simile a quello ottenuto da [22]. Le lacune presenti nelle traccie sono da attribuire all'impossibilità dell'algoritmo di individuare minimi locali in quella regione. Per ogni soggetto si possono notare tre diverse traccie nelle PRTF.

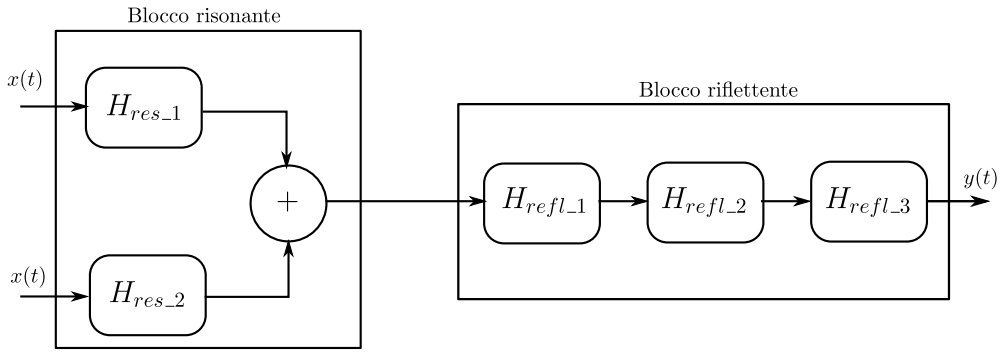


Figura 2.5: Modello con un blocco risonante e uno riflettente

	Freq. centrale [Hz]	Profondità	Freq. Banda [Hz]
notch1	12485	-32.39	110.25
notch2	...	...	...
notch3	...	...	...
notch1	...	...	...
...			

Tabella 2.1: Esempio di matrice utilizzata per rappresentare i notch.

2.4 File contenenti notch e risonanze

L'output del programma scritto in matlab che implementa l'algoritmo [3] è costituito da cinque matrici. Tre contengono informazioni relative rispettivamente alla frequenza centrale, alla frequenza di banda e alla profondità dei tre notch, mentre le altre due alla frequenza centrale e guadagno delle risonanze. Queste matrici non sono logicamente ordinate ed in alcuni punti non presentano alcun valore. É stato necessario modificare la struttura delle matrici, in modo tale che siano ordinate e private di valori mancanti. In particolare, la nuova struttura della matrice deve essere tale che, le $i-$esime righe della tabella per le quali $i \equiv 0(mod\ 3)$ si devono riferire alla prima traccia del notch, quelle che $i \equiv 1(mod\ 3)$ alla seconda traccia del notch ed infine quelle che $i \equiv 2(mod\ 3)$ alla terza traccia del notch. Poiché gli angoli campionati sono 25 (da -45 a 90) il numero di righe è 75. La prima colonna contiene le informazioni relative alla frequenza centrale, la seconda alla profondità e la terza alla frequenza di banda. Per quanto riguarda le risonanze, la struttura del file finale è molto simile. Le righe pari sono relative alla seconda risonanza, mentre quelle dispari alla prima. Le colonne sono solo due in quanto la frequenza di banda è fissata a 5kHz. La prima rappresenta la frequenza centrale, mentre la seconda delinea il guadagno del filtro.

Questa modifica strutturale è stata effettuata con uno script in matlab. Nella figura 2.6 si può osservare che il notch con frequenza più alta presenta dei valori non definiti negli angoli di elevazione tra -40 a -20. Dopo aver processato la matrice di partenza si

	Freq. centrale [Hz]	Guadagno
risonanza1	4500	10.1
risonanza2	...	...
risonanza1	...	...
...		

Tabella 2.2: Esempio di matrice utilizzata per rappresentare le risonanze.

può notare che con un interpolazione lineare sono stati determinati i valori mancanti.

2.5 Filtri

Le informazioni acquisite dalla decomposizione delle PRTF hanno permesso di ricalcare queste risposte in frequenza con due risonanze e tre notch. Quello che si vuole ottenere è un modello composto da due blocchi principali. Il primo, quello risonante, è composto da due filtri risonanti del secondo ordine. In ingresso è presentato lo stesso segnale e l'uscita dei due filtri è sommata per poi essere immessa nel secondo blocco principale: quello riflettente. Questa parte è composta da tre filtri notch del secondo ordine in cascata.

Per determinare univocamente un filtro notch sono necessari tre parametri: f_C, D, f_B. Avendo a disposizione questi dati è facile costruire un filtro del secondo ordine del tipo:

$$H_{nch} = \frac{1 + (1+k)\frac{H_0}{2} + l(1-k)z^{-1} + \left(-k - (1+k)\frac{H_0}{2}\right) z^{-2}}{1 + l(1-k)z^{-1} - kz^{-2}} \tag{2.5}$$

con

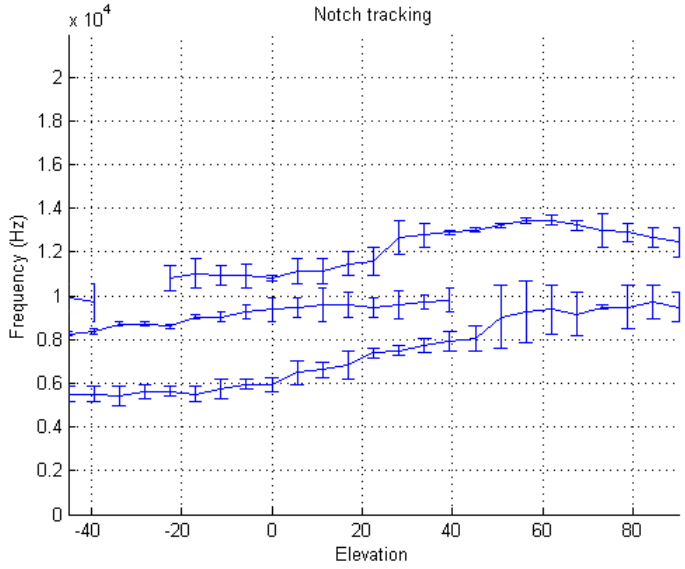
$$k = \frac{\tan\left(\pi \frac{f_B}{f_s}\right) - V_0}{\tan\left(\pi \frac{f_B}{f_s}\right) + V_0} \tag{2.6}$$

$$l = -\cos\left(2\pi \frac{f_C}{f_s}\right) \tag{2.7}$$

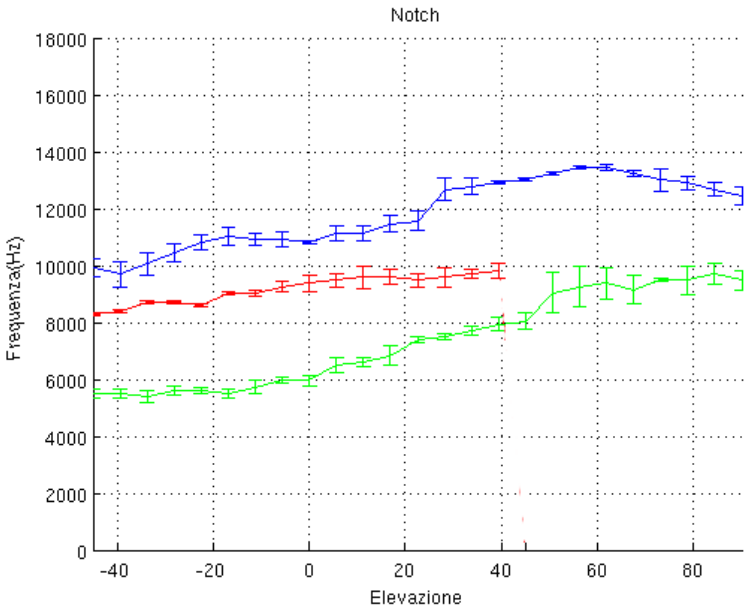
$$V_0 = 10^{D/20} \tag{2.8}$$

$$H_0 = V_0 - 1 \tag{2.9}$$

e f_s frequenza di campionamento. Non tutte le combinazioni dei tre parametri sono realizzabili mediante un filtro IIR di secondo ordine: se il notch che deve essere realizzato è particolarmente stretto e profondo, il filtro produrrà una risposta in frequenza più larga e meno cava, spostando, così, la frequenza centrale del notch. Un altro problema di cui tenere conto è la larghezza di banda. Se essa è eccessiva si possono generare degli spettri alterati. Per ridurre questo fenomeno si utilizza il coefficiente ρ. Dividendo f_B per ρ, la nuova banda passante produrrà un filtro la cui ampiezza del notch sarà



(a) Soggetto 027. Notch derivate dall'algoritmo [3]



(b) Soggetto 027. Notch raffinate a seguito della modifica strutturale effettuata con lo script di matlab

Figura 2.6: Confronto tra i notch determinati dall'algoritmo [3] e quelli raffinati.

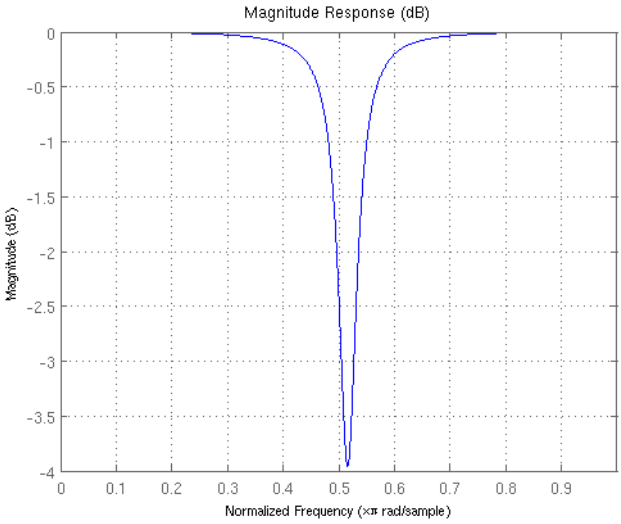


Figura 2.7: Risposta in frequenza del filtro notch con f_C: 11374 Hz D: -3.9 f_B: 699 Hz

ridotta, permettendo di ottenere una larghezza di banda più piccola. Il valore scelto sperimentalmente per ρ è 2.

In questo modello è necessario riprodurre, anche, due risonanze. Un filtro utile per la realizzazione di tali curve è un filtro peak IIR del secondo ordine. Si fissa il valore della larghezza di banda a 5 kHz e si deducono i valori per la frequenza centrale f_C e di guadagno G dai tracciati presenti nelle figure 2.3. La funzione di trasferimento è:

$$H_{res}(z) = \frac{V_0(1-h)(1-z^{-2})}{1+2dhz^{-1}+(2h-1)z^{-2}} \tag{2.10}$$

con

$$h = \frac{1}{1+\tan\left(\pi\frac{f_B}{f_s}\right)} \tag{2.11}$$

$$d = -\cos\left(2\pi\frac{f_C}{f_s}\right) \tag{2.12}$$

$$V_0 = 10^{G/20} \tag{2.13}$$

Questo tipo di filtro ha un inviluppo che si adatta molto bene alle HRTF, purtroppo modifica anche altre frequenze. In particolare attenua pesantemente il modulo al di fuori della frequenza di banda. Questo può essere un problema per le frequenze basse, quelle udibili, mentre non lo è per quelle superiori a 14kHz; infatti quello che si vuole ricostruire sono le PRTF, e queste funzioni non sono importanti per tali frequenze. Per eliminare questo tipo di problema, per la prima risonanza si è utilizzato un filtro peak del secondo ordine che non modifica l'intervallo al di fuori dalla frequenza di banda [28]. Questo filtro peak è il duale di quello utilizzato per i notch e la sua funzione di trasferimento è:

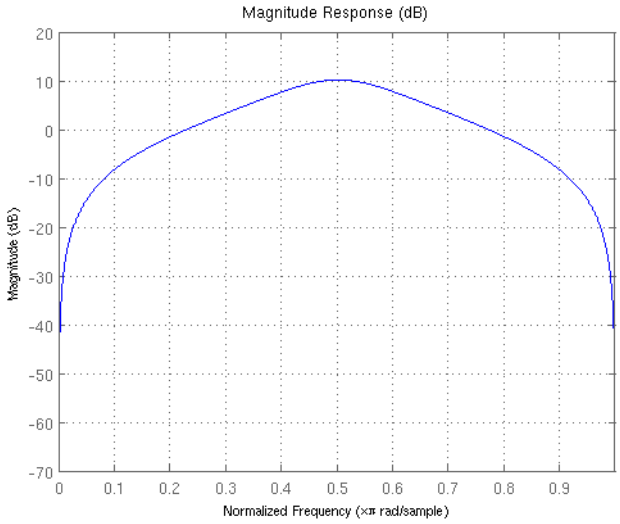


Figura 2.8: Risposta in frequenza di un filtro peak del secondo ordine G: 14.7868 f_C: 11096.9 Hz f_B:5000 Hz

$$H_{res}(z) = \frac{1 + \frac{H_0}{2}(1+k) + d(1-k)z^{-1} + \left(-k - \frac{H_0}{2}(1+k)\right) z^{-2}}{1 + d(1-k)z^{-1} - kz^{-2}} \tag{2.14}$$

con

$$d = -\cos\left(2\pi \frac{f_C}{f_s}\right) \tag{2.15}$$

$$V_0 = 10^{G/20} \tag{2.16}$$

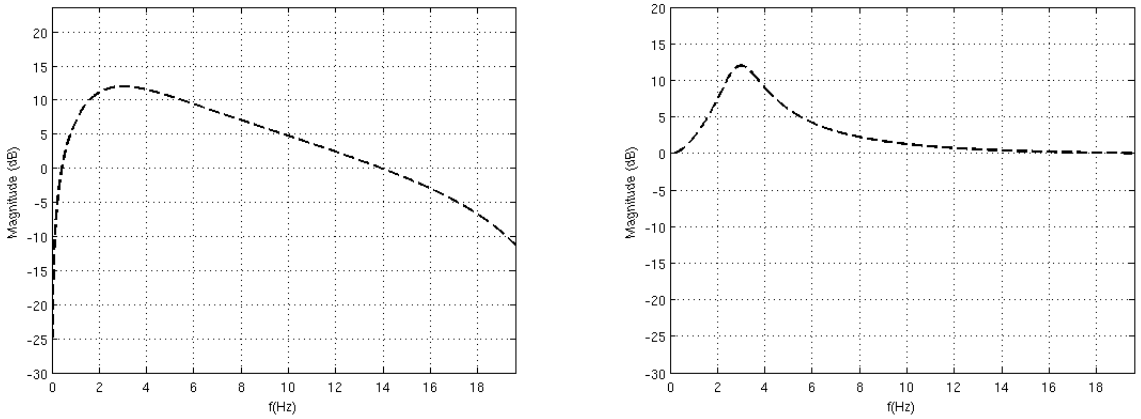
$$H_0 = V_0 - 1 \tag{2.17}$$

$$k = \frac{\tan\left(\pi \frac{f_B}{f_s}\right) - 1}{\tan\left(\pi \frac{f_B}{f_s}\right) + 1} \tag{2.18}$$

Per la seconda risonanza si è utilizzato il filtro peak dell'equazione 2.10. Gli inviluppi di questi due filtri sono rappresentati in figura 2.9. I parametri sono $f_c = 3kHz$, $f_B = 5kHz$ e $g = 12$.

2.6 Sintesi

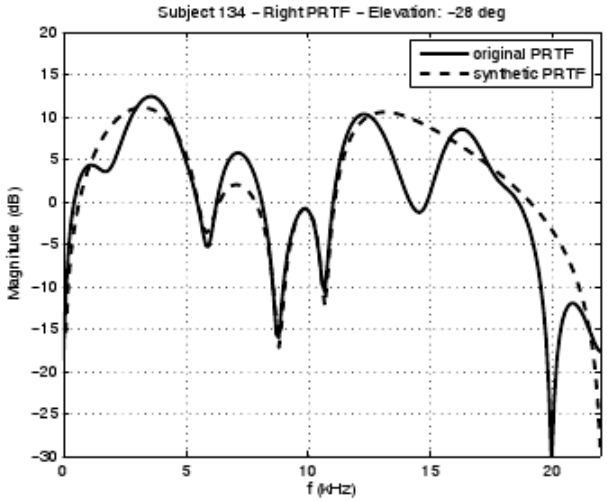
I risultati ottenuti con questo algoritmo sono soddisfacenti. In figura 2.10 è rappresentato un confronto tra le PRTF reali, linee continue, e le PRTF ricostruite linee tratteggiate. Fino a 14kHz la somiglianza tra le due funzioni è buona anche se sono



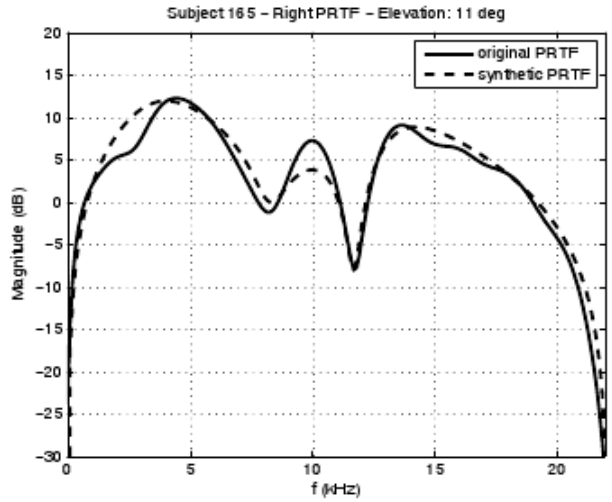
(a) Filtro peak dell'equazione 2.10 (b) Filtro peak dell'equazione 2.14

Figura 2.9: Confronto tra i due filtri Peak IIR del secondo ordine

presenti alcune piccole imperfezioni: i notch profondi presenti a basse elevazioni complicano la procedura di costruzione del filtro. Per esempio se il notch da modellare è particolarmente stretto ed appuntito e vicino ad esso è presente un altro notch, il filtro del secondo ordine potrebbe costruire una risposta in frequenza che non corrisponde alla PRTF originaria nell'intervallo di frequenze tra i due notch. Inoltre il fatto di non aver considerato i notch oltre i 14.5 kHz è un altra fonte di errore. Per frequenze superiori a questa soglia l'approssimazione delle PRTF artificiali è scarsa. Tuttavia poiché l'importanza psicoacustica di tali frequenze è molto bassa, l'incongruenza tra le due funzioni non indice sul risultato finale. Per quanto riguarda le risonanze la scelta di non modificare la larghezza di banda può rappresentare una limitazione alla procedura di sintesi. Un altro problema è rappresentato dall'algoritmo per determinare le frequenze centrali dei peak. Non sempre coincidono con i picchi reali. Il modello produce un risultato soddisfacente soprattutto per PRTF con una o due risonanze principali e notch non troppo pronunciati.



