

TRABAJO ESPECIAL DE GRADO

**DISEÑO E IMPLANTACION DE UN SISTEMA DE
LOCALIZACION INDOOR BASADO EN REDES WI-FI.**

Presentado ante la Ilustre
Universidad Central de Venezuela
por el Dr. Ing. Núñez R., Héctor L.
para optar al título de
Ingeniero Electricista

Caracas, 2008

TRABAJO ESPECIAL DE GRADO

**DISEÑO E IMPLANTACION DE UN SISTEMA DE
LOCALIZACION INDOOR BASADO EN REDES WI-FI.**

Prof. Guía: Roberto Cusani
Tutor Industrial: Massimiliano Giurelli

Presentado ante la Ilustre
Universidad Central de Venezuela
por el Dr. Ing. Núñez R., Héctor L.
para optar al título de
Ingeniero Electricista

Caracas, 2008

DEDICATORIA

Dedico este trabajo a quienes nunca se han apartado de mi lado en todo lo que llevo de vida:

“A Dios y a mis padres: Rosa y Jose Ramon”

AGRADECIMIENTOS

Para empezar quiero agradecer a Dios por ser mi guía espiritual en todos los momentos de mi vida y mandarme a este mundo de la mano, de unos maravillosos padres: Rosa y Jose Ramon, a quienes agradezco de todo corazon por todas sus orientaciones y cuidados, que me permitieron mantenerme por el camino correcto convirtiéndome así en un hombre de provecho y con lo cual ha sido posible alcanzar la mayor de mis metas, convertirme en ingeniero. Agradezco también a mis hermanos Javier, Daniel, Gabriel, Ismael y en particular a mi hermana Raiza, por toda la colaboracion, comprensión y afecto incondicionado en todos estos años. Agradezco a mi mejor amigo y compañero de estudios Ricardo Relvas y toda su familia: Marlene, Sra. Emilia y Sr. Armando, por su apoyo incondicional. Tambien agradezco a mis amigos más cercanos en los ultimos años de la carrera Irene Salas y Johnny Rangel.

Agradezco a la Universidad Central de Venezuela, en particular a la Escuela de Ingenieria Electrica, donde he seguido los primeros pasos de la carrera de ingegnieria. Igualmente, agradezco a la Università di Roma La Sapienza, en particular a la Facoltà d’Ingegneria, por permitirme participar en el convenio de doble titulacion, con lo cual he alcanzado el titulo de la Laurea Specialistica in Ingegneria delle Telecomunicazioni, convirtiéndome en Doctor en Ingenieria de las Telecomunicaciones.

Agradezco a todo el personal de la empresa Thera S.r.l., por darme la oportunidad de seguir mi pasantia profesional en su sede de Roma-Italia, agradezco de manera particular al presidente de la sociedad el Sr. Massimiliano Giurelli, por toda su colaboracion, comprension y confianza en el trabajo realizado. Agradezco al grupo de trabajo del proyecto Dimmi Museo, en el cual estuve trabajando durante toda la pasantia, en particular agradezco al Sr. Luca D’Angelo por su permanente colaboración.

Núñez R., Héctor L.

DISEÑO E IMPLANTACION DE UN SISTEMA DE LOCALIZACION INDOOR BASADO EN REDES WI-FI.

Profesor Guía: Roberto Cusani. **Tutor Industrial:** Massimiliano Giurelli. **Tesi.** Università degli studi di Roma La Sapienza. **INFOCOM.** Ingegneria delle Telecomunicazioni. **Tesis.** U.C.V. Facultad de Ingeniería. Escuela de Ingeniería Eléctrica. **Ingeniero Electricista.** **Opción:** Comunicaciones. **Institución:** Thera S.r.l. **2008.** 126 h. + anexos.

Palabras Claves: Comunicaciones Móviles, Redes Inalámbricas, Estandar IEEE 802.11, Localización Indoor, Algoritmos di Posicionamento, Dimmi Museo.

Resumen. Se plantea el estudio e utilización de redes inalámbricas bajo el estándar IEEE 802.11 para resolver el problema del posicionamiento en ambientes cerrados tales como museos, hospitales, centros comerciales, ferias, aeropuertos, oficinas, entre otros. Es posible conocer la posición exacta de un usuario que posea un dispositivo portátil yel cual se pueda conectar a la red inalámbrica de dicho ambiente, mediante mediciones y observaciones de la señal de radio, proveniente del intercambio de información entre la red y el dispositivo portátil. Para lograr ubicar al usuario mediante las mediciones recopiladas, es necesario utilizar algoritmos di posicionamiento, muchos de ellos citados en la literatura, e mediante un desarrollo via software, obtener una aplicación en grado de estimar la posición del dispositivo portátil y a su vez del usuario. El algoritmo o método elegido para el sistema de localización indoor, se basa en el funcionamiento de las redes neurales artificiales. El resultado final, es un sistema de localización indoor, que será comercializado, y el cual será en grado de localizar a las personas dentro de un museo y a su vez, asociar a una determinada posición, información multimedia referente a las obras presentes en el sitio o información de utilidad para el usuario.

INDICE GENERALE

DEDICATORIA.....	ii
AGRADECIMIENTOS	iii
RESUMEN	iv
INDICE GENERALE.....	v
INDICE DELLE TABELLE.....	viii
INDICE DELLE FIGURE.....	ix
ACRONIMI.....	xii
INTRODUZIONE	1
CAPITOLO I	
LOCALIZZAZIONE UTILIZZANDO RETI WI-FI.....	4
1.1 Localizzazione.....	4
1.2 Caratteristiche delle Comunicazioni Wireless	6
1.3 Caratteristiche delle Reti Wi-Fi	18
1.4 Localizzazione Wi-Fi in ambiente indoor	20
1.5 Point-Based e Area-Based Location	21
CAPITOLO II	
IL POSIZIONAMENTO IN UN SISTEMA DI LOCALIZZAZIONE WI-FI	23
2.1 Caratteristiche del Posizionamento.....	23
2.2 Infrastrutture di Posizionamento.....	24
2.3 Tecniche di Posizionamento.....	26
2.3.1 Proximity Sensing.....	27
2.3.2 Lateration	29
2.3.3 Angle of Arrival	32
2.3.4 Time of Arrival	33
2.3.5 Received Signal Strength Indication	35

CAPITOLO III	
ALGORITMI E METODI DI LOCALIZZAZIONE WI-FI IN AMBIENTE	
INDOOR	37
3.1 Introduzione.....	37
3.2 Metodo Nearest Neighbor in Signal Space (NNSS).....	40
3.3 Metodo Probabilistico.....	42
3.4 Metodo del Centroide	44
3.5 Metodi di Filtraggio	45
3.5.1 Algoritmo Filtro di Kalman	46
3.5.2 Algoritmo Particle Filter	49
3.6 Reti Neurali Artificiali	51
CAPITOLO IV	
TEST DEI METODI DI LOCALIZZAZIONE WI-FI	56
4.1 Specifiche generali del sistema	56
4.2 Fase Sperimentale e Test	66
4.2.1 Test del Metodo NSSS	72
4.2.2 Test del Metodo Probabilistico	76
4.2.3 Test del Metodo del Centroide.....	84
4.2.4 Test Metodo Ibrido	88
4.2.5 Implementazione dei Metodi di Filtraggio.....	95
CAPITOLO V	
IMPLEMENTAZIONE FINALE.....	96
5.1 Analisi del Problema.....	96
5.2 Realizzazione della Rete Neurale	97
5.2.1 Definizione delle funzioni di strato	101
5.2.2 Diagramma di funzionamento della rete neurale.....	105
5.3 Test della Rete Neurale.....	107
CAPITOLO VI	
CONCLUSIONI.....	121
BIBLIOGRAFIA.....	124

APPENDICE	127
1. Caratteristiche Tecniche del Palmare HP modello iPAQ serie hx2700	127
2. Datasheet del Router Sparklan Modello WX-6615GT.....	128
3. Datasheet degli AP NETGEAR Modello WG102.....	129
4. Mappe del Museo Civico di Albano.....	130
5. Caratteristiche di altri Sistemi di localizzazione Wi-Fi applicati in ambienti indoor.....	132
6. Confronto tra diversi Sistemi di localizzazione Wi-Fi applicati in ambienti indoor.....	133
7. Codice in linguaggio C# che gestisce la rete neurale	134

INDICE DELLE TABELLE

Tabella 2.1: Tecniche di Posizionamento e Parametri Misurati.	27
--	----

INDICE DELLE FIGURE

CAPITOLO I

Figura 1.1: Modello generico di un sistema LBS.	06
Figura 1.2: Modulazioni Digitali: ASK, FSK e PSK.....	07
Figura 1.3: Tecniche di Accesso Multiplo: FDMA, TDMA, CDMA.	08
Figura 1.4: Segnale DSSS.	09
Figura 1.5: Segnale FHSS.	09
Figura 1.6: Spettro di una singola portante OFDM.	10
Figura 1.7: Trasmettitore e Ricevitore CCK.	11
Figura 1.8: Patroni di radiazione dell'antenna isotropica e dell'antenna direzionale.	12
Figura 1.9: Antenna a dipolo.....	12
Figura 1.10: Schema di un collegamento radio in spazio libero.	13
Figura 1.11: Fenomeno di Multipath.	15
Figura 1.12: Fenomeni di Riflessione, Diffrazione e Diffusione.....	16

CAPITOLO II

Figura 2.1: Infrastrutture di Posizionamento.....	26
Figura 2.2: Coperture omnidirezionali e settoriali per sensori di prossimità	28
Figura 2.3 Lateration Circolare	29
Figura 2.4: Lateration Iperbolica.	29
Figura 2.5: Principio AoA.	32
Figura 2.6: Misurazione del tempo di propagazione.	33
Figura 2.7: Confronto di fase (segnali analogici).	34
Figura 2.8: Confronto di fase (segnali numerici).....	34

CAPITOLO III

Figura 3.1: Schema semplificato di un filtro di Kalman.	48
Figura 3.2: Schema di una Rete Neurale Artificiale.....	51
Figura 3.3: Schema di una Rete Neurale in modalità winner-take-all.....	54
Figura 3.4: Schema di una Rete Neurale Artificiale in modalità SOM.....	55

CAPITOLO IV

Figura 4.1: Schermata Mappa dell'applicazione Campionatura.exe.	58
Figura 4.2: Schermata Opzioni dell'applicazione Campionatura.exe.	59
Figura 4.3: Schermata Menu dell'applicazione Campionatura.exe.....	60
Figura 4.4: Schermata Posizione dell'applicazione Campionatura.exe.	60
Figura 4.5: Configurazione della modalità Bridge negli AP.....	64
Figura 4.6: Tipologia della rete di localizzazione Wi-Fi.....	65
Figura 4.7: Applicazione di gestione delle mappe.....	66
Figura 4.8: Mappa della sede a Roma di Thera S.r.l.	69
Figura 4.9: Diagramma di Funzionamento del Sistema di Localizzazione Wi-Fi.....	70
Figura 4.10: Prestazioni Algoritmo NNSS.....	73
Figura 4.11: Prestazioni Algoritmo NNSS Modificato	75
Figura 4.12: Prestazioni Metodo Probabilistico utilizzando Istogramma.	77
Figura 4.13: Prestazioni Metodo Probabilistico utilizzando Istogramma (versione modificata).	78
Figura 4.14: Prestazioni Metodo Probabilistico utilizzando una fdp Gaussiana.	79
Figura 4.15: Istogrammi Sala 1, punti 6 e 8	82
Figura 4.16 Istogrammi Sala 4, punti 1 e 2.	83
Figura 4.17: Prestazioni Algoritmo Centroide.	84
Figura 4.18: Prestazioni Algoritmo Centroide Modificato.....	87
Figura 4.19: Prestazioni algoritmo ibrido modificato.....	93

Figura 4.20: Potenza ricevuta per il terminale da un singolo AP al variare la distanza	94
---	----

CAPITOLO IV

Figura 5.1: Architettura di una rete neurale evolutiva	98
Figura 5.2: Diagramma di funzionamento della rete neurale.....	106
Figura 5.3: Modello di Rete Neurale per il sistema di localizzazione	107
Figura 5.4: Prestazioni della Rete Neurale	109
Figura 5.5: Prestazioni della Rete Neurale dopo le modifiche.....	112
Figura 5.6: Effetti sulla stima al variare il numero di campioni.	113
Figura 5.7: Effetti sulla stima al variare il numero di AP.	115

ACRONIMI

AES: Advanced Encryption Standard
AoA: Angle of Arrival
AP: Access Point
ASK: Amplitude-Shift Keying
BSA: Basic Service Area
BSS: Basis Service Set
CCK: Complementary Code Keying
CDMA: Code Division Multiple Access
CF: Compact Framework in .NET Framework
dB: decibel
dBm: decibel misurato in base a 1 miliwatt
DoA: Direction of Arrival
DQPSK: Differential Quadratura Phase Shift Keying
DSSS: Direct Sequence Spread Spectrum
ESS: Extended Service Set
FDMA: Frequency Division Multiple Access
fdp: funzione di distribuzione di probabilità
FHSS: Frequency Hopping Spread Spectrum
FSK: Frequency-Shift Keying
GB: Giga Byte
GHz: Giga Hertz
GPS: Global Positioning System
IIS: Internet Information Services
IP: Internet Protocol
LBS: Location-Based Services

m: metri
MAC: Medium Access Control
MB: Mega Byte
Mbps: Mega bit al secondo
MHz: Mega Hertz
MP3: MPEG-1 Audio Layer 3
NNSS: Nearest Neighbor in Signal Space
OFDM: Orthogonal Frequency Division Multiple
PDA: Personal Device Assistant
PL: Path Loss
PLCP: Physical Layer Convergence Procedure
PSK: Phase-Shift Keying
PSP: Play Station Portable
QAM: Quadrature Amplitud Modulation
RFID: Radio Frequeuncy Identification
RNA: Reti Neurali Artificiali
RSSI: Received Signal Strength Indication
SOM: Self-Organizing Map
SSID: Service Set Identifier
SVG: Scalable Vector Graphics
TDMA: Time Division Multiple Access
TDoA: Time Differential of Arrival
ToA: Time of Arrival
VoD: Video on Demand
VoIP: Voice over IP
WAF: Wall Attenuation Factor
Wi-Fi: Wireless Fidelity
WIPS: Wireless Indoor Positioning System
WLAN: Wireless Local Area Network
XML: eXtended Markup Language

INTRODUZIONE

Attualmente, molti operatori di servizi di telecomunicazioni wireless, oltre a fornire i servizi di base, vogliono avere traccia della propria utenza, per potergli offrire servizi complementari orientati alla localizzazione. Questi servizi sono conosciuti attualmente come LBS (Location-based Services) e possono essere classificati in outdoor e indoor, secondo l'estensione e l'accuratezza del sistema utilizzato per stimare la posizione.

Un esempio di un'applicazione LBS outdoor, è Google Map Mobile, compatibile con dispositivi terminali tali gli smartphone e PDA. L'applicazione sfrutta l'informazione della rete cellulare, per ottenere una stima della posizione di un utente al momento che questo si collega a Internet. Presenta un'accuratezza di circa 1,7 km. Anche è possibile sfruttare servizi di localizzazione GPS, aggiungendo un'antenna collegata tramite bluetooth al terminale. Lo scopo finale di questa applicazione è offrire informazione all'utente in funzione della sua posizione (i negozi più vicini, ricerca indirizzi, indicazioni stradali, ecc).

Un esempio di un'applicazione LBS indoor, potrebbe essere una guida museale, progettata di maniera diversa a quelle che oggi esistono: *Molte persone che vanno in giro dentro di un museo, portano con sé alcun dispositivo portatile di tipo wireless (smartphone, PDA, PSP, ecc). Se questi dispositivi sono dotati di schede Wi-Fi, e si l'ambiente è coperto di una rete Wi-Fi (situazione che attualmente è normale in diversi ambienti pubblici), è possibile via software determinare la posizione dell'utente sfruttando l'informazione scambiata tra il dispositivo portatile e la rete, e poi conoscendo la sua posizione, la rete può inviare al terminale informazione multimediale riguardante ai beni presenti o alla stanza del museo in cui si trova. Anche sarebbe possibile conoscere informazione utile come ubicazione delle toilette,*

scale, uscite, uffici, percorsi eseguibili dentro del museo, ecc. Mediante servizi di messaggiera istantanea l'utente o l'amministratore del sistema potrebbero interagire tra di loro.

Il presente progetto di tesi nasce proprio da questo esempio, come una proposta dall'azienda Thera S.r.l. di sviluppare un sistema di localizzazione Wi-Fi indoor, che possa essere sfruttato per offrire servizi di localizzazione (posizione in tempo reale dell'utente, suggerimento di percorsi tematici, navigazione, ecc) e servizi di divulgazione di contenuto e informazione (guide multimediali, cataloghi di servizi commerciali, ecc). Infatti, Thera S.r.l., ha lanciato un prodotto denominato Dimmi Museo, che utilizza un sistema di localizzazione Wi-Fi, e che per le sue prospettive, potrà essere utilizzato in contesti diversi dai musei. Si pensa anche a sfruttare la cosiddetta localizzazione in ambienti chiusi di tipo ufficio, ospedale, fiera, centro commerciali, parchi giochi, ecc.

I sistemi di localizzazione Wi-Fi indoor, sono di grande interesse nell'attualità, perché l'infrastruttura necessaria per la loro implementazione è poco costosa, questo grazie a che in molti ambienti indoor sono presenti reti Wi-Fi ed inoltre molti dispositivi portatili nell'attualità sono dotati di schede Wi-Fi, le quali permettono ai dispositivi collegarsi alla rete gestita per il sistema di localizzazione e così determinare la loro posizione. Comunque, per la localizzazione indoor, esistono oltre tecnologie diverse a quella Wi-Fi (sensori, ultrasuoni, ecc), ma che per la sua complessità e costi non hanno ancora avuto successo.

Lo stato dell'arte dei sistemi di localizzazione Wi-Fi indoor è ancora in vie di sviluppo e di sperimentazione, ma comunque esistono alcune soluzioni commerciali orientate al *tracking* in ambiente chiusi. Aziende come Ekahau, Aer Scout (Cisco), Navizon (usato per geolocalizzazione, ma che comunque sfrutta i segnali delle reti Wi-Fi) stanno già commercializzando sistemi che lavorano con la

propria infrastruttura Wi-Fi. Altri esempio, ancora in fase di ricerca sono RADAR (Microsoft), Place Lab (Intel), Herecast di tipo Area-Based Location, ecc.

Lo scopo di questa tesi, è studiare alcune delle tecniche di posizionamento trovate nella letteratura (vedere capitolo 2), verificando il loro funzionamento e le limitazioni, per decidere ulteriormente in funzione delle specifiche iniziali del sistema di localizzazione, la tecnica di posizionamento più appropriata.

Inoltre, occorre studiare i diversi metodi e algoritmi proposti nella letteratura (vedere capitolo 3) che utilizzano la tecnica di posizionamento scelta, studiare il loro comportamento e determinare le limitazioni che possono influenzare l'implementazione a livello di software.

Il passo successivo (vedere capitoli 4 e 5), sarà l'implementazione software e test dei metodi studiati, esaminare le sue prestazioni in un ambiente realistico, analizzare i risultati ottenuti e proporre modifiche necessarie per aumentare la *performance*. In funzione dei risultati ottenuti, sarà possibile scegliere il metodo con migliori prestazioni, e questo costituirà il cuore del sistema di localizzazione finale.

Una volta finalizzate tutte le prove necessarie, l'intero sistema sarà testato in un ambiente diverso a quello di sperimentazione (vedere capitolo 6), per verificare il comportamento del metodo di localizzazione a livello di prestazioni e di funzionalità.

In funzione dei risultati ottenuti in tutti i test eseguiti, durante lo svolgimento della tesi, si dovrà essere in grado di conoscere quali sono i fattori che evolvono un sistema di localizzazione Wi-Fi in ambiente indoor, in modo tale di stabilire i passi a seguire per la sua progettazione e implementazione in un ambiente reale.

CAPITOLO I

LOCALIZZAZIONE UTILIZZANDO RETI WI-FI

1.1 Localizzazione^[1]

Fondamentalmente, il termine “localizzazione” è associato a un determinato luogo nel mondo reale. Per esempio, quando le persone fanno un appuntamento, di solito decidono incontrarsi in un determinato luogo (nell’aeroporto, in un bar, in ufficio). In altri casi, si deve indicare l’indirizzo di residenza, per esempio, per ricevere una posta scritta, si deve specificare una posizione che denota il luogo in cui si trova un oggetto o una persona nel mondo reale, le quali (le posizioni) a sua volta appartengono a un insieme di posizioni fisiche (indirizzi).

Nell’attualità, con l’utilizzo di termini come *cyberspace* e *global village*, per fare riferimento a Internet, c’è una ragione per introdurre un nuovo concetto di posizione, dove anche qui, le persone possono interagire tra di loro tramite applicazioni diverse, ed anche possono essere localizzate per mezzo di posizioni o indirizzi virtuali.

Comunque, nell’ambito dei servizi basati sulla localizzazione (in inglese Local-Based Services, LBS), sul qual è sviluppato questo progetto aziendale (Dimmi, Museo), si fa riferimento sempre a una posizione fisica (del mondo reale), e non c’entra per niente il concetto di posizione virtuale, anche se a volte, in alcuni sistemi come la telefonia mobile, questi concetti si mescolano tra di loro, per avere traccia di un utente.

La maniera in cui è possibile rappresentare queste posizioni fisiche, è per mezzo di una descrizione generica della posizione in cui si trova l'utente (in un'area geografica, una regione, in una stanza di un determinato palazzo) oppure mediante una rappresentazione spaziale, cioè, sono indicate le coordinate di dove si trova l'utente.

Un esempio del primo caso, potrebbe essere la localizzazione della cella alla quale appartiene in un determinato momento un terminale di una rete mobile cellulare. Un esempio del secondo caso, sarebbe il classico GPS, o la versione Europea dello stesso sistema denominato Galileo. In quest'ultimo caso, i sistemi di navigazione satellitare danno come risposta le coordinate della posizione dell'utente, con un certo margine di errore. Ovviamente, il tipo di rappresentazione offerta, è funzione della necessità del determinato sistema e della sua utilità per la gestione della posizione.

Tra i sistemi di rappresentazione di coordinate che tipicamente sono utilizzati, ci sono quello cartesiano e quello ellissoidale. Il sistema cartesiano è tipicamente utilizzato, quando la localizzazione è riferita a uno spazio con quota costante, tipicamente ambienti *indoor*, dove non interessa la posizione dell'utente in funzione delle caratteristiche geografiche rispetto alla posizione della terra, invece in ambienti *outdoor*, come succede nel caso dei sistemi di navigazione dove il campo di ricerca è più ampio e più complesso, è utilizzato un sistema di coordinate ellissoidale più adatto alle considerazioni di localizzazione geografica.

Nella figura 1.1, si mostra un modello generico (non standardizzato) che serve a capire come funziona un sistema basato sulla localizzazione (LBS).

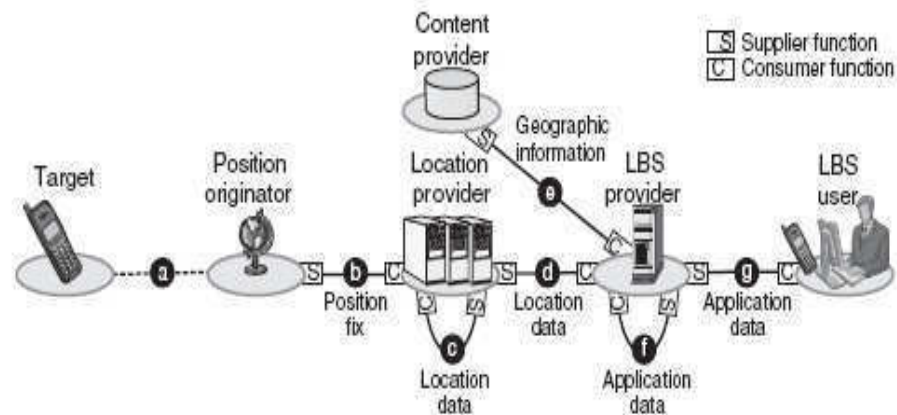


Figura 1.1: Modello generico di un sistema LBS.

