



UNIVERSITÀ
DEGLI STUDI
DI PADOVA

DIPARTIMENTO
DI INGEGNERIA
DELL'INFORMAZIONE

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

CORSO DI LAUREA IN INGEGNERIA INFORMATICA

**POSIZIONAMENTO MEDIANTE BLUETOOTH LOW
ENERGY**

Relatore:

Prof. Stefano Tomasin

Laureando:

Alessandro Leonardi

Co-Relatore:

Laura Crosara

ANNO ACCADEMICO: 2022/2023

Data di laurea: 21 Marzo 2023

Indice

1	Introduzione	1
2	Stato dell'arte	3
2.1	Contestualizzazione	3
2.1.1	Bluetooth Low Energy	3
2.2	Revisione della letteratura	4
2.2.1	RSSI e ToF per misurare la distanza	4
2.2.2	RSSI per il calcolo del posizionamento	7
2.3	Scenari considerati	10
3	Metodo Di Sperimentazione	11
3.1	ASUS TinkerBoard S	11
3.2	Configurazione	11
3.2.1	Librerie e codice utilizzati per le misurazioni	13
3.2.2	Librerie e codice utilizzati per il calcolo della distanza	15
3.2.3	Librerie e codice utilizzati per il calcolo del posizionamento	15
4	Risultati Sperimentali	17
4.1	Calcolo della distanza	17
4.1.1	Acquisizione dei dati	17
4.1.2	Elaborazione dei dati	19
4.1.3	Fitting dei dati e calcolo della distanza	20
4.2	Calcolo del posizionamento	25
4.2.1	Posizionamento ambiente chiuso	25

4.2.2	Posizionamento ambiente aperto	26
4.2.3	Risultati del posizionamento	27
5	Conclusioni	29
	Bibliografia	32

Sommario

Il *Bluetooth low energy* (BLE) è una tecnologia di telecomunicazioni molto adatta alle reti di sensori per l'Internet delle Cose, in quanto è in grado di trasmettere informazioni con un limitato consumo di potenza. In questo lavoro di tesi si utilizza il *received signal strength indicator* (RSSI), ovvero la potenza di segnale ricevuta, ottenuta dalle API del BLE, per ottenere la posizione di un dispositivo. Si presentano i metodi utilizzati per ottenere i valori necessari per il calcolo della distanza e del posizionamento. Nella parte sperimentale si mostrano le criticità del RSSI, in particolare l'errore nella stima della distanza tra due dispositivi, derivante dalla varianza dei valori raccolti. Si procede determinando la posizione di un dispositivo mediante l'algoritmo di trilaterazione. Le misure sono state svolte in condizioni di Line of Sight (LoS), sia in ambiente aperto che in ambiente chiuso.

Capitolo 1

Introduzione

L'obiettivo di questo lavoro di tesi è l'ottenimento della posizione di un dato sensore posto in un terreno agricolo mediante l'utilizzo del *Bluetooth low energy* (BLE), una tecnologia di comunicazione che si basa sul Bluetooth classico, ma è più efficiente dal punto di vista energetico. Molti studi si stanno svolgendo sul BLE, in quanto la richiesta di sistemi privi di cavi, con consumi di energia ridotti e bassi costi è in aumento nel campo dell'Internet delle Cose (Internet of Things, IOT). Il settore di riferimento per questo elaborato è l'agricoltura di precisione, una metodologia agronomica che utilizza tecnologie, come il GPS, i sensori e i calcolatori per analizzare e gestire in modo mirato e preciso ogni fase del processo produttivo agricolo. In questo scenario non è spesso possibile o conveniente posizionare sistemi cablati o energivori, in quanto la fornitura di corrente elettrica e la copertura di rete internet potrebbe essere insufficiente o di difficile reperibilità. Si opta quindi per sistemi di reti wireless a basso costo e con bassi requisiti energetici.

Il numero di sensori ed i campi di applicazione di questi nel settore agricolo sono in forte aumento. Essi possono essere posti sia in campo aperto che all'interno delle serre e coinvolgono molte fasi di sviluppo della pianta: l'analisi del suolo, la semina, l'irrigazione, la fertirrigazione, l'uso di pesticidi e il monitoraggio della salute della

pianta.

Nel settore agricolo i dati raccolti possono variare da un punto all'altro di un terreno e può essere utile la possibilità di collocare i sensori senza doversi preoccupare di registrare le posizioni nelle quali vengono posti. L'obiettivo di questo lavoro di tesi è quello di fornire la posizione di un dato dispositivo in una rete di sensori mediante la raccolta di misure di *received signal strength indicator* (RSSI). Così facendo con un unico modulo Bluetooth si possono sia scambiare i dati raccolti, che determinare la posizione alla quale sono stati raccolti.

Un'altra applicazione del posizionamento mediante BLE è il tracciamento in tempo reale di un mezzo all'interno di una rete di sensori. Se nella macchina agricola è presente un modulo Bluetooth, comunicando con i dispositivi a terra si può determinare la posizione del mezzo in tempo reale.

Nel Capitolo 2 viene riassunto lo stato dell'arte riguardo il BLE, in particolare gli studi condotti per ottenere la distanza tra due dispositivi dall'RSSI, la consistenza dei risultati e la differenza tra questa misura e il Time of Flight. Nella stessa sezione si procede con la revisione del calcolo del posizionamento, effettuato utilizzando l'algoritmo di trilaterazione. Infine vengono presentati gli scenari considerati per questo lavoro di tesi. Nel Capitolo 3 viene presentato il modo in cui è stata condotta la fase sperimentale di questo elaborato. Si parte con la presentazione dei dispositivi utilizzati, proseguendo con la configurazione e il codice di programmazione utilizzato per raccogliere le misurazioni. Nel Capitolo 4 vengono presentati i risultati ottenuti dalla fase sperimentale in entrambi gli scenari considerati. Vengono descritti prima quelli riguardanti la fase di calcolo della distanza, poi quelli per ottenuti dal calcolo del posizionamento. Nel Capitolo 5 vengono infine presentate le conclusioni e i futuri sviluppi che possono susseguirsi a questo lavoro di tesi.

Capitolo 2

Stato dell'arte

2.1 Contestualizzazione

L'utilizzo di sistemi di controllo e gestione nel settore agricolo spesso non trova appoggio in terreni con sufficiente copertura di sistemi elettrici. Sensori sparsi per un campo richiedono l'utilizzo di batterie che possono essere caricate da piccoli pannelli solari. Questo permette inoltre di spostare i sensori, in caso di necessità, in modo agevole. In questo scenario diventa importante riuscire a localizzare in tempo reale questi dispositivi, per associare la misura effettuata (ad esempio, temperatura e umidità) alla posizione in cui viene presa.

Una tecnologia a basso consumo di energia e in grado sia di trasmettere dati rilevati dai sensori, che di permettere la localizzazione è il BLE.

2.1.1 Bluetooth Low Energy

Il BLE è una tecnologia progettata per avere le funzionalità del Bluetooth con un minore consumo di energia. La maggiore efficienza energetica ha cambiato il modo di utilizzare questa tecnologia, e il modo in cui i dispositivi che la supportano

possono essere utilizzati. Il BLE si basa sulla modifica proposta da Nokia [1], in grado di ridurre il consumo di energia, chiamata Bluetooth low-end extension.

