UNIVERSITÀ
DEGLI STUDI
DI PADOVA

DIPARTIMENTO
DI INGEGNERIA
DELL'INFORMAZIONE

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

CORSO DI LAUREA TRIENNALE IN INGEGNERIA
DELL'INFORMAZIONE

"Implementazione di un'interfaccia grafica per l'acquisizione e la visualizzazione real-time della risposta in frequenza da router Wi-Fi commerciali"

Relatore: Prof. Rossi Michele

Laureando: Gallo Antonio

Correlatore: Dott. Meneghello Francesca

ANNO ACCADEMICO 2021 – 2022

Data di laurea 07/03/2022

Abstract

Dato il notevole aumento di interesse circa l'impiego di router Wi-Fi ad uso domestico ed istituzionale nella *human activity recognition* tramite l'estrazione del CSI - risposta in frequenza del canale di comunicazione wireless - si è più recentemente assistito ad un approfondimento sull'utilizzo di devices IEEE 802.11ac nel rilevamento dei movimenti più impercettibili, quale, nello specifico di questo caso, quello respiratorio.

Lo scopo di questa tesi è dunque quello di affinare il processo di estrazione dei dati tramite l'uso dell'applicazione sviluppata ed appoggiata sul software Nexmon, e grazie alla quale si possono visualizzare modulo e fase della risposta in frequenza durante la trasmissione. Ciò rende possibile osservare run-time i dati raccolti durante l'acquisizione ed una supervisione più efficace della progressione dell'estrazione del CSI.

Riconoscimenti

Un sentito ringraziamento alla mia famiglia per il supporto e l'affetto.

A Carmen, che ha sempre saputo starmi vicino.

Contenuti

Introduzione

Questo progetto di tesi nasce da una più ampia ricerca condotta dall'Università degli studi di Padova circa la *human activity recognition* (HAR), impiegando router Wi-Fi ad uso domestico ed istituzionale come sensori capaci di rilevare e riconoscere diverse attività umane, dai movimenti più tracciabili ai più impercettibili.

Nello specifico, il sistema di riconoscimento su cui si basa questo lavoro si potrebbe impiegare per il monitoraggio delle funzioni vitali dei pazienti all'interno di strutture sanitarie come ospedali o RSA.

Lo scopo del progetto riguarda quindi l'utilizzo di devices IEEE 802.11ac nell'estrazione di parametri vitali fondamentali quale - ad esempio - l'attività respiratoria tramite l'acquisizione del CSI (Channel State Information), da cui viene estratta la risposta in frequenza del canale di comunicazione Wireless.

Un ulteriore proposito di questa tesi è inoltre quello di migliorare il processo di acquisizione dei dati tramite l'impiego dell'applicazione sviluppata, la quale si appoggia sul software Nexmon Channel State Information Extractor, un firmware patching framework basato su C che permette, tra le sue molteplici funzioni, l'estrazione del CSI per alcune tipologie di chip, e la visualizzazione di modulo e fase della risposta in frequenza mentre avviene la ricezione dei pacchetti .pcap al router utilizzato: i dati raccolti in run-time rendono possibile supervisionare al meglio la progressione dell'estrazione del CSI.

Infine, il sistema di riconoscimento su cui si basa questo lavoro si potrebbe impiegare per il monitoraggio delle funzioni vitali dei pazienti all'interno di strutture sanitarie come ospedali o RSA.

Capitolo 1

Estrazione della frequenza respiratoria

1.1 Introduzione

Circa 1 miliardo della popolazione mondiale è affetto da patologie respiratorie croniche, che causano oltre il 7% delle morti: è per questo, infatti, che il monitoraggio della respirazione è uno degli accorgimenti più importanti nel controllo della salute e nella diagnosi medica; da ciò, infatti, si possono dedurre diverse informazioni sullo stato biologico delle persone. Tuttavia, fino a poco tempo fa, i metodi utilizzati per la misurazione respiratoria nell'uomo erano piuttosto costosi ed intrusivi e richiedevano contatto fisico tra la persona ed il sensore; ciò provocava un'alterazione della frequenza respiratoria e falsava dunque la precisione della misurazione.
La soluzione nasce con schemi di monitoraggio basati su frequenza radio, che propongono metodi di controllo respiratorio meno intrusivi, i quali tuttavia richiedono attenzione alla condizione del tag posizionato sulla persona o una larga banda wireless, cosa che ne limita l'applicazione.
Per ridurre il costo e le difficoltà, sono stati proposti infine dei metodi basati sull'uso di router Wi-Fi commerciali, argomento di cui tratterà il presente capitolo.
Seppure la tesi verta su altre attività, sembrava opportuno introdurre l'argomento dell'estrazione della frequenza respiratoria quindi l'obiettivo di questo capitolo è quello di dare un'idea di come avvenga.

1.2 Tecniche di estrazione parametri vitali

Gli approcci per il monitoraggio e l'estrazione della frequenza respiratoria tramite Wi-Fi si possono dividere in due categorie, quali:

1. *Received Signal Strength Indicator (RSSI)*: è stato il primo metodo utilizzato nella rilevazione della frequenza respiratoria, presentando tuttavia diverse difficoltà. Innanzitutto, l'RSSI è insensibile ai piccoli movimenti del torace e durante la respirazione potrebbe subire l'interferenza dei rumori. Si richiede poi che i soggetti rimangano vicini ai router Wi-Fi, dando per scontato che la respirazione

di ciascuno sia costante e periodica e rendendo impossibile quindi il suo monitoraggio nel caso di un andamento anormale (apnea, asma, altre patologie).

2. Channel State Information (*CSI)*: rispetto all'RSSI, il CSI è più sensibile alla respirazione umana e viene impiegato per la prima volta nell'indagine di essa in lavori di monitoraggio del sonno (*Wi-Sleep*), successivamente elaborato fino a rendere possibile il tracciamento della respirazione ed il battito cardiaco anche in posizione eretta.
   Questo secondo metodo descritto, è quello che è stato adoperato nel lavoro svolto per la presente tesi e che sarà approfondito nei successivi paragrafi.

1.2.1 Panoramica sul CSI

La differenza di fase e l'attenuazione di potenza contraddistinguono le proprietà di propagazione del segnale. Nello specifico, il CSI caratterizza il canale wireless fornendo la risposta in frequenza, dall'inglese Channel Frequency Response (CFR). Si tratta di una matrice contenente valori complessi da cui possono essere estratti modulo e fase del segnale ricevuto.
In letteratura ci si riferisce al CSI indicandolo con $H(f)$, le informazioni sul segnale inviato sono contenute nel vettore $X(f)$, mentre le informazioni su quello ricevuto sono contenute nel vettore $Y(f)$, il tutto è tenuto insieme dalla relazione $Y(f) = H(f)X(f)$.

1.3 Wi-Fi respiration detection

L'estrazione della frequenza respiratoria tramite Wi-Fi è preferibile ed ottimale in uno spazio senza riflessioni così da identificare solo quelle dovute al soggetto, è tuttavia possibile effettuarla anche all'interno di un luogo chiuso come ad esempio una stanza, con i dovuti accorgimenti e tecniche per il filtraggio delle sole componenti dovute al soggetto.
L'esperimento prevede dunque l'utilizzo di due router, un trasmittente ed un ricevente, situati in due diverse postazioni all'interno dello stesso luogo, e la presenza di uno o più soggetti.

1.3.1 Multipath propagation

La multipath propagation è un fenomeno riguardante le onde elettromagnetiche che si propagano a partire da un'antenna trasmittente: la diffusione nell'ambiente circostante avviene in modo quasi isotropico per

mezzo di onde piane, motivo per cui non è possibile inoltrare il segnale direttamente alla ricevente. Durante il percorso le onde radio si propagano tramite riflessione, diffrazione e dispersione subendo anche attenuazioni e ritardi, che si sommano vettorialmente formando un segnale ricevuto composito. Il fenomeno è rappresentato graficamente in figura 1.