1.2 Caratteristiche delle Comunicazioni Wireless [\[1\]](#), [\[2\]](#), [\[3\]](#)

Prima di andare a vedere come funziona la localizzazione in ambiente wireless è necessario descrivere alcune caratteristiche dei sistemi di comunicazione wireless visto l'influenza di queste sulle prestazioni nella localizzazione dell'utente.

Nel campo delle Tecnologie dell'informazione, i dati sono rappresentati di maniera digitale (numerica), per mezzo di sequenze di bit che possono prendere uno dei due valori discreti possibili: 0 oppure 1 binario. Il vantaggio di questa rappresentazione è la facilità di gestire i segnali numerici a livello di circuiti digitali, per i quali lo sviluppo nel tempo cresce in funzione delle richieste di progettazione. Il mezzo di trasmissione è l'aria, poiché il segnale viaggia o si propaga in forma d'onde elettromagnetiche, di natura analogica, quindi variante in tempo. Queste onde sono prodotte e ricevute per antenne, e per mezzo di opportuni modulatori del segnale collegati a queste, si trasformano i segnali dati in segnali radio. I parametri che definiscono il segnale radio sono l'ampiezza, la frequenza e la fase. Questi valori variano secondo il tipo di modulazione impiegata.

La modulazione è un processo per cui i segnali sono distribuiti in funzione di alcuni dei parametri noti del segnale (ampiezza, frequenza o fase). Questi segnali possono essere processati di maniera analogica o numerica. Nel caso di modulazione numerica si può distribuire i dati in ingresso alla durata di un impulso di informazione. Tra i tipi di modulazioni numeriche ci sono: scorrimento in ampiezza (ASK), scorrimento in frequenza (FSK), scorrimento in fase (PSK), in quadratura (QAM). In altre parole, gli impulsi sono adattati alle variazioni di alcuni di questi parametri. Ad esempio in PSK, il segnale modulato varia tra due valori di fase (0 e π), la trasformazione potrebbe essere che per un segnale binario ingresso con valore logico 1 la fase del segnale modulato non cambia, mentre che per un livello logico 0 presenterà uno sfasamento di π . Lo stesso per il caso ASK e FSK, variando l'ampiezza e la frequenza del segnale modulato rispettivamente, mentre nel caso QAM, se hanno 3 possibili valori di ampiezza e 4 possibili valori di fase per ogni valore di ampiezza, quindi permutando si possono rappresentare segnali analogici con una sequenza binaria di 4 bit.

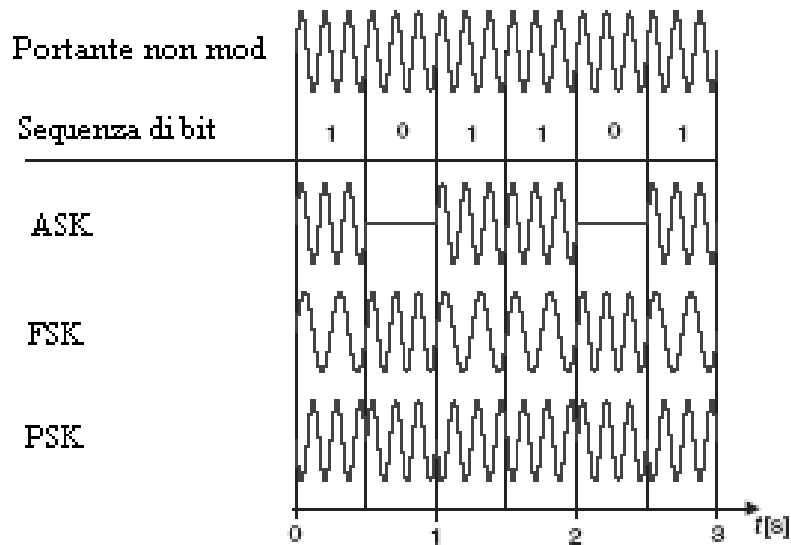


Figura 1.2: Modulazioni Digitali: ASK, FSK e PSK.

Le tecniche di accesso, permettono di distribuire vari utenti dentro di una banda o un canale. Alcune tecniche utilizzate in sistemi wireless sono TDMA (dove gli utenti sono distribuiti in *slot* di tempo su una stessa banda di frequenza), FDMA (dove gli utenti sono distribuiti su canali di frequenza diversa) e CDMA (dove gli utenti sono distribuiti per codici ortogonali tra di loro su una stessa banda di frequenza e magari nello stesso *slot* di tempo).

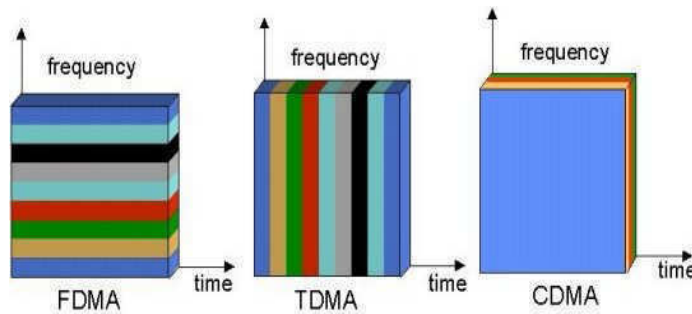


Figura 1.3: Tecniche di Accesso Multiplo: FDMA, TDMA, CDMA.

Esistono altre tecniche trasmissive, orientate al trattamento del segnale numerico, che permettono migliorare l'efficienza spettrale del segnale, con un conseguente aumento della velocità di trasmissione, parametro molto rilevante in alcuni sistemi wireless.

Alcune di queste tecniche trasmissive disperdono l'informazione su una banda molto più ampia di quella necessaria alla modulazione del segnale disponibile, è per far ciò occorre che il trasmettitore allarghi la banda del segnale trasmesso, moltiplicando il proprio flusso informativo (a ritmo f_b) con una sequenza di "chip" caratterizzata da un ritmo di cifra $f_c > f_b$, mentre che in lato ricevitore è ripristinato il segnale originario mediante un calcolo della correlazione rispetto alla medesima sequenza di chip utilizzata nel trasmettitore. Il vantaggio di queste tecniche è che per segnali di utenti diversi, si può avere un codice di *spread spectrum* ortogonale e diverso tra di loro, da qui il fatto che si può sfruttare la correlazione per riottenere il segnale originale.

Tecniche di tipo *spread spectrum* sono Direct Sequenze Spread Spectrum (DSSS) e il Frequency Hopping Spread Spectrum (FHSS). La differenza tra di loro, è che in FHSS si esegue un salto in un intervallo di tempo fisso tra differenti valori di frequenza, quindi per lo stesso istante si presenta più di un segnale utente, invece in DSSS, si ha solo un segnale nello stesso istante di tempo.

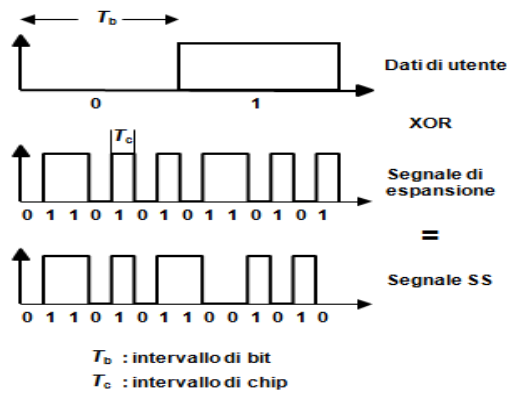


Figura 1.4: Segnale DSSS.

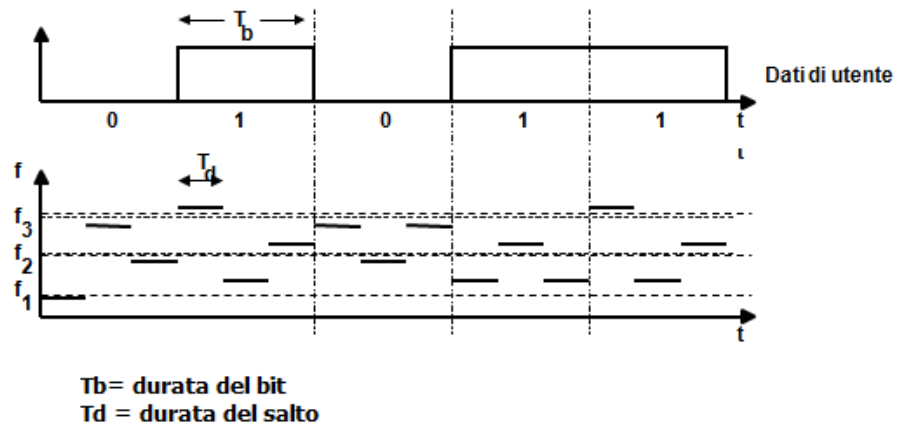


Figura 1.5: Segnale FHSS.

Un'altra tecnica trasmissiva, è Orthogonal Frequency-Division Multiplexing (OFDM), un tipo di modulazione multi - portante, dove il flusso di dati è suddiviso in diverse sottoportanti, uniformemente spaziate e ortogonali tra di loro. Il vantaggio

primario dell'OFDM rispetto agli schemi a singola portante è la robustezza fronte alle variazioni del segnale e alle condizioni di propagazione, a parità di diminuzione della velocità di trasmissione.

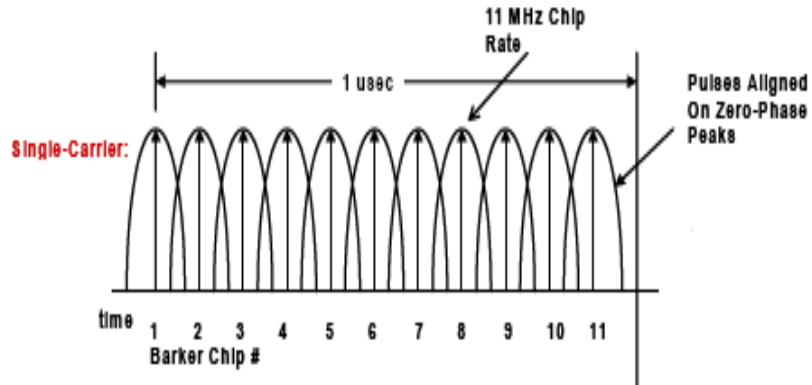


Figura 1.6: Spettro di una singola portante OFDM^[3].

Un'ultima tecnica trasmissiva, che sarà esposta è il Complementary Code Keying (CCK) dove i flussi di dati sono suddivisi in parole di codice, cosicché la trasmissione consta di milioni di *code symbols* al secondo, che sono generati attraverso trasformazioni matematiche che permettono al ricevitore di distinguere tra le diverse parole di codice anche in presenza di interferenza o di *multipath*. I codici complementari sono un insieme di sequenze finite con la stessa lunghezza, tali che per una sequenza qualsiasi il numero di coppie di elementi identici, con una data separazione, coincide con il numero di coppie di elementi diversi, a parità di separazione, per le altre sequenze.

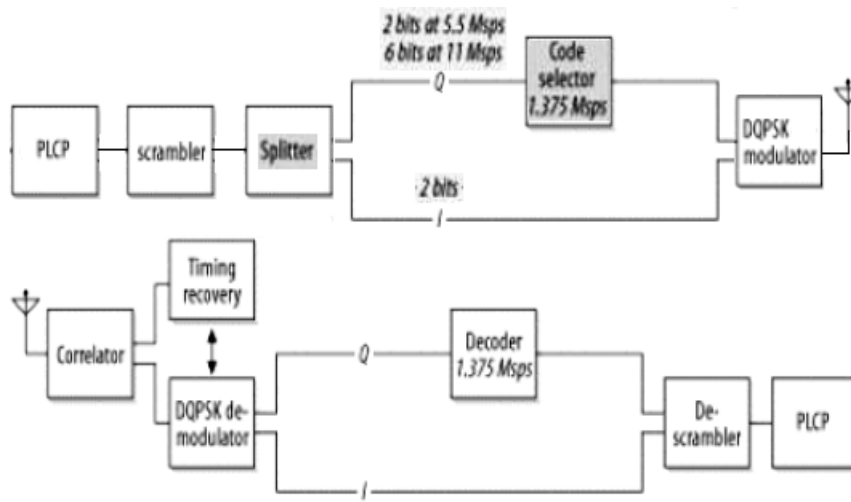


Figura 1.7: Trasmettitore e Ricevitore CCK ^[3].

In quanto alla propagazione del segnale, si sa che un segnale si comporta in maniera diversa in ogni mezzo, a seconda delle proprietà fisiche di questi. I segnali sono sempre oggetti di attenuazioni e di diversi tipi di rumore, che appaiono sostanzialmente sui mezzi con diversi livelli di interferenza, dovuto alla presenza di mezzi non guidati (l'atmosfera, l'acqua, lo spazio, ecc) oppure esposti a interferenze in funzione di come sono disposte le antenne in trasmissione e ricezione.

Le antenne impiegate in questi sistemi differiscono essenzialmente dalla loro direttività, che dipende dal modello di radiazione. La rappresentazione di questa radiazione è funzione dell'angolo di propagazione del segnale. Un parametro che si deriva di questo è il guadagno che può sfruttare il segnale e che varia secondo il patrone di radiazione, definendo la cosiddetta direttività delle antenne. In questo senso, si parla di antenne omnidirezionali quando l'energia è irradiata verso tutte le direzioni, come nel caso delle antenne isotropiche (che irradiano equamente verso tutte le direzioni) e antenne direzionali (la maggior parte dell'energia irradiata si concentra in una direzione in particolare).

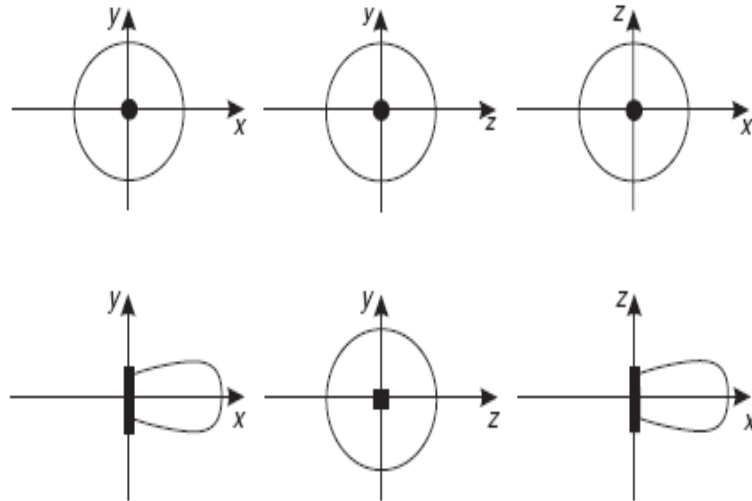


Figura 1.8: Patroni di radiazione dell'antenna isotropica e dell'antenna direzionale.

Le antenne isotropiche sono ideali, e per tanto non esistono. Il tipo di antenna più impiegato e semplice, è l'antenna a dipolo, che consiste in due fili dritti disposti in linea retta (in posizione orizzontale o verticale) con un piccolo scarto di alimentazione in mezzo. La lunghezza di questi fili dipendono dalla lunghezza d'onda dal segnale, lo quale avrà una influenza diretta sull'efficienza della trasmissione.



Figura 1.9: Antenna a dipolo.

Quando i segnali viaggiano attraverso l'aria, sono esposti a una riduzione della loro energia, fenomeno che è indicato come perdita di attenuazione di percorso. Il grado di attenuazione dipende della distanza tra l'antenna e il punto di misura, dalla lunghezza d'onda e dall'ambiente di propagazione. Per spiegare il fenomeno, si parte

di un'antenna isotropica (definita come un singolo punto sullo spazio che irradia la medesima quantità di energia verso tutte le direzioni), perciò il raggio di copertura è visto come una sfera. Un segnale emesso da una tale antenna con una certa potenza P_t arriva attenuato a una certa distanza d dall'antenna trasmittente, dovuto alla dispersione sulla superficie della sfera di radio d . Il livello di attenuazione è tipicamente espresso come il rapporto tra la potenza trasmessa e la potenza ricevuta. In questo caso si assume che le antenne in trasmissione e ricezione, sono allineate con il massimo lungo la loro congiungente (“corretto puntamento”), quindi le funzioni che rappresentano le radiazioni delle antenne rispetto agli angoli zenith e azimuth, (θ, φ) e (θ', φ') , sotto cui l'antenna trasmittente vede all'antenna ricevente e viceversa, avranno un valore unitario per la suddetta condizione di congiungente.

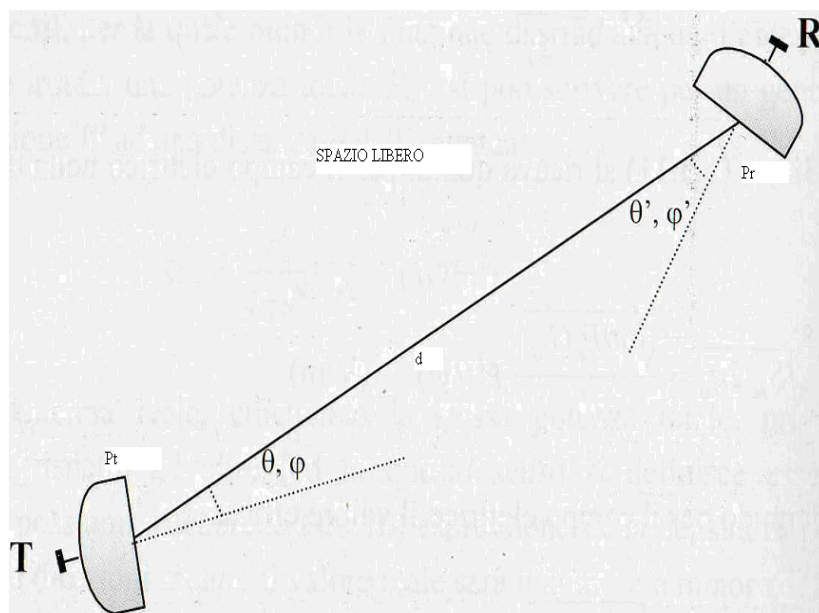


Figura 1.10: Schema di un collegamento radio in spazio libero.

Assumendo un'antenna isotropica e che non ci sono ostacoli tra le antenne trasmittente e ricevente, la relazione tra la potenza ricevuta e quella effettivamente trasmessa, è la seguente:

$$\frac{P_r}{P_t} = \frac{(4\pi d)^2}{\lambda^2}$$

Questa formula è una semplificazione dell'equazione di Friis in spazio libero. Dove d è la distanza tra le antenne, P_t e P_r , le potenze trasmesse e ricevute, e il parametro λ , che rappresenta la lunghezza d'onda a una frequenza specifica f nello spazio libero è determinata mediante la formula $\lambda=c/f$ (dove c è la velocità della luce $\approx 300e6$ m/s). Se ora, si pensa alla regione di campo lontano delle antenne (situazione tipica nella trasmissione di segnali tra due punti a una certa distanza) non va considerata più la potenza trasmessa, invece si prende il valore della potenza ricevuta rispetto a un punto di riferimento (d_o) conosciuto, che è determinata utilizzando l'equazione precedente. Quindi, la potenza ricevuta in qualsiasi punto della zona di campo lontano (d) è calcolato con la seguente equazione:

$$P_r(d) = P_r(d_f) \left(\frac{d_o}{d}\right)^2$$

Tipicamente, in sistemi wireless operando nella banda tra 1 e 2 GHz (com'è il caso sistemi operando con IEEE 802.11), in ambienti indoor, liberi di ostacoli (cioè, in linea di vista diretta), la distanza di riferimento d_o è di un metro. A questo punto, ancora non si ha analizzato il problema in presenza di ostacoli, dove diversi fenomeni di propagazione entrano in gioco (riflessione, diffrazione e scattering) e sono assunti sotto il concetto di propagazione *multipath*.

Il *multipath* è un fenomeno di propagazione originato per le strutture che coprono gli ambienti indoor, tali come: muri, pareti, colonne, pavimenti, ecc. Anche dovuto al ribaltamento del segnale contro ostacoli all'interno di queste strutture (ad esempio i mobili). La natura di queste strutture e ostacoli, cioè i materiali con i quali sono costruiti, sono la causa di questo fenomeno e caratterizzano fortemente

l'ambiente di propagazione del segnale. Il *multipath* è originato dai diversi percorsi in cui un segnale raggiunge l'antenna in ricezione, con diversi gradi di attenuazione e ritardo, risultato della combinazione di altri fenomeni di propagazione (riflessione, diffrazione e scattering).

La manifestazione di alcuni di questi effetti di propagazione, dipenderà principalmente dalla lunghezza d'onda del segnale in relazione alle dimensioni rispettive dell'ostacolo. La riflessione avviene se la lunghezza d'onda del segnale è minore rispetto alle dimensioni dell'ostacolo. La diffrazione descrive l'incurvamento del segnale radio, quando questo colpisce i bordi o gli angoli di un ostacolo, dove la dimensione dell'ostacolo è significativamente più grande della lunghezza d'onda del segnale. Lo scattering o diffusione, rappresenta la dispersione del segnale dovuta agli ostacoli le cui dimensioni sono dello stesso ordine di grandezza della lunghezza d'onda del segnale o meno.

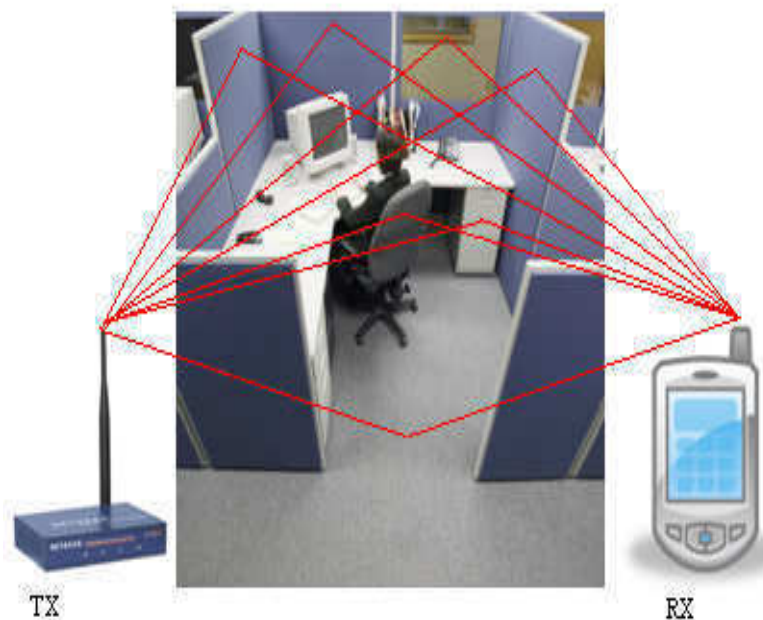


Figura 1.11: Fenomeno di Multipath.