Nel 2010 questa tecnologia venne introdotta ufficialmente dal *Bluetooth special interest group* (BSIG) con il nome di BLE. Il BLE trasmette utilizzando 40 canali ad una frequenza di 2.4 GHz, ad una potenza più bassa che varia da 0.5 mw a 10 mw rispetto alla versione classica trasmette con una potenza da 1mW a 100 mW. Questo comporta una minore copertura e una maggiore influenza alle interferenze. Con il Bluetooth classico si potevano realizzare solo reti di tipo punto a punto, in quanto un dispositivo poteva comunicare con un solo altro dispositivo, mentre con il BLE ogni dispositivo può comunicare con un alto numero di dispositivi. Questo permette la progettazione di diversi tipi di reti come punto-punto, broadcast e di tipo mesh (si veda la Figura 2.1).

Il BLE è così diventato uno dei sistemi di comunicazioni più utilizzati nell'ambito dell'Internet delle cose (Internet of Things, IOT) [2]. Alcune delle applicazioni di studio di questa tecnologia sono il trasferimento di dati, i servizi di localizzazione e le reti di dispositivi interconnessi. La localizzazione viene intesa sia come calcolo di prossimità, sia come calcolo della posizione del dispositivo [3].

2.2 Revisione della letteratura

2.2.1 RSSI e ToF per misurare la distanza

Molti studi si sono susseguiti negli ultimi anni sullo studio della localizzazione mediante l'uso di reti di sensori wireless, sia in ambienti chiusi che all'aperto. Tra i metodi più studiati ci sono il *time of flight* (ToF) e il RSSI [4]. Il ToF è la misura del tempo di propagazione del segnale radio dal trasmettitore al ricevitore. Considerando che i segnali radio viaggiano ad una velocità nota (prossima a quella della luce nel vuoto), è possibile calcolare la distanza percorsa dato il tempo di

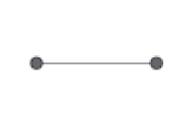
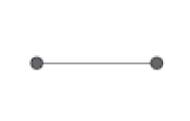
	audio streaming	data transfer	location services	device networks
solution				
topology	 point-to-point	 point-to-point	 broadcast	 mesh
radio	Bluetooth BR/EDR	Bluetooth Low Energy		

Figura 2.1: Bluetooth vs BLE [3]

viaggio (ToF) con la formula:

$$d = \frac{c \text{ToF}}{2}, \quad (2.1)$$

dove c è la velocità di propagazione delle onde radio.

Il ToF non è ottenibile facilmente, infatti bisogna sviluppare un programma che sia in grado di scrivere nel pacchetto del trasmettitore il tempo in cui viene inviato e il ricevente deve istantaneamente sottrarre il tempo in cui l'ha ricevuto, con il tempo scritto nel pacchetto. Si tratta di un calcolo complesso nella scala dei nanosecondi, deve essere quindi accompagnato da un hardware in grado di realizzare questa procedura nel minor tempo possibile per avere una misura accurata del tempo e i due dispositivi devono essere accuratamente sincronizzati prima della trasmissione. La tecnologia per permettere di ottenere una misura del ToF accurata non è solitamente presente nei dispositivi che integrano il BLE. Serve quindi un hardware dedicato e questo rende la soluzione più costosa. Si tratta comunque di una valida risorsa per ottenere un'approssimazione della distanza.

Il RSSI indica la potenza di segnale ricevuto, è facilmente ottenibile utilizzando le API del Bluetooth in quanto viene rilevato automaticamente durante la scansione dei dispositivi. La potenza del segnale ricevuto non è una metrica molto affidabile in quanto, soprattutto in luoghi chiusi, bisogna tenere in considerazione errori causati da multi-path propagation, riflessione e dissolvenza. Inoltre ci sono anche caratteristiche fisiche che possono avere un impatto sui risultati come: caratteristiche fisiche di antenne e il loro orientamento, variazione di potenza di trasmissione, variazione della frequenza di trasmissione e ambiente in cui vengono disposti [5]. Interpretando in modo corretto i valori ottenuti, ad esempio, prendendo la moda dei valori [6], si può ottenere una stima abbastanza precisa della distanza. Considerando la potenza di ricezione del segnale a distanza d_0 , denominata P_0 , è possibile calcolare la distanza percorsa dato l'RSSI con la formula

$$d = d_0 10^{\frac{P_0 - \text{RSSI}}{10n}}, \quad (2.2)$$

dove n è il *path loss exponent*, mentre le misure di P_0 e RSSI sono in decibel.

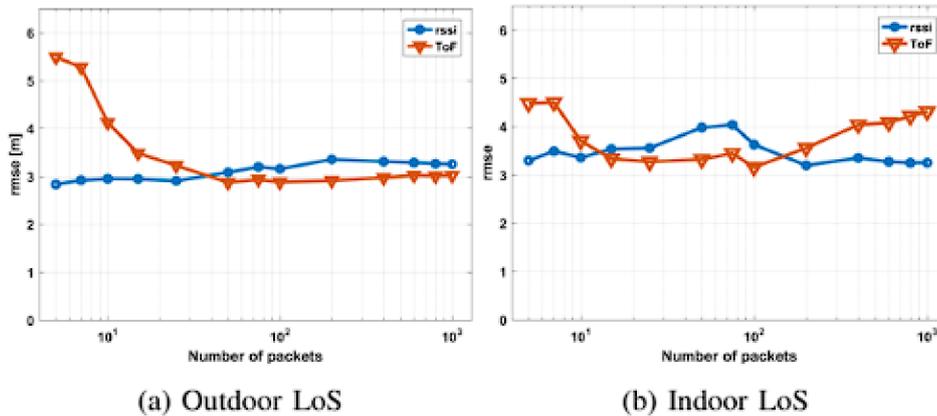


Figura 2.2: Errore quadratico medio (Root Mean Square Error, RMSE) con RSSI e ToF sia all'aperto che al chiuso in condizioni di LoS [4].

Il campo di applicazione delle due misure è diverso, in quanto nel caso del ToF

per ottenere un valore che può essere ritenuto corretto bisogna raccogliere almeno 100 misurazioni, mentre, se si utilizza l’RSSI, il numero di misurazioni non va ad incidere sul valore della radice dell’errore quadratico medio (*root mean square error*, RMSE), come si nota dalla Figura 2.2. Questo rende il ToF meno adatto per applicazioni in tempo reale. Inoltre, per misurare l’RSSI, non serve che i dispositivi siano sincronizzati a differenza del ToF, rendendo quindi difficile l’applicazione di quest’ultimo in scenari con molti dispositivi.

2.2.2 RSSI per il calcolo del posizionamento

Per la determinazione del posizionamento, in questo elaborato si utilizza l’RSSI. Questa soluzione trova già applicazione in ambienti chiusi, come magazzini, ristoranti o case [7][8]. Vengono disposti dei dispositivi indicati come punti di accesso sparsi per l’edificio. Questi rilevano il segnale emesso da altri apparecchi, elaborano i dati ottenuti e applicando regole di trilaterazione (Figura 2.3) con i valori di RSSI ottenuti (Figura 2.2).