(a) Soggetto 134. Elevazione -28



(b) Soggetto 165. Elevazione 11

Figura 2.10: Confronto tra PRTF reali e sintetiche in MATLAB

Capitolo 3

Estrapolazione dei parametri del modello da misurazioni antropometriche

In questo capitolo si illustra la relazione tra i parametri utilizzati per la costruzione dei filtri introdotti nel capitolo 2, per personalizzare le PRTF in funzione dell'ascoltatore, e la forma e dimensione della pinna. Lo scopo è di eliminare la misurazione in laboratorio delle HRTF con costose apparecchiature audio. Sebbene, come più volte sottolineato, le PRTF sono completamente determinate dall'esatta forma della pinna, il fine, è di riuscire a determinare poche caratteristiche antropometriche che sono sufficienti per determinare i parametri dei filtri che compongono il modello. Lavori precedenti hanno suggerito che molte delle caratteristiche delle PRTF hanno una semplice corrispondenza con la dimensione della pinna. In particolare, Shaw [27] ha relazionato la profondità e la larghezza della conca con la profondità delle risonanze, mentre in [22] i notch nelle PRTF sono in corrispondenza con la distanza tra l'ingresso del canale uditivo al bordo della conca.

3.1 Tempo di ritardo t_d

Il tempo di ritardo t_d è legato ai notch presenti nelle PRTF. t_d secondo molti autori [6], [22] ha una semplice spiegazione in termini antropometrici. L'argomentazione che accomuna i ricercatori è quella di considerare il suono come dei raggi direzionali che investono la pinna. L'onda diretta viene riflessa dal bordo della conca. Se $x(t)$ è l'onda diretta allora il segnale $y(t)$ percepito è dato dalla somma del segnale diretto e quello riflesso.

$$y(t) = x(t) + \alpha x(t - t_d(\phi)) \tag{3.1}$$

con α un coefficiente di riflessione. ϕ è l'angolo di incidenza dell'onda diretta che ovviamente incide su t_d. Il ritardo corrisponde alla distanza tra l'ingresso del condotto uditivo e la parte riflettente, per esempio il bordo della conca. Tale distanza w è data da:

$$w = \frac{ct_d}{2} \tag{3.2}$$

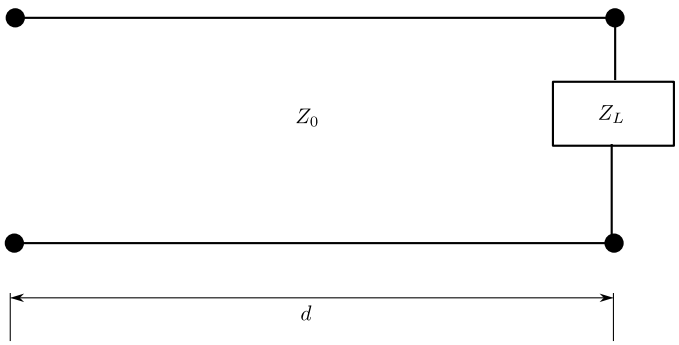


Figura 3.1: Linea di trasmissione.

Si è diviso per un fattore due, perché l'onda sonora dovrà percorrere due volte il tragitto a causa del processo di riflessione. Il ritardo $t_d(\phi)$ causa più notch nello spettro le cui frequenze sono pari a:

$$f_n(\phi) = \frac{(2n+1)}{2t_d(\phi)} = \frac{c(2n+1)}{4w(\phi)},\ n = 0, 1, ... \tag{3.3}$$

La frequenza del primo notch è a:

$$f_0(\phi) = \frac{c}{4w(\phi)} \tag{3.4}$$

Lo studio in [26] ha considerato le immagini delle pinne dei soggetti del database CIPIC. In particolare per ogni soggetto si sono determinati i notch predominanti nelle PRTF corrispondenti. In funzione dell'angolo di elevazione, attraverso la 3.2 si sono disegnate le corrispondenti distanze prendendo come origine degli assi il targo. Applicando questo procedimento al soggetto 40 del database si è osservato uno strano fenomeno. Il punto di riflessione è stato posto molto lontano dalla pinna, esattamente al doppio della distanza dell'elice. L'esperimento è stato ripetuto per più soggetti e si è riscontrato lo stesso comportamento, ma non per tutti i soggetti. Per alcuni, $w/2$ è stato posizionato in vicinanza del bordo della conca e non nell'elice. Questo mostra che, a seconda dei casi è l'elice o la conca, a determinare la riflessione delle onde sonore. Il fatto sorprendente è il fattore due che si presenta costantemente tra i soggetti. Perché si trova $w/2$ e non w in prossimità dell'elice o della conca? Sebbene le interazione tra la pinna e l'onda sonora incidente sono molto complesse, Satarzadeh è riuscito a fornire una spiegazione abbastanza semplice, creando un'analogia con le linee di trasmissione analogiche. Una linea analogica ha una impedenza caratteristica indicata con Z_0, un carico indicato con Z_L, e una data lunghezza d.

Assumendo che sia presente una tensione in ingresso, la tensione presente nella linea è caratterizzata dall'interazione tra le onde dirette e riflesse. Le onde riflesse dipendono dal coefficiente di riflessione che è dato da:

$$\Gamma = \frac{Z_L - Z_0}{Z_L + Z_0} \tag{3.5}$$

Quindi "trasformando" la formula 3.1 al corrispettivo esempio elettronico otteniamo che:

$$V(s) = V_0(1 + \Gamma e^{-j\omega 2d/c}) = V_0(1 + \Gamma e^{-j\omega t_d}) \tag{3.6}$$

cioè la convoluzione tra l'input e il filtro a pettine che descrive i notch. L'interpretazione del coefficiente di riflessione assumeva che il carico e l'impedenza caratteristica fossero puramente reali, con il modulo del carico maggiore del modulo dell'impedenza caratteristica: $|Z_L| > |Z_0|$. Queste condizioni permettono al coefficiente di riflessione di essere sempre nell'intervallo $[0, 1]$, mentre in generale le possibili valori per tale coefficiente possono essere tra -1 e 1. Per chiarire, consideriamo il caso in cui $|Z_L| < |Z_0|$, cioè il coefficiente di riflessione negativo. Per semplicità si assume che il coefficiente di riflessione abbia modulo pari a 1 e fase di π. Il corrispettivo filtro *comb* è del tipo:

$$H_{comb} = 1 - e^{-j\omega t_d} \tag{3.7}$$

poiché $e^{-j\pi} = -1$. I notch si presentano in corrispondenza degli zeri, cioè quando:

$$-e^{-j\omega t_d} = -1 \tag{3.8}$$

Sostituendo l'uguaglianza $e^{-j\pi} = -1$ in 3.8 si ottiene:

$$-e^{-j\omega t_d - \pi} = e^{-j\pi} \quad \text{ovvero} \quad \omega t_d - \pi - \pi = 0 \tag{3.9}$$

$$\omega_n = \frac{2n\pi}{t_d},\ n = 1, 3, 5, ... \tag{3.10}$$

Come abbiamo visto in precedenza per un coefficiente di riflessione positivo invece si ha:

$$\omega_n = \frac{n\pi}{t_d},\ n = 1, 3, 5, ... \tag{3.11}$$

Le equazioni 3.11 e 3.10 differiscono proprio di un fattore due. Assumendo che il carico di impedenza Z_L sia vicina a zero (i.e. il circuito analogico è corto) allora il fattore di riflessione è pari a -1. Purtroppo non è il caso che interessa in questa situazione. L'impedenza della pinna corrisponde al carico mentre l'aria rappresenta l'impedenza caratteristica. L'impedenza della pinna è maggiore rispetto all'aria e quindi il coefficiente di riflessione è positivo. Poiché il circuito che si sta considerando non è corto, l'analogia con le linee analogiche non è applicabile. É possibile comunque, effettuare qualche modifica al modello delle linee analogiche per ottenere dei buoni risultati. Si consideri la natura fisica della pinna. La figura 3.1 mette in evidenza che esiste un confine creato dalla differenza di impedenza, che può produrre una propria riflessione. Questa può quindi introdurre un cambio di fase all'onda riflessa portandola a π. Il fenomeno del cambio di fase può essere osservata nelle PRIR. In particolare si cerca una copia negativa del primo impulso ritardato di metà tempo rispetto al tempo di ritardo calcolato. Satarzadeh è riuscito a identificare il fenomeno della riflessione negativa in 16 soggetti su 20 del database CICIP. L'autore afferma che supporre il coefficiente di riflessione negativo è un assunto valido. Sebbene sia chiaro come il tempo di ritardo t_d sia legato alla forma della pinna, non si è

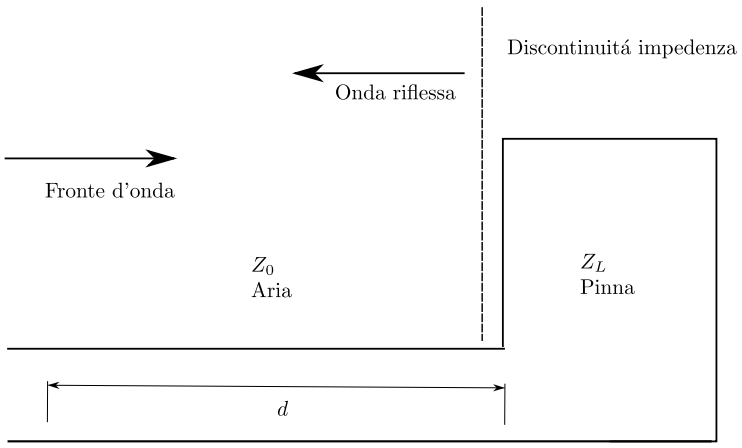


Figura 3.2: Modello fisico della pinna [26]

ancora riusciti ad associare il ritardo ad una distanza fisica. Si è già visto che la distanza che garantisce un'adeguata periodicità è attribuibile all'elice o alla conca a seconda del soggetto. Purtroppo, non esiste un criterio per decide quali dei due elementi della pinna scegliere. Questa difficoltà nasce dalla complessità del processo di dispersione di energia. Il modello precedente semplifica di molto questo processo, poiché le riflessioni vengono considerate monodimensionali, mentre esse operano chiaramente in tre dimensioni. In aggiunta sono presenti anche molte altre riflessioni oltre a quella dominante. Una regola euristica per decidere quale parte della pinna è il maggior riflettore è verificare, da una vista frontale, la curvatura della pinna. Se l'orecchio è ad ansa, l'elice sarà il riflettore principale, altrimenti sarà la conca. Tuttavia in [12] si è osservato che questo metodo non sempre fornisce risultati soddisfacenti ed è tuttora in corso uno studio dei ricercatori del dei per riuscire ad individuare in maniera precisa l'elemento della pinna che provoca la maggior riflessione.

3.2 Prima risonanza

Shaw [27] ha notato che c'è una stretta correlazione tra la frequenza delle risonanze e la profondità e la larghezza della conca. Shaw ha scelto di rappresentare la conca come un cilindro, che è una delle forma geometriche elementari che più assomiglia alla conca. Il cilindro simula bene le risonanze fino a 5kHz e quindi descrive bene la prima risonanza. La frequenza di risonanza del cilindro è stata ampiamente studiata ed è calcolata come:

$$\frac{\lambda_{max}}{4} = d + 0.822r \tag{3.12}$$

con 0.822 termine di correzione finale, d altezza e r raggio del cilindro. In termini di frequenza si ha che:

$$f_{max} = \frac{c}{4(d + 0.822r)} \tag{3.13}$$

con c velocità del suono. Il raggio del cilindro è pari a 1/4 di w calcolato utilizzando la formula 3.2. Si è diviso per quattro poiché w è il doppio della distanza antropometrica e il raggio è la metà di tale valore. Satarzadeh [26] ha verificato la veridicità della formula 3.13 su venti soggetti presenti nel database CIPIC, ponendo d al valore di profondità presente nel database e r pari al risultato della formula 3.2. Non si hanno avuti errori superiori a 1kHz.

3.3 Seconda Risonanza

In letteratura non sono presenti molti risultati o ricerche che mettono in relazione la seconda risonanza e la dimensione della pinna. In [26] si è osservato che la seconda risonanza è legata al filtro *comb* usato per descrivere i notch. Si è osservato che se il riflettore principale è la conca il terzo picco del filtro comb corrisponde alla distanza w, se invece è l'elice il maggior elemento riflettente, il quarto picco del filtro comb corrisponde alla distanza. Questo è sensato poiché rispetto al targo l'elice è più distante della conca. Poiché la distanza è maggiore, lo è anche il ritardo e quindi il filtro comb ha un periodo minore. Minore è la periodicità più sono presenti dei picchi prima che il risonatore raggiunga la frequenza di risonanza. Questa è una buona osservazione perché la predizione del filtro comb permette di determinare simultaneamente sia i notch sia la seconda risonanza. Si cerca una formulazione per la seconda risonanza in funzione del filtro *comb*. I poli o i picchi del filtro comb soddisfano:

$$\begin{aligned} &H_{comb}(e^{j\omega}) = 1 - e^{-j\omega t_d} = 2 \\ &e^{-j\omega t_d} = -1 \\ &\omega_n = \frac{1 + 2n\pi}{2t_d},\ n = 1, 2, 3, ... \end{aligned} \tag{3.14}$$

In questo caso si è utilizzato un coefficiente di riflessione negativo. Utilizzando la 3.14 possiamo determinare la frequenza espressa in Hz della seconda risonanza da:

$$f = \begin{cases} \dfrac{7}{2t_d} & \text{se } \dfrac{w}{2} \simeq \text{conca} \\ \dfrac{9}{2t_d} & \text{se } \dfrac{w}{2} \simeq \text{elice} \end{cases} \tag{3.15}$$

3.4 Personalizzazione

Dalla discussione precedente si evince che sono necessarie solo due misurazioni antropometriche per ricostruire correttamente le PRTF. Una è la profondità l'altra la larghezza della pinna. La definizione di larghezza della pinna dipende dall'elemento dell'orecchio che apporta la maggior riflessione. Come visto questo elemento può essere la conca o l'elice. Non è ancora del tutto chiaro come determinare tale elemento, alcuni autori suggeriscono di sfruttare la curvatura ad ansa dell'orecchio. Se questa curvatura è pronunciata probabilmente l'elemento più riflettente è l'elice altrimenti la conca. Una volta che sono

state determinate queste misurazioni è possibile determinare i parametri necessari per la costruzione del modello descritto nel capitolo 2.

Capitolo 4

Pure Data

In questo capitolo si descrive l'ambiente realtime nel quale è stato sviluppato l'algoritmo di sintesi [4]. Dopo una breve introduzione si analizzano gli elementi del linguaggio (paragrafo 4.3), i due livelli di lavoro dell'ambiente: controllo ed audio (paragrafi 4.4 e 4.5). Successivamente sono mostrate due tecniche per eliminare le discontinuità nei segnali audio (paragrafo 4.6). Infine si presenta un progetto per la creazione di external di Pure Data (paragrafo 4.7).

4.1 Introduzione

Pure Data (Pd) è un linguaggio di programmazione per musica elettronica (DSP). Con il termine DSP (*digital signal processing*) si indica la capacità del computer di manipolare segnali digitali. Pd fu creato dall'ingegnere del software americano Miller Puckette, già sviluppatore del famoso Max/Msp [21]. Pd è open source: non è sviluppato da nessuna azienda e non è in vendita. Il codice sorgente può essere consultato da chiunque gratuitamente (*free*). Come sottolinea più volte Stallman free si deve intendere come libertà di parola e non come birra gratis. Il software libero rispetta, quindi, la libertà d'espressione dell'utente. Una persona con abbastanza competenze informatiche è in grado di modificare il programma a suo piacimento.
Ad oggi molti programmatori, musicisti, ingegneri acustici e compositori si sono uniti a Miller Puckette per continuare lo sviluppo di Pd. Uno svantaggio di questa politica risiede nel impossibilità, per una persona estranea a queste tematiche, di capire (ed utilizzare) questo software. Proprio perché non esiste un'azienda, che ha interessi monetari nella diffusione del programma, l'accessibilità è ristretta ad una piccola cerchia di persone. Nonostante ciò, grazie alla disponibilità immediata del programma, garantita da internet e il democratico avanzamento del progetto, Pd è ottimizzato per l'utilizzo a livelli professionali.
Pd è definito come un ambiente di programmazione grafico real-time per il trattamento dell'audio. Esso utilizza degli oggetti grafici che l'utente posizione e collega sullo schermo. Non si tratta, quindi, di una programmazione basata sulla scrittura di un codice ma della creazione di un flow chart. Gli oggetti inseriti nel programma svolgono svariate funzio-

nalità. Alcuni blocchi sono utili per la gestione dei segnali audio altri per la supervisione dei segnali di controllo. Essi, inoltre, per interagire fra di loro sono collegati attraverso delle linee. L'analogia con il mondo reale è lampante: i dispositivi fisici utilizzati prima dell'avvento del computer music, sono rappresentati dagli oggetti che si posizionano sullo schermo e i cavi sono raffigurati dai collegamenti tra l'entità virtuali. Proprio per questo Pd è definito come un linguaggio di programmazione orientato al flusso di dati (*datastream-oriented programming language*).
Uno degli aspetti fondamentali di Pure data risiede nell'esecuzione real-time. I cambiamenti effettuati in Pd sono istantanei; il comportamento non è uguale ai classici linguaggi di programmazione nei quali il codice deve prima essere processato (es. compilato) prima di poter essere utilizzato. Pd si comporta come uno strumento musicale classico: l'esecutore sente istantaneamente la modifica del suono, a seguito di un suo intervento. Questo rende Pd utilizzabile anche in performance live.
In questi anni Pd è diventato una comunità. Poiché molte persone da tutto il mondo partecipano attivamente allo sviluppo del progetto, sono stati creati dai programmatori dei moduli (external) che permettono l'ampliamento delle funzioni di Pd stesso: video, connessioni Internet, integrazioni con joystick (es WiiMote)... Alcune di queste librerie sono state inserite nella distribuzione ufficiale di Pure Data.