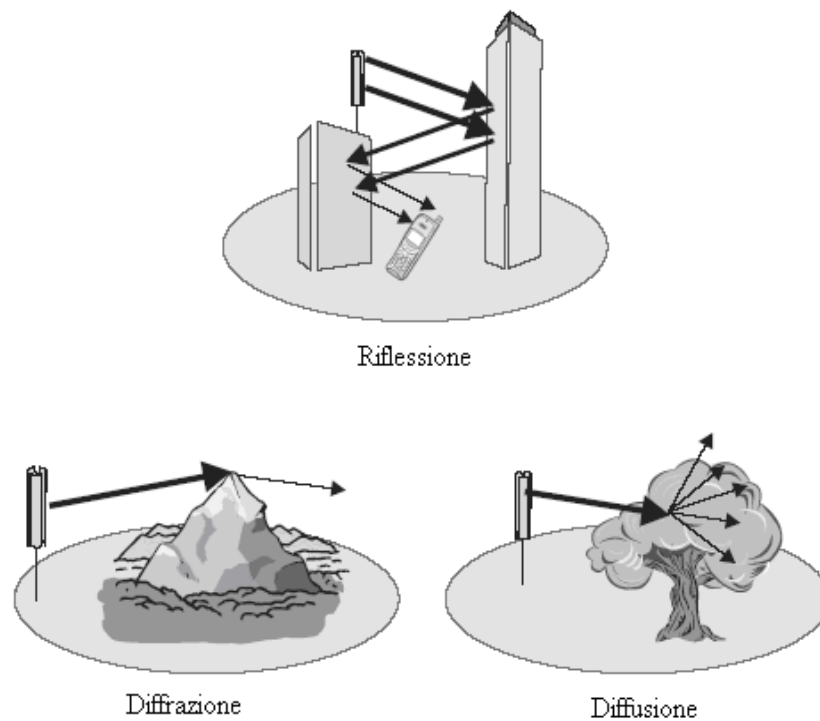


Figura 1.12: Fenomeni di Riflessione, Diffrazione e Diffusione.

Allora, se vengono considerati tutti questi problemi, l'equazione di Friis diventa:

$$P_r(d) = P_r(d_f) \left(\frac{d_o}{d} \right)^\alpha$$

Dove α è un fattore che raggruppa le perdite e le attenuazioni complessive del segnale dovute al *multipath* in un determinato ambiente, e d_f è la distanza di campo lontano della potenza ricevuta dall'antenna. Queste equazioni modellano l'ambiente di propagazione di maniera molto semplice in funzione soltanto dell'attenuazione in spazio libero, e non considerano gli altri fenomeni presenti.

Per i motivi già accennati, sono stati sviluppati modelli sia in ambiente *indoor* sia in ambiente *outdoor*, per determinare le perdite di percorso originate dalle attenuazioni subite per i fenomeni di propagazione presenti. Esempi di questi modelli sono: il modello di Rayleigh utilizzato quando tra trasmettitore e ricevitore non c'è un raggio diretto, e il modello di Rice quando c'è almeno un raggio diretto. Entrambi modelli affrontano il problema di *multipath*, ma hanno delle equazioni molto complesse da valutare. Esistono anche modelli di propagazione in ambiente *indoor*, che sono più semplici da utilizzare. Uno di modelli più utilizzati è il modello di Log-Distance Path Loss^[2], il quale rappresenta il legame tra la potenza ricevuta e la distanza, mediante la seguente equazione:

$$PL(d) = PL(d_o) + 10n \log\left(\frac{d}{d_o}\right) + X_\sigma$$

Dove il valore di n dipende dalle caratteristiche del mezzo e l'ambiente di propagazione, e X_σ rappresenta una variabile aleatoria normalizzata in dB avente deviazione standard σ in dB. $PL(d)$ e $PL(d_o)$ rappresentano le perdite di percorso in qualsiasi punto d a partire della distanza di un punto di riferimento d_o .

Un'adattamento di questo modello è stata proposta per Seidel e Rappaport^{[2],[5]}, ed è rappresentata della seguente equazione:

$$P(d)[dBm] = P(d_o)[dB] - 10n \log\left(\frac{d}{d_o}\right) - \begin{cases} nW * WAF & \text{se } nW \leq C \\ C * WAF & \text{se } nW > C \end{cases}$$

Dove $P(d)$ rappresenta la potenza ricevuta sul punto in studio dove d è la distanza trasmettere-ricevitore, $P(d_o)$ è la potenza in un punto di riferimento d_o . Invece WAF è il Wall Attenuation Factor rappresenta l'attenuazione dovuta ai muri,

n W numero di ostacoli (muri) tra il trasmettitore ed il ricevitore, e C è il massimo numero di ostacoli (muri) dove il WAF comincia ad affettare l'attenuazione. Tipicamente n e WAF dipendono dell'ambiente. Inoltre, il valore di $P(\text{do})$ può essere ottenuto dalle specificazioni di hardware della rete wireless oppure di maniera sperimentale.

1.3 Caratteristiche delle Reti Wi-Fi^{[3], [4], [5]}

Tra le reti WLAN (Wireless Local Area Network), quelle più utilizzate sono le reti Wi-Fi (Wireless Fidelity), che per definizione, seguono le specificazioni dallo standard IEEE 802.11 e sono disponibili in diverse varianti. Un'infrastruttura WLAN all'interno di un edificio si assomiglia molto ad una rete cellulare, che comprende un insieme di celle, ciascuna servita da una stazione base.

La famiglia di standard IEEE 802.11, specificano le caratteristiche a livello fisico e a livello MAC (Medium Control Access) che devono avere i dispositivi dotati di schede Wi-Fi. La famiglia IEEE 802.11 è composta dai seguenti standard:

- IEEE 802.11: opera nella banda di 2.4 GHz, impiegando tecniche di modulazione di tipo DSSS e FHSS, che li permettono di arrivare ad una velocità di circa 2 Mbps.
- IEEE 802.11a: opera nella banda di 5 GHz, impiega uno schema di modulazione di tipo OFDM, che le permette di arrivare ad una velocità di circa 54 Mbps. Non è interoperabile con la versione successiva IEEE 802.11b. Sono a disposizione 12 canali.
- IEEE 802.11b: anche conosciuto come Wi-Fi (Wireless-Fidelity). Opera nella banda di 2.4 GHz, impiega tecniche di modulazione di tipo

DSSS e CCK. Queste le permettono di raggiungere un'alta velocità di trasmissione con un ampio raggio di copertura con lo stesso AP (100 metri @ 11 Mbps). Sono a disposizione 14 canali. A differenza della versione IEEE 802.11a, richiede poche stazioni base per avere un maggiore raggio di copertura.

- IEEE 802.11g: opera nella banda di 2.4 GHz. Complessivamente arriva ad una velocità di trasmissione di circa 54 Mbps. Utilizza lo schema di modulazione OFDM quando la velocità di trasmissione è sopra dei 20 Mbps, altrimenti utilizza le tecniche DSSS e CCK. E' più robusto in quanto a sicurezza dalla versione iniziale IEEE 802.11. E' compatibile con la versione IEEE 802.11b. Sono a disposizione 14 canali. Ha un raggio di copertura simile a la versione IEEE 802.11b. E' quello più impiegato attualmente in combinazione con IEEE 802.11g.

- IEEE 802.11e: è il primo standard orientato a soddisfare una Qualità di Servizio, a seconda del tipo di utenza (residenziale o affare). Sono aggiunte caratteristiche di supporto multimediali per farli interoperabili con le versioni IEEE 802.11b e IEEE 802.11a . Può essere implementata in applicazioni di tipo Video on Demand, Audio on Demand, Voice over IP (VoIP) ed accesso Internet ad alta velocità.

- IEEE 802.11i: questa versione aggiunge il meccanismo di cifratura Advanced Encryption Standard (AES), il quale è più robusto dal tradizionale meccanismo di sicurezza Wi-Fi.

In IEEE 802.11, le stazioni base sono chiamate *Access Point* (AP), e la zona di copertura di un *access point* è denominata Basic Service Area (BSA). L'insieme di tutti i terminali serviti dallo stesso AP è chiamato Basic Service Set (BSS). Diversi BSS, a loro volta, possono essere interconnessi attraverso un'infrastruttura cablata per

formare un Extended Service Set (ESS). L'infrastruttura appena descritta è necessaria per il funzionamento di IEEE 802.11 nella cosiddetta modalità infrastruttura. In alternativa, IEEE 802.11, consente la comunicazione diretta tra i terminali senza infrastruttura nella modalità *ad hoc*. Tuttavia, per il posizionamento, per le ragioni spiegate ulteriormente nella sezione 2.2 , la modalità infrastruttura è necessaria.

1.4 Localizzazione Wi-Fi in ambiente indoor^[1]

Le infrastrutture WLAN, attualmente sono disponibili nella maggiore parte degli edifici pubblici, ed inoltre i venditori di dispositivi mobili come PDA, smartphone, PSP, riproduttori MP3, ecc, includono connettività Wi-Fi su questi dispositivi, rendendo più attraente lo sviluppo dei servizi LBS basati sulla localizzazione *indoor*. Quindi, dovuto all'accessibilità Wi-Fi di tanti dispositivi in giro, non bisogna fare un grosso investimento in infrastruttura, poiché dal lato HW i dispositivi sono economici e dal lato SW l'implementazione è più semplice. Questo rende l'infrastruttura WLAN una soluzione meno costosa rispetto ad altre infrastrutture dedicate.

Un altro vantaggio della localizzazione utilizzando reti Wi-Fi, è che operano meglio in ambienti indoor, sia in grosse zone chiuse, oppure nel caso che si debbano interconnettere due palazzi vicini, quindi il raggio di copertura non è limitato, anzi, c'è la possibilità di scalare, utilizzando le appropriate topologie di reti. I sistemi di comunicazione wireless, come visto nella sezione 1.2, sono affetti di fenomeni di propagazione del segnale che possono causare delle imprecisioni nel calcolo della posizione, ma tuttavia, il segnale radio è più robusto, dal segnale impiegato per altri sistemi come il video, infrarossi o ultrasuoni, che hanno più limitazioni nel caso non ci sia linea di vista diretta per gli ostacoli presenti, oppure per disturbi come la luce solare, luce fluorescenti, ecc.

In pratica, la localizzazione Wi-Fi in ambiente indoor, si basa nella collezione di un certo parametro presente nel segnale radio che viaggia tra gli AP e i terminali, e utilizzando una tecnica di posizionamento, si fa un confronto tra il valore corrente di quel parametro e i valori collezionati, e così mediante un algoritmo o metodo sistematico di localizzazione, sarà possibile determinare la posizione corrente dei terminali. Questi metodo non sempre si basano su un semplice confronto, anche potrebbero impiegarsi trasformazioni matematica per determinare la posizione.

1.5 Point-Based e Area-Based Location^{[5],[6]}

Vi sono sostanzialmente due approcci utilizzati nella localizzazione, quello di tipo puntuale o Point-Based e quello di tipo geografico o Area-Based (per zone, cluster o celle). La localizzazione Wi-Fi spesso utilizza l'approccio basato sul punto. In entrambi approcci, l'obiettivo è quello di restituire la posizione dell'oggetto ad essere localizzato. L'utilizzo di un sistema basato sulla localizzazione di aree, nasce fondamentalmente dall'incertezza derivante dei meccanismi di posizionamento puntuali utilizzati in Wi-Fi. Alcuni vantaggi dell'approccio geografico sono:

- E' in grado di individuare un terminale in un modo più sistematico rispetto all'approccio puntuale. Per esempio, se viene utilizzato questo sistema per determinare dove si trova un oggetto, utilizzando un approccio basato sull'area, l'utente può iniziare la sua ricerca nella zona individuata per il sistema, e con un'alta probabilità si troverà in quell'area o almeno in un'area adiacente. Poi si può espandere la ricerca fino a precisare l'ubicazione puntuale.
- E' più naturale e intuitivo per l'utente individuare un'area geografica. Se il sistema restituisce come risposta il nome di una stanza o di una zona, l'utente

capirà rapidamente la posizione data come risposta, piuttosto che andare a cercare un paio di coordinate su una mappa.

- In termini di prestazioni, un sistema basato sulla localizzazione di aree, sicuramente avrà un *range* di accuratezza e precisione più alto, visto che l'errore generato dipende dalla differenza tra la posizione giusta e quella stimata, quindi in localizzazione puntuale si rischia ad avere meno accuratezza.

Alla fine, quello che tipicamente è fatto, è impiegare entrambi approcci per accelerare la convergenza delle tecniche di posizionamento e velocizzare la risposta del sistema di localizzazione.

CAPITOLO II

IL POSIZIONAMENTO IN UN SISTEMA DI LOCALIZZAZIONE WI-FI.

2.1 Caratteristiche del Posizionamento [\[1\]](#)

Il posizionamento è un processo per ottenere la posizione spaziale di un determinato oggetto o di un utente che sia collegato a un servizio di localizzazione. Esistono diverse tecniche per determinare la posizione, che in generale sono legati ai seguenti fattori:

- I parametri osservati durante il processo di misurazione.
- Il tipo di calcolo realizzato per determinare la posizione.
- Il sistema di rappresentazione (spaziale o descrittivo) della posizione.
- L'infrastruttura.
- Protocolli per la coordinazione del processo di posizionamento.

La parte centrale di un meccanismo di posizionamento è la misurazione di uno o più parametri osservabili, ad esempio: angoli, *ranges*, differenza di *ranges*, velocità, ecc. Questi tipicamente riflettono una relazione spaziale tra un punto in studio rispetto a uno o più punti fissi in una determinata area. I punti fissi

rappresentano quei punti ben conosciuti e individuati per le sue coordinate. Essi sono spesso misurati utilizzando i principi dei segnali radio.

Dopo che questi parametri sono stati definiti e determinati, la posizione istantanea deve essere derivata ulteriormente, considerando il risultato delle misurazioni e le coordinate dei punti fissi. La procedura per determinare la posizione è tipicamente basata in alcuni metodi che sono molto legati ai cosiddetti parametri.

Un metodo di posizionamento determina la posizione istantanea in funzione del sistema di riferimento impiegato, e interagendo con un'infrastruttura distribuita, perché non può farlo di maniera autonoma. Quindi, ogni utente che vuole essere individuato, deve portare con sé un terminale. Poi sulla zona d'implementazione, devono esserci dispositivi che fungono come stazioni base, per assistere al terminale durante il posizionamento e misurare il parametro associato al metodo impiegato.

Inoltre, il posizionamento necessita di essere coordinato e controllato dai protocolli, che gestiscono i componenti del sistema. Generalmente, lo schema di coordinazione si basa nell'utilizzazione di un'unità centrale di controllo, che gestisce la comunicazione tra le diverse stazioni base, così come la comunicazione tra le stazioni base e i terminali.

2.2 Infrastrutture di Posizionamento^[1]

Il posizionamento e le infrastrutture di posizionamento possono essere classificate rispetto a criteri diversi. Le principali distinzioni fatte sono l'infrastruttura integrata oppure infrastruttura *stand-alone*; basata sul terminale oppure basata sulla rete (in termini di chi fa le misurazioni e il calcolo della posizione).

Un'infrastruttura integrata fa riferimento a una rete wireless che è utilizzata sia per la comunicazione sia per il posizionamento. Tipicamente, tali reti sono state inizialmente progettate solo per comunicazione, e ora stanno sperimentando estensioni, per permettere la localizzazione dei propri utenti attraverso meccanismi di posizionamento sui dispositivi mobili, il che vale soprattutto per le reti cellulari. Componenti che possono essere riutilizzati sono le stazioni base e i dispositivi mobili, nonché i protocolli di localizzazione e di gestione della mobilità. L'estensione della rete riguarda l'installazione di nuove versioni software e di componenti di rete per l'esecuzione e il controllo di posizionamento, l'installazione di nuovi componenti hardware per la sincronizzazione degli orologi, per la raccolta di dati di misurazione, e per altri scopi. Un approccio integrato ha il vantaggio che la rete non ha bisogno di essere costruita da zero e che i costi *roll-out* e i costi operativi sono gestibili, tuttavia, gestire il posizionamento sulla rete, risulta oneroso, in termini del traffico aggiunto, e di conseguenza diminuire la capacità offerta per i servizi di base dell'utenza. Nella maggior parte dei casi le misurazioni devono essere fatte sulle interfacce aerea esistenti, la cui progettazione non è stata ottimizzata per il posizionamento, ma per le comunicazioni, e quindi le ulteriori implementazioni potrebbero essere un tanto complicate in alcuni casi.

Un'infrastruttura *stand-alone* opera indipendentemente dalla rete di comunicazione alla quale l'utente è collegato. In contrasto con un sistema integrato, le infrastrutture e le interfacce aerea sono destinate esclusivamente per il posizionamento e la loro progettazione è molto semplice. Il più noto esempio di un'infrastruttura stand-alone è GPS, ma soprattutto per ambiente *indoor*, per esempio, negli aeroporti o uffici, un certo numero di soluzioni proprietarie sono state sviluppate. Gli svantaggi sono elevati costi di roll-out e di funzionamento, poiché gli utenti non possono essere localizzati attraverso un dispositivo mobile standard, richiedendo ulteriori modifiche, e a volte componenti hardware aggiunti proprietari.

Com'è stato detto precedentemente, c'è una distinzione tra posizionamento basato sulla rete e quello basato sul terminale, in termini di chi effettua le misurazioni e chi fa il calcolo della posizione fissa. Un approccio ibrido è dato quando il terminale effettua solo le misurazioni e poi trasmette i risultati alla rete, quindi alla fine la posizione è calcolata dalla rete in funzione di queste misure. Questo è chiamato posizionamento assistito (dalla rete) sul terminale. L'operazione inversa è stata definita, ma è applicata molto raramente: posizionamento assistito (dal terminale) sulla rete. La decisione sulla classe di metodi da utilizzare principalmente dipende dalle caratteristiche dei rispettivi LBS e sulla capacità dei rispettivi dispositivi mobili. Poi, gli operatori di rete devono sapere la difficoltà di fare migrare l'utenza verso dispositivi più adatti ai meccanismi di posizionamento, per cui bisogna puntare su soluzioni di posizionamento basate sulla rete e non sui terminali, facendoli interoperabili con tutti i tipi di terminali.

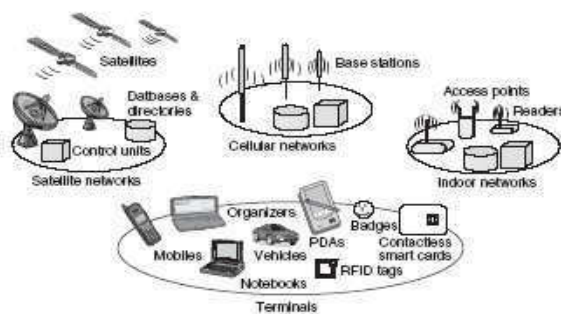


Figure 6.1 Positioning infrastructures.

Figura 2.1: Infrastrutture di Posizionamento.

2.3 Tecniche di Posizionamento [\[1\]](#), [\[5\]](#)

Esistono varie tecniche che possono essere utilizzate per determinare la posizione di un terminale (utilizzate sia in ambienti *outdoor* come in ambienti *indoor*), dove sono sfruttati i valori di certi parametri fisici sensibili alla variazione

della posizione, e che devono essere misurati costantemente per il servizio di localizzazione per avere conto della posizione del terminale. In generale, si basano su misurazioni di distanze, di angoli oppure di pattern di segnali radio.

Nella tabella 2.1, si mostra un quadro riassuntivo delle tecniche, il basamento e i parametri misurati:

Tabella 2.1: Tecniche di Posizionamento e Parametri Misurati.

Positioning method	Observable	Measured by
Proximity sensing	Cell-Id, coordinates	Sensing for pilot signals
Lateration	Range or	Traveling time of pilot signals Path loss of pilot signals
	Range difference	Traveling time difference of pilot signals Path-loss difference of pilot signals
Angulation	Angle	Antenna arrays
Dead reckoning	Position and	Any other positioning method
	Direction of motion and	Gyroscope
	Velocity and	Accelerometer
	Distance	Odometer
Pattern matching	Visual images or	Camera
	Fingerprint	Received signal strength

Nella sezione successiva, vengono descritte le tecniche di posizionamento trovate nella letteratura, tipicamente impiegate in sistemi di localizzazione. Alcune di queste sono: Proximity Sensing, Lateration, Angle of Arrival (AoA), Time of Arrival (ToA) e RSSI.

2.3.1 Proximity Sensing

E' la tecnica più semplice e più diffusa per ottenere la posizione di un terminale, basata nella creazione di zone delimitate per i raggi di copertura di un segnale radio, infrarosso o ultrasuono. La posizione di un terminale è derivata in

questo caso, delle coordinate della stazione base che pilota il segnale dal terminale in direzione *uplink* o dei segnali pilota che sono ricevuti dal terminale in *downlink*. Il principio dei sensori di prossimità si mostra nella figura 2.2.

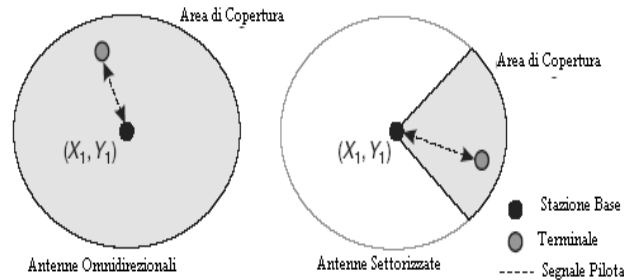


Figura 2.2: Coperture omnidirezionali e settoriali per sensori di prossimità.

Nella figura a sinistra si mostra una configurazione con sensori di prossimità, impiegando antenne omnidirezionali, mentre nella figura a destra si mostra il caso con antenne settorizzate. In questo caso si assume che la posizione del terminale è semplicemente la posizione della stazione base che invia e riceve i segnali pilota.

L'implementazione di questi sensori, si può fare in modi diversi, alcuni di essi standardizzati mentre che altre sono soluzioni proprietarie di certi operatori e fornitori. Comunque, la forma come sono disposti dipenderà anche dal fatto che sia impiegata un'infrastruttura integrata o *stand-alone*. In sistemi *indoor*, il rilevamento basato nella vicinanza è stato implementato in vari progetti di ricerca, ma non è stato finora standardizzato. Esempi sono Active Badge o Wireless Indoor Positioning System (WIPS). Esistono anche prototipi simili che si basano su emissioni di segnali infrarossi e segnali ultrasuono. Anche sono utilizzati tag che emettono e ricevono segnali radio (RFID), rilevabili a distanze nell'ordine dei centimetri.

2.3.2 Lateration

In questa tecnica si può partire dalla conoscenza sia della copertura come delle differenze di coperture di almeno tre stazioni base vicine al terminale in studio. Questa informazione è utilizzata per definire un sistema di n equazioni non lineari, che servono a determinare la posizione del terminale, dove n denota il numero di stazioni di base. Se il posizionamento è basato direttamente sulle coperture, si parla di *lateration circolare* (figura 2.3) poiché il problema a risolvere è l'intersezione dei cerchi con raggio la copertura delle stazioni base limitata al terminale. Invece, se il posizionamento è basato sulla differenza delle coperture delle stazioni base, si parla di *lateration iperbolica* (figura 2.4).

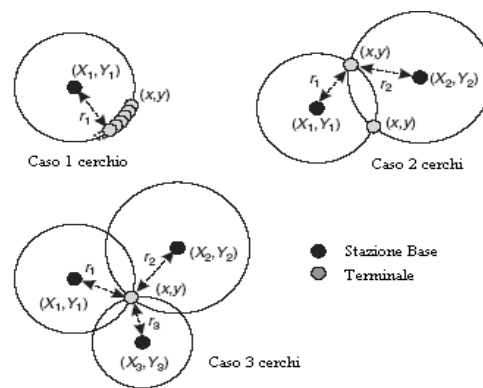


Figura 2.3 Lateration Circolare

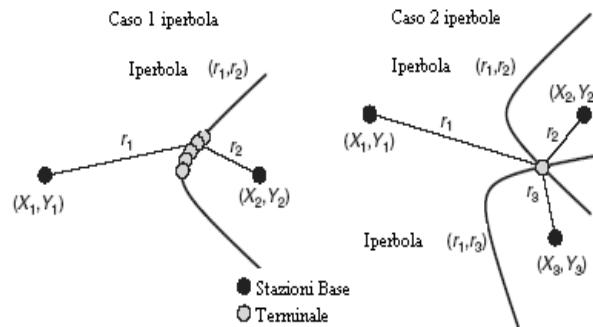


Figura 2.4: Lateration Iperbolica.

Adesso viene dettagliata la *lateration* circolare, per essere quella più impiegata nei sistemi di localizzazione.