Date tre posizioni note (x_1, y_1) , (x_2, y_2) , (x_3, y_3) , in ciascuna è posto un dispositivo in grado di misurare l’RSSI. La componente sull’asse z non viene considerata poiché assumiamo che i dispositivi si trovino tutti alla stessa altezza. Date queste tre posizioni, vogliamo stimare le coordinate $X = (x, y)$ di un quarto dispositivo, posto a distanza d_1 , d_2 , d_3 dai tre punti noti, dotato di trasmettitore bluetooth. La posizione viene calcolata tramite un procedimento iterativo partendo da una posizione di riferimento $X_0 = (x_0, y_0)$. Con l’equazione (2.3) viene mostrato un modello di range, dove ρ_i è lo pseudo range e v rappresenta il rumore stocastico:

$$d_i = \rho_i + v = \sqrt{(x_i - x)^2 + (y_i - y)^2} + v. \quad (2.3)$$

Siccome l’equazione (2.3) è una funzione non lineare, la espandiamo con la serie di

Taylor mantenendo solo i valori lineari. Si ottiene quindi

$$d_i - \rho_{0,i} = \frac{x_0 - x_i}{\rho_{0,i}} \Delta x + \frac{y_0 - y_i}{\rho_{0,i}} \Delta y, \quad (2.4)$$

dove $\rho_{0,i} = \sqrt{(x_i - x_0)^2 + (y_i - y_0)^2}$.

Successivamente, si riprende ciò che è stato spiegato nella Sezione 2.2.1, ovvero che i valori di RSSI sono in relazione con la distanza come definisce l'equazione (2.2), in cui i parametri P_0 , d_0 e n possono essere ottenuti, ad esempio, mediante un fit sui dati sperimentali. Si procede, quindi, con la definizione dell'algoritmo iterativo di trilaterazione.

$$V = BX - l, \quad l = \begin{bmatrix} \hat{d}_1 - \rho_{0,1} \\ \hat{d}_2 - \rho_{0,2} \\ \hat{d}_3 - \rho_{0,3} \end{bmatrix}, \quad X = \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix}, \quad B = \begin{bmatrix} \frac{x_0 - x_1}{\rho_{0,1}}, & \frac{y_0 - y_1}{\rho_{0,1}} \\ \frac{x_0 - x_2}{\rho_{0,2}}, & \frac{y_0 - y_2}{\rho_{0,2}} \\ \frac{x_0 - x_3}{\rho_{0,3}}, & \frac{y_0 - y_3}{\rho_{0,3}} \end{bmatrix}, \quad X_0 = \begin{bmatrix} x_0 \\ y_0 \end{bmatrix}, \quad (2.5)$$

dove (x_i, y_i) si riferiscono alle coordinate dei dispositivi, \hat{d}_i rappresenta la distanza misurata tra la Tinker Board S la quarta e la posizione attuale. Successivamente, il vettore delle coordinate residue, indicato con \hat{X} , viene calcolato come

$$\hat{X} = (B^T B)^{-1} B^T l \quad (2.6)$$

Dopo aver ottenuto le coordinate residue, si può derivare la posizione aggiungendo \hat{X} ai valori iniziali

$$\begin{bmatrix} x \\ y \end{bmatrix} = X_0 + \hat{X} \quad (2.7)$$

Questa procedura viene ripetuta finchè, ad esempio, la norma del vettore delle

coordinate residue \hat{X} è inferiore ad una soglia fissata.

Molteplici sono i casi studio di posizionamento con il BLE, si possono dividere in due macro aree, soluzioni di prossimità e soluzioni di posizionamento. Le prime vengono applicate ad esempio per trovare un oggetto in un luogo, oppure ricevere informazioni riguardo un punto di interesse in cui ci troviamo. Nel secondo caso invece si ha la localizzazione in tempo reale, ovvero il tracciamento, e la realizzazione di percorsi per raggiungere una determinata posizione in un certo ambiente. Il range stimato di correttezza nel quale si possono trovare misure più o meno precise è da 1 ai 10 m.

Una novità importante del BLE è l'angolo di arrivo del segnale. Infatti, dal Bluetooth 5.1 i dispositivi che supportano questa tecnologia avranno la capacità di determinare la direzione da dove il segnale viene trasmesso. In questo elaborato vengono utilizzati dispositivi Bluetooth 4.0, quindi non si considerano le nuove funzionalità aggiunte alle successive versioni di Bluetooth. I continui sviluppi di questa tecnologia ne permettono il miglioramento per quanto riguarda il posizionamento. Infatti, utilizzando le nuove funzionalità si è in grado di ottenere un tracciamento in tempo reale con una precisione nell'ordine dei centimetri [3].

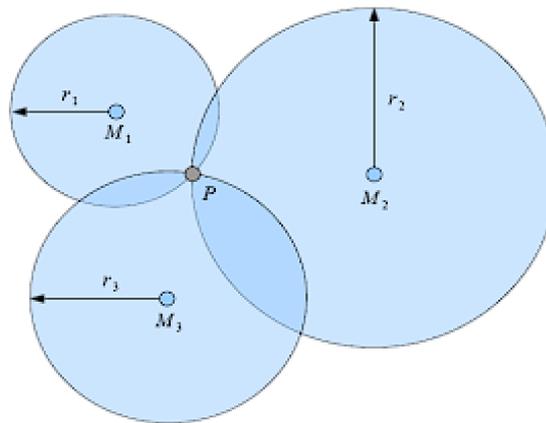


Figura 2.3: Trilaterazione [5]

2.3 Scenari considerati

Per questo elaborato sono stati considerati due scenari statici, ovvero tutti i dispositivi sono fermi durante il rilevamento del segnale. Nel primo caso, gli esperimenti vengono condotti in una stanza di medie dimensioni presso l'edificio del Dipartimento di Ingegneria dell'Informazione dell'Università di Padova. Nel secondo caso, le misurazioni sono state svolte in un ambiente aperto privo di ostacoli, quale un terreno agricolo pianeggiante. In entrambi i casi si è simulata una macchina agricola con a bordo tre dispositivi BLE, a distanza ravvicinata, che scannerizzano continuamente, e un altro apparecchio *a terra* in modalità advertising.

Per ogni scenario si procede con la raccolta di RSSI tra due soli dispositivi. Vengono raccolti più campioni per ogni misurazione, circa 150, iniziando da una distanza molto piccola, 20 cm, procedendo con: 30 cm, 60 cm, 1 m, 1.30 m, 1.60 m, 2 m, 2.5 m, 3 m, 4 m, 5 m, 6 m, 7 m. In questo modo si può ottenere una stima dell'attenuazione della potenza di segnale e riuscire a capire come correlare i valori di RSSI con la distanza a cui si trovavano i due apparecchi.

Si prosegue disponendo tre dispositivi vicini tra di loro per simulare la posizione frontale e posteriore-laterale della macchina agricola. Ad uno di questi si collega un rilevatore GPS collegato alla porta USB. Si pone un quarto dispositivo a distanza scelta e si esegue il codice contemporaneamente su tutti i dispositivi centrali. In questo caso si memorizza il RSSI su tutti e tre i dispositivi e la loro posizione rispetto ad un piano. Si procede con la rielaborazione dei dati al fine di ottenere la posizione del quarto dispositivo, considerando nota la posizione degli altri tre.