4.2 Pd e la open source community

Pd è rilasciato sotto licenza BSD che permette ma non richiede la pubblicazione e la modifica del codice sorgente. É quindi possibile che il codice di Pd sia incorporato in software proprietario[1].
Raymond (autore del libro la cattedrale ed il bazaar) elenca tre punti essenziali che caratterizzano un progetto Open Source: motivazione personale, visibilità del codice, e gerarchia piatta. Mentre i grandi progetti Open Source, come il kernel di Linux, non seguono più questo schema, questi punti rimangono fondamentali per lo sviluppo dinamico di Pure Data.
Molti progetti open source iniziano a svilupparsi quando un programmatore incontra un problema per il quale non esiste ancora una soluzione software adeguata. In questa situazione si verificano due casi. Se il problema non può essere affrontato utilizzando un programma già esistente (eventualmente modificandolo), un nuovo progetto deve essere pianificato. Miller Puckette scelse questa strada quando scrisse la prima versione di Pure data.
Nella maggior parte dei casi il problema è risolvibile mediante l'utilizzo e la riscrittura di programmi open source già presenti. In questo caso, la strategia più efficiente prevede la scrittura di nuove funzioni cosicché esse possano essere aggiunte al programma originario. Poiché è molto probabile che più persone nello stesso periodo affrontino lo stesso problema è sensato pensare che questi sviluppatori cooperino per il fine comune. Questo è il lavoro svolto dalla comunità di Pure Data: espandere l'area applicativa del programma

[1]Spore EA games. `http://lists.puredata.info/pipermail/pd-list/2007-11/056307.html`

scritto da Miller Puckette. La connessione che si instaura tra i membri della comunità è cruciale. Il gruppo è composto da un insieme di persone specializzate in diverse aree che si aiutano a vicenda. L'evoluzione di Pure Data è alimentata dalle necessità e dalle motivazioni individuali, a cui devono essere sommate le interazione tra i membri della comunità. Più le persone comunicano tra di loro, si scambiano opinioni ed in generale si stimolano a vicenda, più il progetto risulta vivo e fervido.
La visibilità del codice sorgente permette a tutti gli sviluppatori ed utenti di influenzare lo sviluppo del progetto. Questo può essere raggiunto dalla semplice segnalazione di un bug o dalla discussione sul mantenimento di un certo frammento di codice. La visione del codice sorgente comporta due vantaggi: gli errori sono trovati e riparati molto velocemente poiché potenzialmente tutti gli utenti sono dei tester e degli sviluppatori. Nuove idee possono essere proposte da tutti i membri, e un processo di apprendimento collaborativo è incoraggiato attraverso l'utilizzo del forum. Entrambi gli aspetti sono molto importanti per la sopravvivenza di Pure Data come programma e come comunità. Per poter aumentare il numero dei partecipanti è necessario che l'architettura del software sia quanto più modulare possibile. Questo permette a più persone di lavorare in parallelo senza aumentare lo sforzo di coordinamento della comunità stessa. Pd è altamente modulabile, grazie all'utilizzo delle external. La possibilità di collegare questi moduli senza dover modificare il core di Pure Data permette uno sviluppo teoricamente illimitato che ha già superato le più rosee aspettative iniziali. Il mantenimento e la descrizione di questi external rimane comunque un problema. L'identificazione e la nomenclatura degli oggetti non è sottoposta a nessun piano specifico. L'unico modo per diminuire i doppioni ed aumentare l'unificazione degli external è realizzare un'adeguata documentazione. L'integrazione degli external permette, quindi, una gerarchia piatta.
Per quanto riguarda il core del programma la situazione è diversa: la gerarchia non è piatta. Pukette indiscutibilmente ricopre un ruolo di rilievo all'interno della comunità. Lui ha scritto il core del programma ed ha influenzato pesantemente l'organizzazione della comunità. Nel frattempo molti altri programmatori hanno contribuito allo sviluppo del nucleo del programma rimanendo, comunque, costantemente in contatto con Puckette, attraverso la mailing list.

Uno studio sulla composizione dei membri della comunità ha rivelato che la totalità dei partecipanti è di sesso maschile e di età superiore ai 20 anni. Più del 65% delle persone è di età compresa tra i 20 e 29 anni. Tutti i partecipanti al progetto sono mossi da un forte senso di partecipazione e coinvolgimento.

4.3 Elementi di Pd

In Pd possono essere definiti diversi tipologie di dati di base: oggetti, messaggi, numeri ed atomi. Gli oggetti sono identificati da delle caselle rettangolari, i messaggi hanno una strana indentazione sulla destra mentre i numeri hanno un riquadro con un angolo destro piatto. Un messaggio molto importante è il `bang`; questo segnale viene utilizzato

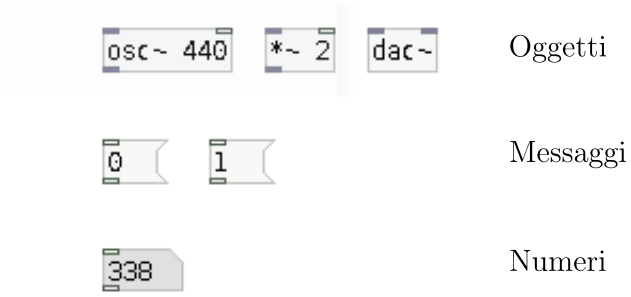


Figura 4.1: Elementi di Pure Data

per iniziare eventi ed immettere dati nei flussi. L'equivalente fisico di questo messaggio potrebbe essere un bottone.

Tutti questi dati hanno degli ingressi (inlet) e delle uscite (outlet). Gli ingressi sono sempre posizionati sulla parte alta del rettangolo, mentre le uscite nella parte bassa. Per poter modificare una patch (il programma di pd) è necessario entrare in *edit mode*. Una volta che la patch è pronta si può passare in modalità di esecuzione (*execute mode*).
Esistono due tipi di "cavi" che si possono utilizzare, per collegare i riquadri, sottili e spessi. I cavi sottili trasportano informazioni riguardanti il controllo degli oggetti, mentre quelli spessi trasmettono segnali. Tutti gli oggetti che producono o gestiscono suoni hanno una tilde $\sim$ dopo il nome (esempio osc$\sim$), gli altri oggetti ne sono sprovvisti. Tutti questi accorgimenti ci permettono di gestire due livelli: quello di controllo, dove solamente messaggi di controllo possono fluire, e quello del segnale, nel quale si gestiscono i suoni.

Semantica

Il testo in un rettangolo inserito in Pd può avere differenti funzioni, sia esso un messaggio, un atomo (numero, simbolo) o un oggetto. Nei riquadri che contengono messaggi, il testo specifica il messaggio che sarà inviato in output. Nei riquadri che contengono atomi il testo cambia al momento dell'esecuzione del programma per mostrare lo stato del riquadro, che può essere o un numero o un simbolo. In un riquadro che contiene oggetti, il testo specifica un messaggio; in questo caso il messaggio viene inviato direttamente a pd che ha il compito di creare l'oggetto indicato dal messaggio stesso. Quando viene aperto un file, tutti gli oggetti sono creati usando il proprio testo come un "messaggio di creazione". Se si modifica il messaggio con un nuovo oggetto, quello vecchio viene distrutto per lasciare spazio all'ultimo. Il selettore del messaggio (la prima parola nel testo) è una direttiva che indica a pd quale oggetto deve creare. Successivamente sono riportati gli argomenti per l'oggetto, se previsti. Al momento della creazione di una patch si dovrebbero immediatamente capire le sue funzionalità. Per questo motivo se un messaggio ricevuto da un oggetto cambia le azioni dell'oggetto stesso, poiché questi cambiamenti non si ripercuotono nella visualizzazione dell'oggetto, questi non saranno salvati nel file che specifica la patch. Allo stesso modo se si cancella e si ricrea un oggetto

questo non sarà conservato ma nuovamente ricreato.
Come precedentemente descritto, la creazione di un oggetto prevede l'utilizzo di messaggi. Messaggi di questo tipo sono formati da un selettore e da uno o più argomenti. Come prima operazione pd controlla l'appartenenza ad una classe del selettore. I riquadri messaggio appartengono tutti alla stessa classe: come input hanno lo stesso segnale e come output producono il valore del proprio stato interno. Gli oggetti, invece, possono avere diverse classi. Ad ogni classe sono associati un inseme di messaggi che possono ricevere in input e gli output che possono produrre.

4.4 Livello di controllo

Poiché Pure Data è un programma real-time è necessario un livello diverso da quello audio, per gestire gli oggetti della patch. Queste computazioni devono poter essere eseguite ad intervalli irregolari di tempo. É fondamentale per ogni computazione creare una corrispondenza con un tempo logico specifico. Il tempo logico indica quale sarà il primo campione audio di uscita che rispecchierà il risultato della computazione. In sistemi non real-time, questo implica che il tempo logico inizia da zero e la sua durata è pari alla lunghezza del segnale da riprodurre. Ad ogni azione di controllo è associato un tempo logico. Quando il tempo di esecuzione raggiunge tale istante temporale, l'azione di controllo viene eseguita, producendo di conseguenza il cambiamento dell'output. Per una corretta computazione, i calcoli di controllo e audio, vengono eseguiti a turno, in un ordine temporale crescente. In un sistema real-time (come pd), il tempo logico, che corrisponde sempre al prossimo campione affetto dalla modifica audio, è leggermente in ritardo rispetto al tempo reale, che è misurato dal campione che attualmente è in uscita dal computer. Anche in questo caso le computazioni audio e controllo sono eseguite alternativamente, secondo l'ordine logico temporale. La ragione per la quale si usa un tempo logico e non un tempo reale è di poter mantenere separati i calcoli eseguiti del computer. Essi, infatti, possono variare per molte ragioni, anche per istruzioni che apparentemente sembrano uguali. Quando si determina l'uscita di un segnale audio, o si sta valutando qualche ingresso di controllo, è necessario che il tempo logico sia lo stesso per tutta la durata delle istruzioni, come se queste fossero eseguite istantaneamente. In questo modo, le computazioni di musica elettronica, se eseguite correttamente, saranno deterministiche: l'esecuzione delle stesse operazioni audio in un sistema real-time e non, produrranno (si spera) lo stesso risultato.

La maggior parte dei software di *computer music* computa l'audio in blocchi. Questo aumenta l'efficienza del programma stesso. Ogni oggetto audio, incorre in un aumento del carico di computazione ogni volta che è chiamato, pari più o meno a venti volte il costo medio per calcolare un unico campione. Se il blocco è composto da 64 campioni (come è per default in pd), l'overhead introdotto è del 30%; se non si utilizzasse questa tecnica (blocco composto da un solo campione) l'overhead sarebbe del 2000%.

Lo scopo principale del livello di controllo è gestire i blocchi audio con valori numerici. In Pure Data sono definite le operazioni matematiche elementari più qualche altra

funzione. Un aspetto da tenere in considerazione è l'ordine con il quale le operazioni vengono svolte. Quando un oggetto di controllo ha più input, è obbligatorio (per ottenere un corretto risultato) inserire dati negli inlet partendo da quello più a destra fino ad arrivare a quello a sinistra. In altre parole un oggetto cambierà il suo output solo quando riceverà un dato dal suo inlet più a sinistra (*hot inlet*). Per poter rispettare questo ordinamento si utilizza l'oggetto *trigger*. Quest'oggetto può ricevere in input qualsiasi variabile di pure data. Esso invia o trasforma in bang l'input verso le sue uscite da destra a sinistra. L'uscita di un oggetto *trigger* è determinata dai suoi argomenti (bang, float, symbol, pointer, list).

4.5 Livello Audio

Come descritto nel capitolo 1 il suono è vibrazione. Nella musica elettronica, vengono utilizzati altoparlanti per generare il suono. Questi strumenti sono dotati di una o più membrane che vibrando, producono il suono. Le vibrazioni di queste membrane sono controllate dal computer. In Pd, l'oggetto `dac~` (*digital audio converter*) gestisce questo procedimento. Il suo compito fondamentale è di trasformare numeri (32 bit) in variazioni di corrente elettrica che, una volta amplificati, producono la vibrazione della membrana dell' altoparlante. La posizione più convessa che la membrana può assumere è associata con il numero 1. La posizione più concava è ovviamente associata con il numero -1. Tutti i possibili valori che la membrana può assumere sono compresi tra 1 e -1. Il rate con il quale vengono inviate informazioni audio, in pd, è per default di 44100Hz.
É necessario comprendere come le informazioni fluiscono dal livello di controllo a quello audio (*flusso di controllo*). Questo flusso di dati è un insieme di numeri (anche nulli) determinati in base alle computazioni di controllo. Essi possono coinvolgere sezioni temporali posizionate ad intervalli, regolari od irregolari. Il più semplice esempio di *flusso di controllo* è composto da una serie di informazioni indicanti intervalli temporali $(..., t(0), t(1), ...)$ di campioni audio. Come unica condizione si vuole che siano disposti in ordine non decrescente. Un *flusso di controllo numerico*, invece, è una coppia di valori che associa a qualche attimo temporale il valore che la funzione di uscita assume in quell'istante : $..., (t(0), x(0)), (t(1), x(1)), ...$
Si può considerare il *flusso di controllo* come un controllore MIDI, che cambia ad intervalli irregolari. In questo caso si indica solo l'istante temporale in cui accade un evento. Il *flusso di controllo numerico* può essere visto come un segnale audio, in cui per ogni istante temporale è indicato che valore deve assumere la funzione. Ma a differenza dei segnali audio in cui il rate è fisso e non varia, il flusso di controllo non ha una frequenza precisa. Quando bisogna convertire un flusso di controllo numerico in un segnale audio si possono utilizzare tre tecniche.

Nella prima si cerca di effettuare la conversione più velocemente possibile. Ogni campione del segnale di output è pari al più recente valore del segnale di controllo. Se per esempio il blocco è lungo quattro ed il flusso di controllo è una onda quadra di periodo

$5, 5$:

$$..., (2, 1), (4.75, 0), (7.5, 1), (10.25, 0), (13, 1), ...$$

i campioni da 0 a 3, poiché a causa della lunghezza del blocco saranno computati al tempo logico 4, avranno un valore di uscita pari a 1 determinato dalla prima coppia. I successivi quattro campioni saranno sempre pari a 1 a causa delle due coppie $(4.75, 0)$ e $(7.5, 1)$. Il più recente ha comunque valore uno. Questo tipo di conversione è appropriato per flussi di controllo che non cambiano molto frequentemente rispetto alla lunghezza del blocco. Il vantaggio di quest'approccio risiede nella sua semplicità di computazione e alla massima velocità di risposta ai cambiamenti. Quando gli aggiornamenti del flusso di controllo sono troppo veloci (sempre rispetto alla lunghezza del blocco) il segnale audio di uscita può essere affetto da disturbi: se la frequenza di Nyquist del segnale di uscita è minore rispetto alla frequenza del segnale in ingresso, l'uscita subisce un effetto di alias ad una nuova frequenza minore rispetto alla frequenza di Nyquist.

Nella seconda conversione ogni nuovo valore del flusso di controllo al tempo t incide sui campioni di uscita successivi al tempo t. Questo metodo è equivalente ad usare la conversione precedente (più veloce possibile) con una lunghezza di blocco pari a 1. Questa soluzione è la migliore nel caso in cui il flusso di controllo cambia rapidamente.

Nella terza conversione si utilizza un interpolazione di due punti per ottenere una maggiore accuratezza. Supponiamo che l'ultimo valore del flusso di controllo sia pari a x, e il punto successivo sia $(n + f, y)$, con n intero e f è una frazione di un'unità temporale $(0 \leqslant f < 1)$. Il primo punto di uscita a risentire del cambiamento sarà il campione con indice n. Invece di impostare il valore d'uscita a y (come nel caso precedente), il suo valore sarà:

$$fx + (1 - f)y$$

In altre parole, y è pari alla media pesata del valore precedente e di quello attuale, il cui peso è maggiore se il suo tempo associato è antecedente e più vicino ad n. Nell'esempio precedente la transizione tra 0 e 1 al tempo 2 produrrà un valore di uscita pari a $0x+1y = 1$ mentre la transizione da 1 a 0 al tempo $4, 75$ produrrà $0, 75x + 0, 25y = 0, 75$ Questa tecnica fornisce una migliore rappresentazione del flusso di controllo, a discapito di un maggior carico computazionale ed un piccolo ritardo. I flussi di controllo numerici possono essere convertiti in segnali audio utilizzando funzioni rampa per addolcire le discontinuità. Quest'approccio è usato solitamente quando un flusso di controllo gestisce l'ampiezza di un segnale. In generale sono necessari tre valori per definire una funzione rampa: tempo di inizio, un valore di arrivo e il tempo di arrivo, di solito espresso come ritardo rispetto al tempo di inizio. In queste situazioni si rimane abbastanza precisi quando si modifica il tempo iniziale e finale in modo da uguagliare il primo campione audio computato al successivo tempo logico. Nelle situazioni reali, in cui la lunghezza del blocco potrebbe essere nell'ordine di millisecondi, questa tecnica è troppo raffinata per il controllo d'ampiezza, in quanto se raggiungessero il valore di arrivo in una frazione di millisecondi prima o dopo, non si potrebbero sentire differenze. Questa tecnica rimane valida quando ci sono molte e veloci ripetizioni del flusso di controllo. Se si effettuano operazioni ripetute

ogni pochi millisecondi, la differenza nel segmento audio attuale produrrà comunque una aperiodicità udibile.

In alcuni casi è utile convertire un segnale audio in un flusso di controllo. Per input si avrà un segnale audio e una collezione di tempi logici. In output si deve produrre un flusso di controllo che combini la sequenza temporale logica con i valori recuperati dal segnale audio. Quest'applicazione può essere utile per esempio se volessimo controllare l'ampiezza di un segnale utilizzando un oggetto line∼. Supponiamo di voler eliminare il suono con un rate fissato piuttosto che in un certo tempo. Per esempio, si ha necessità di riutilizzare una rete con un altro segnale audio riducendo al minimo il tempo di transizione senza artefatti acustici. Una possibile soluzione prevede l'utilizzo di una rampa che smorzi il segnale in minor tempo. Bisogna far in modo che l'oggetto line venga portato a zero in un tempo calcolato in funzione dell'attuale valore di output. Per eseguire questo è necessario campionare l'uscita dell'oggetto line (un segnale audio) in un flusso di controllo. Gli stessi problemi che si devono risolvere nella conversione flusso di controllo in segnale audio, possono essere ritrovate anche in questa trasformazione; si tratta di trovare un buon trade off tra accuratezza e velocità. Vogliamo calcolare un segnale audio (sapendo che esso è composto da blocchi di 4 campioni ciascuno) al tempo logico 6 ed utilizzarlo per cambiare il valore di un altro segnale. Se il valore del campione più recente del segnale è all'indice 3, il primo campione modificabile è il 4. É possibile quindi ritardare l'intera operazione di un solo campione. Rimane comunque impossibile scegliere esattamente quale campione sarà modificato, si ha la certezza che entro 4 campioni la modifica potrà compiersi. Per migliorare l'accuratezza temporale si può ridurre l'immediatezza dell'esecuzione. In generale se il blocco è composto da B campioni, e dato un qualche indice n, è possibile leggere il campione $n - B$ e modificare il campione all'indice n. Si crea un ritardo di B campioni per passare dal livello audio a quello di controllo per tornare nuovamente al livello audio. Questo è il prezzo che bisogna pagare per essere in grado di modificare esattamente il campione ad uno specifico indice. É possibile andare oltre, specificare per esempio una frazione di campione, utilizzando una interpolazione; in questo modo, però, aumenta il ritardo complessivo. Nella maggior parte dei casi quando si tratta con la conversione audiocontrollo la soluzione più semplice è solitamente la migliore.