Nel caso di *lateration* circolare si assume che le distanze tra il terminale e le stazioni base sono conosciute e denotate come r_i con $i=1,\dots,n$, dove n è il numero di stazioni base. Come si vede nella figura 2.3, conoscendo i diversi r_i , è possibile rappresentare tutti i punti che distano r_i da ogni stazione base, mediante il raggio di un cerchio. Se viene preso in considerazione più di una stazione base, le soluzioni possibili vengono ridotte, per l'intersezione degli altri cerchi. Bastano 3 stazioni base, quindi r_i con $i=1,2,3$; per avere una soluzione univoca al problema di posizionamento. Questa procedura anche viene chiamata triangolazione, perché servono 3 stazioni base per determinare le coordinate del punto in studio.

Per il calcolo della posizione del terminale, viene utilizzato il teorema di Pitagora. Se le coordinate dei punti di riferimenti, cioè, la posizione delle stazioni basi in studio, rispetto a un sistema di coordinate cartesiano rispetto sono (X_i, Y_i) , allora, il problema di posizionamento viene determinato, della risoluzione delle seguenti equazioni:

$$r_i = \sqrt{(X_i - x)^2 + (Y_i - y)^2}, \text{ per } i=1,2,3$$

Dove (x,y) , sono le coordinate a determinare del terminale e r_i rappresentano le equazioni dei diversi cerchi intorno al punto in studio. Nel caso , le equazioni a risolvere vengono determinate così:

$$\begin{aligned} A &= (X_1 - x)^2 + (Y_1 - y)^2 = r_1^2 \\ B &= (X_2 - x)^2 + (Y_2 - y)^2 = r_2^2 \\ C &= (X_3 - x)^2 + (Y_3 - y)^2 = r_3^2 \end{aligned}$$

Allora, queste 3 equazioni possono essere ridotte in 2 equazioni della forma:

$$A - B = (X_1 - x)^2 + (Y_1 - y)^2 = r_1^2 -$$

$$(X_2 - x)^2 + (Y_2 - y)^2 = r_2^2$$

$$A - B = ax + by = e$$

$$B - C = (X_2 - x)^2 + (Y_2 - y)^2 = r_2^2 -$$

$$(X_3 - x)^2 + (Y_3 - y)^2 = r_3^2$$

$$B - C = cx + dy = f$$

Dove le costanti a,b,c,d,e,f sono calcolate a partire delle procedure precedenti di somma, moltiplicazione e differenza. Dunque, ci arriva a un sistema di 2 equazioni lineari con 2 incognite, facilmente risolvibile utilizzando il teorema di Cramer apposto per questi casi:

$$x = \frac{\text{Det}\begin{pmatrix} e & b \\ f & c \end{pmatrix}}{\text{Det}\begin{pmatrix} a & b \\ c & d \end{pmatrix}}, y = \frac{\text{Det}\begin{pmatrix} a & e \\ c & f \end{pmatrix}}{\text{Det}\begin{pmatrix} a & b \\ c & d \end{pmatrix}}$$

Sostituendo con le rispettive costanti, diventa un semplice problema di determinanti di matrici.

Questa procedura può essere estesa al caso tridimensionale, con la differenza che entra in gioco una terza coordinata, e si parla di sfere nello spazio che si intersecano tra di loro, determinando la posizione (o le coordinate) del punto in studio. In questo caso è più pratico lavorare con coordinate ellittiche, facendo riferimento alla latitudine e longitudine del sistema terrestre.

2.3.3 Angle of Arrival

E' un'altra tecnica per stimare la posizione di un terminale, dalle coordinate di diverse stazioni base. La differenza rispetto alla tecnica *lateration*, è che qui sono misurati gli angoli tra il terminale e un certo numero di stazioni base. Questa tecnica è anche conosciuta come AoA (Angolo di Arrivo) o DoA (Direzione di Arrivo). Per determinare questi angoli, è necessario che le stazioni base come i terminali siano dotati di *array* di antenne, secondo il tipo di posizionamento (basato sul terminale o basato sulla rete). Tuttavia, oggi, per ragioni di complessità e costi, si preferisce una soluzione orientata alla rete, per la facilità di gestire *array* di antenne nelle stazioni base piuttosto che nei terminali.

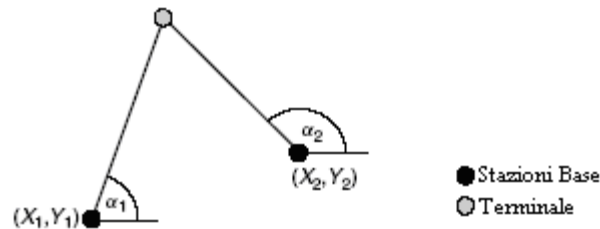


Figura 2.5: Principio AoA.

Il principio di base dietro all'angolazione è illustrato nella figura 2.5. L'angolo di arrivo di un segnale pilota in ingresso è misurato rispetto alla stazione base e una linea che unisce al terminale nella sua posizione istantanea. Se l'angolo di una seconda stazione base è preso in considerazione, si definisce un'altra linea che unisce il terminale con questa seconda stazione base.

L'intersezione di entrambe le linee rappresenta quindi la posizione del terminale. Così, da un punto di vista teorico, è sufficiente per rendere questa tecnica fattibile, misurazioni su due stazioni base, con lo scopo di ottenere una posizione in un piano bidimensionale.

2.3.4 Time of Arrival

In questa tecnica, si applicano i principi di *lateration* sul ritardo che subisce il segnale, quindi si presenta un caso di *lateration* circolare quando vengono presi direttamente i ritardi (Tempo di Arrivo: ToA), oppure si presenta una *lateration* iperbolica quando vengono presi la differenza tra questi ritardi (Tempo Differenziale di Arrivo: TDoA). Anche qua si può parlare di posizionamento orientato al terminale o alla rete, dipendendo del verso in cui si fanno le misurazioni dei ritardi: *downlink* o *uplink*. Nel caso *downlink*, il terminale esamina i segnali pilota ricevuti dalle stazioni base. Invece, nel caso *uplink*, la rete, più precisamente, le stazioni base, ricevono il singolo segnale pilota proveniente dal terminale. Questi segnali pilota possono essere infrarossi, ultrasuoni, però, quelli che vengono utilizzati di più sono i segnali radio.

I metodi utilizzati per determinare i ritardi sono:

- *Pulse ranging*: viene misurato il tempo di propagazione di un singolo impulso. Per il caso ToA deve essere noto il momento in cui è trasmesso questo impulso. Invece, per il caso TDoA, non è necessario, perché il calcolo si fa in base alle differenze di tempo rispetto agli altri segnali arrivati.

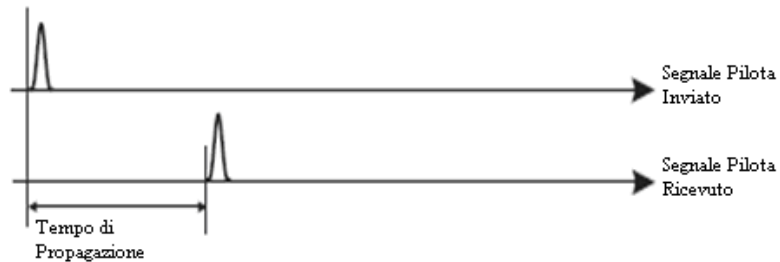


Figura 2.6: Misurazione del tempo di propagazione.

- *Carrier phase ranging*: in questo caso, il segnale ricevuto è confrontato in fase con un segnale di riferimento generato previamente per il

ricevitore. Questa differenza di fase viene a determinare il ritardo di propagazione.

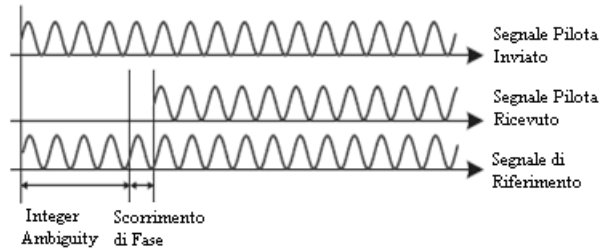


Figura 2.7: Confronto di fase (segnali analogici).

- *Code phase ranging*: anche qua si misura la differenza di fase del segnale ricevuto con un segnale di riferimento, la differenza con il metodo precedente è che i segnali sono modulati con un codice *spreading* noto, e quindi è necessario per il calcolo della fase, determinare la correlazione tra i due segnali. Il valore *Integer Ambiguity* riflette il numero di cicli del segnale di riferimento prima che aggancia il segnale pilota ricevuto.

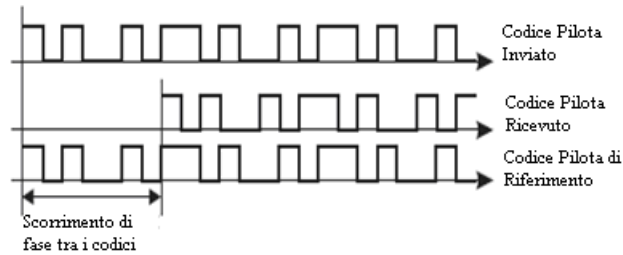


Figura 2.8: Confronto di fase (segnali numerici).

In tutti questi metodi, c'è una grossa dipendenza alla velocità di propagazione del segnale, per cui è necessaria un'esatta sincronizzazione tra gli orologi del terminale e la stazione base coinvolta per il caso ToA, e tra gli orologi delle diverse stazioni base coinvolte per il caso TDoA. La sincronizzazione è un

problema rilevante nel caso dei metodi basati sulla misura dei ritardi. Questo può essere evidenziato con un semplice calcolo, ponendo uno scorrimento nella sincronizzazione di 1 μ s, fra trasmettitore e ricevitore, supponendo che la velocità di propagazione del segnale è pari alla velocità di propagazione della luce si ha che $d = v * t = 1 \mu s * 300 E - 6 m / s = 300$ metri. Per un sistema di localizzazione indoor è poco gestibile un errore di circa 300 metri.

2.3.5 Received Signal Strength Indication

Una tecnica alternativa consiste in creare un registro con i livelli di potenza del segnale ricevuto (RSSI) per ogni stazione base, nei diversi punti fissi scelti nel luogo dove verrà implementata la localizzazione, e poi, mediante alcun approccio matematico, determinare la posizione a partire del valore di potenza istantaneo ricevuto sul terminale dalle diverse stazioni base. L'approccio matematico, tipicamente è una formula che mostra il legame tra i livelli di potenza del segnale e il percorso di propagazione affetto dai fenomeni spiegati nella sezione 1.2. Allora, siccome queste formule sono costituite di parametri che dipendono dalle caratteristiche di propagazione del mezzo che deve percorrere il segnale, è necessario uno studio sperimentale dell'ambiente, per determinare poi il comportamento del segnale in ogni caso e potere stabilire alcun modello di come si propaga il segnale in funzione delle attenuazioni subite.

Esistono, altri metodi per lavorare con i livelli RSSI ottenuti e che sono più accurati, poiché non dipendono dalla propagazione del segnale, la quale a sua volta può diventare un problema piuttosto che una soluzione al problema del posizionamento dovuto ai fenomeni di propagazione che sono difficili da modellare mediante una semplice formula. I metodi alternativi che impiegano RSSI, si basano fondamentalmente in un *pattern matching*, cioè, un confronto tra patroni fissi, cercando quelli che si approssimano di più a questi patroni, secondo una regola di

decisione che può essere di tipo probabilistica, deterministico, ecc. Su questi metodi si parlerà più in dettaglio nel capitolo 3.

Com'è visto, è possibile lavorare con RSSI in diversi modi, inoltre l'informazione relativa ai RSSI è più facile di ottenere in un'infrastruttura Wi-Fi qualsiasi. Sebbene, le tecniche che utilizzano i ritardi sono teoricamente più accurate e precise, nella pratica sono più ingestibile dal punto di vista hardware e software, oltre alla grande sensibilità ai fenomeni di propagazione, tale la riflessione, con serie ripercussione sull'errore di stima. Per queste ragioni, la scelta di questo progetto si orienta nell'utilizzo dei valori RSSI.

CAPITOLO III

ALGORITMI E METODI DI LOCALIZZAZIONE WI-FI IN AMBIENTE INDOOR

3.1 Introduzione

Nella sezione 2.3 si fa riferimento alle tecniche di posizionamento che possono essere impiegate per determinare le coordinate di un terminale provvisto di una scheda Wi-Fi, in un ambiente chiuso (*indoor*) coperto di una rete Wi-Fi, a partire di un insieme di parametri misurati. Questi parametri potrebbero essere il tempo, un angolo o un livello di potenza. Dipendendo dalla scelta, viene fuori una procedura matematica che determina la posizione. Per far ciò, la rete deve essere in grado di gestire questi parametri e disporre delle risorse hardware e software per prelevare i campioni di essi.

Esaminando i requisiti di questo progetto, basato sulla localizzazione Wi-Fi *indoor*, per mezzo di un'infrastruttura semplice, in molti casi, esistente sull'ambiente in studio, la principale restrizione la pongono le schede Wi-Fi disponibili (dei palmari e degli AP) e il sistema operativo su cui operano. In questo caso il sistema operativo in dotazione è Windows Mobile O.S., provvisto sui terminali mobili disponibili, PDA modello HP iPAQ serie hx2790b, e sui quali mediante la piattaforma di sviluppo Microsoft .NET Framework, specificamente librerie della versione compatta (CF), è stato possibile avere accesso all'informazione relativa ai livelli di potenza ricevuti dai diversi AP, i quali tipicamente indipendentemente del modello sono in grado di fornire l'informazione relativa alla potenza, cioè, i valori RSSI.

Allora, indipendentemente del metodo o l'algoritmo scelto, la tecnica che si potrebbe utilizzare, deve essere in grado di supportare questo parametro nei calcoli. RSSI lo fa, ma comunque, si potrebbe utilizzare un approccio ibrido, in cui se fosse in grado di arrivare ad altri parametri attraverso il livello di potenza ottenuto, magari utilizzando le formule di propagazione in ambiente *indoor*, con le quali potrebbe essere possibile determinare le distanze che servono alla tecnica *lateration*, per poi potere applicare una triangolazione. Come si vede, è una tecnica molto malleabile e che può essere diversificata secondo le risorse a disposizione e i parametri osservabili nella rete.

Adizionalmente all'infrastruttura WLAN, determinata per gli AP e i dispositivi terminali, si possono aggiungere sensori di prossimità, con lo scopo di rilevare con maggiore precisione la posizione del terminale in una determinata area. In questo progetto, infatti, sono utilizzati lettori (incorporati ai diversi dispositivi PDA) e schede RFID, soprattutto per individuare univocamente certi oggetti all'interno di un'area. Comunque, questi dispositivi, come sono implementati, non cercano posizionare il terminale, soltanto sono utilizzati per riconoscimento e identificazione di certi oggetti, quindi servono come complemento al servizio di localizzazione.

Allora, in questo caso, in cui è utilizzata la tecnica RSSI, bisogna avere una serie di valori di riferimento (punti *fingerprinting* o impronte digitali) che possono essere presi in una fase previa detta *offline*, e dei quali si conoscono le coordinate e i rispettivi RSSI. In seguito, quando è attivato il sistema di localizzazione, fase *online*, sulla misura della potenza ricevuta si fa una stima della posizione del terminale in funzione dei punti di riferimenti.

I criteri per valutare la qualità di un certo algoritmo o metodo di localizzazione sono:

- Accuratezza e precisione. Con accuratezza s'intende quanto distano i cosiddetti punti fissi più vicini al terminale (oppure la stima della posizione) dalla posizione reale del terminale. Invece, con precisione s'intende quanto distano i punti fissi più vicini al terminale (oppure la stima della posizione) dal valore medio. Questi sono considerati i parametri di qualità più importanti nei sistemi di localizzazione.
- Rendimento e consistenza. Con rendimento si intende l'abilità del metodo di localizzazione di ottenere le posizioni fisse in tutti tipi di ambienti, mentre la consistenza è una misura della stabilità dell'accuratezza in differenti ambienti.
- Overhead. Indica la quantità d'informazione scambiata tra il terminale e il sistema. Va legata direttamente al livello di accuratezza e precisione che si vuole raggiungere. Si può presentare per i meccanismi di segnalazione oppure per il processamento computazionale.
- Consumo di potenza. Si devono considerare le risorse energetiche disponibili per il terminale, inoltre bisogna studiare il consumo durante le operazioni di comunicazioni con il sistema e come si potrebbe risparmiare tale in momenti d'inattività.
- Latenza. Si riferisce all'intervallo di tempo tra una richiesta di localizzazione e la risposta del sistema in termini di una posizione fissa. Tipicamente per sistemi LBS è dell'ordine dei secondi.
- Costi roll-out e costi operativi. I costi roll-out sono quelli necessari per installare l'infrastruttura: le stazioni base, database, unità di controllo, oppure i costi per ampliare l'infrastruttura già disposta come succede nelle reti cellulari. Invece, i costi operativi sono legati alla complessità

del sostenimento e mantenimento dell'infrastruttura impiegata. Nel caso di localizzazione indoor sono minori rispetto al caso di localizzazione utilizzando satelliti.

Sono stati sviluppati diversi metodi e algoritmi a livello software, che a partire dei valori RSSI ottenuti, ritornano una stima sulla posizione dell'utente.

3.2 Metodo Nearest Neighbor in Signal Space (NNSS)^[6]

E' un metodo di localizzazione basato sul concetto delle impronte digitali, in altre parole, conoscendo i valori RSSI dei punti di riferimento di una determinata area, si può stimare la posizione di un terminale dotato di scheda Wi-Fi, che si trova nella stessa area.

Assumendo l punti di riferimenti, ognuno con la sua impronta F , le impronte possono essere rappresentate come $\{F_1, F_2, \dots, F_l\}$, mentre i punti sulla mappa possono essere rappresentati come $\{L_1, L_2, \dots, L_l\}$.

Nella fase *online*, il terminale prende un campione, che contiene i valori misurati di RSSI in quell'istante di tempo. Il terminale se si trova in un'infrastruttura WLAN, sicuramente vede più di un valore RSSI, corrispondenti alle stazioni base presenti in quell'area. Quindi, questo campione, sarebbe un vettore S con i diversi valori RSSI (un RSSI per ogni AP presente), che può essere espresso come $S = \{s_1, s_2, \dots, s_n\}^T$ dove n è il numero di AP presenti. Ogni impronta F associata a una determinata posizione i , può essere rappresentata come $F_i = \{p_1^i, p_2^i, \dots, p_N^i\}^T$, cioè un vettore composto dal valore medio RSSI di ogni AP presente. In pratica, questo metodo fa una semplice selezione dell'impronta più vicina al valore RSSI misurato correntemente. Per far ciò, si deve calcolare la minima distanza spaziale tra il valore

corrente e quelli salvati in fase *offline*. Utilizzando una regola di decisione per questi valori di distanza ottenuti, si può arrivare a quell'impronta che si avvicina di più al valore corrente e quindi questo valore sarebbe la stima della posizione in quell'istante di tempo. Questa regola può essere espressa come segue:

$$D(S, F_j) \leq D(S, F_k), \forall k \neq j$$

Dove, S, si riferisce al vettore con i valori RSSI ottenuti dai diversi AP, F_j sarebbe l'impronta nella postazione corrente, F_k rappresenta le diverse impronte nelle diverse postazioni misurate in fase *offline*, e D rappresenta una funzione che calcola la distanza spaziale.

Uno dei metodi più utilizzati per il calcolo di distanze spaziali è la differenza euclidea, determinata a partire della seguente equazione che espressa la differenza pesata nello spazio:

$$L_p = \frac{1}{N} \left(\sum_{i=1}^N \frac{1}{\omega_i} |s_i - p_i|^p \right)^{1/p}$$

Dove L_p rappresenta la differenza euclidea; N il numero di valori RSSI presenti al momento di calcolare la differenza, quindi, rappresenta il numero di AP presenti; s_i rappresenta l'impronta RSSI per quell'AP e p_i il valore corrente RSSI per lo stesso AP. Per il caso euclideo p = 2 e ω_i = 1. Invece per il caso denominato distanza di Manhattan, il valore di p = 1. Allora, per il caso della differenza euclidea tra le impronte e il valore corrente, è possibile riscrivere l'espressione precedente come:

$$D = \sqrt{(s_1 - p_1)^2 + (s_2 - p_2)^2 + \dots + (s_N - p_N)^2}$$

Uno svantaggio di questo metodo, sta nel fatto che ritorna come soluzione soltanto il valore più vicino, quindi, nel caso in cui questo sia un valore sbagliato, non si può dire più niente sulla posizione, e non c'è modo di agganciare almeno una stima approssimata di questa, perciò bisogna aumentare il *range* di valori più vicini, in maniera tale di avere un insieme di possibili posizioni, dopodiché si prendono questi valori e mediante il calcolo della media, si ottiene la posizione corrente. Ovviamente, più dispersi sono i punti in questo insieme, più errata sarà la stima, quindi l'insieme non deve essere troppo grande.

3.3 Metodo Probabilistico^[6]

Un approccio probabilistico cerca di stimare la posizione in funzione delle impronte digitali ottenute in fase *offline*, utilizzando le definizioni di probabilità condizionata e il teorema di Bayes. Per far ciò, è necessario conoscere la distribuzione di probabilità che meglio modella i valori RSS ottenuti. Questo si potrebbe fare di due maniere, con i valori ottenuti in fase *offline* oppure mediante l'utilizzo di un modello di propagazione per stimare il comportamento del segnale radio.

Per ogni posizione L sulla mappa, si può stimare la funzione densità di probabilità condizionata $P(F|L)$ a partire di un numero elevato di campioni contenenti i valori RSSI che definiscono l'impronta di L . Esistono diverse maniere per determinare la distribuzione di probabilità, la più semplice è la costruzione di un istogramma con i valori RSSI prelevati da ogni AP in ogni punto. Quindi la probabilità di un determinato valore di segnale (o insieme visto che sicuramente ci saranno più di un AP rilevato) di appartenere a una determinata posizione, cioè $P(s|L)$, sarà valutato in funzione della frequenza di apparizione dei valori RSSI dei punti campionati associati a uno stesso AP.

Un'altra possibilità per definire la distribuzione di probabilità, è pensare ad una funzione di distribuzione Gaussiana. In pratica, quello che si fa, è prendere n campioni di RSS per uno stesso AP su quella posizione e poi assumere che hanno distribuzione Gaussiana con media μ e deviazione standard σ . Assumendo poi, uno spazio unidimensionale, è possibile descrivere una funzione di massima verosimiglianza come la somma dei n pesi, assunti come uguali tra di loro, quindi l'equazione risultante è:

$$P(s | L) = \frac{1}{n} \sum_{i=1}^n \left[\frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(s - p_i)^2}{2\sigma^2}\right) \right]$$

In questa equazione s rappresenta il valore RSSI corrente e p_i rappresenta il valore RSSI di un punto campionato in fase *offline*, per uno stesso AP. Prendendo in considerazione multipli AP e assumendo che sono indipendenti tra di loro, si può stimare la probabilità condizionata $P(F|L) = P(s_1|L)P(s_2|L) \dots P(s_N|L)$, di maniera tale di potere stimare la somiglianza di una data impronta con la posizione corrente. Poi è possibile stimare la distribuzione a posteriore, della posizione corrente, condizionata all'impronta. Assumendo che la probabilità a priori $P(L)$ per ogni posizione è nota (inizialmente potrebbero essere tutte uguali), si può applicare il teorema di Bayes per trovare la probabilità condizionata della posizione L , data un'impronta F , come si mostra nella seguente espressione:

$$P(L | F) = \frac{P(F | L)P(L)}{P(F)} = \frac{P(F | L)P(L)}{\sum_{k \in L} P(F | L_k)P(L_k)}$$

Di conseguenza, questo metodo potrebbe scegliere la posizione dell'impronta con la stima più alta della probabilità a posteriore. Assumendo che esistono due posizioni con le sue rispettive impronte $A(L_A)$ e $B(L_B)$. La regola di decisione di Bayes potrebbe scegliere A anziché B poiché:

$$P(L_A|F) > P(L_B|F)$$

$$P(F|L_A)P(L_A) > P(F|L_B)P(L_B)$$

Con informazione addizionale riguardante la distribuzione delle locazioni L , questo metodo sarebbe utile per stimare la posizione. Tuttavia, saranno necessari una grande quantità di campioni durante la fase *offline* dei valori RSSI per ogni postazione per raggiungere un grado di accuratezza accettabile.