Capitolo 3

Metodo Di Sperimentazione

3.1 ASUS TinkerBoard S

I dispositivi forniti dal dipartimento di Ingegneria dell'informazione dell'Università degli studi di Padova sono quattro Tinker Board S. La Tinker Board S [9] è un computer a scheda singola con dimensioni ridotte, dotata di un processore a quattro core con architettura ARM denominato Rockchip RK3288 e du 2 GB di RAM LPDDR3. È presente un'uscita audio, un'uscita video, quattro porte USB 2.0 e un slot micro SD per installare il sistema operativo. La Asus Tinker Board S ha un modulo Bluetooth integrato, con chip Realtek RTL8723BS, che supporta la connessione Bluetooth 4.0. Il chip RTL8723BS combina un'interfaccia wireless LAN 802.11n e un'interfaccia Bluetooth 4.0 in un singolo chip, offrendo una soluzione di connettività wireless completa.

3.2 Configurazione

Inizialmente si è scelto il sistema operativo, Tinker OS, una distribuzione Linux basata su Debian, progettata e sviluppata da Asus per funzionare su hardware

ARM. Ci sono varie versioni di questa distribuzione, ma le ultime uscite nel 2021 non funzionano nelle schede fornite, si è quindi dovuto installare la versione Tinker Board Debian OS V2.1.16. In seguito si è aggiornato BlueZ, che è l'implementazione ufficiale del protocollo Bluetooth per il sistema operativo Linux.

Avendo una certa familiarità con il linguaggio di programmazione Python, si è scelto di installare Python3 e alcune sue librerie, utili allo sviluppo del programma. Per fare ciò è stato utilizzato il protocollo Secure Shell (SSH), che permette di stabilire una sessione remota cifrata, tramite interfaccia a riga di comando con un dispositivo connesso alla stessa rete. Una volta installato il tutto nel primo dispositivo si è copiata l'immagine di sistema e messa nelle micro SD delle altre Tinker Board S. Per essere sicuri di aver configurato adeguatamente le schede sono stati provate le utility fornite da BlueZ come Hcitol e Btmgmt. La prima consente di controllare e configurare il protocollo Bluetooth sui dispositivi, in particolare permette la scansione, la connessione, la verifica dello stato del dispositivo e l'invio di comandi direttamente al controller Bluetooth. La seconda è molto simile ma permette di accedere anche a funzionalità avanzate e di impostare opzioni di configurazione. Hcitol è deprecato dalla versione 5.0 di BlueZ in favore di Btmgmt e Btmon.

Per permettere la visualizzazione dei dispositivi tramite BLE si procede impostando l'interfaccia del bluetooth, nel nostro caso hci0, con il Low Energy in funzione, dando un nome al dispositivo, impostando l'interfaccia in advertising, per essere trovata tramite scansione e impostando l'interfaccia su on:

```
sudo btmgmt -i hci0 le on
sudo btmgmt -i hci0 name NomeDispositivo
sudo btmgmt -i advertising on
sudo btmgmt -i hci0 power on
```

In alcuni dispositivi l'indirizzo MAC è mascherato con uno random in modo tale da non permettere agli altri calcolatori in scansione di identificare la Tinker Board S in advertising. In questo caso è utile avere l'indirizzo MAC vero e quindi bisogna disabilitare l'impostazione di privacy:

```
sudo btmgmt -i hci0 power off
sudo btmgmt -i hci0 privacy off
sudo btmgmt -i hci0 power on
```

Una volta verificato con un secondo dispositivo che il primo venga trovato in modo corretto si può procedere con l'implementazione del codice.

3.2.1 Librerie e codice utilizzati per le misurazioni

Per la stesura del codice si sono incontrati diversi problemi, in quanto le informazioni disponibili su Internet sono molto limitate. Sul sito Internet ufficiale di Bluetooth [10] si trova documentazione indirizzata agli sviluppatori, per gestire al meglio BlueZ utilizzando codice scritto in C e Python. In questo caso per scrivere codice in Python che si interfacci con BlueZ bisogna installare la libreria Bluepy, progettata per offrire un'interfaccia di programmazione per il protocollo Bluetooth Low Energy (BLE) su dispositivi Linux. Questa libreria fornisce funzioni per eseguire la scansione, la connessione e la comunicazione con i dispositivi BLE. In questo caso si utilizza Bluepy per scannerizzare i dispositivi circostanti e salvando alcune caratteristiche di questi dispositivi, come l'indirizzo MAC per identificare il dispositivo trovato e il Received signal strength indication. I dati raccolti vengono salvati per comodità in un file .csv, in questo modo sono facilmente analizzabili.

```
# Intervallo di scan in secondi (s)
intervallo = 0.5
scanner = bluepy.btle.Scanner()
    scanner.scan(intervallo)
```

```

# Itera tutti i dispositivi trovati dallo scanner
    for device in scanner.getDevices():
# Salva i valori di indirizzo MAC e RSSI dei dispositivi
        mac = device.addr
        rssi = device.rssi
# Scrive su un file csv i valori salvati
        with open(csv_file, "a", newline="") as f:
            writer = csv.writer(f)
            writer.writerow( [mac , rssi])

```

Si nota che i dispositivi non vengono trovati in modo costante, questo perchè per risparmiare energia hanno un intervallo di pausa di advertising preimpostato. Le Tinker Board S supportano fino a un minimo di 50 millisecondi di advertising e questo tempo si può cambiare in questo modo:

```

# Indirizzo del dispositivo Bluetooth, 0 dato che eseguo il codice in
  locale
dev_id = 0

# Apre un'interfaccia HCI per il dispositivo Bluetooth
sock = bluez.hci_open_dev(dev_id)

# Imposta i parametri dell'intervallo di advertising
cmd_pkt = struct.pack('<BBHHBBB', 0x08, 0x0006, 0x00a0, 0x00a0, 0x00,
    0x00, 0x00)
bluez.hci_send_cmd(sock, 0x08, 0x0006, cmd_pkt)

# Abilita l'advertising del dispositivo
cmd_pkt = struct.pack('<BB', 0x01, 0x00)
# Il tempo viene definito con 0x00a0 = 160 unita di 0.625 ms
bluez.hci_send_cmd(sock, 0x08, 0x000a, cmd_pkt)

```

3.2.2 Librerie e codice utilizzati per il calcolo della distanza

Una volta raccolti i dati necessari, ovvero i valori RSSI dei dispositivi, si può calcolare la distanza, mediante la (2.2). Lo si fa procedendo con il calcolo della media lineare dei valori di RSSI.

```

for mac, rssi_values in data.items():
    color = colors.get(mac, 'btck')
    if distance is not None:
        counter = Counter(rssi_values) # conta i valori di RSSI
        sample_total = len(rssi_values)
        cmap = LinearSegmentedColormap.from_list("", ["white",
            color])
        mean_rssi = statistics.mean(rssi_values)
        mean_dbm = 10 * mt.log10(mean_rssi)
        computeddistance = 0.0009275 * 10**((2.081 - mean_dbm)
            /(10*1.796))

```

In seguito utilizzando la libreria di Python Matplotlib, viene stampato un grafico che mostra l'andamento delle distanze trovate rispetto alla distanza reale alla quale erano posti i dispositivi.