4.6 Continuità e discontinuità nei segnali di controllo

Gli algoritmi di sintesi variano notevolmente nel modo in cui gestiscono le discontinuità nei segnali del livello di controllo. Se i controlli variano in maniera continua un approccio a generazione di inviluppo ADSR funziona più che bene. Nel caso in cui si presentasse una discontinuità nella massima ampiezza di una sequenza di note, l'ADSR creerebbe una rampa dal vecchio valore a quello nuovo evitando le discontinuità. Ci sono casi in cui il generatore di inviluppo cambia il timbro di un suono: quindi una modifica dell'attacco (per esempio con l'introduzione della rampa) potrebbe compromettere il risultato finale del suono, ed in alcune situazioni questo è inaccettabile.

Ci sono alcuni parametri di controllo che non possono in alcun modo essere cambiati in modo continuo, per esempio, il passaggio tra due sezioni di due wavetable.
Quando non è possibile utilizzare la tecnica di inviluppo ADSR, per eliminare le discontinuità si possono impiegare due diversi approcci: *muting* e *ramp and switch*.

4.6.1 Muting

Questa tecnica consiste nell'applicare un inviluppo all'ampiezza del segnale di output. In particolare l'ampiezza del segnale deve essere velocemente portata a zero, con una rampa, un istante prima del cambiamento del parametro che comporta la discontinuità e riportare l'ampiezza al livello originario subito dopo (se necessario). Il tempo permesso per il *muting* deve essere breve (in modo da disturbare il meno possibile il suono precedente) ma non troppo per causare artefatti acustici in output. In generale un tempo di 5ms è accettabile. Nel caso in cui sia necessario effettuare un muting in un valore di ampiezza diverso da zero è necessario riportare al valore originario l'ampiezza con una rampa ascendente. Il tempo impiegato per portare a zero l'ampiezza non deve essere pari al tempo utilizzato per portarla al livello originario. Questi parametri devono essere impostati di volta in volta ascoltando il segnale di uscita.
In generale questa tecnica risulta difficilmente applicabile in situazioni real-time, cioè quando non conosciamo a priori il momento in cui ci sarà il cambiamento del parametro che comporta la discontinuità.

4.6.2 Switch and ramp

Attraverso questa tecnica si cerca di rimuovere le discontinuità sintetizzando una discontinuità opposta che si oppone e cancella quella originaria.
Il vantaggio di questo approccio, rispetto al muting, risiede nel fatto che non è necessario sapere in anticipo il momento del cambiamento del parametro. Inoltre, ogni artificio acustico prodotto da questa tecnica sarà nascosto dal nuovo suono prodotto. La figura 4.2 mostra come si può realizzare questa tecnica. Il riquadro indicato con i puntini può contenere un qualunque algoritmo di sintesi, per il quale vogliamo eliminare le discontinuità. Quando si modifica il valore di controllo (rappresentato dal generatore di inviluppo ADSR più in alto), si azzera e si innesca l'altro generatore di inviluppo ADSR per cancellare la discontinuità. Alla discontinuità viene sottratto l'ultimo valore sintetizzato proprio prima di azzerarlo. Per far ciò si misura il livello del generatore ADSR verso cui bisogna arrivare. Questo valore è pari al livello corrente meno la discontinuità. Questi due segnali sono sommati ed il valore attuale è salvato attraverso l'oggetto `snapshot`. Il generatore di inviluppo che cancella la discontinuità è azzerato al suo nuovo valore e successivamente portato a zero con una rampa. L'oggetto sommatore in basso a sinistra addiziona il segnale sintetizzato con il segnale di cancellazione della discontinuità.

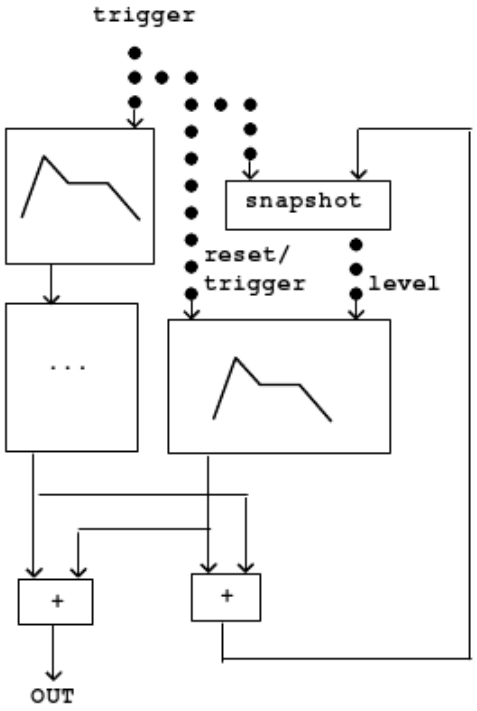


Figura 4.2: Blocco per il switch and ramp

4.7 Flext

Come è ben noto tutti gli elementi di un sistema real-time devono essere quanto più veloci possibili per mantenere una latenza relativamente bassa. Questo è il motivo principale per cui gli external, in generale, sono realizzati in linguaggi binari pre compilati che sono caricati all'interno del sistema. Il linguaggio che più viene utilizzato per questo tipo di applicazioni è il C/C++. Tutti i sistemi real-time, estensibili tramite moduli (Max/MSP, Pd ...), forniscono un insieme di interfacce (API) per la comunicazione tra moduli e sistema. Per esempio un modulo per definire la sua struttura (quanti inlet e quanti outlet avrà) dovrà chiamare la rispettiva funzione del API. D'altro canto il sistema informerà il modulo che un messaggio gli è stato inviato usando le funzioni di *callback*. Queste funzioni sono dei veri e propri punti di accesso all'external, in modo che essa possa inviare e ricevere informazioni al sistema real time. Un modulo dichiara l'esistenza di tali funzioni di callback al momento della sua creazione. Ogni messaggio che può essere ricevuto da un modulo ha bisogna di una specifica funzione di callback.
Il modo di realizzare queste API è diverso per ogni sistema: questo rende le external poco portabili. Un sistema, quindi, non è in grado di usare moduli realizzati per altri programmi, sebbene le funzionalità dell'external siano previste per entrambi i sistemi. Come se ciò non bastasse, nemmeno il codice sorgente C/C++ scritto per il modulo esterno è lo stesso per sistemi differenti.

Nasce, quindi, la necessita di trovare un modo per uniformare la scrittura del codice: flext [13]. Esso fornisce un insieme di API indipendenti dal sistema in cui l'external sarà utilizzata. É importante realizzare ciò senza modificare il codice di origine. L'external nel momento della compilazione verrà tradotta nella versione utilizzata dal sistema. Flext esegue tutto il lavoro di traduzione tra il codice del programma e l'interfaccia del sistema real-time in base alle callback che bisogna invocare. Le external basate su flext sono scritte in C++, e fanno uso dell'ereditarietà. Gli oggetti derivati da una stessa classe madre possono ereditare tutte le sue caratteristiche ed eventualmente modificarle in base alle nuove esigenze. Oltre a ciò, esistono numerose funzioni previste da flext che facilitano enormemente il compito di sviluppare un external. Le attività principali delle external prevedono la gestione di segnali digitali. In questo caso è necessario ereditare dalla classe `flext_base` e modificare a seconda delle necessità il metodo `m_signal`. Tutto il codice necessario per la costruzione di external in questa tesi è stato scritto in flext. É possibile quindi utilizzare questi external anche in ambienti diversi da Pure Data. Le external create sono state compilate sia in windows, sia in ubuntu 9.04. Flext è rilasciato sotto licenza GPL ed è stato scritto da Thomas Grill.

Capitolo 5

Realizzazione

In questo capitolo si analizzano i processi implementativi che hanno portato alla creazione del modello per le *Pinna related transfer function* (paragrafo 5.1). Successivamente si osservano i risultati che si sono ottenuti con quattro soggetti presenti nel database CIPIC [1], paragrafo 5.2.

5.1 Calcolo del segnale di uscita

Le sezioni che è necessario implementare, per il calcolo del segnale di uscita, sono quelle relative al blocco riflettente e risonante, come illustrato in figura 2.5. Si è deciso di realizzare una unica *external* per la rappresentazione del filtro multiNotch, mentre si è costruito un unico blocco per ogni filtro peak. I dati relativi alle risonanze e a i notch sono contenuti in due file di testo.

Blocco multiNtc∼

Consideriamo il blocco `multiNtc∼`. Si è creata una classe di supporto `Notch`, attraverso la quale è possibile definire un punto di notch attraverso le tre caratteristiche principali, ottenute dall'algoritmo descritto nel capitolo 2: frequenza centrale f_C, la profondità del notch D e l'ampiezza di banda f_B. Una delle operazioni da svolgere nel costruttore della classe `multiNtc∼`, è di ottenere i dati relativi ai notch posizionati nei punti di elevazione fissati: da -45 a $+90$ con passo $5,625$. Al massimo per ogni angolo di elevazione si considerano tre notch distinti. In totale, quindi possono esserci $3 \cdot 25 = 75$ notch. La variabile che include tutti questi dati è un array di 75 elementi della classe Notch. Essa contiene tutte le informazioni necessarie per realizzare i filtri notch. Il file dal quale è possibile leggere questi dati è un file di testo del tipo `notch_165.txt`, l'ultimo numero è l'identificatore del soggetto nel database CIPIC. Ogni riga di questo file di testo contiene i dati relativi al notch secondo questo ordine: f_C D f_B. Il blocco `multiNtc∼` ha tre ingressi: uno per il segnale audio, due numeri, il centrale indica l'angolo di elevazione prescelto, da -45 a 90, mentre quello più a destra l'identificatore del soggetto del database. Fondamentalmente questo numero sarà utilizzato per l'apertura dei file contenenti

le informazioni dei filtri da costruire. Al momento del passaggio al blocco `multiNtc` di un nuovo valore di elevazione, pure data chiama la funzione di callback associata al metodo `setElevation`. A seconda del nuovo valore di elevazione vengono modificati i coefficienti dei filtri, e di conseguenza la risposta in frequenza globale. Attraverso le formule presenti nella sezione 2.5 è possibile determinare univocamente i coefficienti del numeratore e denominatore delle risposte in frequenza del `multiNtc`$\sim$. Una volta calcolati tutti i coefficienti dei tre filtri notch è sufficiente effettuare una convoluzione tra di essi. Date due sequenze $x[n]$ di lunghezza N e $h[n]$ di lunghezza M la loro convoluzione è una sequenza di lunghezza $N + M - 1$ così calcolata:

$$y[n] = \sum_{k=-\infty}^{\infty} x[n-k]h[k] = x[n] * h[n] \tag{5.1}$$

ponendo a zero i punti in cui la sequenza non è definita. Applicando 5.1 ai coefficienti dei filtri notch a coppie otteniamo i coefficienti del filtro finale.

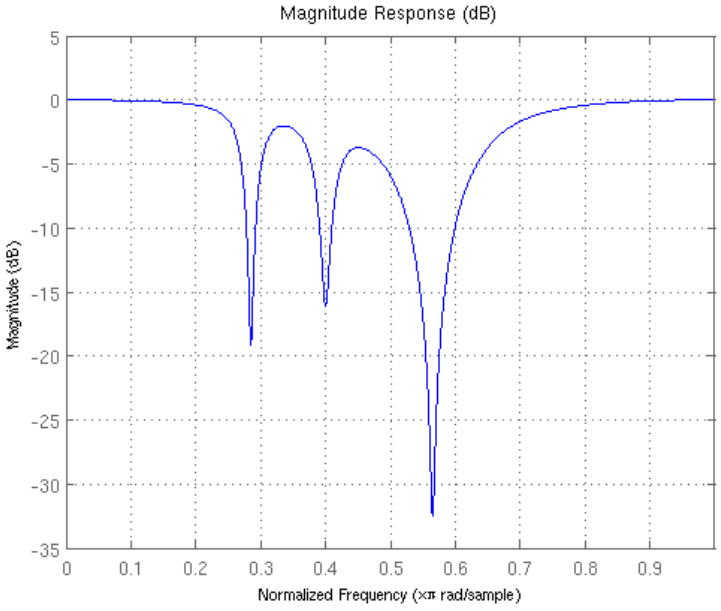


Figura 5.1: Risposta in frequenza del filtro multiNtch con elevazione -45.

I coefficienti del filtro rappresentati in figura 5.1 sono:
num = $[0.672666, -0.973214, 1.983493, -1.703077, 1.947571, -0.934594, 0.631697]$
den = $[1.000000, -1.437278, 2.363997, -1.644029, 1.496166, -0.529579, 0.375264]$.
Essi sono in accordo con i valori presenti nel file `notch_165.txt` per il soggetto 165 (KEMAR).

f_C [Hz]	D	f_B [Hz]
6284.25	-18.5	110.25
8820	-14.5	220.5
12458.25	-32.4	110.25

Tabella 5.1: Frequenze centrali, profondità e ampiezza dei notch del soggetto 165 per elevazione -45

Interpolazione Lineare

Non avendo a disposizione tutti i possibili valori di notch per ogni angolo da -45 a 90 è necessario interpolare i valori sconosciuti di f_C, D e f_B. In questo modo è possibile ridurre le discontinuità nelle funzioni di trasferimento dei filtri, causa dei fastidiosi artifici acustici, come click e fruscii. L'interpolazione qui usata è quella lineare [20], in cui campioni adiacenti sono connessi tra di loro attraverso una linea dritta. In generale dati due punti nel piano cartesiano (x_a, y_a) e (x_b, y_b), la retta che li unisce è del tipo:

$$\frac{y - y_a}{x - x_a} = \frac{y_b - y_a}{x_b - x_a} \tag{5.2}$$

dato un punto $x_a < x < x_b$ e risolvendo questa equazione per y otteniamo il valore della funzione in x:

$$y = y_a + (x - x_a)\frac{y_b - y_a}{x_b - x_a} = \frac{(x - x_a)y_b + (x_b - x)y_a}{x_b - x_a} \tag{5.3}$$

.

L'interpolazione lineare su un insieme di punti $(x_0, y_0), (x_1, y_1), ...$ è definita come la concatenazione delle interpolazioni lineari tra ogni coppia di punti dell'insieme. Il risultato è una spezzata, con derivate discontinue proprio nei punti dell'insieme. La funzione appartiene quindi alla classe di differenziabilità C^0. L'errore che si commette utilizzando l'interpolazione lineare è definito come:

$$R_T = f(x) - p(x) \tag{5.4}$$

con $p(x)$ il polinomio dell'interpolazione lineare, cioè:

$$p(x) = f(x_0) + \frac{f(x_1) - f(x_0)}{x_1 - x_0}(x - x_0) \tag{5.5}$$

Può essere dimostrato, utilizzando il teorema di Rolle, che se $f(x)$ ha una derivata seconda continua, l'errore è limitato da:

$$|R_T| \leq \frac{(x_1 - x_0)^2}{8} max_{x_0 \leq x \leq x_1} |f''(x)| \tag{5.6}$$

L'approssimazione lineare tra due punti peggiora con la derivata seconda. Intuitivamente questo è corretto: più la funzione è "curva", più il modulo della derivate seconde è maggiore, più l'errore commesso con l'interpolazione lineare cresce.

Segnale di uscita dei filtri

Nel momento in cui i coefficienti del numeratore e denominatore sono stati determinati, bisogna calcolare l'uscita del filtro IIR in funzione dell'ingresso presente in quell'istante [15]. Dalla seguente funzione di trasferimento generale

$$H(z) = \frac{P(z)}{D(z)} = \frac{p_0 + p_1 z^{-1} + ... + p_M z^{-M}}{d_0 + d_1 z^{-1} + ... + d_N z^{-N}} \tag{5.7}$$

si può notare che la computazione dell $n-$esimo campione di output richiede la conoscenza di svariati campioni passati della sequenza di output; la realizzazione di un filtro IIR causale richiede una qualche forma di feedback. In particolare un filtro di ordine N è caratterizzato da $2N + 1$ coefficienti e richiede, in generale, $2N + 1$ moltiplicatori e $2N$ sommatori. Consideriamo un filtro IIR del secondo ordine, come un unico filtro notch utilizzato per la costruzione del multiNotch. Esso è del tipo:

$$H(z) = \frac{P(z)}{D(z)} = \frac{p_0 + p_1 z^{-1} + p_2 z^{-2}}{1 + d_1 z^{-1} + d_2 z^{-2}} \tag{5.8}$$

Per passare da 5.7 a 5.8 è sufficiente normalizzare i coefficienti. Realizzando una cascata di due filtri con:

$$H_1(z) = \frac{W(z)}{X(z)} = p_0 + p_1 z^{-1} + p_2 z^{-2} \tag{5.9}$$

e

$$H_2(z) = \frac{Y(z)}{W(z)} = \frac{1}{D(z)} = \frac{1}{p_0 + p_1 z^{-1} + p_2 z^{-2}} \tag{5.10}$$

Nel dominio del tempo:

$$w[n] = p_0 x[n] + p_1 x[n-1] + p_2 x[n-2] \tag{5.11}$$

e

$$y[n] = w[n] - d_1 y[n-1] - d_2 y[n-2] \tag{5.12}$$

Quindi con cinque moltiplicazioni, quattro addizioni, due valori precedenti dell'input e dell'output è possibile determinare l'n-esimo campione di output dell'uscita. Nell'implementazione in flext l'uscita dei filtri è stata realizzata nella Prima forma diretta (*Direct Form I*).

La prima forma diretta ha queste proprietà:

- Può essere vista come un filtro a due zeri seguita in serie da una sezione a due poli.
- Nelle maggior parte dei calcoli in virgola fissa (come il complemento a due) non c'è possibilità che si verifichi un overflow interno del filtro. Poiché nel filtro, è presente solamente un punto di somma, e poiché il numero in virgola fissa successivo al massimo numero rappresentabile è quello più piccolo (esiste quindi una sorta di naturale cerchio rappresentativo), fintantoché il risultato finale $y(n)$ è nell'intervallo di valori previsto, l'overflow è evitato, anche nel caso in cui i risultati intermedi della somma siano affetti da overflow.