3.4 Metodo del Centroide [\[5\]](#), [\[7\]](#)

Questo metodo si basa nel raccogliere un insieme di valori RSSI, per ogni punto fisicamente ubicato sulla mappa, e poi determinare la distribuzione probabilistica normalizzata a partire del peso dei valori RSSI associati a ogni punto. L'algoritmo stima la posizione in funzione della distribuzione di RSSI dei punti misurati in fase *offline* e li confronta con i valori RSSI misurati correntemente. Quelli più vicini determinano, per mezzo del calcolo di una media ponderata, la posizione stimata del terminale.

Allora, sia $p(i)$ la probabilità della posizione x_i appartenente ad un insieme \bar{X} , il quale contiene la lista di tutte le coppie di coordinate dei punti campionati, ordinate in modo decrescente in base alla probabilità ad esse associata data dalla seguente relazione

$$P = \frac{\sum_{i=1}^N p(i) \bar{X}}{\sum_{i=1}^N p(i)}$$

dove, x_i corrisponde all' i -esimo valore dell'insieme \bar{X} e N il numero totale di elementi dell'insieme. Per determinare la probabilità $p(i)$, può essere utilizzata la funzione di distribuzione Gaussiana, ovvero:

$$p(i) = \prod_{j=0}^M \left[\frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{s_j - \bar{s}_j}{2\sigma^2}\right)^2 \right]$$

dove, M rappresenta il numero di AP, σ la deviazione standard, s_j il valore di potenza ricevuto dal j -esimo AP ed infine \bar{s}_j rappresenta la media della potenza ricevuta dai campioni nella fase *offline* dall'AP j -esimo.

3.5 Metodi di Filtraggio ^{[8], [9]}

Nei sistemi di posizionamento *indoor*, le fluttuazioni dei valori RSSI portano certi problemi nell'accuratezza della stima. Quando il terminale ha bisogno di essere tracciato continuamente, i filtri possono essere utili per aggiustarsi ai percorsi seguiti e così ridurre l'errore nella stima.

Un filtro di Kalman è comunemente utilizzato in applicazioni di tracciamento. Lo svantaggio di questo filtro è dato principalmente per il fatto che definisce un modello lineare, il quale non si avvicina al comportamento reale del parametro in studio, che in questo caso si riferisce al segnale descritto mediante diversi valori RSSI (livelli di potenza) e che dipendono del canale radio il quale non ha un comportamento lineare nel tempo. Esistono due versioni del filtro di Kalman che cercano di allineare i comportamenti non lineari di sistemi che stimano la posizione, questi sono: Extended Kalman Filter (EKF) e Unscented Kalman Filter (UKF). La versione estesa EKF utilizza un approccio differenziale, invece la UKF

utilizza trasformazioni deterministiche intorno ad una variabile aleatoria. Tuttavia sono più affidabili in quei sistemi che hanno un comportamento lineare. Inoltre per queste versioni bisogna fare una procedura di integrazione per ottenere la traccia della posizione, lo quale potrebbe diventare un calcolo molto oneroso dal punto di vista computazionale.

Un'alternativa sarebbe utilizzare un particle filter, conosciuto anche come un metodo sequenziale di tipo Monte Carlo. Le operazioni che gestisce sono più semplici e può essere implementato per risolvere problemi di natura non lineare e di distribuzioni non Gaussiane. Considerando informazioni addizionali come la distanza percorsa e informazione sulla mappa, si potrebbe migliorare le prestazioni di questo filtro. Si assume che l'informazione sulla mappa è non lineare e distribuita nello spazio, mentre la distanza percorsa caratterizza solo una parte del comportamento del movimento, per esempio, potrebbe non considerarsi l'informazione sull'orientazione. A continuazione, sono specificati come operano questi filtri dal punto di vista matematico e fisico.

3.5.1 Algoritmo Filtro di Kalman

Il Filtro di Kalman, che si mostra è del tipo EKF, e modella un processo sotto controllo in tempo discreto, utilizzando le seguenti equazioni lineari differenziali stocastiche, per un dato vettore stato $x(k)$, un dato vettore misura $z(k)$ e un vettore controllo $u(k)$:

$$\begin{aligned}x(k) &= Fx(k-1) + Bu(k) + w(k-1) \\z(k) &= Hx(k) + v(k)\end{aligned}$$

Dove le matrici F , B e H , definiscono la transizione lineare e la fase di misurazione, mentre i vettori aleatori w e v rappresentano il processo e il rumore

misurato rispettivamente, che sono assunti come distribuzioni normali [N(media, deviazione standard)], bianche e indipendenti, cioè:

$$p(n) \propto N(0, Q)$$

$$p(v) \propto N(0, R)$$

Dove le matrici Q e R sono le covarianze dell'errore sullo stato e dell'errore sulle misure, rispettivamente. Per un sistema basato sull'utilizzo dei valori RSSI, si possono definire i parametri come segue:

$$x(k) = \begin{bmatrix} x(k) \\ y(k) \\ v_x(k) \\ v_y(k) \end{bmatrix}, z(k) = \begin{bmatrix} x_{RSS}(k) \\ y_{RSS}(k) \end{bmatrix}$$

$$F = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, B = 0, H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

$$w(k) = \begin{bmatrix} \frac{\Delta t^2}{2} & 0 \\ 0 & \frac{\Delta t^2}{2} \\ \Delta t & 0 \\ 0 & \Delta t \end{bmatrix} \begin{bmatrix} a_x(k) \\ a_y(k) \end{bmatrix}, v(k) = \begin{bmatrix} e_x(k) \\ e_y(k) \end{bmatrix}$$

Dove x e y sono le coordinate della posizione sugli assi x e y rispettivamente; v_x e v_y rappresentano le velocità rispetto agli assi x e y; a_x e a_y rappresentano le accelerazioni rispetto agli assi x e y, e sono legate a una distribuzione aleatoria con le stesse caratteristiche del rumore; z è la posizione stimata a partire dalle misure dei valori RSSI. Allora per stimare lo stato \hat{x} può essere calcolato utilizzando le seguenti equazioni ritorsive di predizione e correzione:

$$\begin{aligned}\hat{x}^-(k) &= F\hat{x}(k-1), \\ P^-(k) &= FP(k-1)F^T + Q, \\ K(k) &= P^-(k)H^T (HP^-(k)H^T + R)^{-1}, \\ \hat{x}(k) &= \hat{x}^-(k) + K(k)(z(k) - H\hat{x}^-(k)), \\ P(k) &= (I - K(k)H)P^-(k)\end{aligned}$$

Dove $P^-(k)$ è la matrice di covarianza corrispondente alla predizione dello stato, mentre $P(k)$ è la matrice di covarianza, corrispondente allo stato stimato considerando anche le misure correnti dei valori RSSI.

La seguente figura mostra, mostra uno schema di come funziona il filtro di Kalman. Si divide in due strati, quello visibile dove stanno i parametri esterni al sistema, e quello nascosto derivato dalla procedura di calcolo che modella il funzionamento del filtro secondo lo stato.

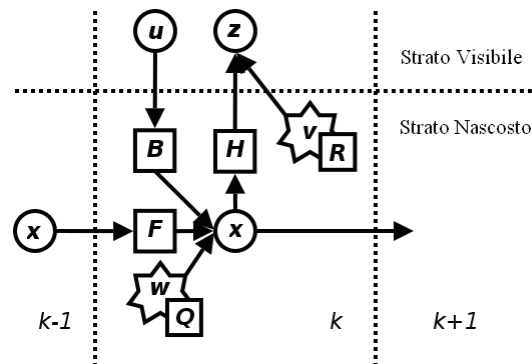


Figura 3.1: Schema semplificato di un filtro di Kalman.

3.5.2 Algoritmo Particle Filter

Nel caso di filtro di particelle, a differenza del filtro di Kalman, si fa una stima a partire della funzione densità di probabilità (fdp) a posteriori dello stato $x(k)$ data le osservazioni precedenti di $Z(k)$ utilizzando la seguente equazione:

$$p(x(k) | Z(k)) \approx \sum_{i=1}^N w^i(k) \delta(x(k) - x^i(k))$$

Dove $x^i(k)$ è il campione i -esimo del punto o della particella della probabilità a posteriore, mentre w^i è il peso della particella.

Esistono 2 forme di implementare un algoritmo basato sul filtro di particelle: *Sampling Importance Resampling* (SIR) e *Sequential Importance Sampling* (SIS). La differenza ci sta nella fase di *resampling* fatta soltanto nel caso SIR, e per questa generalità non si entrerà nel dettaglio dell'algoritmo SIS.

A continuazione, si descrive di maniera molto semplice, i passi di un algoritmo generico SIR:

- Inizializzazione: sono campionate N particelle $\{x^i(0), i = 1 \dots N\}$ a partire della fdp iniziale $p(x(0))$, che generalmente all'inizio, rende a tutte le particelle uniformemente distribuite, cioè, con la stessa probabilità.
- Predizione: per ogni particella $x^i(k)$, data una nuova particella $x^i(k+1)$ a partire della fdp transitoria $p(x(k+1)|x^i(k))$.
- *Importance Sampling*: per ogni nuova particella $x^i(k+1)$, si calcola $w^i(k+1) = p(z(k+1)|x^i(k+1))$.

- Normalizzazione e *Resampling*: i pesi sono normalizzati e finalmente *re-sampled*. Nel passo di *resampling*, le particelle con peso più basso sono cancellate e quelle con peso più alto sono raddoppiate, facendo così che ogni particella abbia lo stesso peso.

Un'osservazione rispetto alla fdp transitoria $p(x(k+1)|x^i(k))$ ed il suo aggiornamento $p(z(k+1)|x^i(k+1))$, è che devono essere funzioni note, e non necessariamente devono essere di tipo Gaussiano. Per il caso RSSI, i pesi possono essere determinati tramite le seguenti equazioni:

$$w^i(k+1) = p(z(k+1) | x^i(k+1)) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\left[\frac{(x^i(k+1) - x_{RSSI}(k+1))^2 + (y^i(k+1) - y_{RSSI}(k+1))^2}{2\sigma^2} \right]}$$

In questa equazione è assunta che la posizione stimata attraverso i valori RSSI è una distribuzione Gaussiana con media la posizione esatta oppure una stima di essa.

Quello che manca per definire, è il modello dinamico^[10] che rappresenta il comportamento delle particelle su in un sistema di coordinate cartesiane.

$$X_i = X_{i-1} + V_i \Delta t \cos(\alpha_i) + n_{i-1}$$

$$Y_i = Y_{i-1} + V_i \Delta t \sin(\alpha_i) + n_{i-1}$$

$$v_i = \left| N(v_{i-1}, 1ms^{-2} \Delta t) \right|, \text{ con } v = [0, 10ms^{-1}]$$

$$\alpha_i = N(\alpha_{i-1}, 2\pi - a \tan\left(\frac{\sqrt{v_i}}{t}\right) \Delta t), \text{ con } \alpha = [0, 2\pi]$$

Dove le coordinate (X_i, Y_i) descrivono le posizioni delle particelle in un determinato momento; v_t e α_t rappresentano la velocità e la direzione delle particelle all'istante t ; n_{t-1} rappresenta il rumore del sistema con distribuzione Gaussiana. Questi valori di v_t e α_t , dovrebbero essere ottenuti mediante sensori di misura dell'inerzia, e per questo le formule mostrate, rappresentano soltanto un modello euristico del suo comportamento.

3.6 Reti Neurali Artificiali^[11]

Le Reti Neurali Artificiali (RNA) o semplicemente Reti Neurali sono sistemi di processamento dell'informazione, distribuiti e paralleli, ispirati alla struttura e al funzionamento delle reti neurali biologiche. Questi sistemi sono concepiti per emulare certe funzioni eseguite dal cervello umano, quali riconoscimento dei patterns, classificazioni, apprendimento, memoria e l'astrazione della conoscenza. ecc. Queste reti sono formate dall'insieme di neuroni artificiali collegati tra loro in diverse modalità, dove ogni collegamento è caratterizzato da un peso per il neurone che sta a valle verso il neurone a monte. Questi neuroni rappresentano degli eventi esterni. Le grandezze, mostrate nella figura 3.2, caratterizzano una rete neurale generica.

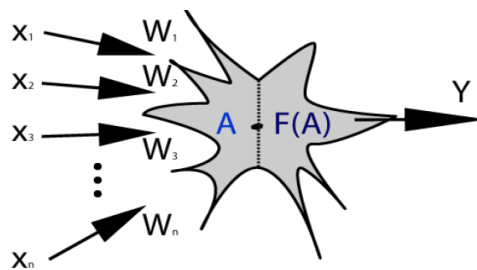


Figura 3.2: Schema di una Rete Neurale Artificiale.

- Il vettore $X = (X_1, \dots, X_n)$ rappresenta il vettore degli ingressi al neurone che possono essere a loro volta le uscite di altri neuroni quando sono presenti

delle sinapsi. Il vettore $W = (W_1 \dots W_n)$ è il vettore dei pesi, che rappresenta cioè il peso da attribuire alla sinapsi corrispondente.

- A non è altro che la Funzione di attivazione che è la funzione che stabilisce il livello di attivazione del neurone artificiale come conseguenza degli ingressi ricevuti. In corrispondenza al piano biologico questa funzione identifica gli insiemi dei valori che assume il Potenziale di Membrana a Riposo (PMR) come conseguenza della sommatoria spazio-temporale dei potenziali postsinaptici. Le funzioni frequentemente utilizzate sono: funzione identità, funzione rampa, funzione Gaussiana, ecc.

- $F(A)$ invece è la funzione di propagazione che stabilisce quando il neurone è autorizzato a dare un'uscita, corrisponde quindi alla funzione di uscita.

Le reti neurali competitive costituiscono un'importante variante delle reti neurali artificiali in cui la rete cerca di scoprire caratteristiche e pattern speciali dei dati di ingresso, senza che ci sia alcuna supervisione esterna. Questo tipo di rete utilizza algoritmi d'apprendimento senza supervisione per eseguire compiti come:

- Clustering. Il clustering (raggruppamento) consiste in raggruppare o dividere un insieme di dati in sottoinsiemi separati (cluster), in modo tale che i pattern appartenenti a uno stesso cluster siano molto simili fra essi, mentre i pattern appartenenti a cluster diversi siano il più possibile differenti.

- Vector quantization. Questo tipo di compito è molto simile al clustering. Lo spazio di entrata è diviso in diverse regioni connesse. Ciascuna regione è rappresentata utilizzando un unico vettore. Ciascun pattern nello spazio di entrata appartiene a una regione particolare (quella che possiede il vettore più vicino al pattern).

- Distribuzione di probabilità. Il metodo consiste in ottenere una descrizione riassuntiva di grandi volumi di dati estratti da una distribuzione di probabilità. La descrizione potrebbe consistere in un insieme di punti dello spazio di dati, tale che la distribuzione di questi punti costituisca un'approssimazione della distribuzione originale.
- Estrazione di caratteristiche. Il metodo consiste in scoprire le caratteristiche più importanti (cioè quelle che possiedono maggiore variazione) nello spazio di dati. Seguendo questo metodo, pattern appartenenti a cluster differenti possono essere individuati da alcune caratteristiche comuni. L'estrazione di caratteristiche porta a una riduzione della dimensione dei pattern di entrata.

Queste caratteristiche, permettono di affermare l'utilità delle reti neurali, nella gestione di un sistema di localizzazione, dove i neuroni in questo caso rappresentano il legame tra un determinato punto noto in fase *offline* mediante le sue coordinate e mediante i suoi valori RSSI. L'approccio a strati, agevola l'analisi della stima e potrebbe aumentare la precisione del sistema giacché a differenza degli algoritmi precedenti, non si basa soltanto in una procedura matematica per stimare la posizione, anzi, prima di ritornare la posizione, si devono eseguire una serie di analisi, che man mano, riducono l'insieme di possibile soluzioni, e poi questi risultati potrebbero essere utilizzati per aggiornare l'informazione del sistema oppure aggiustarsi alle condizioni reali dell'ambiente in cui si implementa, quindi questi punti fissi, potrebbero essere cancellati o aggiornati in funzione delle caratteristiche aggiuntive di analisi della rete neurali.

Tra queste caratteristiche aggiuntive si ha, l'evoluzione e l'apprendimento, che si riferiscono alla capacità della rete di cambiare per conto proprio e la capacità di

adattarsi al tempo, ed anche di reagire fronte a situazioni già rilevate precedentemente. Esempio di reti neurali sotto questi approcci sono:

- Le reti winner-take-all costituiscono il principale approccio per eseguire l'apprendimento competitivo. Questo meccanismo presuppone l'utilizzo dello spazio neurale (strati di neuroni) per la rappresentazione delle categorie, le quali sono organizzate su questo spazio in una maniera spontanea e data dall'interazione con gli eventi esterni. L'idea base di questo principio di organizzazione è la seguente: i neuroni o insieme di neuroni che rappresentano una porzione dello spazio neurale, competono fra essi per rispondere alla presentazione di una particolare categoria di eventi esterni. La gara si basa sui diversi livelli di attività raggiunti dai diversi neuroni, data la presentazione di un evento esterno, nonché in una struttura di inibizione laterale esistente fra neuroni.

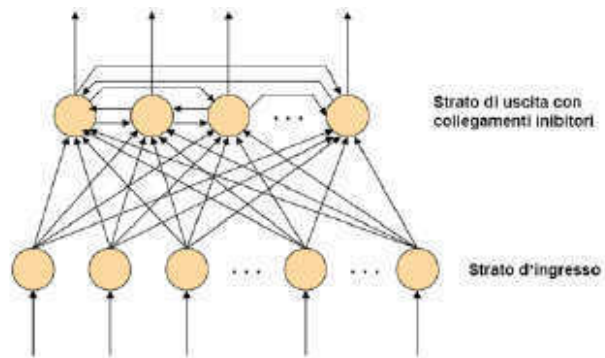


Figura 3.3: Schema di una Rete Neurale in modalità winner-take-all.

- Le reti SOM (Self-Organizing Map), combinano un principio di apprendimento competitivo con una strutturazione topologica dei neuroni, in modo che i neuroni adiacenti tendano ad avere vettori di pesi simili. Nella rete SOM i neuroni tendono ad avere vettori di pesi che prendono le caratteristiche dello spazio vettoriale di entrata. La relazione di vicinanza si

traduce nella prossimità fra i neuroni nello spazio Euclideo, pur essendo i vettori di pesi iniziali arbitrari.

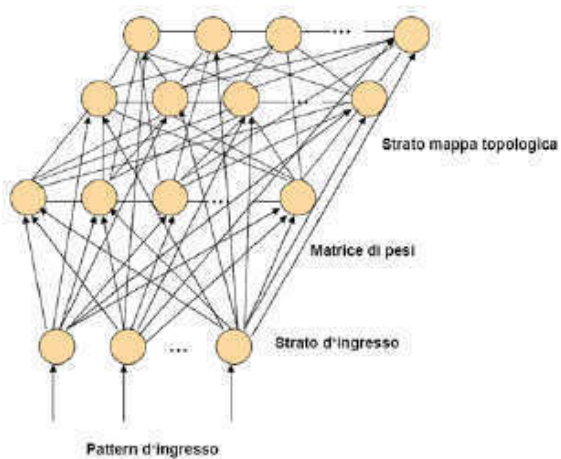


Figura 3.4: Schema di una Rete Neurale Artificiale in modalità SOM.

CAPITOLO IV

TEST DEI METODI DI LOCALIZZAZIONE WI-FI

4.1 Specifiche generali del sistema

Questa tesi riguarda la progettazione e la realizzazione di un sistema che permetta di stimare la posizione di un apparato radiomobile portatile, in specifico un palmare, all'interno di un ambiente chiuso, basandosi sulla connessione radio dell'apparato stesso a un numero ristretto di AP con tecnologia Wi-Fi presenti all'interno dell'ambiente.

L'infrastruttura impiegata è del tipo *network-based*, cioè la rete è chi esegue la localizzazione a partire dei valori misurati di RSSI. A questo scopo è utilizzato un server remoto il quale gestisce le funzioni di rete: esecuzione dell'algoritmo di posizionamento, salvataggio dei punti campionati, salvataggio delle mappe, collegamenti, ecc.

Per dare inizio alla progettazione, è necessario definire le risorse hardware e software a disposizione, le quali influenzano direttamente lo scenario e l'analisi delle soluzioni possibili al problema del posizionamento indoor. In linee generali il sistema è costituito da:

- I terminali mobili sono palmari provvisti di scheda Wi-Fi (compatibili almeno con lo standard IEEE 802.11b), operando con Windows Mobile O.S.

- La rete Wi-Fi, è costituita da diversi dispositivi AP compatibili con lo standard IEEE 802.11b. Questi devono essere in grado di restituire il RSSI, il Service Set Identifier (SSID) e l'indirizzo MAC.
- La piattaforma di sviluppo SW, è Microsoft .NET Framework, in ambiente Visual Studio 2005, sotto Windows XP Pro O.S.
- Un server remoto di alte prestazioni per lavorare con le diverse applicazioni in gioco (Processore Core Duo, memoria RAM superiore a 2 Gb, scheda di rete LAN, unità Disco Rigido, unità CD/DVD-ROM, porte USB, altre porte e uscite standard). Sistema operativo in corso Windows Server 2003, compatibile con la piattaforma .NET Framework.

Com'è stato spiegato all'inizio del capitolo 3, si deve lavorare con i valori RSSI originati dagli AP che conformano la rete, quindi serve un'applicazione che sia in grado di campionare dal dispositivo terminale i valori RSSI che li arrivano in qualsiasi istante di tempo. A questo scopo, è stata sviluppata un'applicazione (chiamata "Campionatura") in azienda, sulla piattaforma Microsoft .NET, eseguibile sui palmari con Windows Mobile OS, e la quale ha le seguenti caratteristiche:

- Un'interfaccia grafica, Mappa, dove sono caricate le mappe in formato SVG, delle zone dove si desidera implementare la localizzazione. Sulle mappe a disposizione c'è la possibilità di selezionare i punti dove si vuole campionare, data la possibilità di navigazione sulla mappa stessa. Per ogni punto scelto a caso, con una precisione di 10 pixel (valore di *default*), si può associare un nome, dopodiché comincia la campionatura. In caso di errore, c'è la possibilità di eliminare i punti, oppure il suo aggiornamento con più campioni.

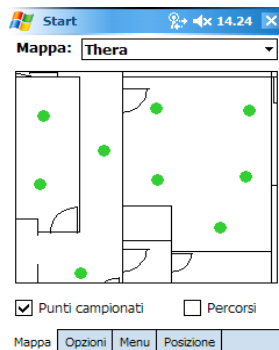


Figura 4.1: Schermata Mappa dell'applicazione Campionatura.exe.

- Un'etichetta con le Opzioni che possono essere configurate: un “Filtro AP” che controlla l'identificativo SSID degli AP, in modo tale di campionare solo quelli associati alla stessa rete; il “Numero di campionature”, che può variare in decine e va da 10 fino 100 campionature; “Intervallo campionature”, che stabilisce l'intervallo di attesa tra un campione e quello successivo, e può variare da 1 secondo fino 10 secondi; uno “Zoom mappa”, per ingrandire o diminuire le dimensioni delle mappe sullo schermo del palmare, varia di un fattore di 1x (valore di *default*) fino a 5x; la “Grandezza del punto”, che stabilisce quanti pixel coprono i punti sullo schermo, che può essere da 1 fino a 10 oppure impostato per default in Auto; la “tolleranza del clic”, che stabilisce quanto è grande in pixel i punti segnati sottoforma di palline e va da 1 pixel fino a 10 pixel; poi è specificata una leggenda per differenziare un punto da campionare, un punto campionato oppure un punto annullato.