3.2.3 Librerie e codice utilizzati per il calcolo del posizionamento

Per il posizionamento si utilizzano le stesse librerie utilizzate per il calcolo della distanza (si veda la Sezione 3.2.2). Una volta determinata la distanza tra ciascun dispositivo posto sulla macchina agricola e il dispositivo a terra, si caricano i dati ottenuti su uno script in Matlab per calcolare la posizione, utilizzando il procedimento descritto nella Sezione 2.2.2.

Capitolo 4

Risultati Sperimentali

4.1 Calcolo della distanza

4.1.1 Acquisizione dei dati

Scenario al chiuso

Il luogo scelto per la prova in ambiente chiuso è un'aula del Dipartimento di Ingegneria dell'Informazione dell'Università di Padova. La stanza ha dimensioni di circa $8\text{ m} \times 8\text{ m}$. La misura della distanza viene effettuata utilizzando due Tinker-Board S [9] con modulo Bluetooth 4.0 e Low Energy come descritto nel Capitolo 3. Ciascuna di queste viene posta al di sopra di un tavolo alto circa 90 cm da terra. A priori è stato deciso di misurare il RSSI a determinate distanze, elencate nella Tabella 4.1.

Tabella 4.1: Distanze a cui sono stati posti i dispositivi nello scenario al chiuso.

Distanza (m)									
0.2	0.4	0.6	1	1.3	1.5	2	2.5	3	4



Figura 4.1: Foto dell'esperimento in campo aperto

Viene misurata con una cordella metrica la distanza tra i due dispositivi e viene eseguito il programma fino al raggiungimento 150 campioni. Ogni misurazione viene registrata in file `.csv` con le seguenti colonne: [Tempo, MAC Address, RSSI].

Scenario all'aperto

Il luogo scelto per la prova sperimentale con BLE e Bluetooth classico in ambiente esterno è un terreno agricolo privo di ostacoli. Il suolo è pianeggiante e attorno sono presenti delle basse coltivazioni orticole. I dispositivi alimentati da due batterie portatili, sono stati posti al di sopra di due scatole di cartone ad un'altezza di circa 60 cm (Figura 4.1). Sono stati raccolti 150 campioni per ogni distanza sia con BLE che con il Bluetooth classico. Le distanze alle quali è stato misurato il RSSI elencate nella Tabella 4.2.

Tabella 4.2: Distanze a cui sono stati posti i dispositivi nello scenario all'aperto.

Distanza (m)												
0.2	0.4	0.6	1	1.3	1.5	2	2.5	3	4	5	6	7

Ogni misurazione viene salvata in un file `csv` con le seguenti colonne [Tempo, MAC Address, RSSI].

4.1.2 Elaborazione dei dati

Elaborazione dei dati al chiuso

Una volta raccolte le misurazioni si è proceduto con l'elaborazione dei dati. Eseguendo il codice descritto nella Sezione 3.2.2, si ottiene un grafico che mostra una linea che collega la media dei valori per ogni distanza, data dai campioni di RSSI raccolti. Vengono stampati nel grafico anche dei cerchi colorati, caratterizzati da un'intensità di colore variabile. L'intensità rappresenta la densità di un certo valore rilevato. Più il colore è intenso più campioni hanno lo stesso valore (Figura 4.2).

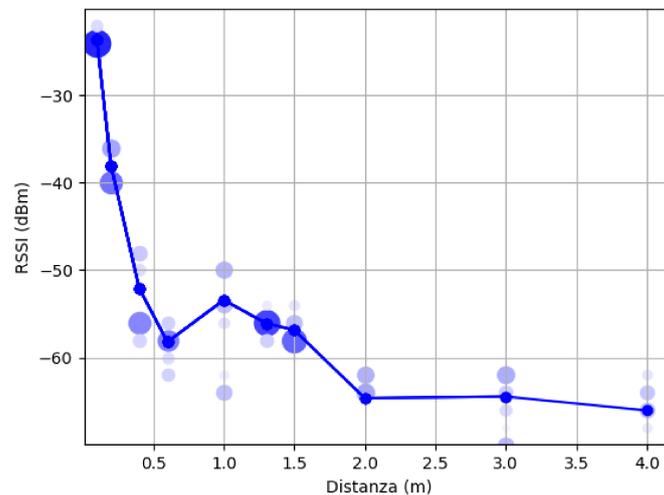


Figura 4.2: Dati ottenuti in ambiente chiuso utilizzando il BLE. Il grafico rappresenta il valore di RSSI per ogni distanza.

Elaborazione dei dati all'aperto

In seguito alla raccolta dati, mantenendo separati i valori ottenuti dal BLE da quelli ottenuti con Bluetooth classico, si esegue il codice descritto nella Sezione 3.2.2 per ognuno di questi. Si ottengono due grafici che mostrano l'andamento del RSSI in relazione all'aumento della distanza. In particolare si mostra con una linea

l'andamento della media dei valori di RSSI e con dei cerchi colorati si rappresentano invece i valori. In base alla densità di ciascun dato, viene aumentata l'intensità del colore del cerchio che lo rappresenta (Figure 4.3 e 4.4). Un terzo grafico mostra la comparazione dell'andamento della media lineare ottenuta con il BLE e quella ottenuta con il Bluetooth classico (Figura 4.5).

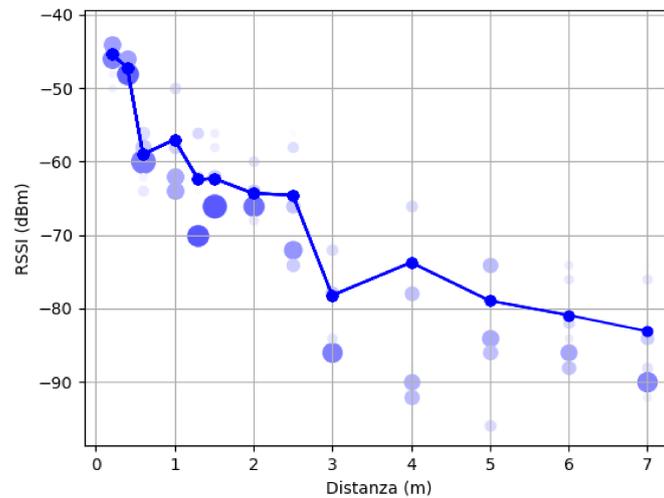


Figura 4.3: Dati ottenuti in ambiente aperto utilizzando il BLE. Il grafico rappresenta il valore di RSSI per ogni distanza.

4.1.3 Fitting dei dati e calcolo della distanza

Al fine di ottenere un'approssimazione della distanza ricavata dai valori RSSI, si esegue, per ogni scenario, un fitting dei dati, ovvero si stimano i parametri P_0 , n , e d_0 della funzione matematica

$$y = P_0 - 10n \log_{10} \left(\frac{x}{d} \right), \quad (4.1)$$

che meglio legano le distanze alla relativa lista di valori di RSSI medi misurati. Si procede prima con i dati raccolti nello scenario al chiuso (Figura 4.6), poi con

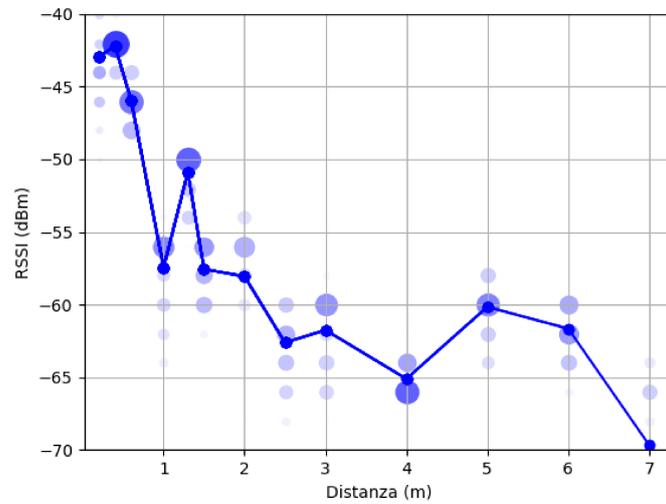


Figura 4.4: Dati ottenuti in ambiente aperto utilizzando il Bluetooth classico. Il grafico rappresenta il valore di RSSI per ogni distanza.