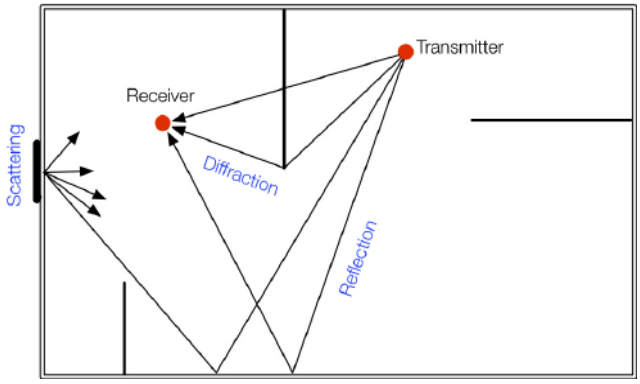


Figura 1: Indoor Multipath Propagation | fonte: One Stage Indoor Llocation Determination Systems

1.3.2 Single-person respiration sensing tramite CSI

Nel caso specifico di questa tesi, la misurazione avviene all'interno di un luogo chiuso, all'interno del quale, i due router – trasmittente e ricevente – vengono situati in due punti precisi di esso, mentre al soggetto, a seconda dell'esperimento, viene richiesto di assumere diverse posizioni sempre tra i due devices.

Nel momento in cui il soggetto respira, il movimento del torace causa la variazione della lunghezza di percorso dell'onda elettromagnetica emessa dal trasmittente e riflessa al ricevente.

Generalmente, l'escursione del petto è di 5-12 mm, e bisogna tenere conto del fatto che questa apparentemente lieve differenza in realtà comporta grandi modifiche sulla fase del segnale ricevuto: esistono ripercussioni legate alla riflessione multipla (sia sul petto del soggetto, sia sull'ambiente circostante) del segnale inviato, così come precedentemente illustrato dal fenomeno del multipath propagation.

Considerando un'onda radio con lunghezza d'onda λ, quando essa si muove lungo un percorso di lunghezza d, la fase trasla di una quantità pari a $2\pi d/\lambda$.

Il segnale ricevuto, per effetto della multipath propagation, può quindi essere espresso come $\sum_{p=0}^{P-1} A_p e^{-j2\pi d_p/\lambda}$, dove p *è l'*identificativo del percorso e A_p è il coefficiente di attenuazione di ogni percorso.

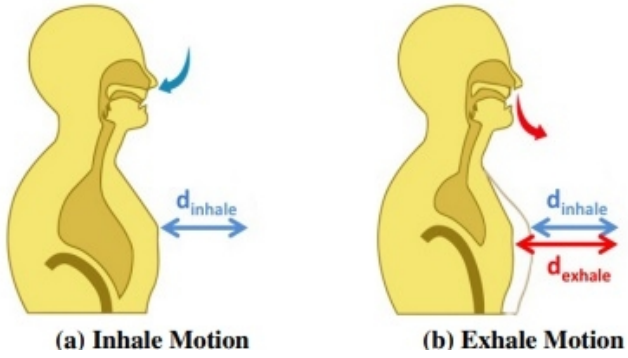


Figura 2: Il movimento del petto durante la respirazione | fonte: voce 2 bibliografia

Il CSI è dunque la sovrapposizione delle componenti provenienti da ogni percorso attraversato dall'onda radio emessa dalla trasmittente, e si può dividere in componenti statiche e dinamiche.

Le componenti statiche rappresentano la variazione, costante nel dominio del tempo, sull'onda radio trasmessa da tutto ciò che all'interno della stanza ha una posizione statica, compreso il soggetto; il suo movimento toracico, invece, genera tramite riflessione delle modifiche sul percorso delle onde e sulla loro fase: tali modifiche, variabili nel tempo, rappresentano le componenti dinamiche.

Il CSI totale può quindi essere separato nelle componenti:

$$H(f,t) = H_s(f,t) + H_d(f,t) = H_s(f,t) + A(f,t)e^{-j2\pi d(t)/\lambda}$$

dove $H_s(f,t)$ è la componente statica, $A(f,t)$, $e^{-j2\pi d(t)/\lambda}$ e $d(t)$ sono rispettivamente l'attenuazione, lo sfasamento e la lunghezza del percorso della componente dinamica.

1.3.3 Multi-person respiration sensing

In questo paragrafo si discute brevemente la più complessa questione della *multi-person respiration sensing*. Si consideri il caso in cui i soggetti esaminati si trovino molto vicini, i loro segnali riflessi sono mescolati, e ciò ne rende difficile la distinzione.

Il multi-person respiration sensing può essere modellato con successo tramite un problema di *Blind Source Separation* (BSS) basato sul CSI.

Il BSS cerca di ricostruire segnali sorgenti sconosciuti ***s(t)*** a partire dai segnali ricevuti ***x(t)***, risultanti dalla mescolanza dei segnali ***s(t)*** passando attraverso un canale di propagazione di cui non si dispongono informazioni.

Il problema BSS può essere efficientemente risolto da una tecnica chiamata *Independent component analysis* (ICA) se sono sono soddisfatti i seguenti assunti: i segnali non devono avere statistica Gaussiana, sono mutuamente e statisticamente indipendenti, e la loro mescolanza è di natura lineare.

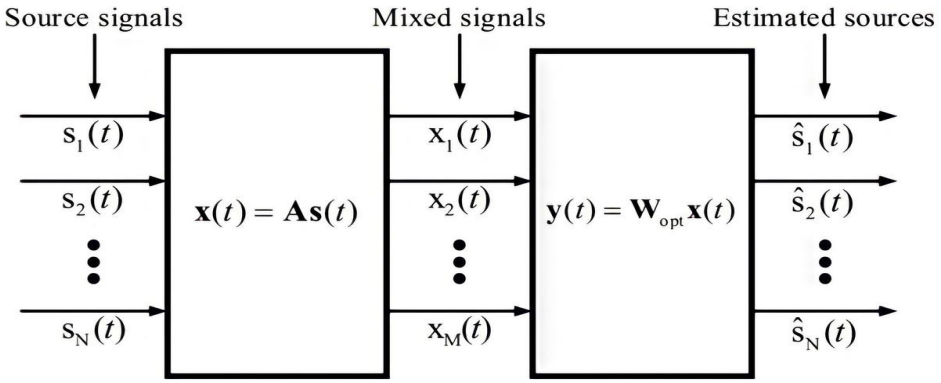


Figura 3: Illustrazione del funzionamento di ICA per problemi BSS | fonte: voce 4 bibliografia

Il processo di mescolanza può essere scritto come:

$$\mathbf{x}(t) = A\,\mathbf{s}(t)$$

dove A è una matrice sconosciuta di dimensioni $M \times N$, mentre le componenti del vettore $\mathbf{s}(t) = \left[s_1(t)\ s_2(t)\ \ldots\ s_N(t)\right]^T$ sono N segnali sconosciuti.

Il metodo ICA tenta di trovare una matrice W_{opt} di dimensione $N \times M$ tale che

$$\mathbf{y}(t) = W_{opt}\,\mathbf{x}(t)$$

dove $W_{opt} = \widehat{A}^{-1}$è un approssimazione della matrice originaria A e $\mathbf{y}(t) = \hat{s}(t)$ è un'approssimazione dei segnali sorgenti.

Capitolo 2

Modulazione del segnale

2.1 Introduzione

In questo capitolo saranno approfondite alcune delle caratteristiche principali degli elementi che si utilizzeranno e a cui si farà riferimento nell'esperimento proposto da questa tesi, così da favorire la comprensione dei processi su cui si è lavorato.