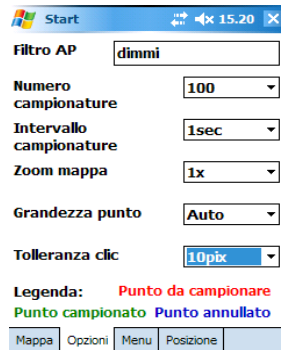


Figura 4.2: Schermata Opzioni dell'applicazione Campionatura.exe.

– Poi c'è l'etichetta Menu con le seguenti opzioni: indirizzo IP associato al server remoto (WS, che significa Web Server, termine utilizzato nell'intorno .NET, per riferirsi a un codice eseguito da remoto mediante pubblicazione in un server locale) dove sono archiviati i punti dopo che sono stati salvati durante il processo di campionatura; indirizzo IP associato al computer dove si trova l'algoritmo di stima della posizione (POS), in fase di test si riferisce al computer di sviluppo (dove sarà possibile eseguire il codice in step, analisi degli errori, correzioni, raffinamenti, ecc.) mentre in fase finale farà riferimento al server remoto stesso; c'è un'opzione "Sincronizza mappe", che carica tutti i file di mappa che si trovano sul palmare; c'è un'opzione "Sincronizza punti campionati", che salva i punti dopo che sono stati campionati; e per ultimo l'opzione "Esci", per chiudere l'applicazione.

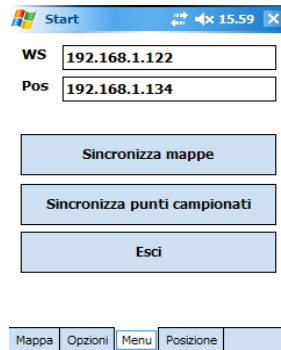


Figura 4.3: Schermata Menu dell'applicazione Campionatura.exe.

- Nell'ultima etichetta Posizione, si visualizza la mappa corrente dove è rappresentata la stima della posizione. E' un'interfaccia interattiva che permette di sapere la posizione del palmare in tempo reale e in funzione dei risultati di stima dei metodi di localizzazione impiegati.

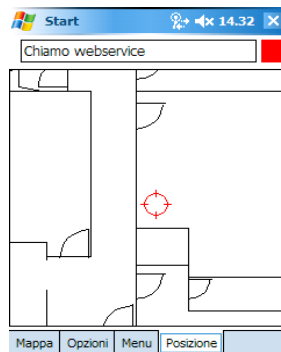


Figura 4.4: Schermata Posizione dell'applicazione Campionatura.exe.

Il palmare su cui è stata installata l'applicazione di campionatura è un HP modello iPAQ serie hx2700, simile a quello mostrato nella figura 4.5. Questo palmare è stato scelto, perché possiede una scheda Wi-Fi che li permette di collegarsi, ad una rete wireless. Include un slot Compact Flash, che permette l'utilizzo dei lettori RFID, compatibili con questa tecnologia. Inoltre, è compatibile con Windows Mobile

O.S., sul quale è possibile sviluppare applicazioni per questi dispositivi che lavorano sulla piattaforma .NET CF (Compact Framework).

Quindi in pratica, il palmare, attraverso l'applicazione di campionatura, è in grado di misurare in qualsiasi istante di tempo, il valore di potenza che gli arriva di un determinato AP. E' possibile che nello stesso momento sia in grado di misurare il livello di potenza di più di un AP oppure non registrare nessun valore per alcun AP. Quindi, non è una campionatura regolare, poiché dipende dalla risposta degli AP e del canale di propagazione che è variante in tempo.

L'applicazione di campionatura utilizza una libreria denominata OpenNETCF versione 2.0, con la quale è possibile lavorare attraverso Visual Studio 2005 e mediante linguaggio C#, e la quale è compatibile con dispositivi mobili che operano con Windows Mobile O.S., quindi è una versione compatibile con .NET Framework ma in versione compatta (CF). Grazie ad una estensione di questa libreria e utilizzando certe classi a disposizione, è stato possibile misurare la potenza che ricevuta dagli AP presenti nella rete Wi-Fi.

Il funzionamento a grosso modo dell'applicazione di campionatura sarà descritto a continuazione in pseudo codice:

```
while count1 < Scans Total    'Ciclo di durata numero di campionature impostato all'inizio (Scans Total)

                                Get_Adapter_Collection();    'Ritorna una Lista di
                                oggetti contenenti tutti i nome degli adattatori Wi-Fi
                                'presenti sul palmare (Adapter Name)

                                for count2=1:Number of adapters    'In questo ciclo si fa una ricerca degli adattatori di
                                rete
                                    'presenti nel palmare cercandosi quelli di tipo
                                    'IsWireless, cioè, di tipo Wi-Fi.
                                if Adapter Name==IsWireless
                                    Get_AdapterName();
```

```

Get_AccessPoint_Collection ()      'Ritorna una Lista di oggetti contenenti tutti i nomi degli adattatori Wi-
Fi che
                                     'vede il palmare (AccessPoint Name)

For count3=1:Number of AP's      'In questo ciclo si fa una ricerca degli AP's che
vede il
                                     'palmare cercandosi quelli con nome lo stesso impostato
                                     'sul Filtro AP

if AdapterName=="Filtro AP"
    Get_MAC();      'sono presi l'indirizzo MAC, la potenza, e si
                    stampa
    Print_Stamp;   'il valore corrente del tempo
    Get_Power();

```

Una volta che finisce di campionare, i valori sono salvati in formato XML e passati al server remoto, presentando la seguente struttura:

```

<?xml version="1.0" encoding="utf-8" ?>
- <WIFIScanResults>
- <Map Id="42">
- <Point Name="SCOL" X="445" Y="24">
  <Adapter Name="SWLD23N1" MAC="00027858D6EF0000" />
  <Scans Total="100" Done="98" />
- <AccessPoint Name="" MAC="001B2FA7F0EF" DIST="433,295511169917">
  <Timestamp>06/02/2008 17.12.14</Timestamp>
  <Power>-84</Power>
  </AccessPoint>
- <AccessPoint Name="" MAC="001B2FA5F20F" DIST="563,798722949955">
  <Timestamp>06/02/2008 17.12.14</Timestamp>
  <Power>-31</Power>
  </AccessPoint>
.
.
.
  </Point>
</Custom>
</WIFIScanResults>

```

Si può osservare, che in questo formato, l'identificativo SSID diventa *AccessPoint Name* per il caso degli AP e *Adapter Name* per il caso della scheda del palmare. Ci sono anche altre informazioni: *Point Name*, che sarebbe il nome

associato al punto campionato con le rispettive coordinate della loro posizione descritte su un sistema di riferimento cartesiano bidimensionale (X e Y); *Scans Total*, che sarebbe il numero di campionature, quelle impostate e quelle effettivamente compiute (*Done*); l'indirizzo *MAC* associato alla scheda Wi-Fi; la distanza *DIST* che c'è tra gli AP e i punti campionati; informazione temporale *Timestamp*, associata al momento in cui si fa la misura del livello di potenza *Power* associato all'AP.

L'infrastruttura rete Wi-Fi, è definita attraverso gli AP Wi-Fi (compatibili con IEEE 802.11b), di diversi fabbricanti e caratteristiche, ma che comunque apportano l'informazione necessaria che serve al processo di campionatura e di alla localizzazione. Questi parametri sono: SSID, l'indirizzo MAC associato alla scheda Wi-Fi dell'AP, ed il RSSI, cioè il livello di potenza del segnale radio dell'AP. Questi parametri vengono presi durante la campionatura e costituiscono una stringa che cambia in ogni intervallo di tempo. La struttura di questa stringa che poi viene trasformata in formato XML, presenta la struttura già mostrata in precedenza.

Questo server remoto è collegato via cavo ad un router Wi-Fi che funge come coordinatore della rete, permettendo così scambiare (mediante la tecnologia wireless) informazione sui punti campionati, con il palmare in maniera remota. Questo router è un Sparklan modello WX-66156T era già nell'ambiente in studio, ed il suo compito era permettere ai dispositivi mobili provvisti di schede Wi-Fi, accesso alla rete di dati aziendale e accesso Internet.

Mentre i dispositivi utilizzati come AP, e dispersi sull'ambiente in studio, sono 5 Netgear modello WG-102, comprati appositamente per le sue alte prestazioni e per le diverse funzionalità disponibili, tra le quali c'è la possibilità di collegamenti diretti tipo bridge verso un nodo centrale, in maniera tale di potere gestire un specie di roaming o handover, che verrà deciso in funzione del livello di segnale più forte in una determinata zona, cioè, il palmare quando deve collegarsi al server remoto, lo farà direttamente con l'AP (o router) che abbia il livello di potenza più alto in quel

momento, con lo quale si agevola la velocità nella trasmissione dei dati e il tempo di risposta del sistema di localizzazione. Inoltre, questi AP fungono come *repeater* del segnale radio, in modo tale di mantenere un collegamento continuo con il server remoto. Questo è possibile, mediante le impostazioni delle stazioni base Wi-Fi.

Un esempio di impostazione delle funzionalità *bridge* e *repeating*, per le stazioni base Wi-Fi, si mostra nella figura 4.7.

The image shows a configuration window titled "Advanced Access Point Settings". Under the "Access Point Mode" section, there is a checked checkbox for "Enable Wireless Bridging and Repeating on Security Profile 1". Below this, three radio buttons are present: "Wireless Point to Point Bridge", "Wireless Point to Multi-Point Bridge" (which is selected), and "Repeater with Wireless Client Association". The "Wireless Point to Multi-Point Bridge" option includes a sub-option "Enable Wireless Client Association" which is unchecked, and four rows of "Remote MAC Address" fields, each with a grid of input boxes. The "Repeater with Wireless Client Association" option includes "Parent AP MAC Address" and "Child AP MAC Address" fields, each with a grid of input boxes. "Apply" and "Cancel" buttons are located at the bottom right of the window.

Figura 4.5: Configurazione della modalità Bridge negli AP.

La tipologia di rete descritta si assomiglia ad una rete a stella, dove in questo caso, il nodo centrale sarebbe il router che si collega al server remoto via cavo Ethernet, e gli estremi sono gli altri AP che coprono la copertura Wi-Fi nell'ambiente, e si collegano al router in modo Wireless mediante un bridge. In questo caso, si ha un bridge punto - multi - punto in direzione router/AP, mentre nel verso opposto il bridge è di tipo punto - punto.

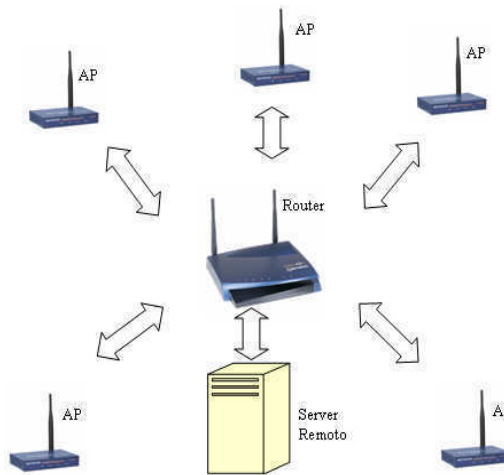


Figura 4.6: Tipologia della rete di localizzazione Wi-Fi.

Utilizzando una piccola applicazione eseguibile su Microsoft Windows O.S. (anche sviluppata in azienda), che costruisce e gestisce le mappe del sito dove si vuole implementare il sistema di localizzazione, se è in grado di associare sulla mappa in formato SVG diversi oggetti utili al posizionamento indoor, tra questi i più importanti sono le posizioni delle stazioni base.

Questa applicazione, permette tra l'altro introdurre la posizione degli AP, informazione che potrebbe essere utile per l'algoritmo di posizionamento. Comunque, l'applicazione per il momento è più rilevante per il progetto aziendale e commerciale di posizionamento, quindi non viene dettagliato il suo funzionamento se non quanto è necessario. Nella figura 4.6, si mostra l'interfaccia grafica di questa applicazione.

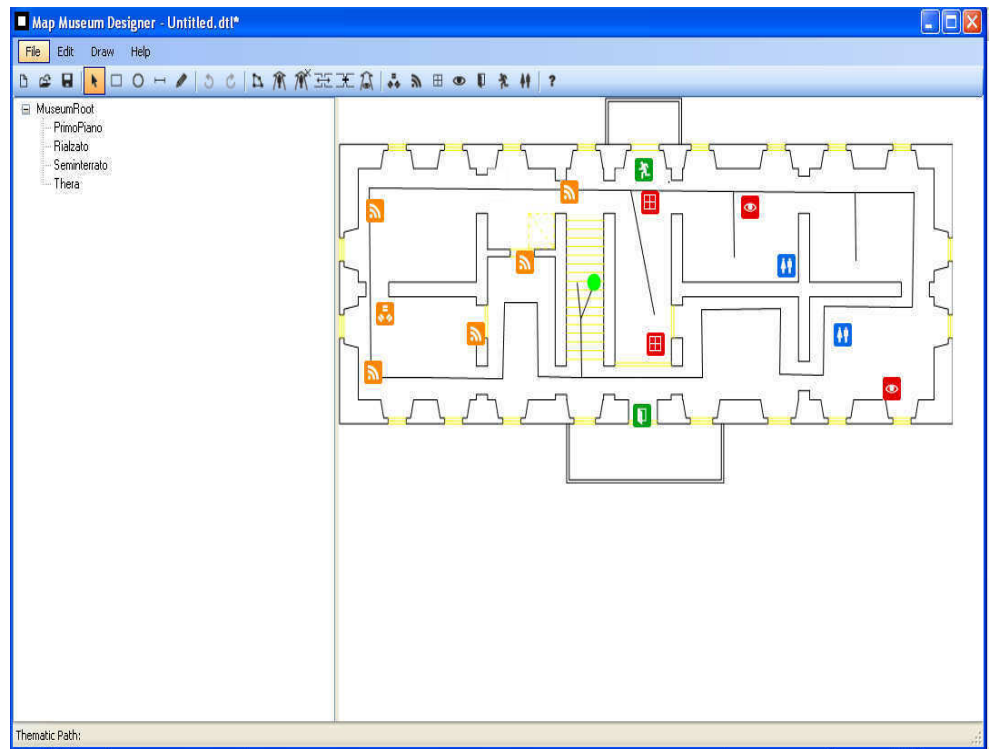


Figura 4.7: Applicazione di gestione delle mappe

4.2 Fase Sperimentale e Test

La sperimentazione consiste nella implementazione e valutazione delle prestazioni di alcuni dei metodi specificati nel capitolo 3. La sperimentazione è suddivisa in fase *offline* e fase *online*. La fase *offline* si riferisce alle impostazioni iniziali comuni e indipendenti dal metodo impiegato. Invece la fase *online* varia secondo le caratteristiche implementative di ciascun metodo utilizzato.

Nella fase *offline* vengono campionati i punti fissi e che saranno noti per poter stimare la posizione in tempo reale del palmare. Di questi punti si conosce la

posizione in termini delle coordinate X e Y, e vengono scelti in maniera arbitraria per gli amministratori del sistema attraverso l'applicazione di campionatura andando sull'interfaccia Mappa. Questi possono essere scelti in maniera regolare sulle diverse zone in cui si divide l'ambiente indoor, oppure in funzione della quantità di informazione che deve essere localizzata, oppure in funzione delle limitazioni dell'algoritmo di posizionamento impiegato.

In questo caso, è stato scelto innanzitutto il numero massimo di campioni che permette l'applicazione di campionatura (100 campioni), questo visto la grossa dipendenza nelle prestazioni degli algoritmi, che aumentano con il numero di campioni prelevati per ogni punto in fase *offline*. Successivamente, si mostrerà sì veramente questo fattore influisce sulle prestazioni del sistema di posizionamento.

Un'altra osservazione riguardo alla fase di campionatura *offline*, è che la campionatura su ogni punto, non deve essere effettuata collocando sulla stessa posizione il palmare, anzi, è consigliato misurare verso tutti i sensi, cioè, girare il palmare rispetto ad uno stesso punto ed asse, verso nord, sud, est ed ovest. Questo è necessario dovuto alla variabilità del canale radio in ambiente indoor, sottoposto a tante riflessioni che potrebbe capitare facilmente, una variazione del livello di potenza ricevuto per il palmare, tale di creare problema di accuratezza nella stima della posizione. Poi, bisogna calcolare una media per ogni punto, in funzione dei campioni misurati nei quattro sensi.

Il numero di punti campionati per ogni zona varierà secondo le dimensione delle zone dentro l'ambiente, in questo caso viene fatta una distinzione generica in due tipi di zone, quelle tipo ufficio (più piccole) dove si prenderanno al meno 2 punti distanti 2 metri (misurati in forma radiale); e quelli tipo sala (più grosse), dove si prenderanno al meno 4 punti distanti 2 metri (anche misurati in forma radiale) e allocati verso gli angoli. Poi c'è una zona di tipo corridoio stretta meno di 1 metro, che circonda altre zone con una distribuzione di punti più grossa e sovrapposti, lo

quale potrebbe creare problemi di stabilità nella stima, e per questo solo si prenderanno 1 punto ogni 4 metri (lungo il percorso dei corridoio), cercando quelli più lontani di altri punti campionati in zone adiacenti. Come si vede, questa è una scelta euristica, che poi si vedrà se avrà qualche influenza sulla stima della posizione.

Alla fine, sono stati campionati 22 punti, e per ciascuno di loro, sono stati prelevati circa 100 valori di RSSI per ogni AP presente in quel momento ed in quella posizione, perché non è detto che gli AP siano visibili tutto il tempo, data la variabilità del canale radio. Questo potrebbe essere verificato, in fase di compilazione, sul *debug* del codice relativo al posizionamento, mettendo degli *step* di pausa, giusto nel momento in cui sono caricati i punti campionati, si può osservare che non tutte le liste di RSSI hanno sempre 100 valori. Questo non affetta i calcoli in fase *offline*, visto che si può lavorare semplicemente con la media al momento di confrontare con i valore correnti di RSSI. Tra ogni campione prelevato si ha una separazione temporale di 2 secondi. Questi campioni sono stati mediati in ogni verso e sono salvati sul server nel formato descritto precedentemente per l'applicazione campionatura. Questa procedura rimane la stessa per ogni metodo di posizionamento a implementare.

Di questi 22 punti campionati, furono prescelti 10 punti per essere utilizzati in fase di test, per il calcolo delle prestazioni dei diversi metodi ed algoritmi implementati. A continuazione vengono elencati secondo la zona di appartenenza: punti 1 e 2, appartenenti alla sala 4; punti 3 e 4, appartenenti alla sala 1; punto 5, appartenente all'ufficio 1; punto 6,7 e 8, appartenenti alla sala 3; punto 9, appartenente alla sala 2 e punto 10, appartenente al corridoio

Nella figura 4.8, si mostra la mappa dell'ambiente in studio, in questo caso la sede di Roma dell'azienda Thera S.r.l.:

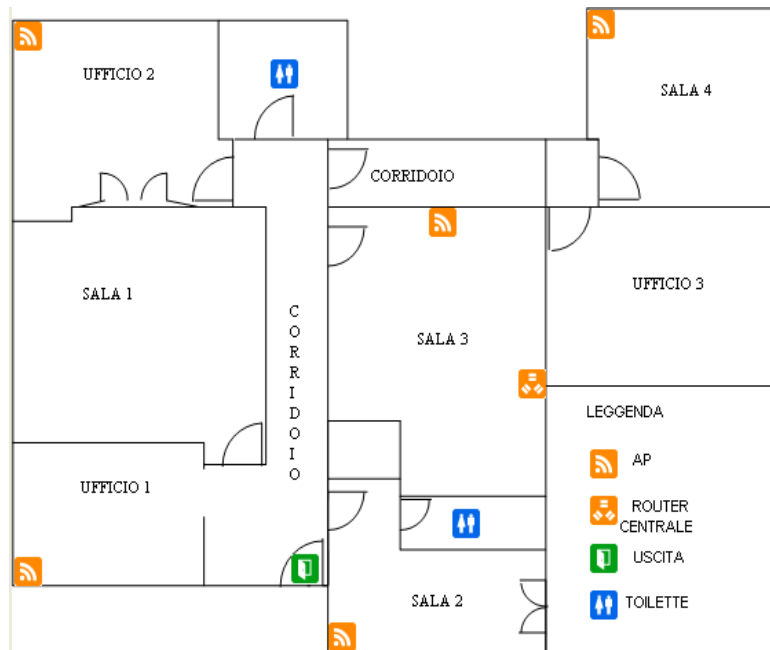


Figura 4.8: Mappa della sede a Roma di Thera S.r.l.

Lo studio si centra in queste zone, perché sono quelle più accessibili per fare le prove dentro l'azienda, e così sarà più facile confrontare le prestazioni tra i diversi metodi, anche in funzione della zona. Inoltre, nella zona di corridoio non si è scelto un punto campionato in specifico, anzi, vengono considerati tutti i punti lungo il percorso descritto per i corridoi.

La fase *online*, riguarda l'esecuzione del posizionamento mediante l'interazione tra il terminale e la rete. In sintesi, il server remoto è in attesa dei valori RSSI in funzione degli AP rilevati dal palmare. Questi valori sono mandati in continuazione, sempre che il dispositivo sia acceso, attraverso il router centrale che si collega via wireless con il terminale, dopodiché questi passa i valori via Ethernet al server remoto. Una volta ricevuti questi valori, il server è l'incaricato di stimare (via software) la posizione del terminale, per far ciò si utilizza una funzione che estrae i valori campionati del server remoto (con il formato XML riportato prima). Poi questi

valori sono gestiti secondo il metodo di posizionamento. Il valore restituito per il server, sono un paio di coordinate che rappresentano la posizione attuale del terminale, che potrebbero essere tra quelle dei punti fissi (approccio statico) presi in fase di campionatura, oppure una stima ponderata per cui ritorna un punto aleatorio (approccio dinamico), non compreso tra i punti campionati, ma che comunque dovrebbe essere quello che si approssima meglio alla posizione esatta del palmare. che saranno poi visualizzati sulla schermata Posizione dell'applicazione di campionatura.

Il componente più importante della fase online è il server remoto, di tipo *push*, perché è l'incaricato in questo caso di generare richieste dei servizi disponibili, utilizzando protocolli di comunicazioni compatibile con Internet (HTTP). In questo caso, come si lavora con la piattaforma .NET Framework, sul server remoto ci sarà l'IIS (Internet Information Services) installato, che permette lavoro con servizi web (web service), sui quali è possibile sviluppare, eseguire e pubblicare applicazioni accessibili in modo remoto. In questo modo, qualsiasi dispositivo che sia in grado di collegarsi ad Internet, potrà accedere ai servizi disponibili.

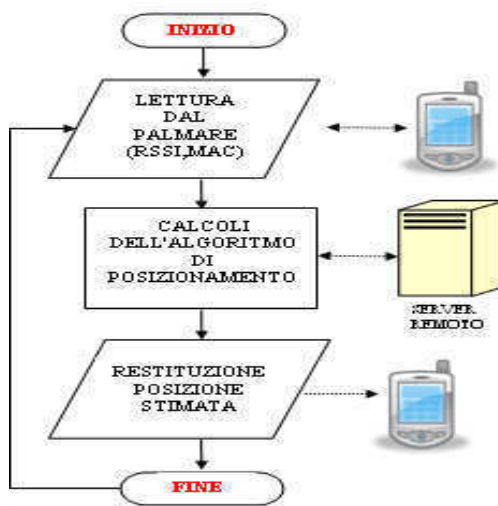


Figura 4.9: Diagramma di Funzionamento del Sistema di Localizzazione Wi-Fi.