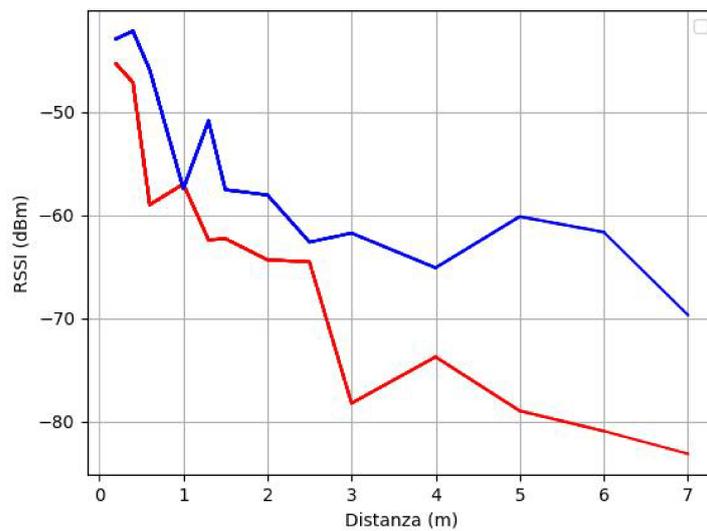


Figura 4.5: Dati ottenuti in ambiente aperto utilizzando il BLE e il Bluetooth classico. Il grafico rappresenta la media dei valori di RSSI per ogni distanza. La linea rossa collega i valori ottenuti dal BLE, la linea blu collega i valori ottenuti dal Bluetooth classico.

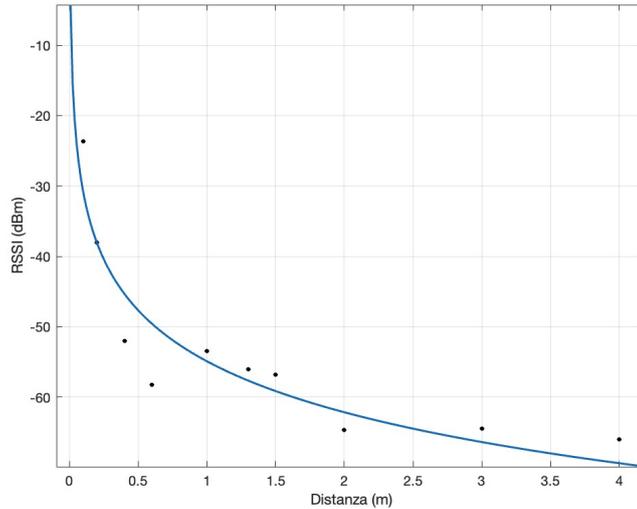


Figura 4.6: Linea di fit dei dati di RSSI ottenuti in un ambiente chiuso mediante il BLE, secondo l'equazione (2.2), con $P_0 = -54$, $n = 2$, $d = 0.001$.

quelli ottenuti nel scenario all'aperto con BLE (Figura 4.7) e infine con i dati ottenuti con il Bluetooth classico (Figura 4.8). Si ottengono così, per ciascuno scenario considerato, i parametri P_0 , n , e d_0 stimati e l'errore quadratico medio rappresentati nella Tabella 4.3.

Tabella 4.3: Parametri ottenuti dal fitting dei dati e il relativo RMSE.

Scenario	P_0	n	d_0	RMSE
BLE all'aperto	0.872	2.363	0.004372	5.389
BLE al chiuso	0.5775	2.105	0.001193	4.317
BT all'aperto	2.081	1.796	0.0009275	3.614

Una volta ottenuti questi valori si sostituiscono alla (2.2) per ottenere la distanza stimata. Successivamente, vengono mostrate le distanze stimate in relazione alla distanza reale alle quali sono state misurate. In ordine vengono raffigurati i grafici corrispondenti gli scenari al chiuso (Figura 4.9) e all'aperto (Figura 4.10) con BLE, poi quelli dei dati relativi al Bluetooth classico nel scenario all'aperto (Figura 4.11).

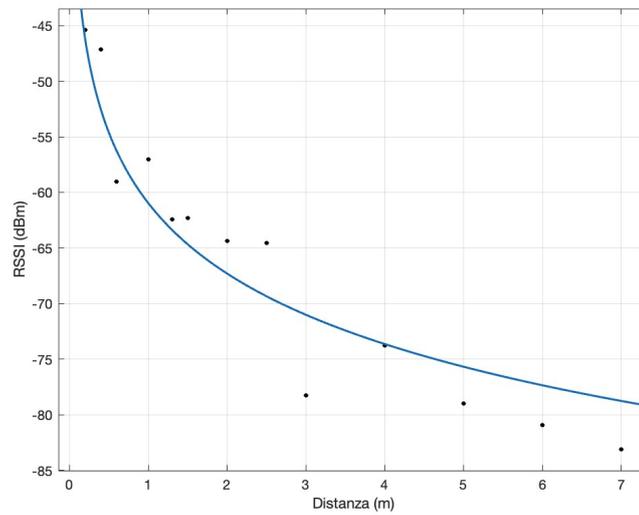


Figura 4.7: Linea di fit dei dati di RSSI ottenuti in un ambiente aperto mediante il BLE, secondo l'equazione (2.2), con $P_0 = -51.333$, $n = 2$, $d = 0.001$.

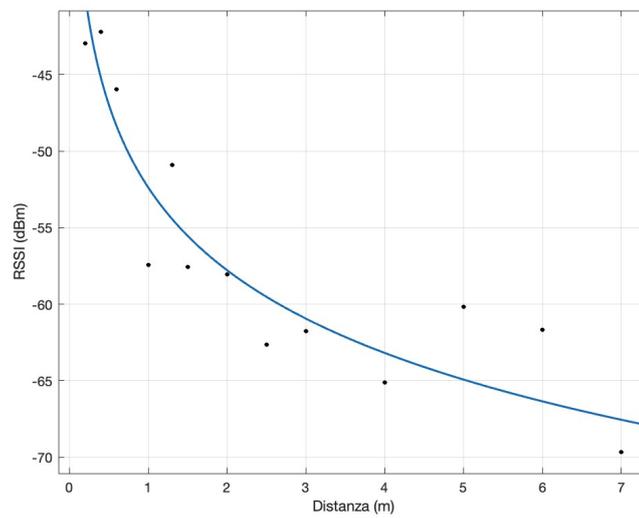


Figura 4.8: Linea di fit dei dati di RSSI ottenuti in un ambiente chiuso mediante il Bluetooth classico, secondo l'equazione (2.2), con $P_0 = -54$, $n = 2$, $d = 0.001$.