2.2 OFDM

Nell'ambito delle telecomunicazioni, ci si riferisce all'Orthogonal frequency-division multiplexing (OFDM, figura 1.1) per indicare una tecnica di trasmissione radio che consiste in una modulazione a multi-portante, in cui i sottocanali sono mutuamente ortogonali tra di loro e sui quali la trasmissione avviene in parallelo. Il vantaggio principale dell'OFDM è la possibilità di permettere la comunicazione anche in condizioni pessime del canale, inclusi i canali con una banda relativamente stretta.
Il segnale, prima di essere trasmesso, è rappresentato come un flusso di bit ed è sottoposto ad una modulazione QAM. I dati risultanti passano attraverso un convertitore serie/parallelo, il quale restituisce un flusso di M numeri complessi $X[0], X[1], \ldots, X[M-1]$, rappresentanti gli M simboli, ovvero le componenti discrete nel dominio della frequenza. Per ottenere i campioni nel dominio del tempo si effettua una DFT (Discrete Fourier Transform) inversa implementata tramite IFFT (Inverse Fast Fourier Transform), ottenendo così M campioni OFDM, la cui sequenza $a_k = \left[a_{k,-\frac{M}{2}}, \ldots, a_{k,\frac{M}{2}-1}\right]$ forma un simbolo OFDM.

$$a_{k,m} = \frac{1}{\sqrt{M}} \sum_{n=0}^{M-1} X[n] e^{\frac{j2\pi nm}{M}}, -\frac{M}{2} \leqslant m \leqslant \frac{M}{2} - 1$$

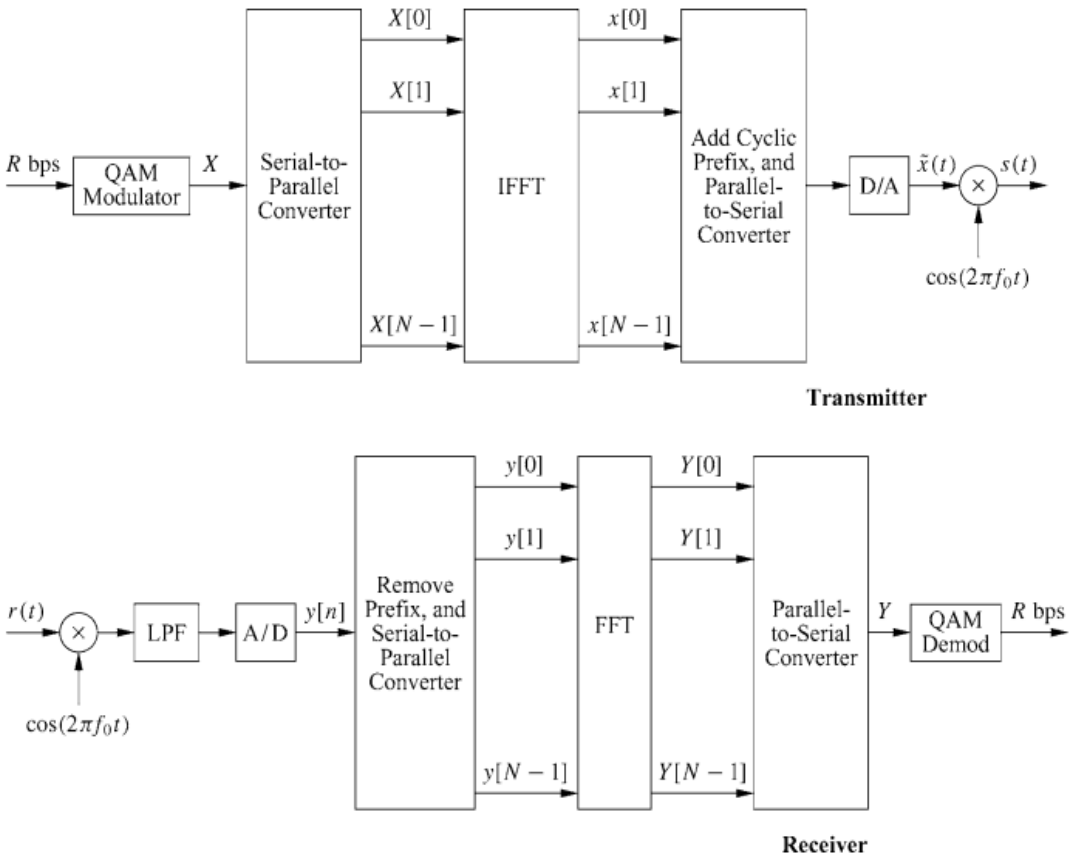


Figura 4: Schema a blocchi della modulazione OFDM | fonte: voce 5 bibliografia, p. 387

Viene poi aggiunto il prefisso ciclico, per ridurre l'interferenza inter-simbolo, e i campioni risultanti sono ordinati dal convertitore parallelo/serie e passati attraverso un convertitore digitale/analogico ottenendo così il k-esimo simbolo OFDM e il segnale completo $\tilde{x}(t)$ in banda base che, per essere trasmesso, viene traslato alla frequenza della portante:

$$\tilde{x}_k(t) = \sum_{m=-\frac{M}{2}}^{\frac{M}{2}-1} a_{k,m} e^{j2\pi mt/T}$$

Considerando la risposta impulsiva del canale $h(t)$ ed il rumore additivo $n(t)$, il segnale ricevuto è

$$y(t) = \tilde{x}(t) * h(t) + n(t)$$

dove l'operazione

$$\tilde{x}(t) * h(t) = \int \tilde{x}(\tau) h(t-\tau) d\tau = \int \tilde{x}(t-\tau) h(\tau) d\tau$$

è la convoluzione tra i due segnali, con τ variabile di tempo su cui si effettua l'operazione.
Il convertitore analogico/digitale rende nuovamente il segnale discreto a cui viene rimosso il prefisso ciclico ed i campioni restanti passano attraverso il convertitore serie/parallelo ed il modulo FFT. L'uscita viene convertita da parallelo a serie e passata attraverso un demodulatore QAM per ricostruire i simboli trasmessi.

Le M sottoportanti sono spaziate di $\Delta f = \frac{1}{T}$ Hz, dove T è il tempo di simbolo della OFDM senza considerare il prefisso ciclico, la cui durata è T_{CP}.

Nello specifico, i dispositivi che seguono lo standard IEEE 802.11ac, operano su una banda da 80 MHz ed utilizzano M = 256 sottocanali, $\Delta f = 312.5\ kHz$, $T = \frac{1}{\Delta f} = 3.2\mu s$, $T_{CP} = 0.8\mu s$ $\bar{T} = T + T_{CP} = 4\mu s$ quindi il segnale completo in banda base risulta essere:

$$\tilde{x}(t) = \sum_{k=0}^{K-1} \widetilde{x_k}(t) \xi(t - k\bar{T})$$

dove

$$\xi(t) = \begin{cases} 1 \text{ se t} \in \left[-T_{CP} - T\frac{M}{2} ; T\frac{M}{2} \right] \\ 0 \text{ altrimenti} \end{cases}$$

e il segnale trasmesso attraverso il canale Wi-Fi è ottenuto traslando $\tilde{x}(t)$ alla frequenza portante f_c

$$S_{TX} = e^{j2\pi f_c t} \tilde{x}(t)$$

A causa del fenomeno della propagazione multipath all'antenna ricevente vengono raccolte P copie del segnale: ogni copia è caratterizzata da una propria attenuazione $A_p(t)$ e da un proprio ritardo di propagazione $\tau_p(t)$.
Trascurando il rumore Gaussiano bianco il segnale ricevuto è:

$$s_{RX}(t) = \sum_{p=0}^{P-1} A_p(t)\, s_{TX}\left(t - \tau_p(t)\right)$$
$$= e^{j2\pi f_c t} \sum_{p=0}^{P-1} A_p(t)\, e^{-j2\pi f_c \tau_p(t)} \tilde{x}\left(t - \tau_p(t)\right)$$

E la sua rappresentazione in banda base è $y_s(t) = S_{RX}(t) e^{-j2\pi f_c t}$.