- Il numero dei delay necessari è doppio. Quindi la *Direct Form I* non è in forma canonica rispetto ai delay. In generale è sempre possibile implementare un filtro di ordine N utilizzando solamente N delay.
- Come per tutte le strutture in forma diretta dei filtri (quelle in cui i coefficienti sono dati dai coefficienti della funzione di trasferimento), i poli e gli zeri possono essere molto sensibili agli errori di arrotondamento dei coefficienti del filtro. Questo non è un problema per una sezione del secondo ordine, ma può diventare una complicazione con un ordine del filtro superiore. Questa è la medesima sensibilità numerica che affligge le radici dei polinomi in relazione con l'arrotondamento dei coefficienti polinomiali. Questa sensibilità tende ad aumentare quando le radici sono molto vicine tra di loro, a differenza di quando esse sono sparse. Per minimizzare questa sensibilità delle volte si preferisce fattorizzare la funzione di trasferimento in serie e paralleli di sezioni del secondo ordine.

Poiché i filtri considerati in questo modello, sono causali, i valori precedenti all'istante temporale iniziale sono posti a zero. Senza questo accorgimento i filtri, potrebbero non essere stabili, causando risultati imprevisti. Il calcolo dell'output del filtro deve essere effettuato ad ogni ciclo dsp e deve quindi essere inserito nel metodo:
`m_signal(int n, float *const *in, float *const *out)`

- `n` : Indica il numero di campioni per blocco, per default 64
- `*ins` : É il vettore di lunghezza n contenente i campioni del segnale di input
- `*outs` : É il vettore che alla fine del metodo dovrà contenere i campioni del segnale di uscita.

L'equazione 5.12 è stata implementata con questo codice: gli array `num` e `den` contengono i coefficienti del numeratore e del denominatore, rispettivamente. `x` e `y` contengono i valori precedenti del segnale di input e di output.

```
 while (n--){
    *outs= (num[0]) * (*ins);
    x[0]=*ins++;
    for(i=1;i<ord;i++){
        *outs+=num[i]*x[i];
        *outs-=den[i]*y[i];
    }
    for(i=6;i>=1;i--){
        x[i]=x[i-1];
        y[i]=y[i-1];
    }
    y[1]=*outs++;
}
```

La costruzione del filtro peak è simile a quella del multiNtch. Gli input sono costituiti dal segnale audio, da f_C, da f_B (costante a 5000Hz) e da G . Le operazioni di recupero dei dati dal file `resonance_xxx.txt` e di interpolazione lineare, sono state implementate direttamente in Pure Data.

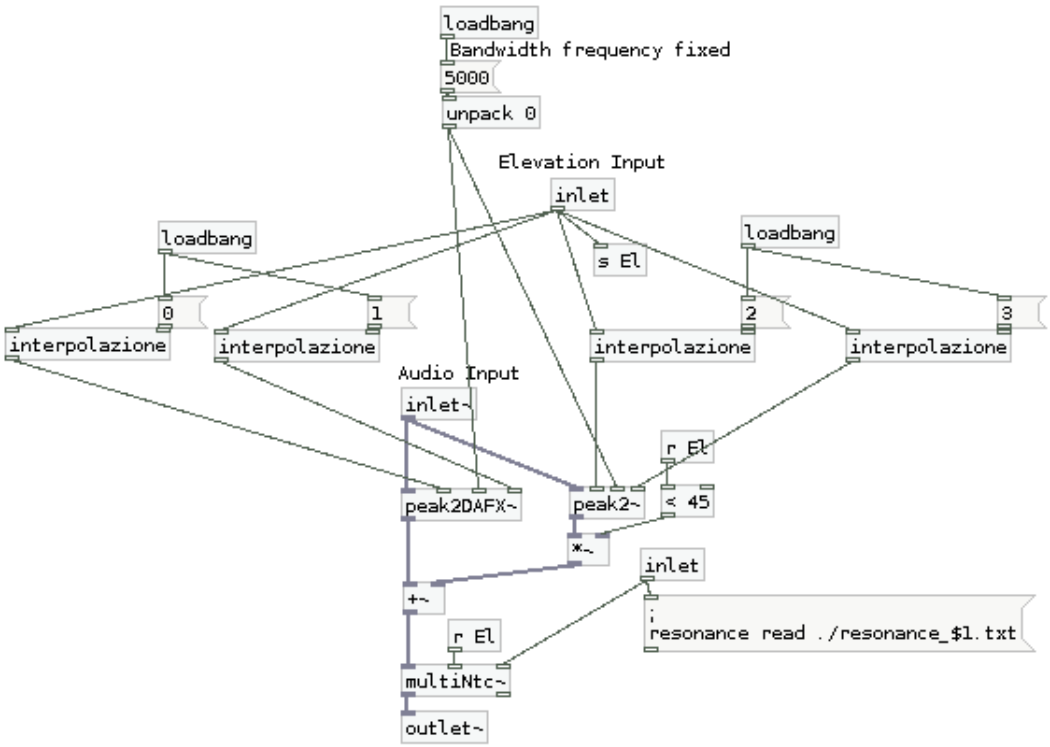
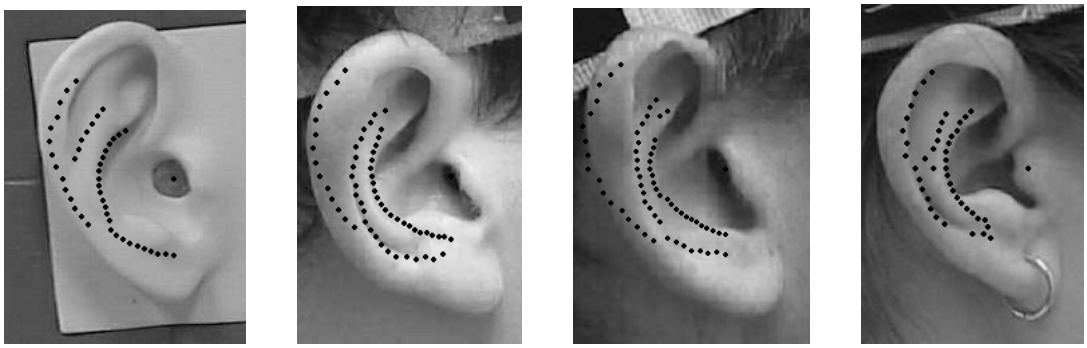


Figura 5.2: Realizzazione in Pure Data del modello Riflettente-Risonante

In figura 5.2 è mostrata la realizzazione del modello del capitolo 2 in Pure Data. Il tracciato più spesso indica il percorso che il segnale audio deve compiere; passa attraverso i due peak in parallelo. Le uscite di questi filtri sono sommate e il risultato è immesso all'ingresso del filtro multiNtc. I parametri dei filtri peak sono calcolati nel blocco interpolazione. Esso riceve il valore di elevazione attuale, determina i punti di altezza conosciuti, nel file `resonance_xxx.txt`, in cui è compreso (determina cioè (x_a, y_a) e (x_b, y_b) della 5.2) ed applica l'interpolazione lineare.

5.2 Risultati

Le prove verificano l'attendibilità dell'implementazione su quattro soggetti del Database CIPIC: 010, 027, 134, 165 (KEMAR). Dopo aver determinato attraverso l'algoritmo presentato nel capitolo 2 i notch e le risonanze, si creano i file `notch_xxx.txt` e



(a) soggetto 165 (b) soggetto 134 (c) soggetto 027 (d) soggetto 010

Figura 5.3: Pinne dei soggetti analizzati nel Database CICIP

`resonance_xxx.txt`. Gli ultimi 3 caratteri rappresentano il soggetto presente nel database CIPIC. Successivamente si confronta la PRTF reale, con la risposta all'impulso ottenuta da Pure Data. Questa risposta all'impulso è determinata immettendo nel modello un semplice impulso, con un blocco *dirac*∼. L'uscita del modello è salvata su un file di testo, che viene processato per determinare la risposta in frequenza. I grafici seguenti mostrano la sovrapposizione tra la PRTF misurata, presente nel CIPIC (linea continua) e la PRTF determinata dalla risposta all'impulso registrata con pure data. Le PRTF misurate sono state ottenute finestrando opportunamente le HRTF ottenute dal database CIPIC. É per questo motivo che nelle frequenze limite, intorno ai 0 Hz e 20 kHz, tali funzioni tendono a meno infinito in dB. In generale le PRTF reali per queste funzioni non devono modificare in alcun modo il segnale in ingresso. Il loro guadagno deve essere 0 dB. La pinna è infatti insensibile alle basse frequenze. A differenza delle PRTF finestrate le PRTF sintetiche devono rispettare questo vincolo. Per gli angoli di elevazione superiori a 45 gradi si è deciso di eliminare la seconda risonanza. La motivazione a questa scelta è dettata da una euristica. Da un punto di vista analitico le PRTF sintetizzate in questa maniera hanno un andamento più naturale. Sono comunque necessari degli ulteriori esperimenti percettivi che confermino o rafforzino questa intuizione.

5.2.1 Soggetto 165

Il soggetto 165 del database CICIP è il manichino KEMAR con pinne piccole. Questo è il soggetto che in prima analisi dovrebbe non creare grossi problemi: nel capitolo 1 si è visto che la sua HRTF non è caratterizzata da molti picchi e valli. La testa è completamente sferica e simmetrica e le pinne non presentano molte caratteristiche antropometriche tipiche degli esseri umani. I notch e le risonanze determinate con l'algoritmo presentato nel capitolo 2 sono graficamente visualizzate nelle figure 5.4 e 5.5.

La risonanza segnalata oltre i 16 kHz non viene considerata e nella fase di post processing viene eliminata. Si analizzano, ora, alcuni angoli di elevazione (-45, $-16,875$, 0, 90) più in dettaglio, con l'ausilio delle risposte all'impulso ottenute in pure data.

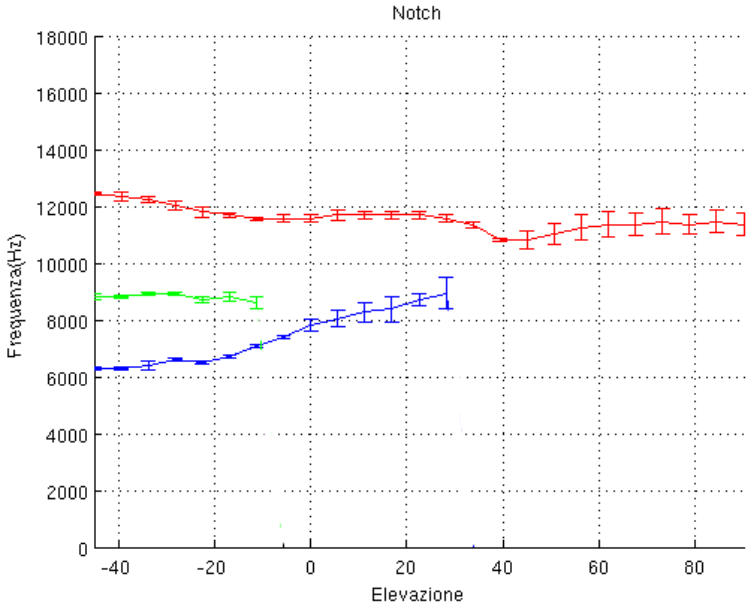


Figura 5.4: Notch (sogg. 165) determinati dall'algoritmo presentato nel capitolo 2

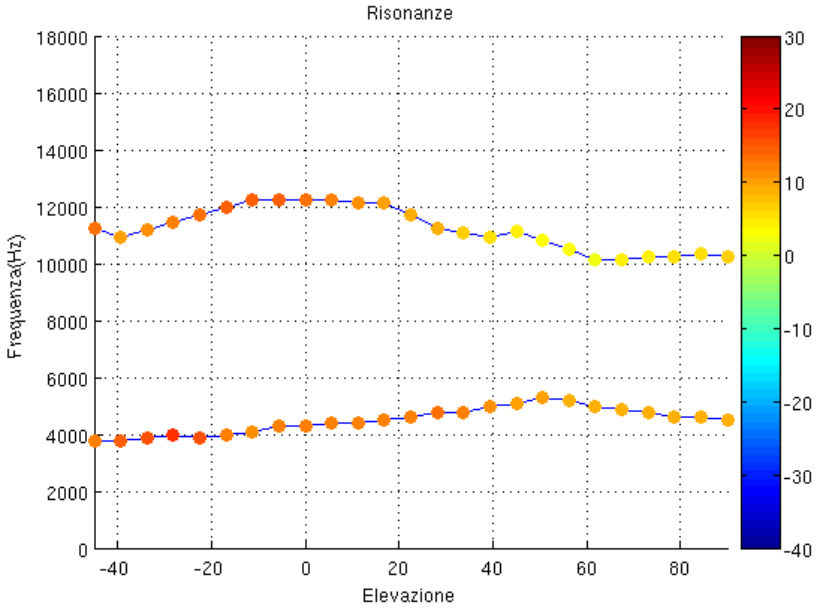


Figura 5.5: Risonanze (sogg. 165) determinati dall'algoritmo presentato nel capitolo 2

Elevazione -45

Si faccia riferimento alla figura 5.6. L'intervallo di frequenze nel quale si ricerca la massima corrispondenza possibile è tra i 4 e 12 kHz. Al di fuori di questo intervallo, la pinna non influenza la HRTF e quindi non è richiesta una buona approssimazione. La corrispondenza tra le due funzioni di trasferimento è soddisfacente, anche se si possono notare alcune imperfezioni. I notch molto profondi a queste basse elevazioni complicano

la procedura di creazione del filtro notch. Infatti se il notch da approssimare è particolarmente profondo e stretto, il filtro del secondo ordine produrrà un notch meno aguzzo e più ampio la cui larghezza di banda può interferire con i notch vicini. In questo modo si sottostima il modulo della risposta in frequenza tra i due notch. Questo fenomeno si può osservare tra gli 8 e 12 kHz. Anche il modello per le risonanze può provocare degli errori di approssimazione. La limitazione imposta sulla larghezza di banda, fissata a 5kHz, può produrre errati segnali di uscita, soprattutto a basse frequenze. Più la frequenza centrale della risonanza si avvicina allo zero più alcune frequenze potrebbero subire un boost che le HRTF non prevedono.

Per di più, l'estrazione delle frequenze centrali con il metodo di identificazione ARMA non sempre è consistente: le frequenze reali centrali non coincidono con quelli derivati.

Elevazione -16,875

In figura 5.7 si nota il fenomeno dell'interazione tra i due notch alla frequenza centrale tra gli 8÷10 kHz. Le risonanze sono state ricostruite correttamente. Il primo notch è troppo poco profondo. Questo sembra essere causato da un'incorretta analisi delle PRTF. La prima risonanza ha un guadagno di 11 dB; il primo notch ha una depressione di -16 dB. Ad una prima approssimativa analisi il notch dovrebbe posizionarsi intorno ai -10 dB, come effettivamente è rappresentato in figura. La PRTF reale invece ha un notch posto a circa -16 dB nettamente più profondo di quanto l'algoritmo di analisi ha determinato.

Elevazione 0

Il risultato per questo angolo di elevazione è abbastanza buono. Dai dati di analisi il primo notch ha una profondità di circa 13 dB, posizionato a circa 8 kHz. In tale frequenza il contributo apportato dalla somma delle risonanze è pari a 10 dB. La PRTF sintetizzata ha un notch con profondità -3 dB, consistente con quanto dichiara l'analisi. La PRTF reale però ha un notch di qualche dB inferiore ai -5 dB. Questo errore si ripercuote successivamente, aumentando di qualche dB l'inviluppo seguente. La seconda risonanza ha un guadagno un po' troppo elevato. Si faccia riferimento alla figura 5.8.

Elevazione 90

Globalmente la PRTF artificiale approssima bene quella reale (figura 5.9). La prima risonanza è ben stimata e la seconda poichè l'angolo di elevazione è superiore ai 45 gradi è stata eliminata. Questo piccolo dettaglio ha permesso di migliorare notevolmente la PRTF sintetica. Nel caso non si utilizzasse questo artificio le due risonanze entrerebbero in conflitto sommandosi tra di loro e causando un boost di qualche dB per le alte frequenze. Eliminando la seconda risonanza il problema scompare. Come ultimo appunto si può notare che la frequenza centrale del notch sempra essere spostata troppo a sinistra.