Dallo schema mostrato precedentemente, quello che succede, è che il palmare, attraverso del programma di campionatura, si collega al server e trasmette i valori RSSI correnti associati ad ogni AP presente nella rete, individuati mediante l'indirizzo MAC, informazione che anche è trasmessa. L'algoritmo di posizionamento, prende questi due array (Power di tipo *int* e MAC di tipo *string*), e li confronta con i valori salvati nel server, corrispondenti ai punti campionati in fase *offline*. Per mostrare le prestazioni di ciascun metodo durante i test, si determinano l'accuratezza e la precisione, in funzione delle posizioni restituite dall'algoritmo finale, su diverse posizioni scelte intorno ai punti fissi selezionati in fase di campionatura (sono stati scelti 10 punti distribuiti aleatoriamente nell'ambiente), e durante un intervallo di 4 minuti, misurando verso tutti i sensi. La formula^[1] utilizzata per il calcolo della statistica dell'accuratezza è definita in funzione della deviazione standard, ed è la seguente:

$$\sigma_{acc} = \sqrt{\frac{\sum_{i=0}^n (Z_i - Z_t)^2}{n}}$$

In questa formula, Z_i rappresenta le coordinate associate alla stima ottenuta in ogni istante di tempo, mentre Z_t rappresenta le coordinate associate alla vera posizione del terminale. Il numero di misurazioni è rappresentato da n . Utilizzando la stessa formula si potrebbe calcolare anche la statistica della precisione, soltanto cambiando Z_t per la media del valore campionato. A partire dalla deviazione standard si definisce anche la deviazione standard relativa come il rapporto tra σ_x (in questo caso σ_{prec}) e la media aritmetica dei valori (in questo caso Z_i), che tipicamente viene espressa in forma percentuale per un miglior confronto. Allora, la formula per determinare la precisione diventa:

$$\sigma_{prec} (\%) = \frac{\sqrt{\frac{\sum_{i=0}^n (Z_i - \bar{Z})^2}{n}}}{\bar{Z}_i} \times 100\%$$

4.2.1 Test del Metodo NSSS

In questo caso è stimata la posizione mediante il calcolo delle differenze euclidea, tra i valori RSSI correnti e i valori RSSI salvati sul server, rispettando i vincoli dei MAC, cioè, sono confrontati i valori RSSI dei rispettivi AP.

Dopo che tutte le differenze sono calcolate mediante le formule descritte nella sezione 3.2, sono confrontate tra di loro, rimanendo quella con il valore più basso. Alla fine, si cercano le coordinate associate al punto che ha generato la differenza più bassa. Queste coordinate corrispondono alla stima della posizione corrente del palmare.

Come si sa, nella fase *offline*, per ogni punto si ha una serie di vettori che rappresentano i valori RSSI (al massimo 100 campioni) ricevuti da ogni AP, e visto che il valore corrente tipicamente è composto da un solo valore RSSI, si ha una inconsistenza tra le proporzioni dei vettori, per cui la soluzione più immediata è prendere la media dei campioni dei punti fissi, e poi fare la differenza con il valore corrente rispettivo.

Per accelerare il processo di ricerca e diminuire l'onere computazionale, si può implementare un meccanismo di discriminazione, in funzione del raggio di copertura degli AP. Quindi, in ogni zona dell'ambiente indoor, ci saranno uno o più AP con un livello di potenza noto e superiore rispetto agli altri, è che caratterizzano di maniera univoca la zona (localizzazione basata sull'area). Di conseguenza, durante il

processo di ricerca, sono scartati quelli punti, che non rappresentano lo stesso comportamento (in termini di potenza) dei valori RSSI correnti, diminuendo il numero di calcoli di differenza, è aumentando l'accuratezza del sistema, perché ora va andare a controllare quelli punti sulla zona di appartenenza più idonea, e sono scartati punti con differenza bassa, che potrebbero puntare verso una stima totalmente erronea del terminale. Un modo di fare questa discriminazione, e aggiungendo un pezzo di codice, che verrà messo prima di calcolare la differenza, composto da un ciclo che confronti i valori RSSI correnti con i valori RSSI dei punti campionati, senza andare a fare la differenza. Questo confronto può essere limitato per un valore di tolleranza, oppure di maniera discreta, cioè andando a vedere gli AP con segnale più forte in quel momento e cercando quali corrispondono ad essa condizione tra i punti salvati sul server. I risultati ottenuti sono riassunti nella figura 4.10, che mostra il comportamento dell'algorithm in tempo reale.

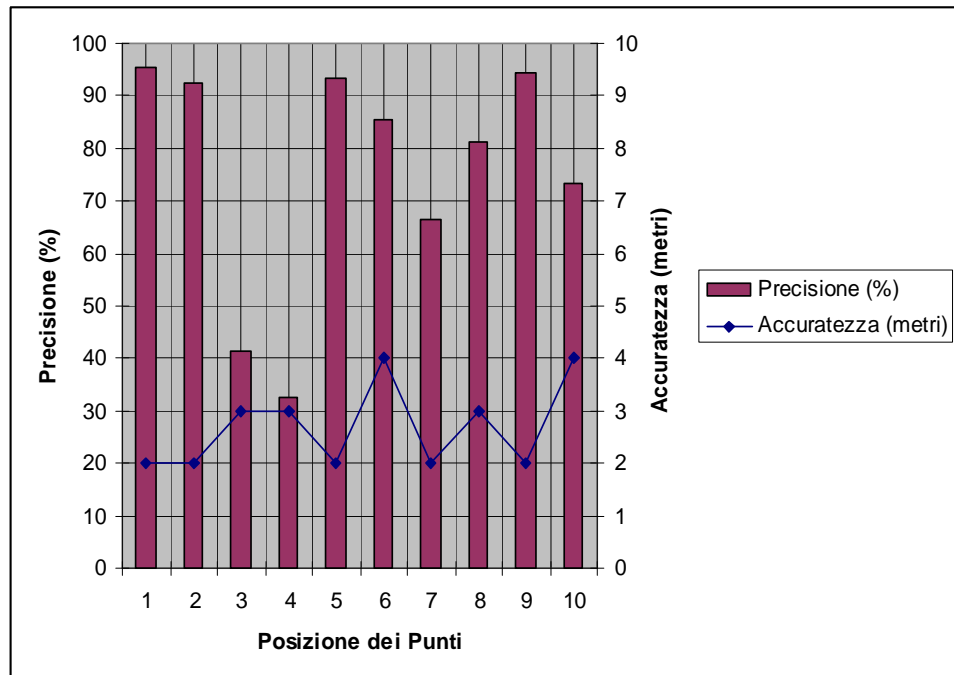


Figura 4.10: Prestazioni Algoritmo NNSS.

Le osservazioni sono le seguenti:

- Nella zona corridoio, visto che è stretta per meno di 1 metro, la stima della posizione segnale era sempre sbagliata, non solo rispetto alla vera posizione del terminale, anche rispetto alla zona. Questo dovuto alla vicinanza delle altre zone. Quindi sicuramente non essendoci muri sufficientemente grossi per limitare il segnale tra una zona ed un'altra, era possibile durante l'esecuzione dell'algoritmo che punti molto vicini potevano entrare nel *range* di posizioni stimate, sebbene appartenenti ad una zona adiacente.
- Nelle zone non coperte direttamente per un AP (punti 3,4), oppure dove c'era sovrapposizione delle coperture radio di tanti AP, la stima era sbagliata, data la sovrapposizione tra i pattern di radiazioni delle antenne e la mancanza di alcun meccanismo per discriminare i segnali. Per questo è consigliabile l'utilizzo di antenne del tipo direzionale, evitando così la sovrapposizione create per la radiazione delle antenne omnidirezionali. Comunque, per i costi elevati di queste genere di antenne non vengono utilizzate in questo progetto.
- Molti di questi problemi di precisione, sono dovuti anche alle fluttuazioni veloci del mobile durante la procedura di stima, per lo spostamento del terminale, quindi il canale varia tanto rapidamente, che non si può avere una stima affidabile. Questo capitava anche quando l'utente in sospeso, si girava su se stesso, verso un qualche senso.
- Un ultimo fattore che viene a influenzare le prestazioni di questo algoritmo sono le attenuazioni del canale radio, in presenza di utenti in una determinata zona, lo quale non corrisponde alle condizioni iniziali di campionatura in fase *offline*, perché in quella fase, l'ambiente era totalmente

vuoto. Utenti in sospeso o spostandosi arbitrariamente nell'ambiente, generano più fenomeni di riflessione.

- Le prestazioni di questo algoritmo sono un'accuratezza media di 2,7 metri ed una precisione del 75%, considerando tutti i fattori precedentemente spiegati.

Ora si vedrà l'effetto di modifica dell'algoritmo, mediante un filtraggio delle zone che lo renderebbe più accurato, preciso e veloce nella ricerca della posizione stimata. La figura 4.11, mostra le prestazioni dell'algoritmo NNSS, versione modificata.

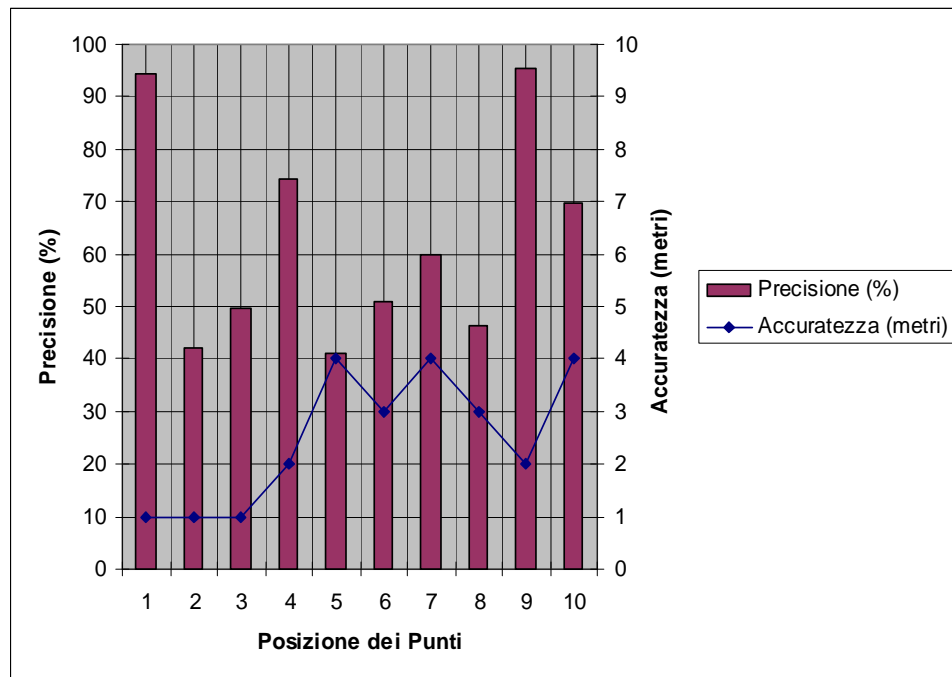


Figura 4.11: Prestazioni Algoritmo NNSS Modificato

Le osservazioni sono le seguenti:

- Il problema della copertura nelle zone dove non è presente almeno un AP (punti 3, 4), peggiora durante la discriminazione per zone, perché l'algoritmo restringe il numero di soluzioni possibili. Invece per il resto delle zone si ha un lieve miglioramento: 2,5 metri di accuratezza media ed una precisione del 63%.
- I problemi esposti nel caso precedente non vengono risolti, quindi sono ancora presenti.

4.2.2 Test dei Metodi Probabilistici

In questo caso invece, quello che deve fare l'algoritmo è riuscire a determinare la probabilità del segnale corrente di appartenere ad alcune delle postazioni fissate in fase *offline*, quindi il valore corrente di RSSI può essere confrontato sia utilizzando un istogramma, che rappresenta la distribuzione dei valori campionanti, oppure per mezzo di un'approssimazione ad una funzione di distribuzione Gaussiana, calcolando la probabilità condizionate in questo ultimo caso, con le formule descritte nella sezione 3.3. Quindi immerso nel codice, ci sarà una funzione che determina la probabilità secondo la frequenza dell'istogramma associato ad ogni punto campionato, oppure una funzione che sapendo a priori la media e la varianza del vettore RSSI di un AP specifico, calcola il valore della probabilità con l'approccio Gaussiano. Nel caso dell'istogramma, la probabilità di un valore RSSI, al di fuori dell'intervallo definito, è fissata ad un valore di probabilità 10 volte minore di quello più basso presente.

Allora, come è stato discusso nella sezione 3.3, i valori di probabilità ottenuti per qualsiasi dei due approcci non basta per determinare la posizione. Quindi considerando che gli AP sono indipendenti tra di loro, la probabilità associata ad una

determinata posizione, è il risultato della moltiplicazione tra i diversi valori ottenuti per ogni AP. Alla fine, simile a come si faceva nel metodo NNSS, il risultato ritornato sarà quello con probabilità più alta rispetto agli altri.

E' possibile anche in questa metodologia, avere un problema simile al caso del NNSS, e cioè si potrebbe giungere ad una probabilità bassa rispetto alle altre, puntando però verso una zona diversa a quella vera. Allora, viene implementata nuovamente la discriminazione degli AP secondo la zona, così viene anche diminuito il campo di ricerca e aumenta la precisione della stima. La funzione dentro il codice che fa questo filtraggio è la stessa implementata in NNSS. Ora si mostrano rispettivamente, i risultati dei test per il caso di probabilità determinata mediante l'istogramma sia la versione originale, sia quella modificata dove si fa la discriminazione in funzione del comportamento degli AP in ogni zona (figure 4.12 e 4.13).

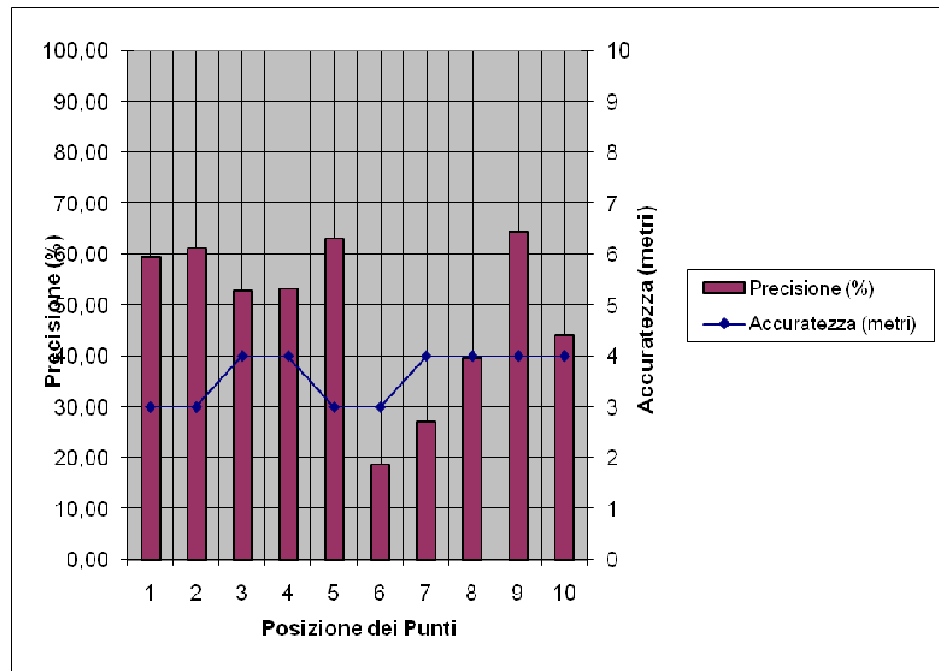


Figura 4.12: Prestazioni Metodo Probabilistico utilizzando Istogramma.

Innanzitutto, il comportamento dell' algoritmo basato sul metodo probabilistico utilizzando l' approccio dell' istogramma per il calcolo delle probabilità singole, ha basse prestazioni rispetto all' algoritmo NNSS, con un' accuratezza media di 3,6 metri ed una precisione media del 48%.

A continuazione vengono rappresentate le prestazioni per il caso modificato, dove si implementa la discriminazione per zone. Insomma, non si raggiungono grossi miglioramenti rispetto alla versione originale.

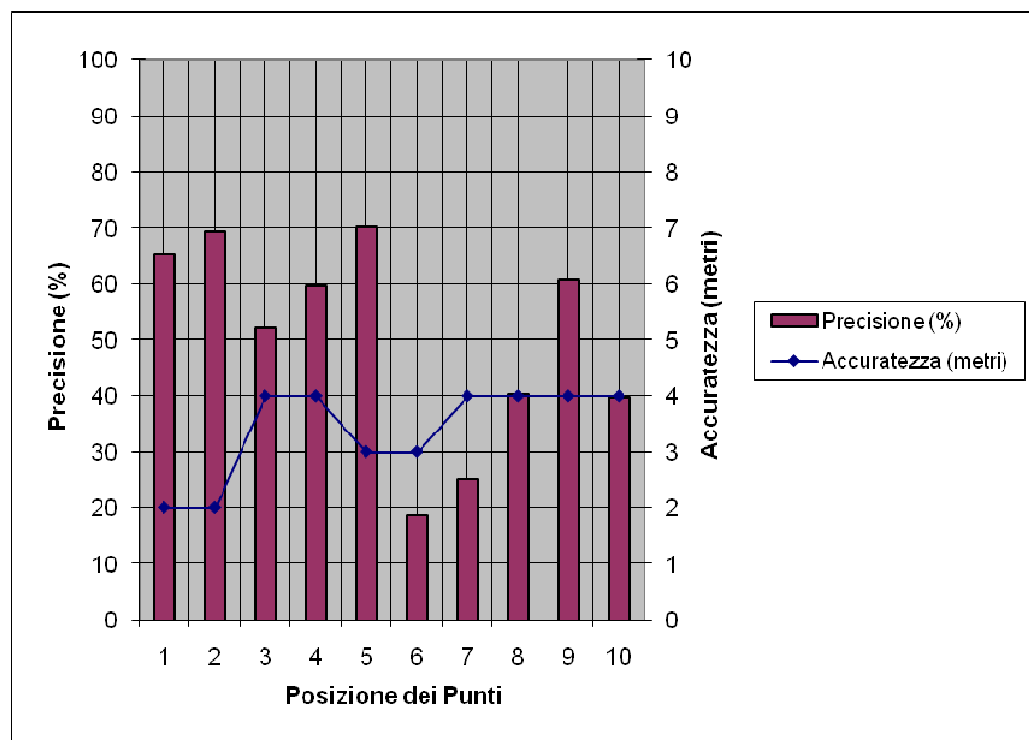


Figura 4.13: Prestazioni Metodo Probabilistico utilizzando Istogramma (versione modificata).

Le osservazioni generali sono le seguenti:

- La stima è molto imprecisa per le zone che non sono coperte direttamente da un AP, per cui il codice ritorna valori che oltrepassano la zona di appartenenza, quindi non è consigliato il suo utilizzo per rilevamento puntuale.
- La variazione del canale radio, che ancora non è stata risolta continua ad influire sui risultati.

La figura 4.14 riporta le prestazioni dell'approccio probabilistico, ma ora per il calcolo delle probabilità è utilizzata la funzione di distribuzione Gaussiana, ed inoltre si valuta direttamente il caso modificato, considerando la discriminazione delle zone.

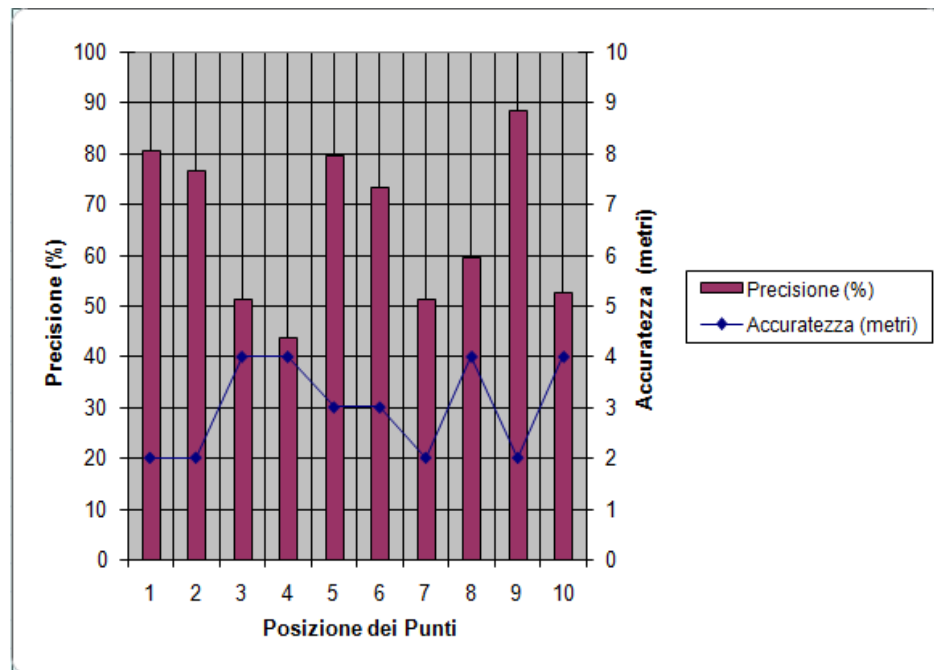


Figura 4.14: Prestazioni Metodo Probabilistico utilizzando una fdp Gaussiana.

In linee generali, utilizzando la distribuzione Gaussiana, si guadagna un po' di stabilità nella stima, quindi questo approccio è più consigliabile di quello utilizzando la propria distribuzione del segnale rilevata in fase *offline* mediante gli istogrammi. L'accuratezza media è di circa 3 metri e la precisione media del 65%.

Le osservazioni, sono le seguenti:

- Rispetto all'algoritmo NNSS, numericamente le prestazioni sono leggermente minori, quindi è meno robusto alle variazioni del segnale e lo spostamento del terminale.
- Il problema della variazione del canale radio non è stato ancora risolto, quindi questo influisce sulle prestazioni dell'algoritmo.

A questo punto, è possibile verificare il comportamento della distribuzione dei valori RSSI in alcuni dei punti campionati sull'ambiente in studio, e di questa maniera chiarire i dubbi rispetto alla migliore approssimazione della funzione distribuzione di probabilità, determinando così quella che meglio si adatta al comportamento del segnale radio Wi-Fi. Per ottenere queste distribuzioni, sono stati presi i valori RSSI, di due punti diversi, di due zone diverse dell'ambiente in studio, tra quelli salvati sul server remoto. Una volta, che si ha accesso a questa informazione, con i vettori ottenuti, si costruisce la statistica in forma di istogramma. I risultati, visualizzate nelle figure 4.15 e 4.16, permettono di affermare che la distribuzione data per ogni AP in ciascun punto è variabile e dipenderà da quanto lontano si trova un AP del punto in osservazione. Comunque, si osserva che più vicino si trova l'AP del punto in questione, più la somiglianza ad una distribuzione Gaussiana, più lontano dagli AP, più aleatoria la distribuzione. Inoltre, le distribuzioni dei valori RSSI sugli istogrammi, quasi mai si assomigliano per punti diversi in zone diversi, comunque, potrebbero esistere minime coincidenze statistiche nei casi in cui i punti si trovino nella stessa zona, oppure in zone non coperte

direttamente dagli AP. Queste piccole deviazioni, sono derivate dei fenomeni di propagazione del segnale, provocando errori sull'accuratezza della stima della posizione. Insomma, grazie alle simmetrie presenti negli istogrammi, e la generalità dei casi, risulta più conveniente utilizzare un'approssimazione Gaussiana per determinare le distribuzione di probabilità del segnale radio Wi-Fi nell'ambiente in studio, senza dimenticare le importanti osservazioni fatte precedentemente.