Al ricevitore, per raccogliere e decodificare l'informazione trasportata da un simbolo OFDM alla volta viene utilizzata una finestra rettangolare $[k\bar{T},\ k\bar{T}+T]$.

Senza perdere di generalità si assume $k=0$ e d'ora in avanti lo si omette dalle seguenti equazioni.

Il simbolo trasmesso è calcolato ricorrendo alla trasformata di Fourier del segnale all'interno della finestra di ricezione:

$$\hat{a}_m = \int_{\bar{T}}^{\bar{T}+T} y(t)\, e^{-\frac{j2\pi mt}{T}}\, dt$$

$$= \sum_{p=0}^{P-1} A_p(t)\, e^{-j2\pi f_c \tau_p(t)} \sum_{b=0}^{M-1} a_b e^{-\frac{j2\pi b\tau_p(t)}{T}} \times \int_{\bar{t}}^{\bar{T}+T} e^{\frac{j2\pi(b-m)t}{T}}\, dt$$

In tale equazione si considerano costanti l'attenuazione ed il ritardo sulla p-esima copia del segnale ricevuto, ponendo quindi $A_p(t) = A_p$ e $\tau_p(t) = \tau_p$ e ottenendo l'espressione

$$a_m T \sum_{p=0}^{P-1} A_p e^{-j2\pi\left(f_c + \frac{m}{T}\right)\tau_p}$$

dove la sommatoria nell'ultima linea rappresenta la risposta in frequenza del canale Wi-Fi,

$$H_m(n) = \sum_{p=0}^{P-1} A_p e^{-j2\pi\left(f_c + \frac{m}{T}\right)\tau_p(n)}$$

mentre *n* è la variabile di tempo discreto per cui si dispone dei campioni di $H_m(n)$.

2.3 Offset di fase nel canale Wi-Fi

Gli hardware artifacts (interferenze, imprecisioni di processo costruttivo, fisica dei conduttori) rendono la risposta in frequenza (CFR) raccolta dai router Wi-Fi leggermente deviata dal modello ottenuto nel paragrafo 1.3, introducendo offset di fase tra cui i seguenti:

- *carrier frequency offset (CFO),* dovuto alla differenza tra la frequenza portante del segnale trasmesso e di quello misurato al ricevitore. Il CFO è parzialmente corretto al ricevitore;
- *sampling frequency offset (SFO),* dovuto all'imperfetta sincronizzazione tra trasmettitore e ricevitore;
- *packet detection delay (PDD),* dovuto al tempo richiesto per recuperare i simboli modulati dal segnale ricevuto;
- *phase-locked loop phase offset (PPO),* dovuto all'errore di sincronizzazione tra la fase generata dal segnale periodico prodotto dal PLL (un circuito in grado di sincronizzare le fasi) e quello del segnale in ingresso al ricevitore;
- *phase ambiguity (PA),* dovuto alla differenza di fase (in multipli di π) tra le antenne, che in condizioni statiche dovrebbe rimanere costante.

Considerando questi contributi, l'espressione completa per la fase del percorso *p*-esimo del segnale ricevuto è:

$$\bar{\phi}_{p,m} = -2\pi\left(f_c+\frac{m}{T}\right)\tau_p+\phi_{CFO}-\frac{2\pi m\left(\tau_{SFO}+\tau_{PDD}\right)}{T}+\phi_{PPO}+\phi_{PA}$$

Si noti che mentre i contributi all'offset di fase dovuti ai termini CFO, SFO e PDD assumono sempre lo stesso valore per ogni antenna, le fasi relative al PLL ed al PA sono specifiche per ogni antenna.

Per la correzione degli offset di fase si può osservare che ogni percorso considerato nell'equazione finale della risposta in frequenza del canale Wi-Fi ottenuta nel paragrafo precedente è affetta dalla stessa differenza di fase $\phi_{offs,m} = \phi_{CFO}+\phi_{PPO}+\phi_{PA}-\frac{2\pi m\left(\tau_{SFO}+\tau_{PDD}\right)}{T}$

Quindi l'espressione definitiva della risposta in frequenza è:

$$\widehat{H}_m(n) = H_m(n)e^{\phi_{offs,m}}.$$

Capitolo 3

Indicazioni pratiche sull'utilizzo e procedimento

3.1 Introduzione

Nel presente capitolo seguiranno indicazioni su come effettuare la configurazione della rete e come utilizzare il software Nexmon.
Doveroso specificare che le istruzioni sono relative al chip bcm4366c0, utilizzato per questo lavoro, ed i comandi elencati da eseguire tramite terminale sono specifici per Linux Ubuntu. Il software Nexmon si considera già scaricato.

3.2 Nexmon

Nexmon è un firmware patching framework basato su C, attraverso il quale, per alcune tipologie di chip - tra cui bcm4366c0 di produzione Broadcom - è possibile l'estrazione del CSI; in particolare la struttura dati utilizzata per il suo salvataggio è un array 3D contenente la risposta in frequenza per ogni antenna (fino ad un massimo di quattro in trasmissione e ricezione) e per ogni sottoportante.
Relativamente ai segnali, è stata utilizzata la modulazione OFDM già precedentemente illustrata. Nexmon fa uso di 242 canali sui 256 disponibili definiti dallo standard WiFi 802.11ac, in particolare i sottocanali utili per la lettura dei valori della CFR sono compresi negli intervalli {-122, …, 2} e {2, …, 122}.

3.3 Set-up sperimentale

3.3.1 Introduzione set-up

Sono stati utilizzati due router WiFi Asus RT-AC86U, uno come trasmittente e l'altro come ricevente, il trasmittente opera con una sola antenna mentre il ricevente con quattro.
I due devices sono posizionati, come già introdotto nel primo capitolo, in due posizioni fisse all'interno di una stanza, tra i quali sono situati i soggetti da cui estrarre i parametri di interesse.

3.3.2 Correzione offset di fase relativi all'hardware

I due router sono dotati di tre antenne situate sulla parte esterna del dispositivo ed una interna, ciò che si è fatto è stato scollegare quella interna ed utilizzarla come addizionale esterna (tramite un particolare cavo), le quali si rendono disponibili ad un uso per la trasmissione/ricezione dei segnali necessari allo svolgimento dello specifico progetto.

Figura 5: Router ASUS RT-AC86U

La vera e propria acquisizione del CSI è preceduta dalla rimozione degli offset di fase dovuti alla circuiteria.

Durante la fase preliminare si connette quindi l'antenna del router tramite SMA (SubMiniature version A) connector ad un attenuatore (40 dB) collegato all'ingresso dello splitter.

Esso restituisce quattro copie del segnale in ingresso, ognuna con una potenza ripartita in base al numero di uscite. Gli SMA connector vengono poi collegati alle antenne del router ricevente.

Si acquisisce il CSI in modalità weird: sottraendo l'offset di fase misurato durante questa operazione alla fase misurata successivamente sul canale radio si corregge l'errore dovuto alla componente hardware.

Figura 6: Splitter, annessi cavi SMA e attenuatore in ingresso

3.3.3 Motivazioni di configurazione sperimentale

Per rendere più semplice la stima del tempo di arrivo e dell'angolo di incidenza dell'onda al ricevitore, è stato scelto di posizionare le antenne ad una distanza $d = \lambda/2$, dove λ è la lunghezza dell'onda radio emessa.