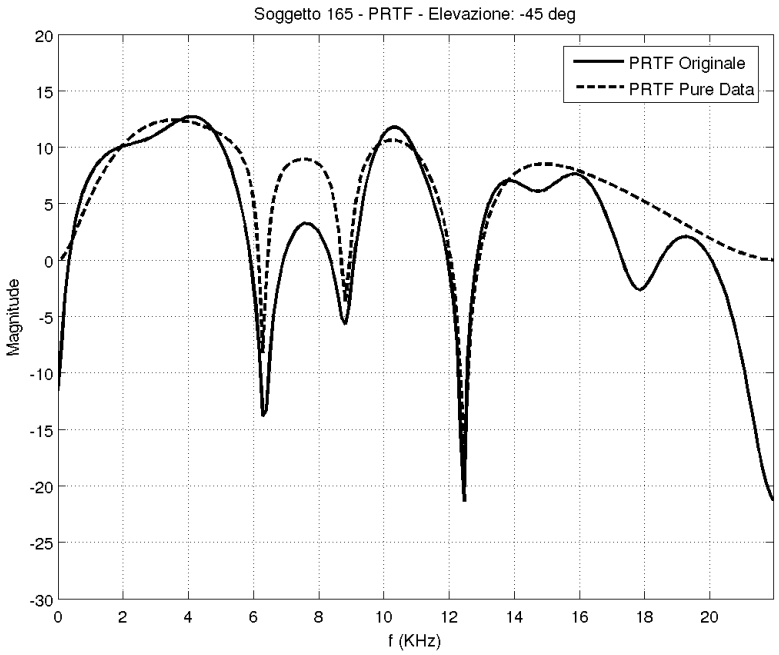


Figura 5.6: Soggetto 165. Elevazione -45

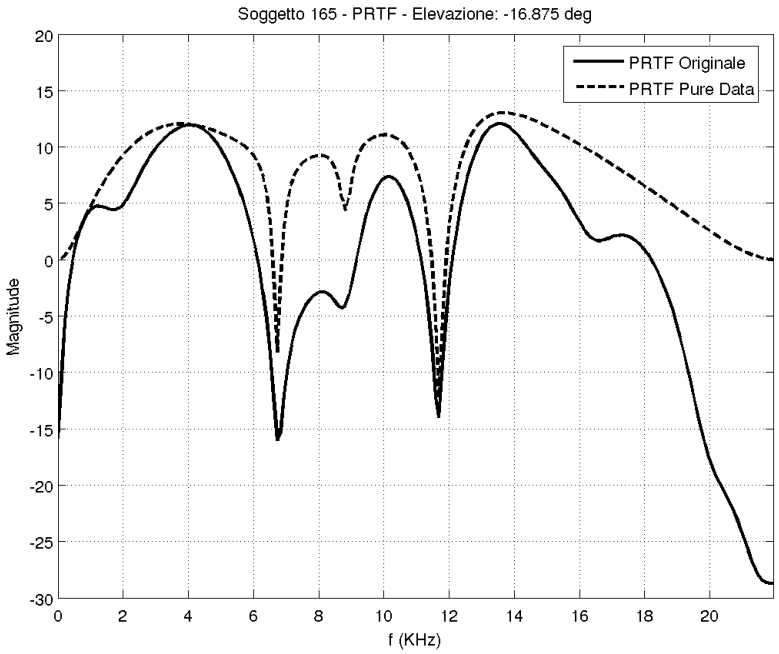


Figura 5.7: Soggetto 165. Elevazione -16,875

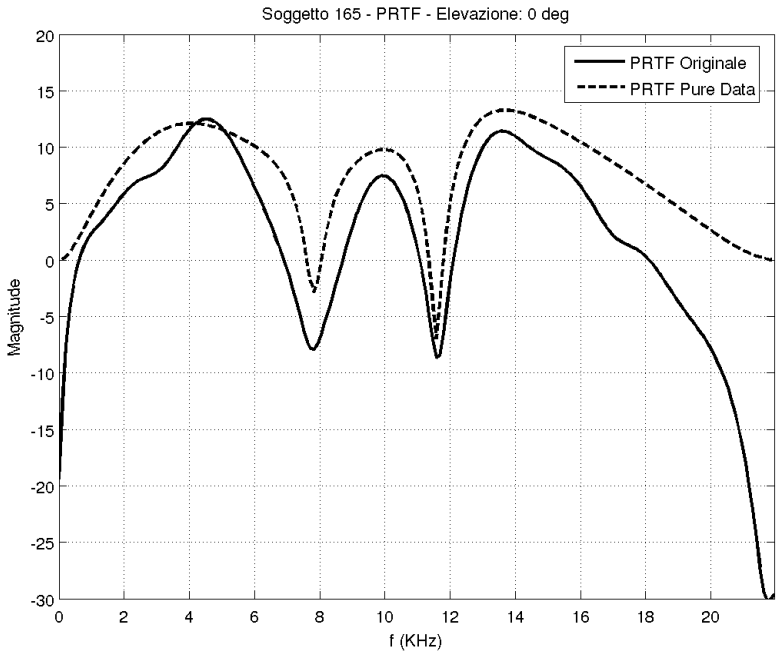


Figura 5.8: Soggetto 165. Elevazione 0

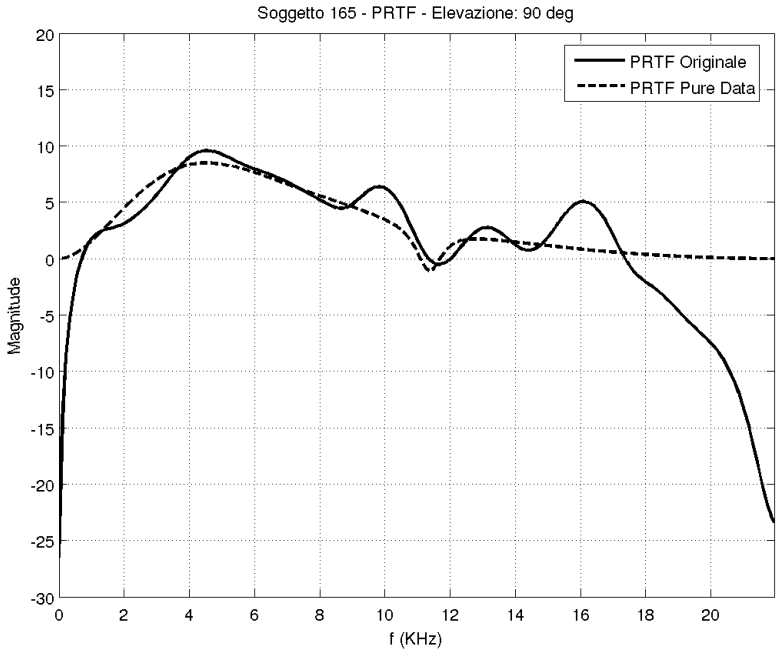


Figura 5.9: Soggetto 165. Elevazione 90

5.2.2 Soggetto 134

Il soggetto 134 del database CICIP è un uomo con pinne caratterizzate da un elice ed antielice abbastanza pronunciate. L'algoritmo presentato nel capitolo 2 si comporta abbastanza bene con questo soggetto. I risultati prodotti sono rappresentati dalle figure 5.10 e 5.11.

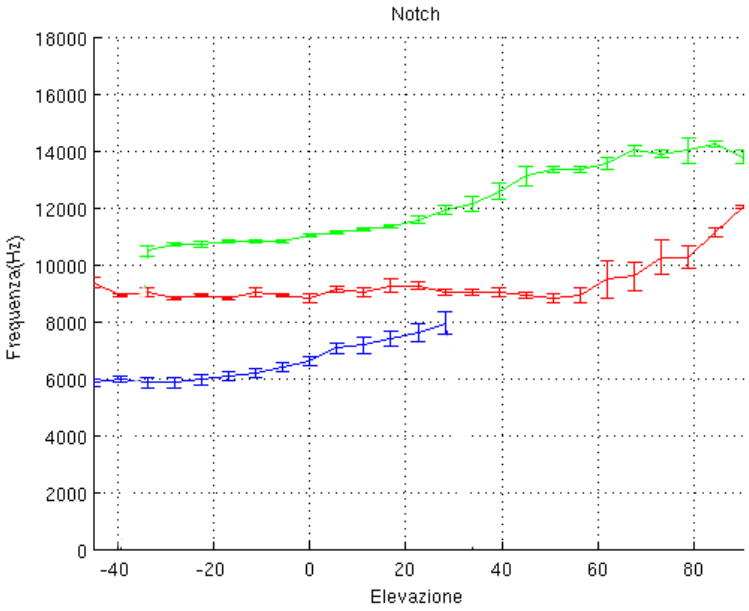


Figura 5.10: Notch (sogg. 134) determinati dall'algoritmo presentato nel capitolo 2

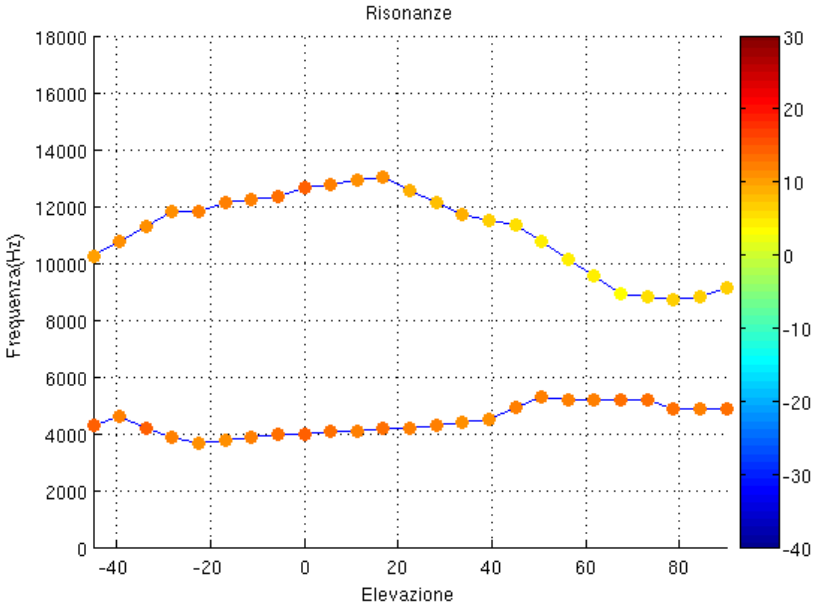


Figura 5.11: Risonanze (sogg. 134) determinati dall'algoritmo presentato nel capitolo 2

Elevazione -45

Come si può vedere dalla figura 5.12 la PRTF è caratterizzata da un numero di valli e picchi molto più numeroso rispetto al soggetto 165. Tutte queste caratteristiche non sono ben rappresentate. L'inviluppo complessivo è comunque ben riprodotto. Il primo notch ha una profondità di 13,7 dB posizionato ad una frequenza di 5,8 kHz. La prima risonanza ha un guadagno di 14 dB a 4,3 kHz. Complessivamente la PRTF sintetica ha il primo notch con una profondità di 0 dB, valore congruo ai valori determinati dall'algoritmo di analisi. La seconda risonanza è troppo pronunciata. I dati in ingresso indicano che ha un guadagno di 10 dB, invece la PRTF sintetica è di qualche dB ancora superiore. La PRTF reale ha un guadagno ben più basso, circa 5 dB. Il problema potrebbe essere risolto sintetizzando i notch con frequenza centrale superiore ai 14 kHz. In questa PRTF ne è presente uno proprio a 14,2 kHz.

Elevazione -16,875

Osservando la figura 5.13 si può affermare che per questa elevazione le risonanze sono ben riconosciute e riprodotte. Il primo notch on è abbastanza profondo. Il contributo apportato dalle risonanze è di circa 6 dB pari al valore di profondità del primo notch. La PRTF sintetica infatti ha nel punto minimo nell'intorno di quella frequenza un valore pari a 0 dB. Si verifica il fenomeno di scarsa approssimazione tra due notch consecutivi. Probabilmente per ottenere una maggiore approssimazione anche nelle frequenza comprese tra 8 e 10 kHz sarebbe stato opportuno aumentare la frequenza di banda del secondo notch. L'inviluppo tra il secondo e terzo notch ha un guadagno di qualche dB superiore alla PRTF reale.

Elevazione 0

Quest'angolo di elevazione è caratterizzato da un notch molto profondo e stretto a 11kHz. A 6,6 kHz la somma delle due risonanze produce un valore in modulo pari a 12 dB. Il notch posto alla stessa frequenza ha un valore di d pari a -9 dB. La PRTF sintetica ha un guadagno, nell'intorno di questa frequenza pari a 2,5 dB, in accordo con quanto rappresentato in figura 5.14. La PRTF misurata ha un guadagno inferiore ai -5 dB. Il guadagno delle due risonanze sembra sovrastimato, questo causa un aumento dell'inviluppo della PRTF sintetica di qualche dB.

Elevazione 90

Tra tutti gli angoli di elevazione questa dovrebbe essere la PRTF più semplice da riprodurre. La prima risonanza è approssimata bene. Il notch intorno ai 14 kHz è troppo poco profondo, questo causa un aumento del modulo per le alte frequenze di qualche dB. Figura 5.15.

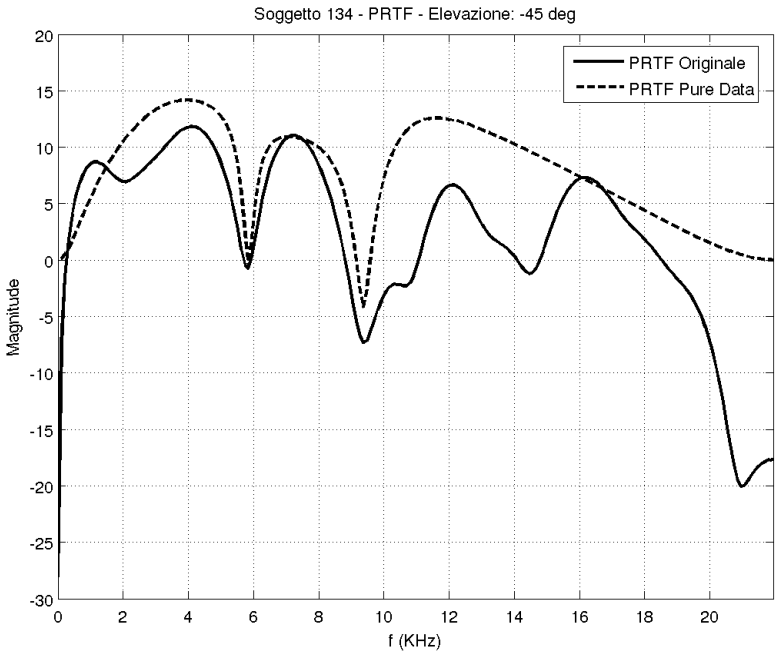


Figura 5.12: Soggetto 134. Elevazione -45

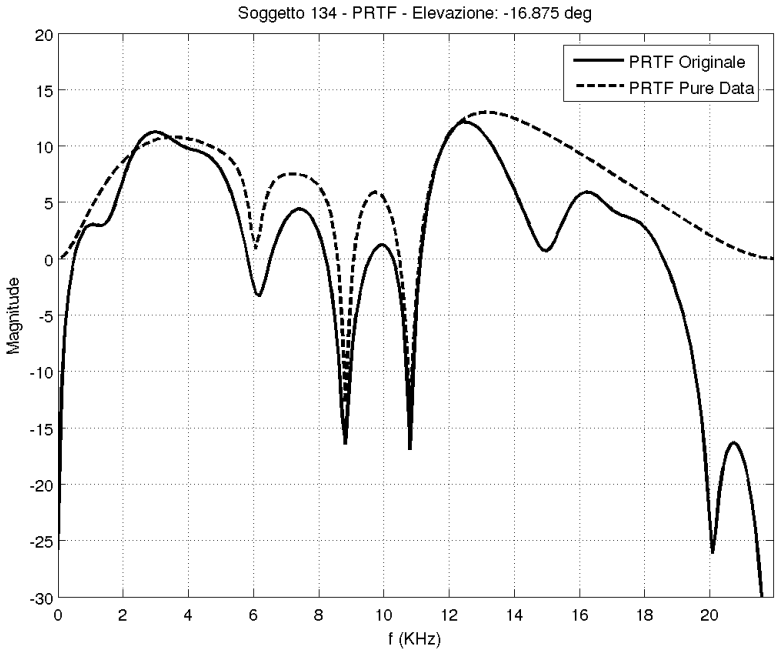


Figura 5.13: Soggetto 134. Elevazione -16,875

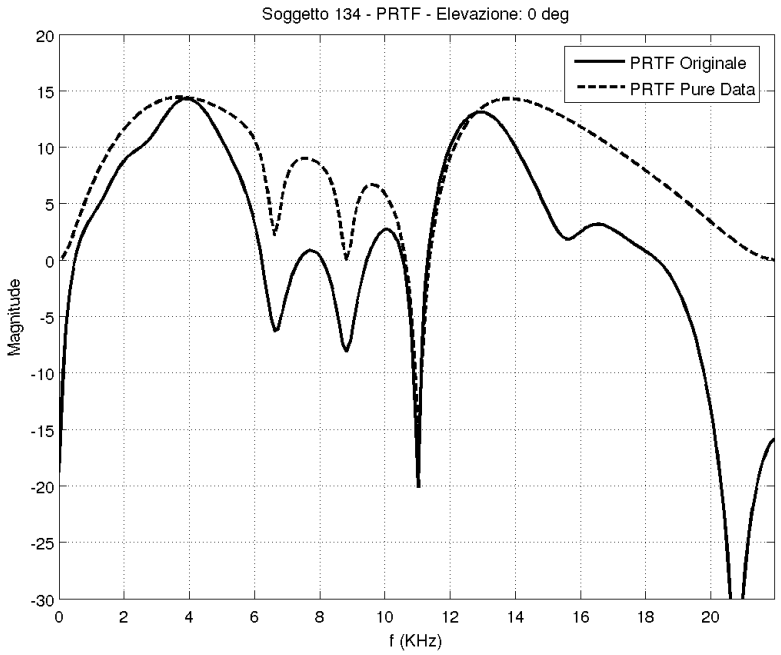


Figura 5.14: Soggetto 134. Elevazione 0

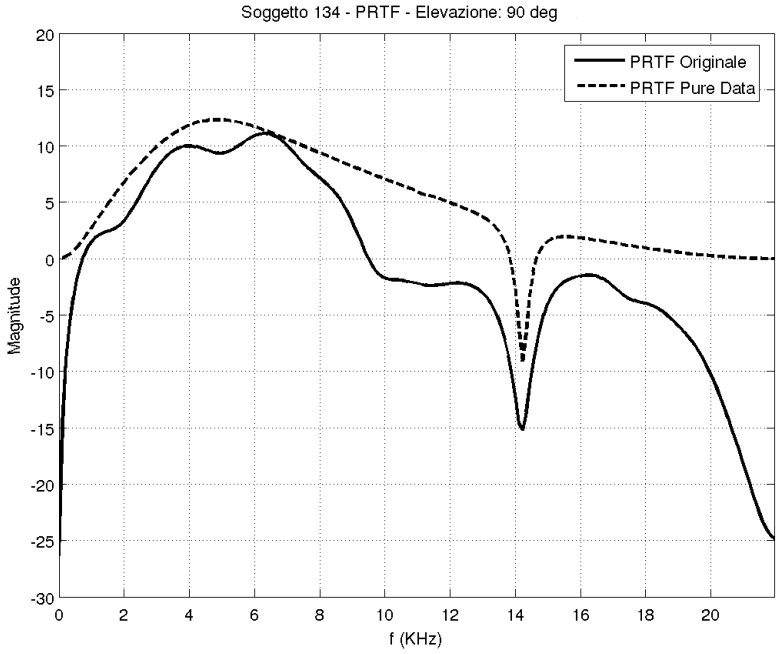


Figura 5.15: Soggetto 134. Elevazione 90

5.2.3 Soggetto 027

Il soggetto 027 è caratterizzato da un elice più avvolgente tra tutti i quattro soggetti. L'antielice è poco pronunciato. I risultati prodotti dall' algoritmo del capitolo 2 sono rappresentate dalle figure 5.16 e 5.17.