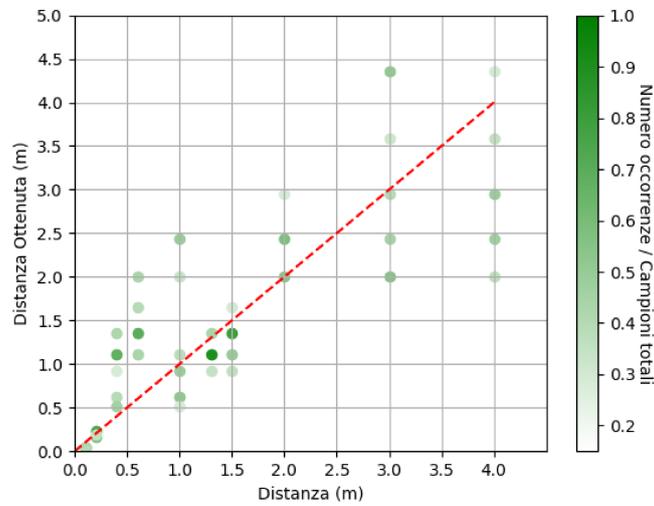


Figura 4.9: Distanza ottenuta per ogni valore di RSSI, secondo (2.2), in relazione alla distanza reale alla quale erano posti i due dispositivi. I dati sono stati raccolti mediante il BLE in uno spazio chiuso.

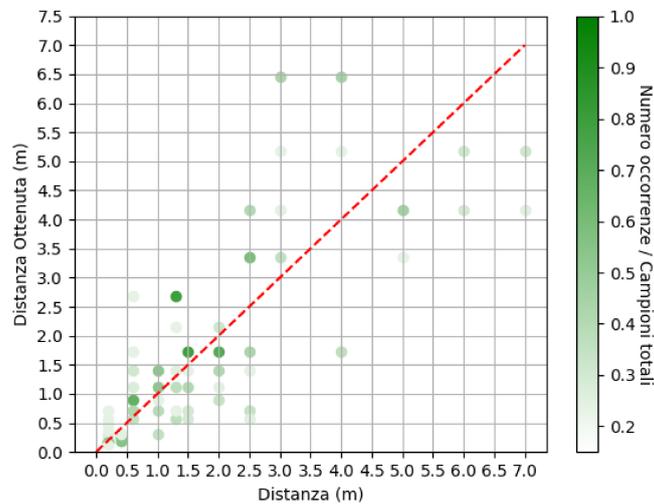


Figura 4.10: Distanza ottenuta per ogni valore di RSSI, secondo (2.2), in relazione alla distanza reale alla quale erano posti i due dispositivi. I dati sono stati raccolti mediante il BLE in uno spazio aperto.

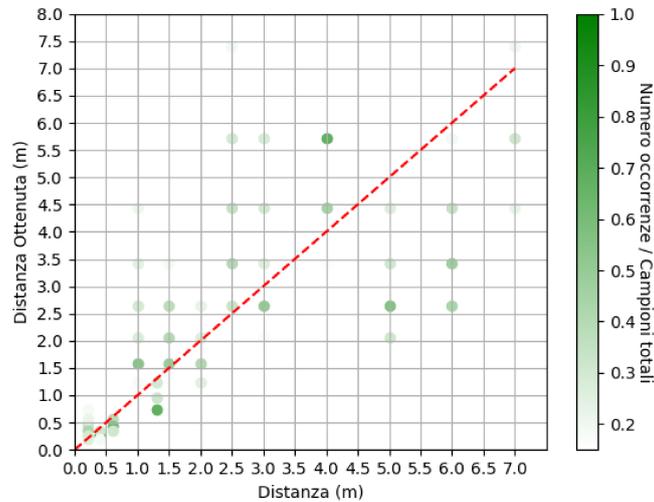


Figura 4.11: Distanza ottenuta per ogni valore di RSSI, secondo (2.2), in relazione alla distanza reale alla quale erano posti i due dispositivi. I dati sono stati raccolti mediante il Bluetooth classico in uno spazio aperto.

4.2 Calcolo del posizionamento

Per il posizionamento si utilizzano quattro dispositivi. Tre Tinker Board S sono posizionate con una disposizione triangolare e sono fisse. La quarta viene spostata per ogni misurazione. Si avvia sui tre dispositivi fissi il codice utilizzato per il calcolo della distanza descritto nella Sezione 3.2.2. Una volta raccolti i dati si procede con l'algoritmo di trilaterazione descritto nella Sezione 3.2.3. I dispositivi durante gli esperimenti condotti per questo elaborato si trovavano tutti alla stessa altezza dal suolo, così da semplificare le formule non dovendo considerare l'asse z .

4.2.1 Posizionamento ambiente chiuso

L'esperimento in ambiente chiuso è stato condotto nello stesso luogo in cui si è svolto quello relativo alla distanza in ambiente chiuso descritto nella Sezione 4.1.1. I tre dispositivi fissi sono stati posti su una scrivania, mentre quello mobile lo si

spostava in un'altra adiacente, con le medesime dimensioni e stessa altezza dal pavimento. Si esegue il codice descritto nella Sezione 3.2.2 sui tre dispositivi fissi, mentre il quarto viene configurato in modalità advertising. Si raccolgono così i dati di RSSI, si calcola la distanza e con l'algoritmo di trilaterazione (si veda la Sezione 2.2.2) otteniamo la posizione della quarta Tinker Board S.

4.2.2 Posizionamento ambiente aperto

L'esperimento in ambiente aperto è stato condotto, come per il calcolo della distanza, si veda la Sezione 4.1.1, in un campo agricolo privo di ostacoli (Figura 4.12). In questo caso sono stati utilizzati quattro dispositivi, posti sopra delle scatole di eguale altezza. Tre di questi dispositivi hanno posizione fissa e rappresentano la macchina agricola, mentre il quarto viene spostato per ogni misurazione. Vengono raccolti i valori di RSSI e calcolata la distanza. In seguito applicando l'algoritmo di trilaterazione si ricava la posizione della quarta Tinker Board S.



Figura 4.12: Foto che mostra la disposizione dei dispositivi durante l'esperimento in un ambiente aperto.

4.2.3 Risultati del posizionamento

Di seguito vengono presentati i risultati ottenuti per il posizionamento in ambiente chiuso e ambiente aperto. Il codice utilizzato, descritto nella Sezione 3.2.3, ci permette di determinare una posizione approssimata di un dispositivo data la distanza di questo da altre tre Tinker Board S. Ogni freccia ha una lunghezza uguale a quella ottenuta dal calcolo della distanza tra ogni dispositivo fisso a quello mobile e direzione verso la posizione trovata \hat{P}_i . Come si può notare le linee non si intersecano, questo è dato dalla varianza delle misure di RSSI raccolte, come si è mostrato nella Sezione 4.1.3. Di seguito i grafici che raffigurano i risultati ottenuti dal calcolo del posizionamento nello scenario al chiuso (Figura 4.13) e in quello all'aperto con BLE (Figura 4.14).