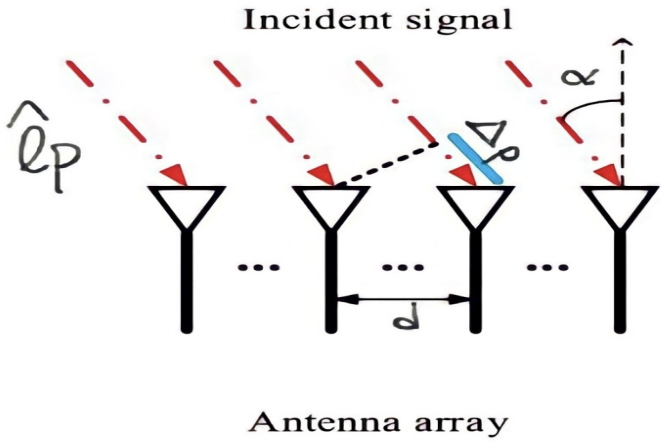


Figura 7: Configurazione antenne per la ricezione del segnale | fonte: voce 4 bibliografia

Con riferimento alle singole sottoportanti della modulazione OFDM, l'espressione del CSI al ricevitore può essere scritta come:

$$H_m(t) = \sum_{p=0}^{P-1} A_p e^{-j2\pi(f_c + m/T)\widehat{\tau_p}}$$

dove m è l'indice della sottoportante, p l'indice del multipath, P il numero di percorsi, A_p l'attenuazione, T il tempo di campionamento, $\widehat{\tau_p}$ il ritardo di propagazione, $f_c = 5775\,MHz$ la frequenza portante.

Assumendo che il trasmittente sia sufficientemente distante dal ricevente, è possibile sfruttare l'ipotesi che il trasmittente sia stato posto ad infinito, quindi è possibile considerare le onde in arrivo all'array di antenne tutte parallele tra di loro.

L'angolo di arrivo è indicato con α, mentre la distanza tra le antenne è indicata con d.

Denotando con l_p la lunghezza relativa al percorso del segnale ricevuto dalla prima antenna, la differenza di percorso per arrivare alle successive è pari a $k\,\Delta\,p$, dove $k = 0,1,2,3$ è l'indice di antenna,

mentre $\Delta p = d\,sen(\alpha)$ è la differenza di percorso tra un'antenna e la seguente.
Rappresentando con τ_p il ritardo di propagazione del segnale alla prima antenna, con $\widehat{l_p}$ la lunghezza di percorso per giungere sulla k-esima antenna, allora $\widehat{l_p} = l_p + k\Delta_p = c\tau_p + kdsen(\alpha)$, dove c è la velocità della luce.
Quindi il tempo di propagazione relativo alla k-esima antenna può essere riscritto come $\widehat{\tau_p} = \frac{\widehat{l_p}}{c} = \tau_p + \frac{kdsen(\alpha)}{c}$ e la risposta in frequenza sulla singola sottoportante $m$diventa

$$H_m(t) = \sum_{p=0}^{P-1} A_p e^{-j2\pi(f_c + m/T)\left(\tau_p + \frac{kdsen(\alpha)}{c}\right)}$$

dove osservando che il termine m/T è trascurabile ed esplicitando $f_c = c/\lambda$ si arriva alla relazione:

$$H_m(t) = \sum_{p=0}^{P-1} A_p e^{-j\pi(2f_c\tau_p + ksen(\alpha))}$$

3.3.4. Configurazione router Wi-Fi

Dopo aver collegato i router Asus RT-AC86U tra di loro ed al computer tramite cavo ethernet, portarsi nella directory del pc dove sono contenuti gli script di bash, lanciando il terminale nel relativo percorso eseguire:

1. reload.sh, che caricherà la firmware patch di nexmon all'interno dei dispositivi.
2. config.sh, che configura automaticamente le impostazioni di rete necessarie alla trasmissione wireless, tra cui indirizzo ip e client/server della connessione.
3. send_collect_new.sh, che apre la connessione, genera il traffico dati e imposta l'acquisizione dei pacchetti tramite tcpdump delle tracce generate da Nexmon; dopodiché Wireshark viene usato per dirottare i pacchetti pcap acquisiti dal router all'interno di una directory del pc.

3.4 Esecuzione dell'applicazione real-time

Per poter utilizzare gli script matlab forniti dal gruppo di ricerca Nexmon è necessario in via preliminare poter accedere al file unpack_float.c, il quale si occupa di filtrare il file pcap che contiene le informazioni sul canale. Per poter usufruire quindi di questo codice sorgente, è necessario usare il comando per la generazione del mex file, il quale rende possibile l'integrazione del codice C nell'ambiente di sviluppo matlab.

Per generare il mex file deve essere installato sul proprio dispositivo l'ambiente di sviluppo per la programmazione in C. In questo caso, sul dispositivo in uso è installato un compilatore GNU C Compiler (GCC), il quale rende possibile la creazione del mex file tramite il comando "mex unpack_float.c" posizionandosi nella giusta directory, ovvero quella in cui il main file del programma matlab va lanciato.

Capitolo 4

Codice Matlab e plot del CSI

4.1 Introduzione

L'obiettivo dell'applicazione sviluppata è la sequenziale acquisizione e visualizzazione run-time del file pcap durante la sua ricezione, acquisito per mezzo dello script di bash send_collect_new.sh già precedentemente citato.

Disponendo dei pacchetti pcap, è possibile estrarne il CSI tramite la funzione unpack_float.c, resa disponibile da Nexmon.

Nei seguenti paragrafi sarà presentato il codice prodotto, tra cui gli script e le funzioni matlab, annessi i plot relativi al modulo e alla fase della risposta in frequenza.

Per proteggere i diritti d'autore è stato scelto di presentare solamente la parte di codice risultante dal mio lavoro.

4.2 Save_Data

Questo script rende possibile reperire il file pcap all'interno della directory in cui è situato, inoltre è tramite il codice presente che si rende possibile l'acquisizione real-time: la chiamata alla funzione load80MHzstudio permette di ottenere le informazioni sul CSI, che viene successivamente salvato in un array 3D in una determinata posizione di memoria.

```
1
2   close all
3   clc
4   clear
5
6   % addpath("../functions")
7
8   %% configuration
9   BW = 80;                % bandwidth
10
11  routers_csi = string(3);
12  routers_csi_num = 3;
13
14  VSA='0003';
15  VSB='000';
16
17  subfolder_name = "161221";
18
19  d = dir(strcat("C:\Users\Antonio\Documents\MATLAB\TESI\traces\", subfolder_name));
20  mkdir(strcat("C:\Users\Antonio\Documents\MATLAB\TESI\mat_files\", subfolder_name));
21
22  FILEA = strcat("C:\Users\Antonio\Documents\MATLAB\TESI\traces\", subfolder_name, "\trace", routers_csi, ".pcap");
```

```
packets = 0;
processedPackets = 0;
loopNum = 1;
dim = 0; offset = 24;
file = dir(FILEA);
while dim ~= file.bytes
    dim = file.bytes;

    [~, ~, ~, ~, ~, ~, ~, csi_store, packetsA, ~, offset] = load80MHZstudio(FILEA, BW, VSA, offset);

    if (packets == 0)
        packets = length(csi_store);
        leftPackets = packets;
        K = length(csi_store{1,1}.core{1,1}.nss{1,1}.data);
        N = csi_store{1,1}.mask_to_process;
        csi_data = zeros(packets, K, N);
        toa_packets = packetsA;
    else
        leftPackets = length(csi_store);
        csi_data = padarray(csi_data, leftPackets, 'post');
        toa_packets = [toa_packets' packetsA']';
        processedPackets = packets;
        packets = packets + leftPackets;
    end


    for ii = 1:leftPackets
        for jj = 1:N
            csi_data(processedPackets+ii,:,jj) = csi_store{1,ii}.core{1,jj}.nss{1,1}.data;
        end
    end

    %visualizzazione sequenziale della risposta in frequenza
    %acquisita, in modulo e fase
    T = 0.004;                      %tempo di acquisizione per ogni frame [s]
    pkts_acq_time = [0:packets-1]*T; %vettore dei tempi di acquisizione
    subcarrier_plot(csi_data, pkts_acq_time, N, 100, loopNum);
    heatmap(csi_data, N, loopNum);

end

save(strcat("C:\Users\Antonio\Documents\MATLAB\TESI\mat files\", subfolder_name, "\csi data"), "csi data", "toa packets");
```