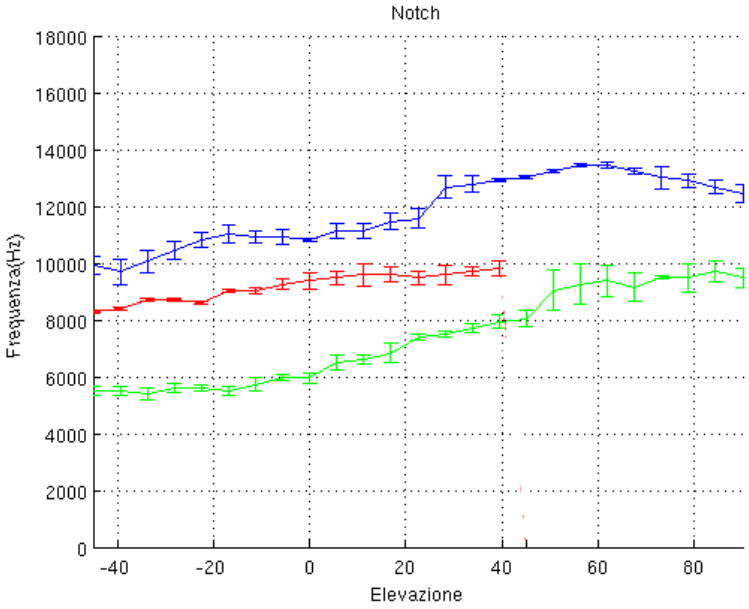


Figura 5.16: Notch (sogg. 027) determinati dall'algoritmo presentato nel capitolo 2

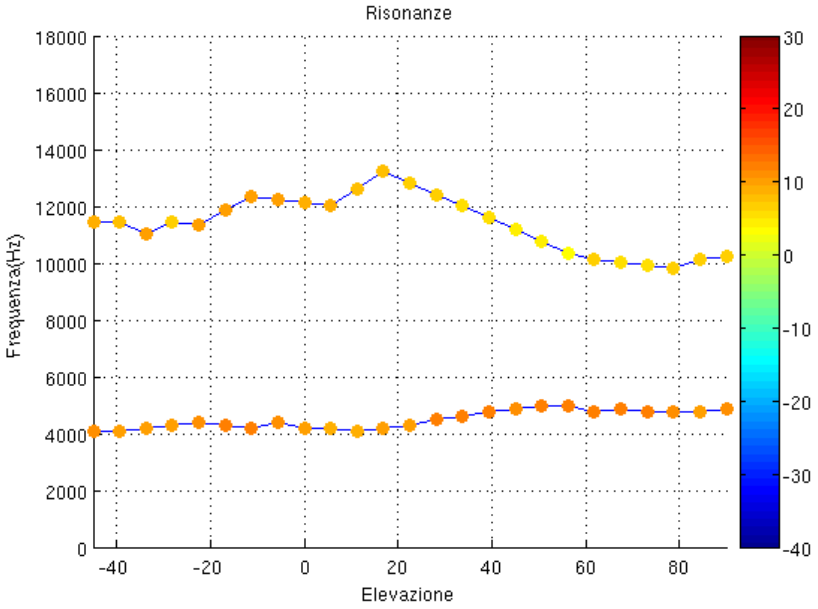


Figura 5.17: Risonanze (sogg. 027) determinati dall'algoritmo presentato nel capitolo 2

Elevazione -45

La difficoltà principale in questa elevazione è rappresentata dal profondo e stretto notch a 8kHz. Il filtro del secondo ordine non riesce a rappresentare correttamente questo notch senza provocare delle ripercussioni non gradite nelle frequenze vicine. Infatti la regione compresa tra i 6 e 10kHz e quella meno precisa. La frequenza centrale della prima risonanza non è stata correttamente determinata. Figura 5.18

Elevazione -16,875

Nel complesso quest'angolo di elevazione è ben approssimato. L'unico problema è presente nei notch: Il secondo notch è posizionato a circa 9 kHz. Il contributo delle risonanze in modulo in tale frequenza è circa 9 dB. Il notch ha una profondità di 20 dB. In totale si ha che il guadagno complessivo è pari a -11 dB. La PRTF reale in quel punto ha invece un valore pari a circa -15 dB. La seconda risonanza ha un guadagno superiore ai 10 dB, maggiore degli 8 dB presenti nella PRTF misurata. Figura 5.19.

Elevazione 0

L'intervallo di frequenze tra i 10 e 12 kHz è quella più problematica e meno precisa.L'algoritmo di analisi individua un notch con frequenza di banda molto stretta, 110 Hz, a 10,8 kHz. La PRTF misurata non ha un notch in quella frequenza. La seconda risonanza ha un guadagno di 8 dB ma l'interazione con la prima porta il guadagno totale a circa 10 dB. Nettamente superiore a quanto richiesto. Figura 5.20

Elevazione 90

In quest'angolo di elevazione l'inviluppo complessivo è bene rappresentato. Il guadagno è ben rappresentato. Esso diminuisce con il crescere della frequenza, in accordo con la PRTF reale. Per questa elevazione sono presenti solo due piccoli notch ($d = -2dB$): una a 12,5 kHz, l'altra a 9,5 kHz. Figura 5.21.

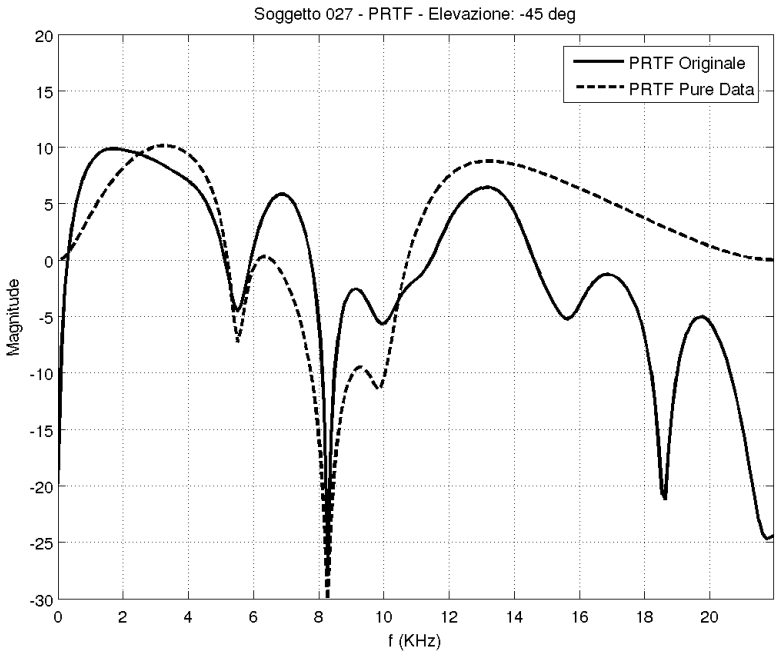


Figura 5.18: Soggetto 027. Elevazione -45

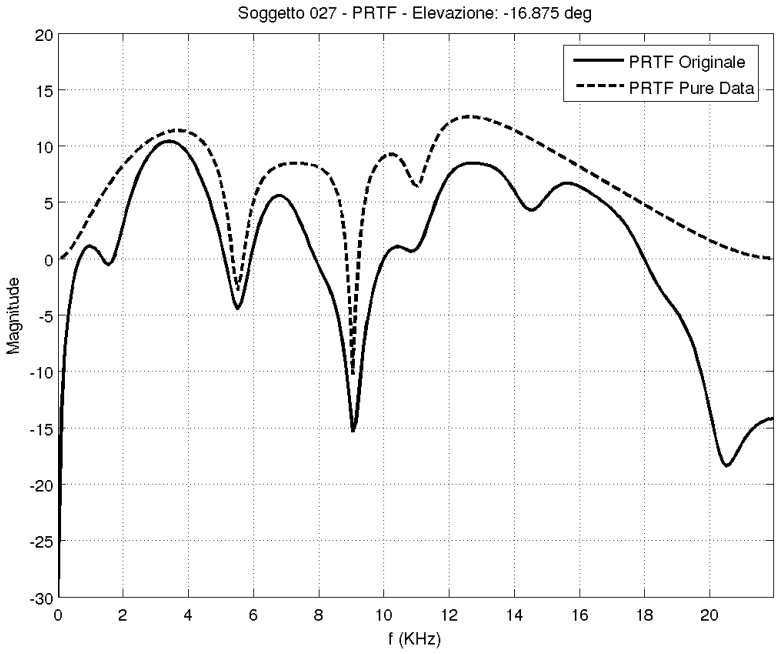


Figura 5.19: Soggetto 027. Elevazione -16,875

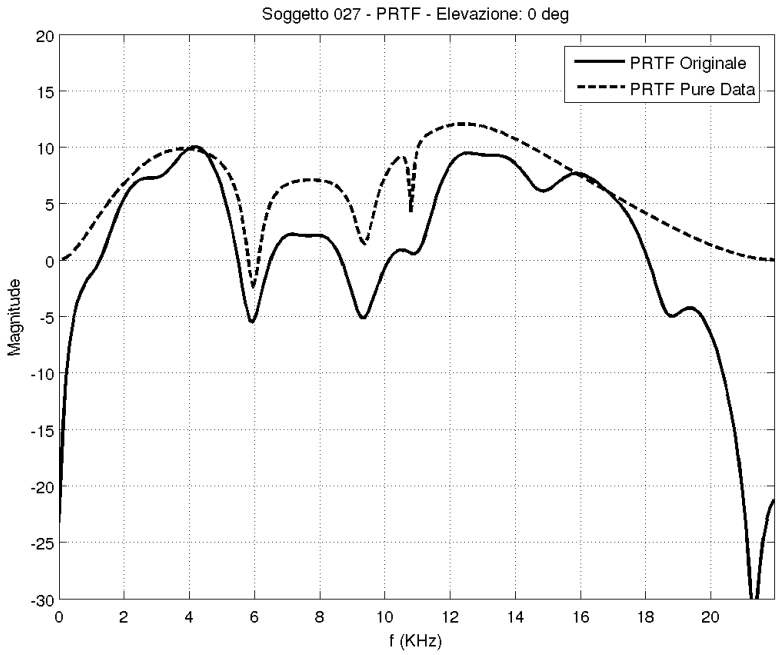


Figura 5.20: Soggetto 027. Elevazione 0

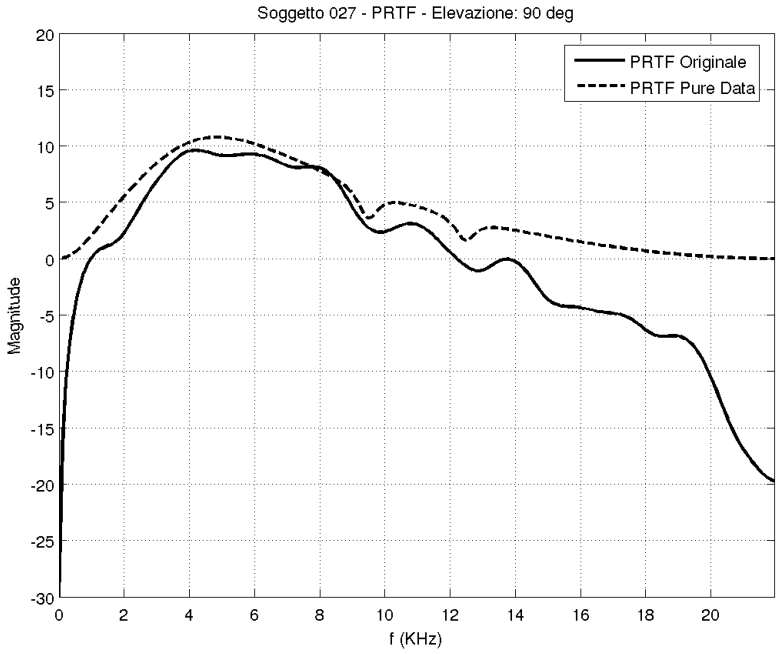


Figura 5.21: Soggetto 027. Elevazione 90

5.2.4 Soggetto 010

L'antitargo di questo soggetto è il più pronunciato, ed è caratterizzata da una conca abbastanza piccola. I risultati prodotti dall' algoritmo del capitolo 2 sono rappresentate dalle figure 5.22 e 5.23.

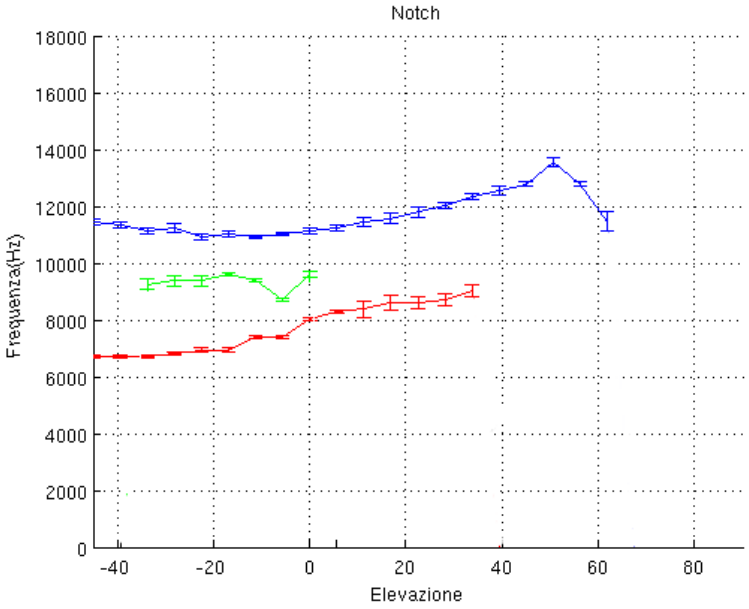


Figura 5.22: Notch (sogg. 010) determinati dall'algoritmo presentato nel capitolo 2

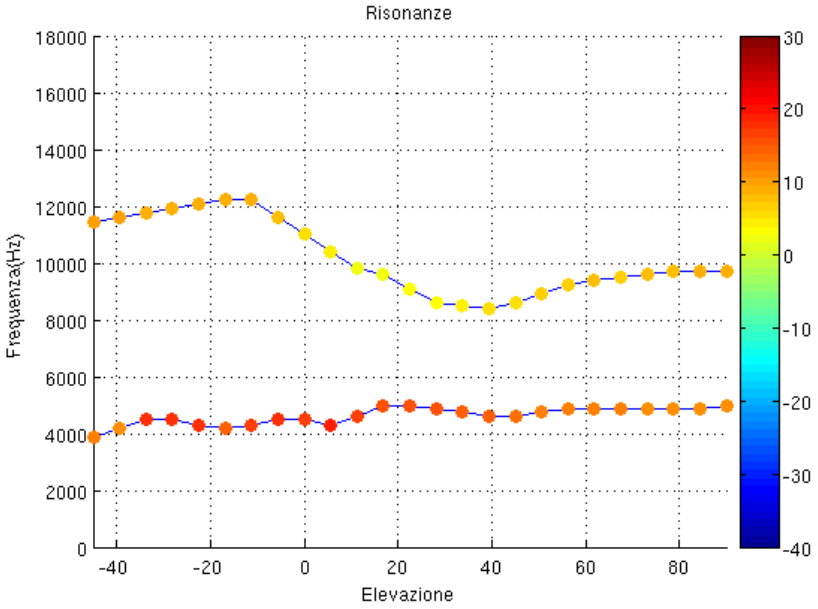


Figura 5.23: Risonanze (sogg. 010) determinati dall'algoritmo presentato nel capitolo 2

Elevazione -45

Per quest'angolo di elevazione, figura 5.24, l'approssimazione della PRTF reale è abbastanza soddisfacente. Rimangono i problemi relativi al secondo notch. I dati ricavati dall'analisi indicano che il secondo notch è posizionato a 11,5 kHz con una profondità di 20 dB. Il contributo delle risonanze nel guadagno totale è pari a 12,3 dB. In totale quindi si ha una profondità di circa 7 dB, che è leggermente superiore a quanto è stato misurato. Il problema sembra essere quindi una sovrastima del modulo della seconda risonanza.

Elevazione -16,875

In generale per questo angolo di elevazione l'inviluppo della PRTF è stato ricostruito abbastanza bene. La prima risonanza è troppo poco pronunciata. Il secondo notch con frequenza centrale pari a 11 kHz e profondità 24 dB è sottostimato. Il contributo delle due risonanze a 11 kHz è pari a 8 dB. Complessivamente il modulo è pari a -16 dB, meno di quanto indicato dalla PRTF reale. A causa di questa sottostima, il proseguo della PRTF sintetica ha un inviluppo di qualche dB superiore a quanto desiderato. Figura **??**.

Elevazione 0

Il primo notch non è abbastanza profondo. L'algoritmo di analisi suggerisce una profondità di 15 dB. Considerando che la somma delle due risonanze per la frequenza centrale del notch, 8 kHz, ha un valore pari a 13 dB, il modulo della PRTF sintetica ha un guadagno di -2 dB. La PRTF reale in quella frequenza ha un valore del modulo pari a -20 dB. Questo causa ripercussioni per tutto il seguito della funzione. L'inviluppo è corretto ma è di molti dB superiore al dovuto. Il problema principale sembra essere un'errata valutazione della profondità del notch da parte dell'algoritmo di analisi e da una cattiva interazione tra le risonanze. Figura 5.26.

Elevazione 90

La specifica che prevede di non considerare i notch oltre i 14kHz ha provocato per quest'angolo di elevazione un errore grossolano. Come si può notare nella figura 5.27 è presente un notch a 15kHz con larghezza di banda molto elevata che influenza l'andamento della funzione anche a frequenze più basse. Poiché questo notch non viene rappresentato nel modello, la PRTF sintetica ha molta più energia rispetto a quella reale.