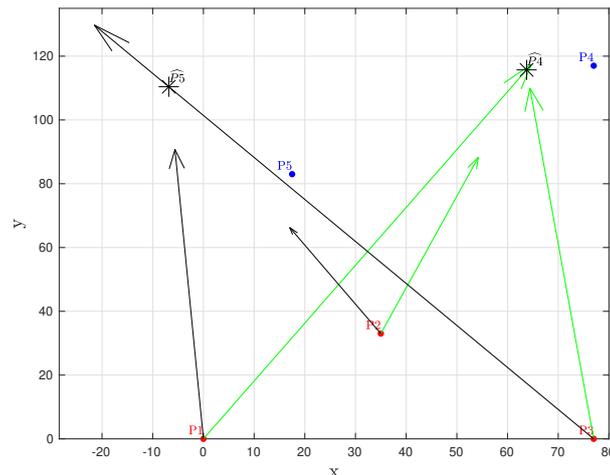


Figura 4.13: Piano in cui sono posti i dispositivi nell'esperimento condotto in ambiente chiuso. P1, P2 e P3 sono le posizioni dei dispositivi fissi. P4 e P5 sono le posizioni in cui si trovava il quarto dispositivo nella prima e seconda misurazione. Mentre \hat{P}_4 e \hat{P}_5 sono le posizioni stimate ottenute con l'algoritmo di trilaterazione. Le linee sono ottenute dalla distanza rilevata dai dispositivi. Le frecce verdi hanno direzione verso il punto P4, mentre quelle nere hanno direzione verso il punto P5.

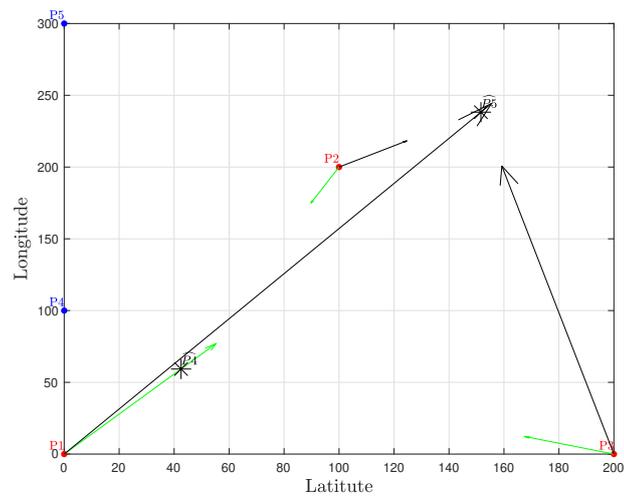


Figura 4.14: Piano in cui sono posti i dispositivi nell'esperimento condotto in ambiente aperto. P1, P2 e P3 sono le posizioni dei dispositivi fissi. P4, P5, P6 e P7 sono le posizioni in cui si trovava il quarto dispositivo nella quattro misurazioni. Mentre \hat{P}_4 , \hat{P}_5 , \hat{P}_6 e \hat{P}_7 sono le posizioni stimate ottenute con l'algoritmo di trilaterazione. Le linee sono ottenute dalla distanza rilevata dai dispositivi. Le frecce verdi hanno direzione verso il punto P4, mentre quelle nere hanno direzione verso il punto P5.

Capitolo 5

Conclusioni

I risultati ottenuti per il calcolo della distanza tra due dispositivi evidenziano un minore errore quadratico medio nelle misurazioni nello scenario all'aperto rispetto a quello al chiuso. Ciò è dovuto alla presenza di ostacoli, come muri e pavimenti, che possono attenuare il segnale e causare riflessioni multiple. Inoltre, la presenza di altre fonti di interferenza, come dispositivi elettronici, può causare ulteriori distorsioni nel segnale. Dai dati ottenuti con il fitting dei dati per il calcolo della distanza si nota un minore errore nel caso del Bluetooth classico rispetto alla sua versione meno energivora, questo per la più alta potenza di trasmissione, che permette un'ulteriore riduzione dell'errore rispetto al BLE. Per quanto riguarda il calcolo del posizionamento di un dispositivo, si può notare che con la sola misura del RSSI è difficile ottenere una buona stima della posizione. I risultati ottenuti rispecchiano la grande varianza dei risultati trovati nel calcolo della distanza. Questo può essere determinato da diverse cause, come il tipo di antenna presente nelle Tinker Board S, l'ambiente in cui si è svolta la campagna sperimentale, o da altri fattori di propagazione del segnale. Questo elaborato si presta per sviluppi futuri, come la realizzazione di metodi per migliorare la consistenza dei dati raccolti, riducendo la varianza delle misure raccolte in favore ad una migliore precisione della

posizione ottenuta. Ma anche applicazioni in tempo reale, ad esempio per tracciare il percorso eseguito da una macchina agricola in una rete di sensori dotati di BLE.

Bibliografia

- [1] M. Honkanen, A. Lappetelainen, and K. Kivekas, “Low end extension for bluetooth,” in *Proceedings. 2004 IEEE Radio and Wireless Conference (IEEE Cat. No.04TH8746)*, pp. 199–202, 2004.
- [2] M. Woolley, “The Bluetooth low energy primer,” *Bluetooth SIG*, 2022. Disponibile online: https://www.bluetooth.com/wp-content/uploads/2022/05/Bluetooth_LE_Primer_Paper.pdf, (Ultimo accesso: 9/03/2023).
- [3] The Bluetooth Special Interest Group, “Enhancing Bluetooth location services with direction finding,” 2019. Disponibile online: https://www.bluetooth.com/wp-content/uploads/2019/03/1901_Enhancing-Bluetooth-Location-Service_FINAL.pdf, (Ultimo accesso: 9/03/2023).
- [4] D. Giovanelli and E. Farella, “RSSI or time-of-flight for bluetooth low energy based localization? an experimental evaluation,” in *2018 11th IFIP Wireless and Mobile Networking Conference (WMNC)*, pp. 1–8, 2018.
- [5] A. Awad, T. Frunzke, and F. Dressler, “Adaptive distance estimation and localization in WSN using RSSI measures,” in *10th Euromicro Conference on Digital System Design Architectures, Methods and Tools (DSD 2007)*, pp. 471–478, 2007.

- [6] V. Gao, "Proximity and RSSI," Disponibile online: <https://www.bluetooth.com/blog/proximity-and-rssi>, (Ultimo accesso: 9/03/2023).
- [7] P. Octaviani and W. Ce, "Inventory placement mapping using bluetooth low energy beacon technology for warehouses," in *2020 International Conference on Information Management and Technology (ICIMTech)*, pp. 354–359, 2020.
- [8] L. Liu, B. Li, L. Yang, and T. Liu, "Real-time indoor positioning approach using iBeacons and smartphone sensors," *Applied Sciences*, vol. 10, no. 6, 2020.
- [9] A. C. Inc., "Tinkerboard s," Disponibile online: <https://www.asus.com/it/networking-iot-servers/aiot-industrial-solutions/all-series/tinker-board-s/>, (Ultimo accesso: 9/03/2023).
- [10] The Bluetooth Special Interest Group, "Bluetooth study guide," Disponibile online: <https://www.bluetooth.com/bluetooth-resources/?types=study-guide>, (Ultimo accesso: 9/03/2023).