4.3 readpcap

Per raggiungere l'obiettivo dell'acquisizione real time e per migliorare l'efficienza, si ricorre all'istruzione fseek, già contenuta all'interno dell'ambiente di sviluppo MATLAB. Aggiungendo al pacchetto readpcap.m - reso disponibile da Nexmon – la funzione leftFrames e sfruttando l'offset in bytes su cui ci si porta alla fine di ogni acquisizione successiva del file pcap durante la sua generazione sul supporto di memoria, diviene possibile spostare la posizione da cui acquisire i rimanenti pacchetti. L'inizio della lettura del file .pcap comincia, secondo la funzione fromstart (contenuta nel file readpcap.m) dal byte 24; sfruttando questa informazione e guardando la dimensione di ogni singolo frame in byte, si può calcolare un offset per ottenere la nuova posizione per l'acquisizione dei bytes, e di conseguenza dei frames.

```
    function frames = leftFrames(obj, offset)
        i = 1;
        frames = cell(1);
        fseek(obj.fid, offset, -1);
        while true
            frame = obj.next();

            if isempty(frame)
                break;
            end

            frames{i} = frame;
            i = i + 1;
        end
    end
```

4.4 load80MHzstudio

La principale modifica apportata a questa funzione consiste nell'aggiunta sia in input che in output dell'offset per tenere traccia dei pacchetti già processati; inoltre è stato aggiunto un controllo sull'integrità dell'header per evitare errori in fase di letture del pcap.

```
k = 1;
n = length(p.leftFrames(offset));
fseek(p.fid, offset, -1);
while (k < n)
    if NICE == 1
      for jj = 1:length(output) + 1
          fprintf(1, '\b');
      end
    end
    charjj = mod(k, length(chars)) + 1;
    output = sprintf(' %.0f/100', k / n * 100);
    if NICE == 1
      fprintf(1, '%c%s', chars(charjj), output);
    end

    %Controllo integrità frames
    f = p.next();
    if (f.header.incl_len == 4*length(f.payload))
        k = k + 1;
    else
        fseek(p.fid, offset, -1);
        continue;
    end
```

4.5 CSI plot

Le funzioni subcarrier_plot e heatmap, generano rispettivamente i plot relativi alla sottoportante selezionata e appunto l'heatmap. In entrambi i casi è stata implementata la correzione degli offset di fasee per ogni successiva acquisizione, i plot vengono salvati nelle directory subcarrier_plot e heatmap_plot per potervi accedere in un secondo momento.

Sono state predisposte al plot solamente le sottoportanti che Nexmon usa per l'acquisizione dei valori.

4.5.1 Correzione offset di fase via software

In riferimento alla trattazione matematica presente nel capitolo 1 riguardo i percorsi statici e dinamici, si approfondirà qui la questione legata alla correzione degli offset.

Osservando che la perfetta sincronizzazione nel dominio del tempo tra le due ricetrasmittenti difficilmente viene conseguita con successo, per ottenere una migliore stima della fase può essere intrapreso mediante le seguenti considerazioni.

Indicando con $e^{-j\theta_{offset}}$ lo sfasamento dovuto ai termini SFO e CFO, ogni campione contenuto all'interno del CSI può essere rappresentato come:

$$H(f,t) = e^{-j\theta_{offset}}\left(H_s(f,t)+A(f,t)e^{-j2\pi d(t)/\lambda}\right) = e^{-j\theta_{offset}}\left(S(f)e^{-j\alpha(t)}+A(f,t)e^{-j2\pi d(t)/\lambda}\right)$$

dove $H_s(f,t)$ è la componente statica, $S(f,t)$ è l'attenuazione $\alpha(t)$ è uno sfasamento della componente statica e $A(f,t)e^{-j2\pi d(t)/\lambda}$è la componente dinamica.

Il termine di sfasamento $e^{-j\theta_{offset}}$è lo stesso per tutte le antenne, quindi prendendo come riferimento il CSI per una delle antenne e moltiplicandolo per il complesso coniugato del CSI delle rimanenti, si ottiene assumendo come riferimento la prima antenna:

$$\begin{aligned}
H_{cm}(f,t) &= H_1(f,t)\overline{H}_2(f,t) \\
&= \left(S_1(f,t)e^{-j\alpha_1(t)}+A_1(f,t)(f)e^{-j2\pi d_1(t)/\lambda}\right)\left(S_2(f,t)e^{j\alpha_2(t)}+A_2(f,t)e^{j2\pi d_2(t)/\lambda}\right) \\
&= \underbrace{S_1S_2e^{-j(\alpha_1-\alpha_2)}}_{1}+\underbrace{S_2e^{j\alpha_2}A_1(f)e^{(-j2\pi d_1/\lambda)}}_{2} \\
&+ \underbrace{S_1e^{-j\alpha_1}A_2(f)e^{j2\pi d_2/\lambda}}_{3}+\underbrace{A_1(f)A_2(f)e^{-j2\pi(d_1-d_2)/\lambda}}_{4}
\end{aligned}$$

dove $H_{cm}(f,t)$è il risultato della moltiplicazione coniugata, $H_1(f,t)$ è il CSI della prima antenna e $\overline{H}_2(f,t)$ è il coniugato del CSI della seconda antenna.

Considerando che $H_{cm}(f,t)$ è composta da 4 componenti, il prodotto delle componenti statiche (1) può essere considerato costante e il prodotto di quelle dinamiche (4) è un valore trascurabile.

I termini rimanenti, 2 e 3, sono due prodotti della componente statica di un'antenna e della componente dinamica di un'altra: esse contengono le informazioni di interesse sulla respirazione.

Un approccio simile è stato adottato anche nei seguenti paragrafi.

4.5.2 subcarrier_plot

La funzione permette la stampa dei grafici per modulo e fase nel dominio del tempo, in base agli input è possibile selezionare per quale antenna e quale sottoportante effettuare il plot.

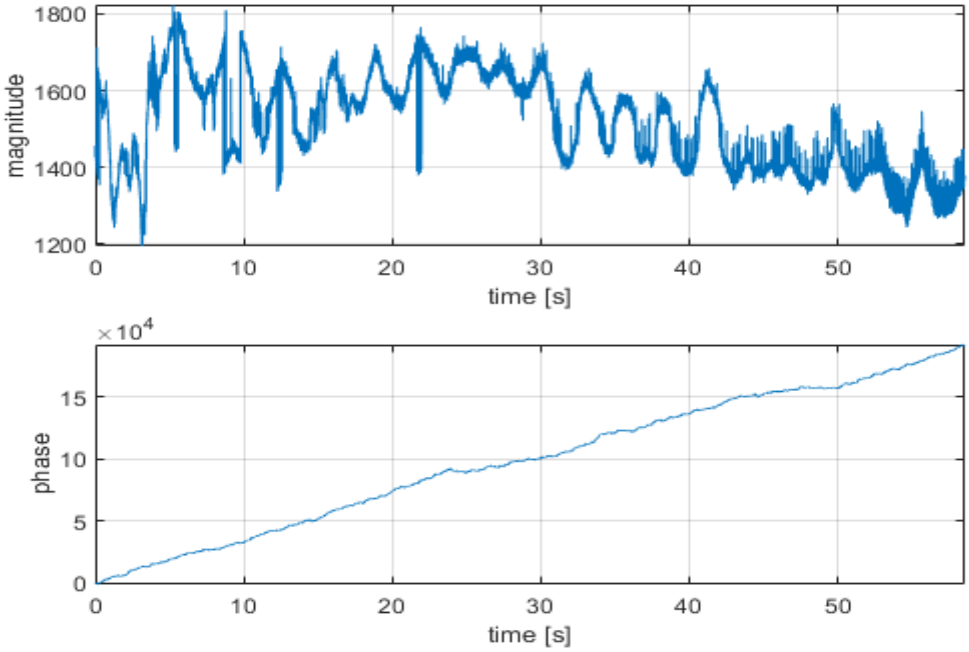
Per rendere migliore la visualizzazione della fase è stato applicato l'unwrap lungo la dimensione dei campioni della singola sottoportante.