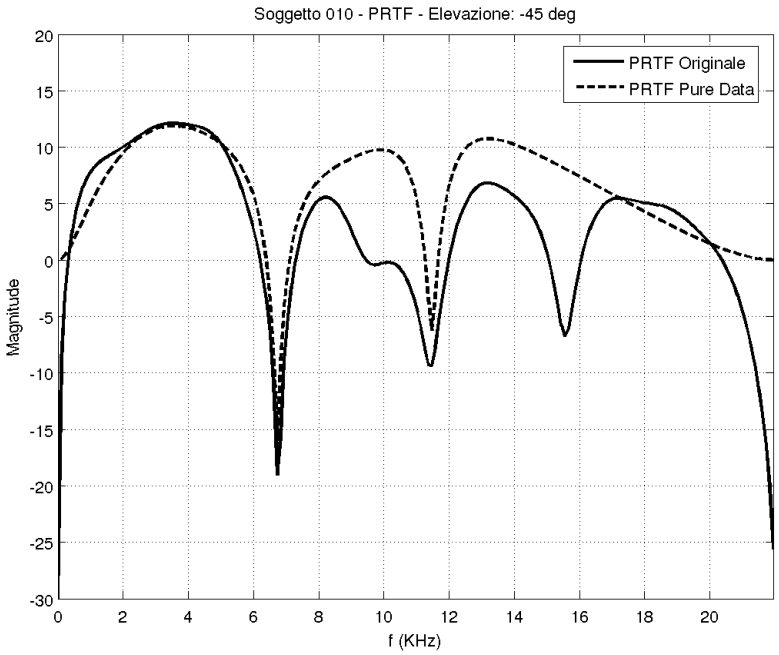


Figura 5.24: Soggetto 010. Elevazione -45

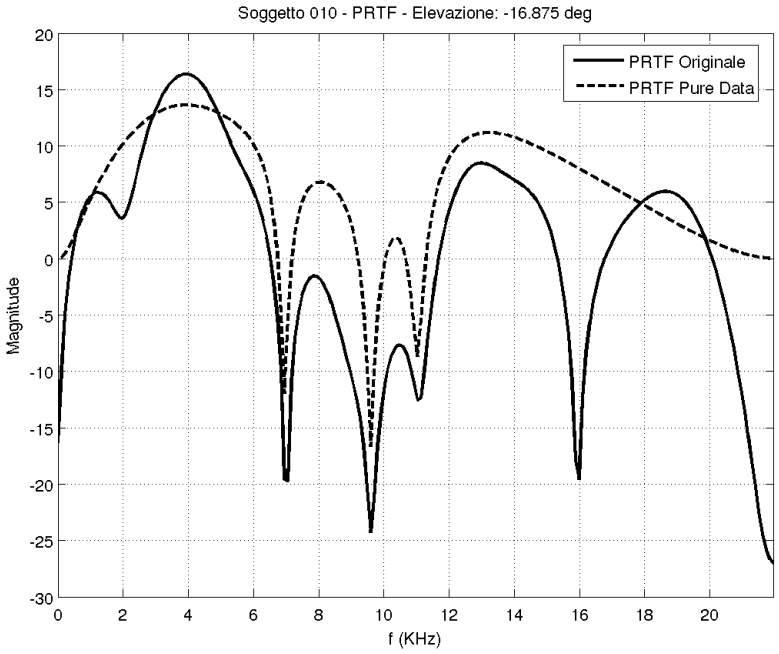


Figura 5.25: Soggetto 010. Elevazione -16,875

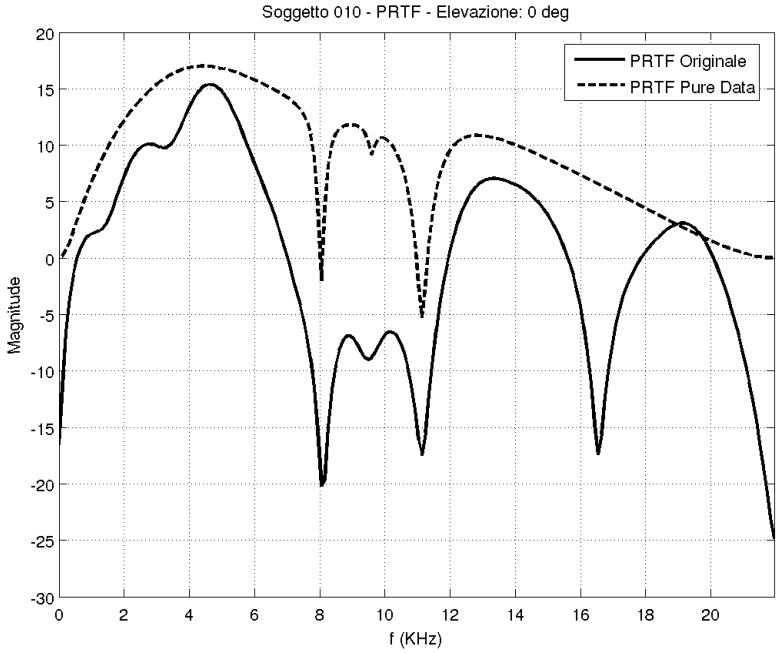


Figura 5.26: Soggetto 010. Elevazione 0

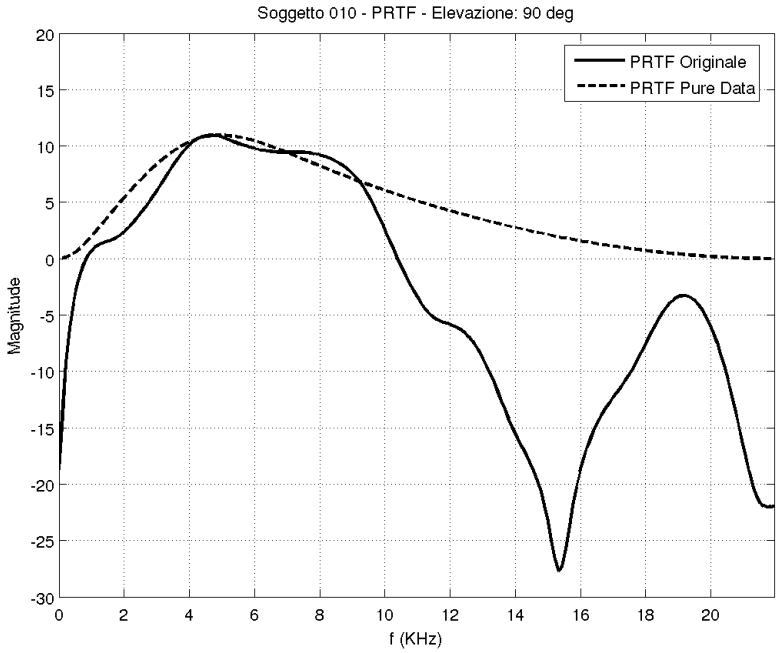


Figura 5.27: Soggetto 010. Elevazione 90

Capitolo 6

Conclusioni

La scomposizione delle PRTF in due filtri peak e tre filtri notch è una buona scelta implementativa. Da un punto di vista analitico le PRTF sintetiche approssimano abbastanza bene quelle reali. Dalle risposte in frequenza osservate nel capitolo 5 si possono notare degli errori ricorrenti. In generale sopra i 45°di elevazione la somma delle due risonanze crea una banda passante con guadagno sovrastimato. Il progetto iniziale prevedeva l'utilizzo dello stesso filtro peak, identificato dall'equazione 2.10, per entrambe le risonanze. Purtroppo le frequenze al di fuori della frequenza di banda, soprattutto quelle basse, sono tagliate. L'utilizzo di tale filtro nelle frequenze in cui opera la prima risonanza causa una variazione dello spettro generale che può portare ad un'incorretta percezione spaziale. La pinna è insensibile alle basse frequenze, le PRTF non devono modificare in alcun modo il segnale in ingresso; il guadagno deve essere quindi pari a 0 dB. Per questo motivo si è utilizzato il filtro peak definito dall'equazione 2.14. Per la seconda risonanza si è deciso di mantenere il filtro dell'equazione 2.10. Purtroppo nelle frequenze in cui opera la seconda risonanza, la prima ha un guadagno di 0 dB. Le due risonanze si sommano provocando quindi un guadagno totale eccessivo (soprattutto per gli angoli di elevazione superiori a 45 gradi). Sono state individuate tre possibili soluzioni:

- Quando l'angolo di elevazione supera i 45 gradi si modella la sola prima risonanza con il filtro di equazione 2.14. Il contributo della seconda risonanza potrebbe essere irrisorio. Analiticamente i risultati sono buoni rimane da verificare percettivamente la validità dell'affermazione.

- Si potrebbe immettere in serie al blocco che sintetizza la prima risonanza un opportuno filtro passa basso, in modo da eliminare l'interazione con la seconda risonanza. Il problema in questo caso è di comprendere il tipo di filtro da usare.

- Si interpreta il guadagno della seconda risonanza come grandezza relativa, in funzione del guadagno già percepito dal filtro che sintetizza la prima risonanza nella banda di intesse del secondo peak.

Si è deciso di realizzare il primo punto ed i risultati analitici sono buoni. Rimangono da effettuare dei test psicoacustici per verificare la bontà del modello.

Dallo studio degli errori ricorrenti riscontrati nell'osservazione dei risultati ottenuti del capitolo 5, si può pensare di migliorare l'algoritmo di analisi. In particolare la sezione che riguarda la componente riflettente delle PRTF è particolarmente critica. Un notch troppo poco profondo può causare un errore che si ripercuote per le successive frequenze. Per avere una prima validazione audio del modello presentato nel capitolo 2 è stato necessario implementare un blocco in pure data che rappresenti la diffrazione della testa. Il risultato ottenuto è il modello strutturale presentato in 1.4. A questo scopo si è realizzato un filtro del primo ordine presentato in [11] e ripreso in 1.4.1. Si è impostato α_{min} a 0,1 e θ_{min} pari a 150. Questo tipo di modello è abbastanza semplice; principalmente si comporta come un panner tra orecchio destro e sinistro. L'aggiunta del blocco che rappresenta la pinna, produce, comunque, una colorazione dello spettro che garantisce un aumento di esternalizzazione. Questo fatto sottolinea quanto sia importante la modellazione della pinna nella spazializzazione del suono.
Come prima stima della validità del modello si sono utilizzati i valori di notch e risonanze derivati dal soggetto 165 del database CIPIC. Si può affermare che tra tutti i soggetti presenti nel database CIPIC il manichino KEMAR è quello che può essere utilizzato dal maggior numero di persone, poiché è quello che ha le HRTF meno particolarizzate.
Introducendo nel modello alcuni suoni, tra cui il ronzio di un ape, il rombo di un elicottero e il suono di un pianoforte, il risultato è abbastanza soddisfacente. Modificando l'angolo di azimuth la sensazione che la fonte sonora si muova nel piano orizzontale è buona. Come affermato prima, la sola aggiunta del blocco che simula la pinna aumenta notevolmente la sensazione di esternalizzazione. Cambiando, invece, l'elevazione si possono osservare alcuni fenomeni. Per alcuni angoli compresi tra -40 e -17 l'individuazione della fonte sonora risulta più complicata. Il suono sembra provenire da un'area più estesa, diffusa, difficilmente individuabile. In alcuni casi risulta difficile capire da dove proviene il suono, senza qualche indizio visivo che aiuti a chiarire la posizione della sorgente. Utilizzando parametri determinati da altri soggetti, per esempio il 010, la percezione di elevazione risulta (per me) un po' più accurata. Per ottenere un'ulteriore conferma sulle potenzialità del nuovo modello realizzato in Pure Data si è effettuata una comparazione con un'external che realizza un compito simile: earplug~. Questo blocco, utilizza la risposta all'impulso del manichino KEMAR per determinare, attraverso una convoluzione, i segnali da riprodurre alle due orecchie. La risposta all'impulso utilizzata in earplug~, non è quella presente nel database CIPIC, ma proviene da altri rilevamenti. Essendo, tuttavia, il manichino KEMAR utilizzato nelle due misurazioni lo stesso, ed assumendo che le operazioni di registrazione della risposta all'impulso siano state effettuate adeguatamente, si può supporre che la sensazioni prodotte da earplug~ e da questo modello siano percettivamente comparabili. Il risultato prodotto da earplug~ è abbastanza convincente, la sensazione che la fonte sonora si muova lungo il piano mediano è buona fino ad un angolo di elevazione pari a 70, poi si percepisce il suono all'interno della testa. La difficile distinzione dell'intera gamma di angoli di elevazione può essere ricercata in errori percettivi dovuti alla non individualizzazione delle HRTF. Questi rimangono comunque dei test qualitativi: dovrebbe essere prevista una più robusta validazione psicoacustica.

6.1 Sviluppi Futuri

Il modello delle PRTF utilizza l'interpolazione lineare per determinare i parametri necessari alla ricostruzione delle PRTF negli angoli di elevazione non campionati. Questa assunzione non è sempre valida, perché si è visto che alcuni notch non seguono un andamento lineare, ma seguono delle traiettorie predefinite. Un possibile miglioramento può essere quello di computare queste traiettorie per rendere le ricostruzioni delle PRTF più consistenti, anche negli angoli non campionati. Per poter effettuare un miglior confronto con earplug$\sim$ sarebbe utile poter ricostruire il file contenente la risposta all'impulso di earplug$\sim$ con le misurazioni effettuate sul manichino KEMAR nel database CIPIC. Così facendo si potrebbero confrontare psicoacusticamente i due modelli, per valutarne i pregi e difetti.
Per diminuire i fenomeni di *frontback confusion* ed aumentare la sensazione di esternalizzazione è opportuno prevedere un sistema di *head tracking*, utilizzando, per esempio, il *phasespace*, un sistema di *motion capture* presente in laboratorio. Come si è visto la sensazione di internalizzazione diminuisce drasticamente se si fornisce la possibilità all'ascoltatore di effettuare dei piccoli movimenti della testa, per individuare la posizione della fonte sonora. Un altro aumento considerevole nell'accuratezza della localizzazione del suono si può ottenere aggiungendo un blocco per sintetizzare fenomeni di riverberazione e riflessione, causati dall'ambiente in cui l'ascoltatore è immerso.

Per aumentare l'utilità di questo prototipo è vantaggioso prevedere un sistema di individualizzazione delle PRTF attraverso il recupero di informazioni relative alle risonanze ed ai notch, dalle immagini delle pinne degli ascoltatori. Alcuni studi sono già stati effettuati [24], ed i risultati nei soggetti del database CIPIC sono buoni. L'errore massimo nel posizionamento di risonanze e notch si aggira attorno ad 1kHz.

Bibliografia

[1] R. V. Algazi, R. O. Duda, D. M. Thompson, and C. Avendano. The cipic hrtf database. *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2001.

[2] F. Avanzini and G. De Poli. *Algorithms for Sound and Music Computing*. 2010. Disponibile all'indirizzo `http://smc.dei.unipd.it/education.html`.

[3] F. Avanzini, S. Spagnol, and M. Geronazzo. Esitimation and modeling of pinna-related transfer functions. *Int. Conf. Digital Audio Effects (DAFx-10)*, pages 431–438, Graz, Settembre 2010.

[4] F. Avanzini, S. Spagnol, and M. Geronazzo. Structural modeling of pinna-related transfer functions. *In Proc. Int. Conf. Sound and Music Computing (SMC2010)*, pages 422–428, Barcelona, July 2010.

[5] A. Barreto, K.J. Faller, N. Gupta, and N.Rishe. Time and frequency decomposition of head-related impulse resopnses for the development of cusomizable spatial audio models. *WSEAS Transaction on Signal Processing*, 2(11):1465–1472, 2006.

[6] D. W. Batteau. The role of the pinna in human localization. *Proc. R. Soc. Londra*, pages 158–180, 1967.

[7] D. R. Begault. *3D Sound for Virtual Reality and Multimedia*. Academic Press Inc., 1994.

[8] J. Blauert. *Spatial Hearing: Pyschophysics of Human Sound Localization*. MIT Press, 2nd edition, 1996.

[9] C. I. Cheng and G. H. Wakefield. Introduction to head-related transfer functions (hrtfs): Rappresentation of hrtfs in time, frequency, and space. *J. Audio Eng. Soc.*, 49(4):231–249, Aprile 2001.

[10] R. O. Duda, R. V. Algazi, R. P. Morrison, and D. M. Thompson. Structural composition and decomposition of hrtf. *IEEE ASSP Workshop on Applications of Signal Processing to Audio and Acustic*, pages 103–106, 2001.

[11] R. O. Duda and C. P. Brown. A structural model for binaural sound synthesis. *IEEE Trans. Speech Audio Process*, 6(5):476–488, 1998.

[12] M. Geronazzo, S. Spagnol, and F. Avanzini. Fitting pinna-related transfer functions to anthropometry for binaural sound rendering. *In Proc. IEEE International Workshop on Multimedia Signal Processing (MMSP'10)*, pages 194–199, Ottobre 2010.

[13] T. Grill. *flext. C++ programming layer for cross-platform develpment of PD and Max/MSP externals.* Disponibile all'indirizzo `http://puredata.info/Members/thomas/flext/`.

[14] W.M. Hartmann and A. Wittenberg. On the externalizzation of sound images. *J. Acustic. Soc. Am*, 99(6):3678–3688, 1996.

[15] Julius O. Smith III. *Introduction to digital filters with audio applications.* Settembre 2007. Disponibile all'indirizzo `https://ccrma.stanford.edu/ jos/filters/filters.html`.

[16] G. S. Kendall. The decorrelation of audio signals and its impact on spatial imagery. *Computer Music Journal*, 19(4):71–87, Dicembre 1995.

[17] D. J. Kistler and F. L. Wightman. A model of head-related transfer function based on principal components analysis and minimum pahse reconstruction. *J. Acustic. Soc. Am*, 91(3):1637–47, Marzo 1992.

[18] A. Kulkarni and H. S. Colburn. Role of spectral detail in sound-source localization. *Nature*, 369:747–749, 1998.

[19] A. Kulkarni, S.K. Isabelle, and H. S. Colburn. On the minimum-phase approximation of the head-releated transfer functions. *IEEE ASSP Workshop on Applications of Signal Processing to Audio and Acustic*, pages 84–87, Ottobre 1995.

[20] S. K. Mitra. *Digital Signal Processing: A Computer Based Approach.* McGraw-Hill, third edition, 2005.

[21] M. Puckette. *The Theory and Technique of Electronic Music.* 2006. Disponibile all'indirizzo `http://www-crca.ucsd.edu/ msp/techniques.htm`.

[22] V.C. Raykar, R. Duraiswami, and B. Yegnanarayana. Extracting the frequencies of the pinna spectral notches in measured head related impulse responses. *J. Acustic. Soc. Am*, 118(1):364–374, 2005.

[23] Lord Rayleigh. On our percepition of sound direction. *Philos. Mag*, 1907.

[24] P. Satarzadeh, R. V. Algazi, and R. O. Duda. Phisical and filter pinna models based on antrhopometry. *Proc. 122nd Convention of the Audio Engineering Society*, 2007.

[25] J. F. Schouten. The perception of timbre. *Reports of the 6th International Congress on Acoustics*, 1968.

[26] E. A. G. Shaw. A study of physical and circuit models of the human pinnae. Master's thesis, University of California Davis, 2006.

[27] E. A. G. Shaw, T.R. Anderson, and R. H. Gilkey. *Binaural and Spatial Hearing in Real and Virtual Environments*, chapter Acustical features of human ear, pages 25–47. Lawrence Eribaum Associates, 1997.

[28] Udo Zolzer. *Digital Audio Effects*. John Wiley & Sons, 2002.

Ringraziamenti

Rigrazio i miei genitori che mi hanno sempre incoraggiato, accompagnato e appoggiato in tutti i miei anni di studio. Questo traguardo è stato raggiunto soprattutto grazie a loro.

Desidero ringraziare, in particolare, il Professore Federico Avanzini, Simone Spagnol e Michele Geronazzo che mi hanno aiutato moltissimo nella realizzazione di questa tesi, e mi scuso per le innumerevoli mail.

Voglio ringraziare l'università di Padova e quella di Aberdeen per avermi permesso di partecipare al progetto Erasmus durante i miei studi universitari. Grazie ai compagni di università. Senza le sessioni intense di studio (e divertimento) sarebbe stata molto più dura. Infine dico grazie a tutti i miei amici perché ci sono.