Segue il codice:

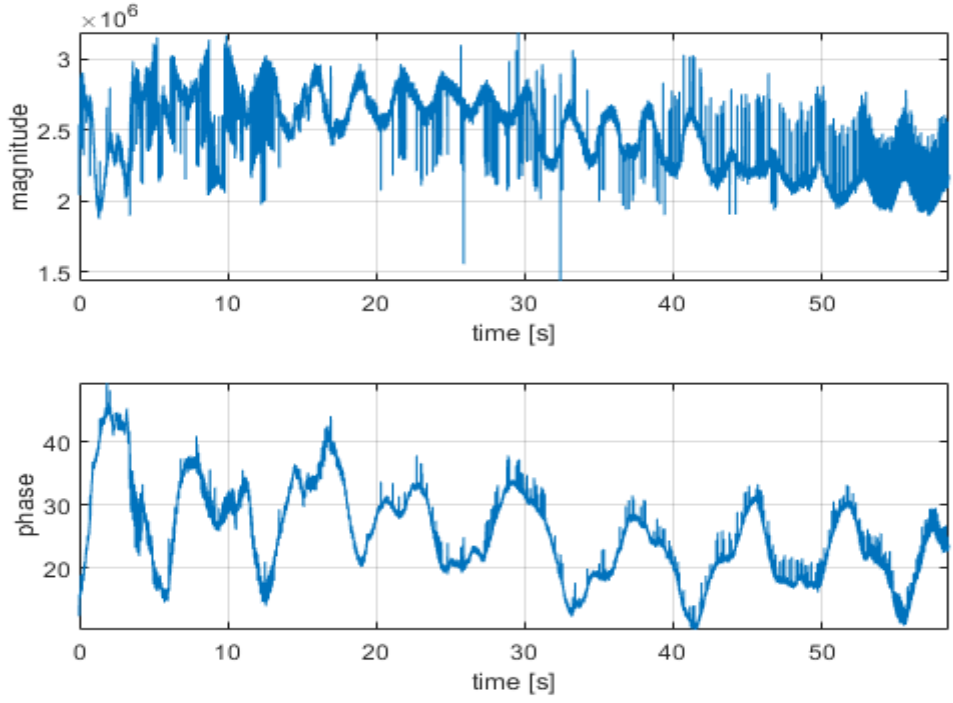
```
function subcarrier_plot(csi_data, pkts_acq_time, Nant, Nsubcarrier, loopNum)
    mkdir(strcat('C:\Users\Antonio\Documents\MATLAB\TESI\subcarrier_plot\loop_num_', num2str(loopNum)));
    h = figure();
    for antenna = 1:Nant
        %risposta in frequenza antenna
        W = csi_data(:,[7:127, 131:251],antenna);

        %plot modulo
        subplot(2,1,1);
        MAGNITUDE = abs(W(:, Nsubcarrier));
        plot(pkts_acq_time, MAGNITUDE);
        grid on;
        xlabel('time [s]');
        ylabel('magnitude');
        xlim([0 pkts_acq_time(end)]);
        ylim([min(MAGNITUDE) max(MAGNITUDE)]);

        %plot fase
        subplot(2,1,2);
```

```
        subplot(2,1,2);
        PHASE = rad2deg(unwrap(angle(W(:, Nsubcarrier))));
        plot(pkts_acq_time, PHASE);
        grid on;
        xlabel('time [s]');
        ylabel('phase');
        xlim([0 pkts_acq_time(end)]);
        ylim([min(PHASE) max(PHASE)]);

        savefig(h, strcat('C:\Users\Antonio\Documents\MATLAB\TESI\subcarrier_plot\loop_num_', ...
            num2str(loopNum),'\magnitude_phase_', num2str(antenna), '.fig'));
    end

    %correzione fase: eliminazione offset di fase
    refAntW = conj(csi_data(:,[7:127, 131:251],4)); %CFR dell'antenna di riferimento per la correzione
    for antenna = 1:Nant-1
        corrOffsetW = refAntW.*(csi_data(:,[7:127, 131:251],antenna));

        %plot modulo
        subplot(2,1,1);
        corrMAGNITUDE = abs(corrOffsetW(:, Nsubcarrier));
        plot(pkts_acq_time, corrMAGNITUDE);
        grid on;
        xlabel('time [s]');
        ylabel('magnitude');
        xlim([0 pkts_acq_time(end)]);
        ylim([min(corrMAGNITUDE) max(corrMAGNITUDE)]);

        %plot fase
        subplot(2,1,2);
        corrPHASE = rad2deg(unwrap(angle(corrOffsetW(:,Nsubcarrier))));
        plot(pkts_acq_time, corrPHASE);
        grid on;
        xlabel('time [s]');
        ylabel('phase');
        xlim([0 pkts_acq_time(end)]);
        ylim([min(corrPHASE) max(corrPHASE)]);

        savefig(h, strcat('C:\Users\Antonio\Documents\MATLAB\TESI\subcarrier_plot\loop_num_', ...
            num2str(loopNum),'\magnitude_phase_correct', num2str(antenna), '-4.fig'));
    end
    close(h);
end
```

Di seguito è mostrato il plot del CSI, per l'antenna 1 e per la sottoportante 100:



Dopo la correzione sulla fase assumendo come riferimento l'antenna 4:



4.5.3 heatmap

La funzione permette la stampa dei grafici del modulo e della fase per sottoportanti e campioni, con la possibilità di selezionare il numero di antenna.

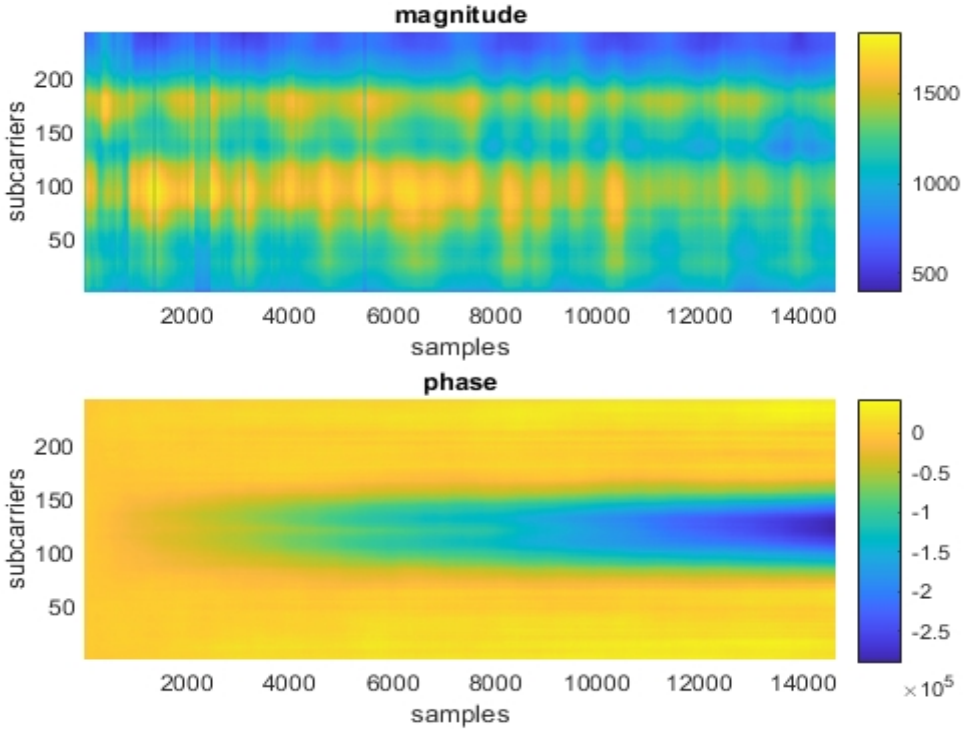
Per rendere migliore la visualizzazione della fase è stato applicato l'unwrap lungo la dimensione delle sottoportanti.

Segue il codice:

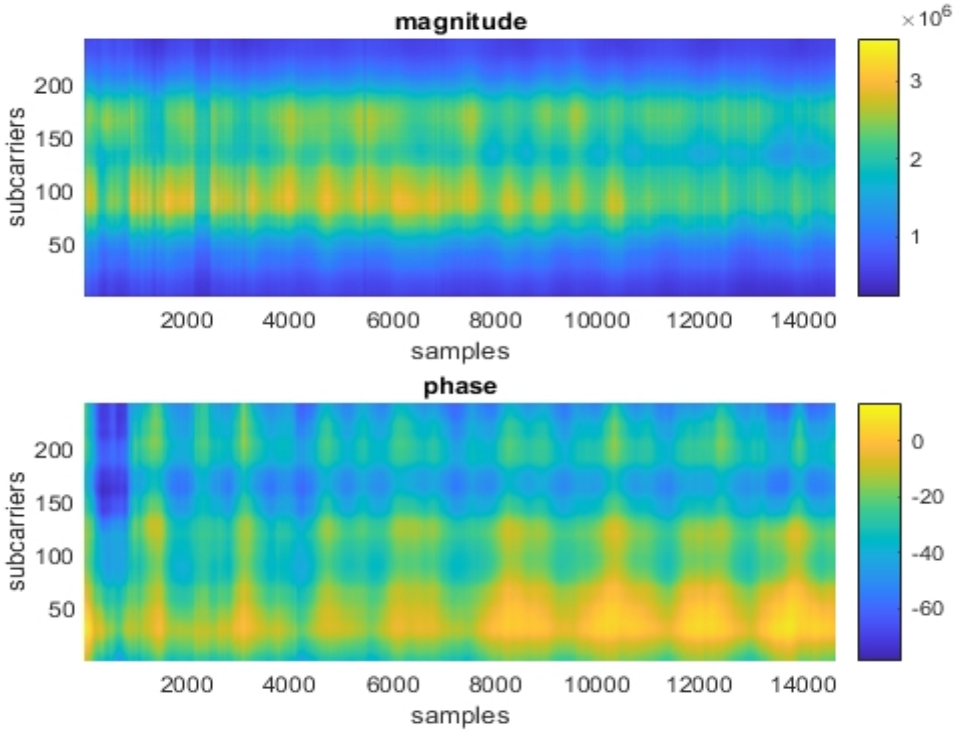
```
function heatmap(csi_data, Nant, loopNum)
    mkdir(strcat('C:\Users\Antonio\Documents\MATLAB\TESI\heatmap_plot\loop_num_', num2str(loopNum)));
    h = figure();
    for antenna = 1:Nant
        %trasposta della risposta in frequenza antenna
        W = csi_data(:,[7:127, 131:251],antenna)';

        %heatmap modulo
        subplot(2,1,1);
        m = pcolor(abs(W));
        set(m,'EdgeColor','none');
        grid on;
        title('magnitude');
        xlabel('samples');
        ylabel('subcarriers');
        colorbar;

        %heatmap fase
        subplot(2,1,2);
```

```
        subplot(2,1,2);
        p = pcolor(rad2deg(unwrap(angle(W), [], 2)));
        set(p,'EdgeColor','none');
        grid on;
        title('phase');
        xlabel('samples');
        ylabel('subcarriers');
        colorbar;

        savefig(h, strcat('C:\Users\Antonio\Documents\MATLAB\TESI\heatmap_plot\loop_num_', ...
            num2str(loopNum),'\heatmap_magnitude_phase', num2str(antenna), '.fig'));
    end

    %correzione fase: eliminazione offset di fase
    refAntW = conj(csi_data(:,[7:127, 131:251],4)); %CFR dell'antenna di riferimento per la correzione
    for antenna = 1:Nant-1
        corrOffsetW = (refAntW.*(csi_data(:,[7:127, 131:251],antenna)))';

        %heatmap modulo
        subplot(2,1,1);
        m = pcolor(abs(corrOffsetW));
        set(m,'EdgeColor','none');
        grid on;
        title('magnitude');
        xlabel('samples');
        ylabel('subcarriers');
        colorbar;

        %heatmap fase
        subplot(2,1,2);
        p = pcolor(rad2deg(unwrap(angle(corrOffsetW), [], 2)));
        set(p,'EdgeColor','none');
        grid on;
        title('phase');
        xlabel('samples');
        ylabel('subcarriers');
        colorbar;

        savefig(h, strcat('C:\Users\Antonio\Documents\MATLAB\TESI\heatmap_plot\loop_num_', ...
            num2str(loopNum),'\heatmap_magnitude_phase_correct', num2str(antenna), '-4.fig'));
    end
    close(h);
end
```

Di seguito è mostrato l'heatmap per l'antenna 1:



Dopo la correzione sulla fase assumendo come riferimento l'antenna 4:



Conclusione

Come enunciato nel capitolo introduttivo, nello sviluppo di questa tesi sono stati coinvolti gli script forniti da Nexmon insieme alla firmware patch per i router Wi-Fi e sono state apportate alcune modifiche al codice sorgente di partenza al fine di migliorare le prestazioni di acquisizione dei dati nell'estrazione della frequenza respiratoria; sono stati usati diversi strumenti hardware per la rimozione degli offset di fase.
Inoltre sono stati prodotti degli script per la visualizzazione real-time di modulo e fase dei dati contenuti all'interno del CSI.
La prossima sfida sarà quella di sviluppare dei metodi più accurati per l'estrazione del battito cardiaco ed altri parametri di interesse in modo da disporre di un tool completo per arrivare a monitorare l'insieme più grande possibile di parametri che si possono rilevare grazie alla conoscenza del CSI e grazie all'uso di hardware potenzialmente accessibili alla maggior parte della popolazione.

Bibliografia

[1] Zeng Y., Wu D., Gao R., Gu T.,Zhang D., *FullBreathe: Full Human Respiration Detection Exploiting Complementarity of CSI Phase and Amplitude of WiFi Signals, S*eptember 2018

[2] Adib F., Mao H., Kabelac Z., Katabi D., Miller R.C., *Smart Homes that Monitor Breathing and Heart Rate,* Massachusetts Institute of Technology, April 2015

[3] Saad Al-Ahmadi A., Abd Rahman T., *One Stage Indoor Location Determination Systems,* October 2012

[4] Zeng Y.,Wu D., Xiong J., Liu J., Liu Z., Zhang D., *MultiSense: Enabling Multi-person Respiration Sensing with Commodity WiFi,* September 2020

[5] Meneghello F., Garlisi D., Dal Fabbro N., Tinnirello I., Rossi M., *Environment and Person Independent Activity Recognition with a Commodity IEEE 802.11ac Access Point,* 17 marzo 2021

[6] Goldsmith A., *Wireless Communications,* Cambridge University Press, 2005

[7] Schulz M., Wegemer D., Hollick M., *Nexmon: The C-Based Firmware Patching Framework*, 2017; available online: https://nexmon.org

[8] Secure Mobile Networking Lab (SEEMOO); Multi-mechanisms Adaptation forthe Future Internet (MAKI); LOEWE centre emergenCITY; Technische Universität Darmstadt; University of Brescia, *Nexmon Channel State Information Extractor*, GitHub, 2021

[9] RT-AC86U Website; available: https://www.asus.com/it/Networking-IoT-Servers/WiFi-Routers/ASUS-WiFi-Routers/RT-AC86U

[10] Ward L., Rhode & Schwarz, *802.11ac Technology Introduction: White Paper,* March 2012

[11] Xiong J., Jamieson K., University College London, *ArrayTrack: A Fine-Grained Indoor Location System,* 2013