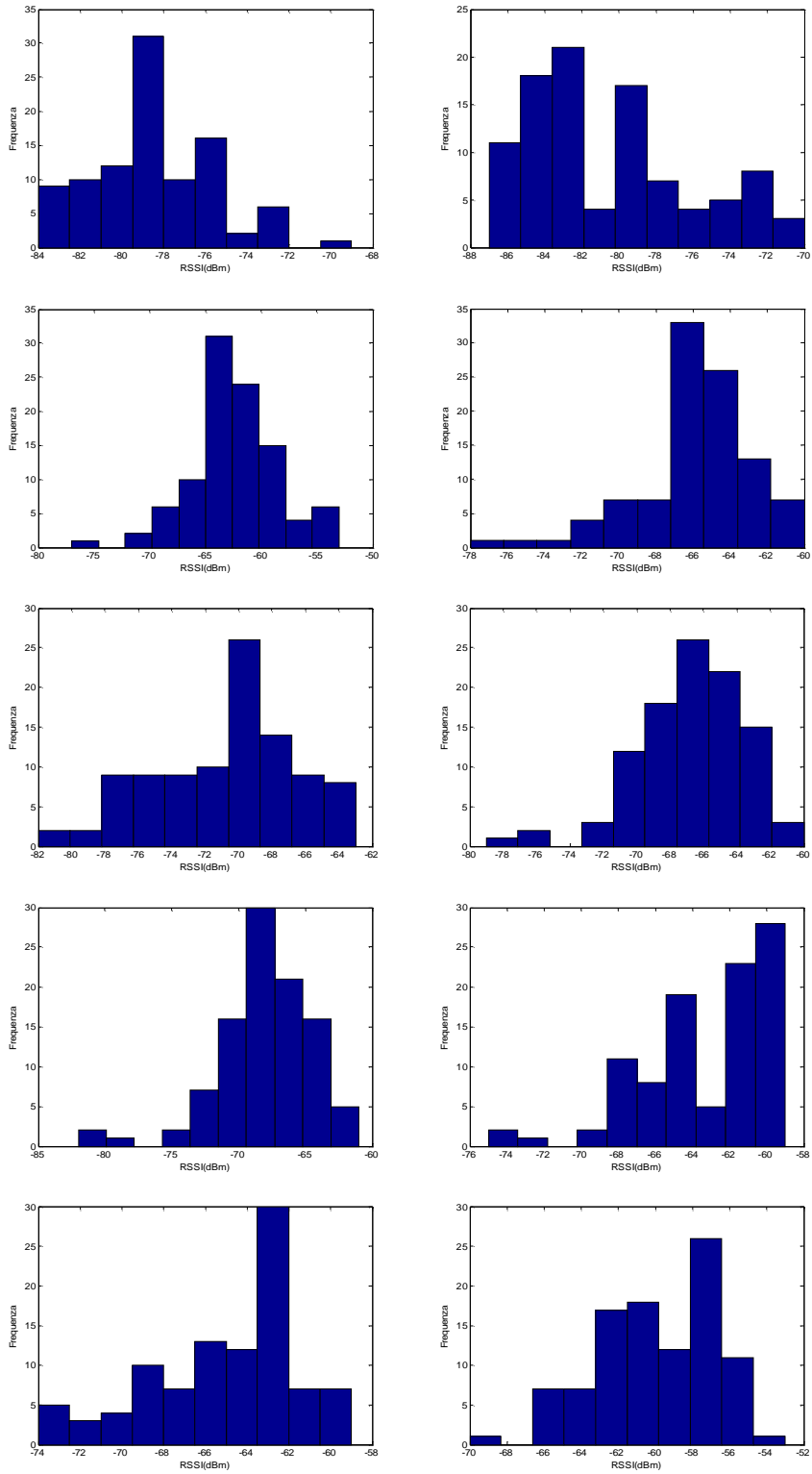


Figura 4.15: Istogrammi Sala 1, punti 6 e 8

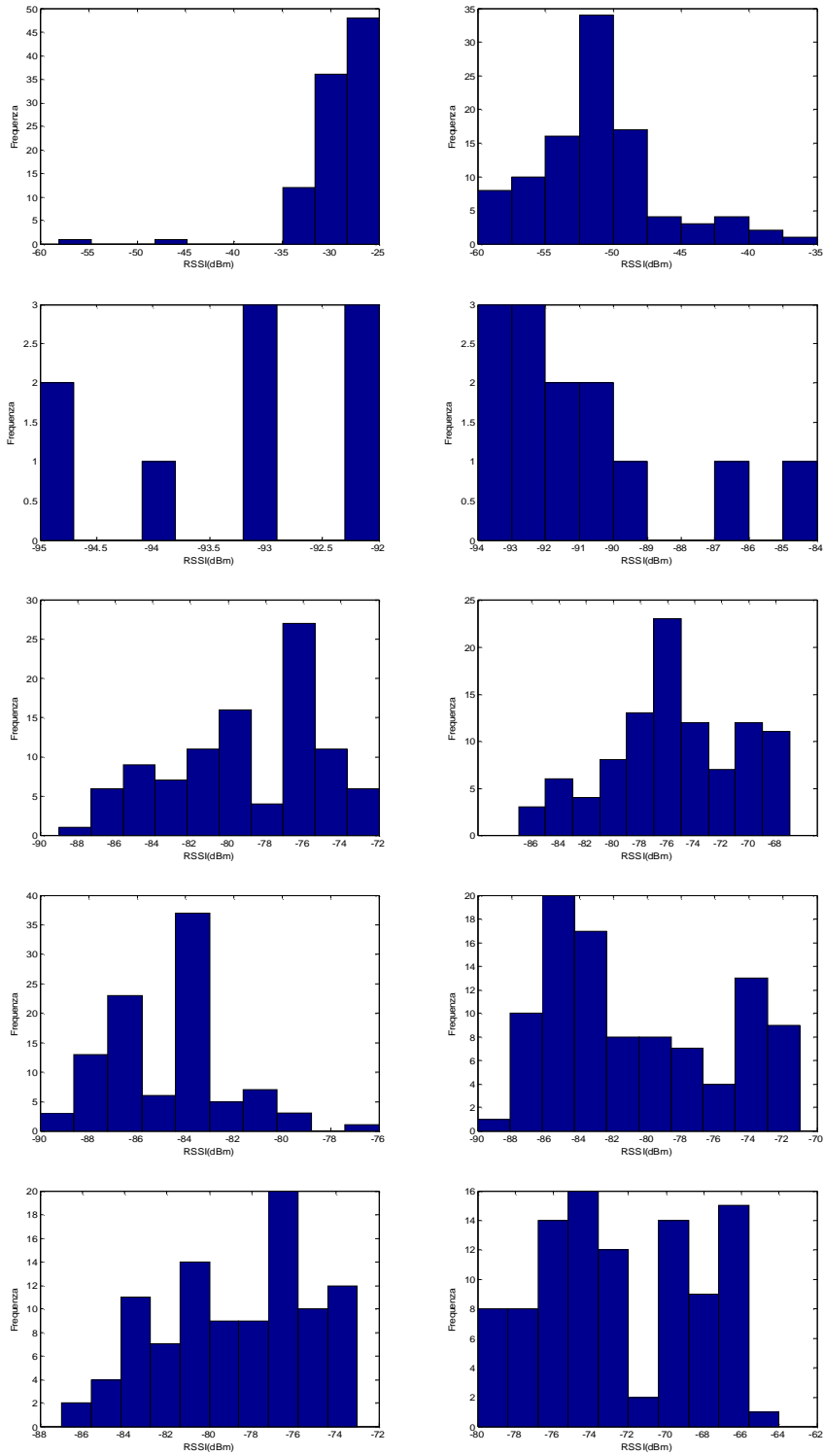


Figura 4.16 Istogrammi Sala 4, punti 1 e 2.

4.2.3 Test del Metodo del Centroide

In questa parte, sarà implementato in codice C#, il metodo del centroide, metodo che stima di maniera probabilistica la posizione del terminale, che a differenza dei metodi precedenti da come risposta un punto che è ponderato a partire di quelli fissi, quindi la sua esecuzione è più dinamica, adatto a risolvere problemi di spostamento del terminale, nel caso in cui serva un tracciamento a partire del segnale Wi-Fi ricevuto dal terminale.

Per fare questa ponderazione, si parte delle formule descritte nella sezione 3.4, supponendo per il calcolo delle probabilità una funzione di densità Gaussiana. Poi, si andrà a giocare con i numeri di punti fissi che entrano nella ponderazione, vista la dipendenza di questo valore con l'accuratezza e la precisione della stima.

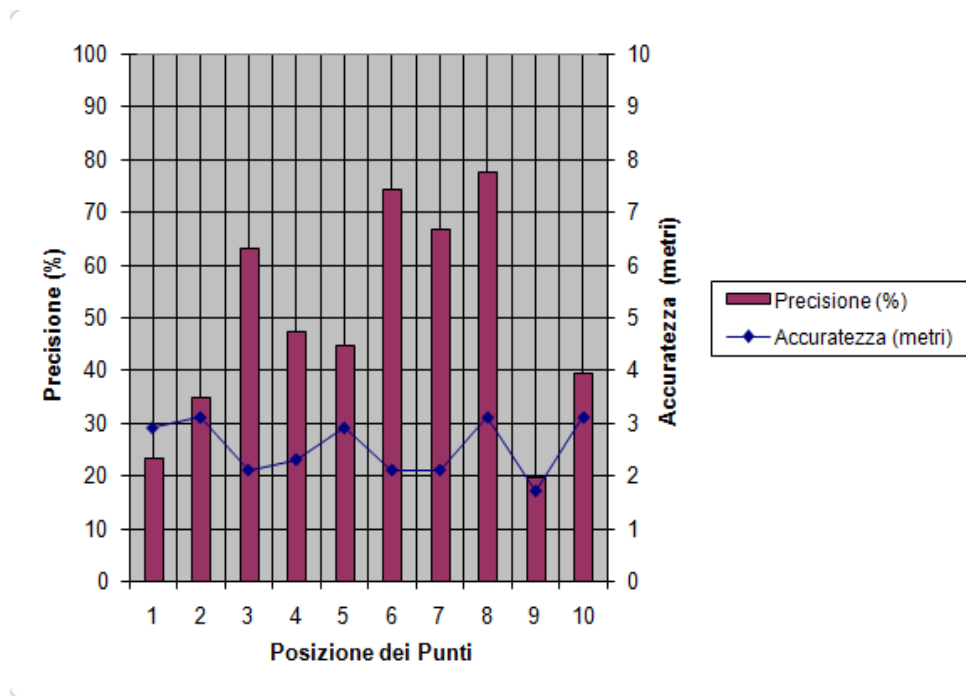


Figura 4.17: Prestazioni Algoritmo Centroide.

In linee generali, dalla figura 4.17, si osserva che questo algoritmo, presenta una precisione del 50%, inferiore a quella del NNSS che ha dato migliori risultati (75%). Dall'altro canto offre per la maggioranza dei casi un'accuratezza media di 2,3 metri, migliore che nel caso NNSS (di 2,7 metri rispettivamente). Comunque, non si possono paragonare direttamente gli algoritmi perché la procedura di stima è diversa in ogni caso, nel centroide è più dinamica, per cui è più adatto al tracciamento come si diceva prima, invece nel caso NNSS e nei metodi probabilistici, il risultato della stima è uno dei punti già fissato in precedenza. Le osservazioni riscontrate sul comportamento dell'algoritmo implementato sono le seguenti:

- Gli stessi problemi della variabilità del canale radio e lo spostamento del terminale, affettano la stima della posizione. Questa influenza crea dei problemi grossi sull'individuazione di una zona specifica dell'ambiente in studio, data la dipendenza del calcolo probabilistico ai valori correnti della potenza, quindi con un'accuratezza notevole (circa 0.5 metri) può puntare verso un punto fuori della zona reale di appartenenza, ma che comunque è adiacente. Il problema si presenta tipicamente in regioni limitrofi (vicini ai muri, porte). Questo non accade con la stessa frequenza negli algoritmi statici, perché la risposta è funzione diretta di un punto fisso.

- Gli errori di accuratezza possono capitare facilmente in tutte le implementazioni fatte finora, dovuto alla variabilità del canale radio e l'instabilità del segnale ricevuto per il palmare. I segnali correnti ricevuti dal palmare, sono tipicamente caratterizzati da un solo valore di RSSI per ogni AP visibile, bisogna aumentare la quantità di valori correnti RSSI per avere una certezza nel comportamento del segnale, il quale varia continuamente nel tempo. Una soluzione sarebbe aumentare il numero di campioni analizzati del segnale corrente. Questo potrebbe creare ritardi nella stima, per i tempi di attesa nel collegamento al server remoto e la esecuzione del codice. Comunque, si possono prendere tanti campioni del segnale corrente,

dipendendo dalla criticità del sistema in dare una risposta all'utente, cioè, dipenderà dal servizio di localizzazione che sarà implementato.

– Un'altra limitazione legata all'instabilità del segnale la pone l'intervallo di campionatura che ha un valore minimo di 1 secondo, e poi il tempo in collegarsi al server remoto (circa 1 secondo), tempi durante i quali sono scartati possibili cambiamenti nei livelli di potenza che potrebbero servire a determinare il comportamento del segnale con maggiore accuratezza. Questo non è facilmente gestibile perché dipende dal tempo di risposta delle schede Wi-Fi disponibili (dei terminali e degli AP) e della tempistica nei protocolli di comunicazione tra i dispositivi.

Per dimostrare, l'influenza dell'instabilità del segnale corrente sulla stima, sarà fatta una modifica sul codice dell'algoritmo del centroide, aumentando il numero di valori di RSSI per ogni AP per il segnale corrente. Si ha scelto un valore arbitrario di 3 RSSI per ogni AP, anziché 1 come era finora. Questo affetterà sicuramente il tempo di attesa nella stima di almeno 3 volte il tempo necessario prima della modifica. La figura 4.18, riporta le prestazioni dell'algoritmo di centroide modificato.

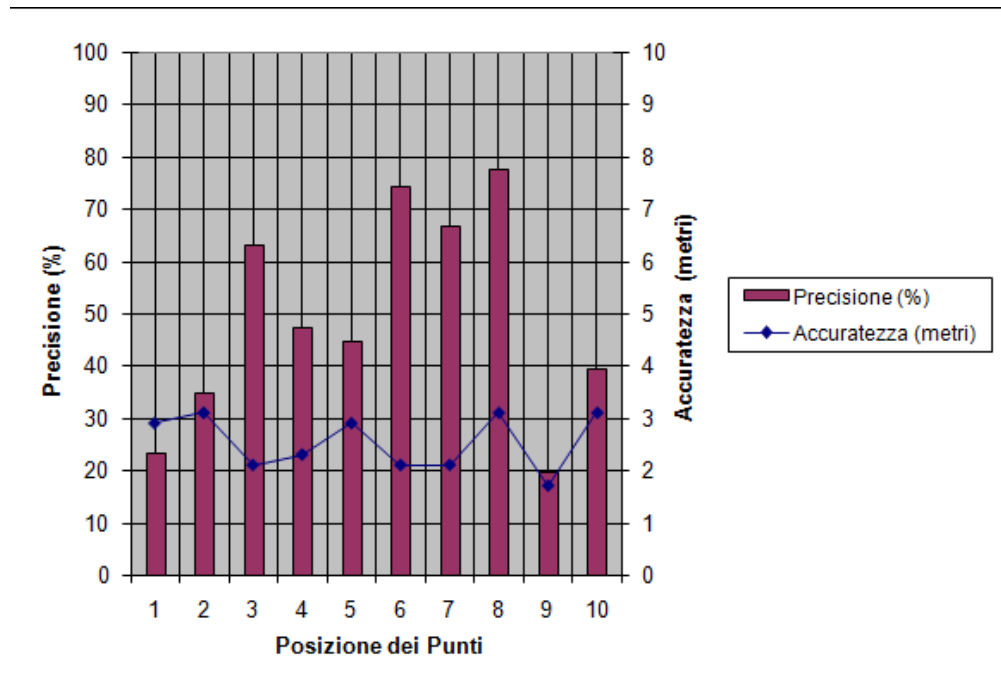


Figura 4.18: Prestazioni Algoritmo Centroide Modificato.

Le prestazioni in generale sono aumentate: 2,1 metri di accuratezza e 61% di precisione. Grazie alle caratteristiche del linguaggio di programmazione, è stato possibile evitare il ritardo di attesa nella stima, mediante una pila che accumula i valori correnti di RSSI ogni volta che è fatta la richiesta di posizionamento, in modo che solo all'inizio, bisogna aspettare un tempo di circa 6 secondi, poi l'algoritmo fa un riciclaggio, cioè, riutilizza alcuni dei valori precedenti per fare la stima, così non è necessario aspettare per altri 3 nuovi valori del segnale corrente ogni volta che si procede con la stima. Le osservazioni riscontrate sono:

- Nelle zone più strette dove tipicamente non ci sono tanti punti fissi, si ha una precisione tollerabile dal punto di vista numerico, ma intollerabile dal punto di vista di grafico, perché punta verso le zone vicine, quindi bisogna in quelle zone aumentare il numero di punti fissi, con una disposizione legata alle dimensioni.

- Se il terminale si sposta velocemente e verso diverse zone, le fluttuazioni nei valori di potenza cambiano così fortemente, che la stima punterà verso posizioni giuste presentando però un ritardo rispetto alla posizione corrente del terminale.
- Dentro dell'algoritmo c'è un parametro che controlla il numero di punti usati durante la ponderazione della stima. In questo caso è difficile lavorare con un valore generico, perché si è verificato che questo parametro influisce sulla stima di una determinata zona e del numero di punti che si trovano intorno. In una zona piccola, i punti fissi ponderati sono pochi, perciò sicuramente i punti delle zone adiacenti entrano nella ponderazione, diminuendo così l'accuratezza. Per questo si deve cercare un modo di aggiustare il parametro in funzione della zona: più grande per zone ampie e più piccolo per zone piccole. Un'altra soluzione, è fare un campionamento *offline* proporzionale tra zone di dimensioni diverse, e impostare il parametro al valore minimo della zona più piccola più 1. Nell'implementazione è stato fissato con un valore di 4.

4.2.4 Test Metodo Ibrido

Questo metodo, è una proposta che utilizza la tecnica di *lateration*, nominata nella sezione 2.3, nella modalità triangolazione, cioè si parte dalla conoscenza di tre punti vicini al punto corrente, si calcola la distanza di questi punti fissi verso il punto corrente mediante un modello rappresenti le perdite per propagazione del segnale, in questo caso derivato dell'equazione di Seidel e Rappaport per ambiente indoor, dopodiché con la conoscenza delle coordinate dei punti fissi e le loro distanze, basta risolvere il sistema di equazioni (derivato della triangolazione) che permette determinare la posizione approssimata del punto corrente.

Innanzitutto per potere utilizzare la cosiddetta equazione, bisogna conoscere le caratteristiche degli AP e di propagazione dell'ambiente radio, o al meno stimarle di maniera empirica, cioè andando a misurare l'attenuazione ottenuta nell'ambiente reale, per poi stabilire una statistica dei parametri che servono per la suddetta equazione di propagazione. Per far ciò, si richiama la seguente equazione:

$$P(d)[dBm] = P(d_o)[dB] - 10n \log\left(\frac{d}{d_o}\right) - \begin{cases} nW * WAF & \text{se } nW \leq C \\ C * WAF & \text{se } nW > C \end{cases}$$

Dove $P(d)$ rappresenta la potenza ricevuta sul punto in studio, d è la distanza trasmettere-ricevitore, $P(d_o)$ è la potenza ricevuta in un punto di riferimento d_o . Invece WAF è il Wall Attenuation Factor rappresenta l'attenuazione dovuto ai muri, nW numero di ostacoli (muri) tra il trasmettitore ed il ricevitore, e C è il massimo numero di ostacoli (muri) dove il WAF comincia ad affettare l'attenuazione. Tipicamente n e WAF dipendono dell'ambiente.

Assumendo $nW=1$, perché saranno presi in considerazione i punti che si trovano nella zona a dove appartiene (o almeno adiacenti) il punto corrente, quindi ci sarà al massimo un muro tra l'AP (o gli AP) e il terminale, quindi in questo caso solo ci resta determinare il valore di WAF di maniera empirica. Della stessa maniera, dovrà essere determinato il valore di n , data la dipendenza con l'ambiente.

Rispetto ai valori di riferimento, si deve considerare che in questo modello è consigliato utilizzare una distanza di riferimento d_o di circa 1 metro, parametro tipicamente noto nelle caratteristiche tecniche degli AP. In questo caso invece, data la conoscenza dell'ubicazione degli AP sull'ambiente in studio e dei punti fissi, con la rispettiva statistica di potenza, è possibile determinare di maniera diretta il valore di

d_0 , dato del calcolo della distanza tra il punto di ubicazione degli AP e i punti fissi più vicini al punto in stima; ed anche è possibile ricavare $P(d_0)$, derivato della media delle statistiche degli RSSI ottenuti in fase di campionatura.

Con tutto questo, è possibile determinare la distanza alla quale si trova il punto in studio degli altri a partire dei valori correnti RSSI, che in questo caso prendono il posto di $P(d)$.

Per determinare, i parametri WAF e n , si è eseguita la seguente procedura, per ogni zona dell'ambiente in studio:

- Posizionare un AP, in un posto noto, puntato in linea di vista diretta verso il terminale, senza ostacoli, a diverse distanze $d_{xi}=\{0,0.5,1,1.5,2,2.5,3\}$ metri.
- Per ogni posizione d_x , si ha misurato la potenza ricevuta P_{rx} per il terminale dall'AP.
- Utilizzando l'equazione che modella le perdita per propagazione, con i valori di d_{xi} e i rispettivi P_{rx} , si possono determinare i parametri WAF e n , mediante un'approssimazione.

Questa stessa procedura è stata eseguita nel Sistema di Localizzazione Wi-Fi, denominato RADAR^[12], sviluppato per Microsoft Research, basato sulla stessa formula di propagazione. Inoltre, per determinare i parametri WAF e n , sono utilizzati i valori ottenuti delle misurazioni di potenza in funzione della distanza, sostituiti nella equazione per ogni caso, dopodiché sono stimati i valori che meglio soddisfano la cosiddetta equazione, pesati in media, applicando sostituzioni ricorsivi, che finiscono al ottenere i parametri che meglio rappresentano il comportamento delle attenuazioni del segnale in funzione della distanza.

Una volta sostituiti i valori di potenza e distanza, nella equazione, per tutti i casi presi in considerazione (diverse zone dell'ambiente in studio, diverse quantità di campioni, ecc), la media ottenuta per i valori di WAF e n che meglio si adattano alla maggioranza dei casi è: WAF = 16,05 dB e $n=2,7$.

Ora, l'equazione di propagazione che serve a determinare questi valori di distanza d , diventa:

$$d[m] = do[m] \times 10^{\left(\frac{P_{RSS\text{Imean}}(do)[dB] - P_{RSS\text{corrente}}(d)[dBm] - 16\text{ dB}}{27}\right)}$$

Allora, si è in grado di determinare le distanze che servono alle equazione che determinano la triangolazione. Le equazione utilizzate in questo caso per eseguire la triangolazione sono quelle riportate nella sezione 2.3.2.

L'algorithmo implementato implementa due procedure di stima. Nel caso in cui non ci siano i tre punti necessari per la triangolazione è eseguito l'algorithmo NNSS per la stima della posizione.

Nel caso della triangolazione, dentro del database si cercano i punti che più si avvicinano ai valori correnti di RSSI individuando gli AP con segnale più forte in quella zona. Questo serve a determinare $P_{RSSI\text{mean}}(do)$ che dipende della media dei valori RSSI dell'AP con il segnale ricevuto più forte di tutti i RSSI ricevuti per il terminale. Poi, si determina $d[m]$ con l'equazione precedente. Questo procedura si ripete per i punti che si trovano più vicino al terminale.

Siccome, questo meccanismo di ricerca dei punti più vicini, è molto generica, bisogna creare un qualche metodo di filtraggio orientato ad aumentare la convergenza e accuratezza del metodo.

Una procedura già utilizzata è l'individuazione della zona di appartenenza in cui si trova il terminale, così solo sono presi in considerazione i punti di quella zona per effettuare la triangolazione.

Per risparmiare codice, ed assicurare una possibile risposta nella stima della posizione, in fase di ricerca e confronto con i punti fissi, sono effettuate due procedure sistematicamente, una orientata ad individuare gli AP con segnali più forte dentro dei valori di RSSI dei diversi punti fissi, e quella tipica per il calcolo delle differenze tramite NNSS. Dopodiché, con il valore di NNSS in riserva, si procede con il calcolo della triangolazione e se questo calcolo non è eseguibile (per esempio mancanza di un punto), la stima definitiva corrisponderà alla soluzione NNSS. Potrebbe approfittarsi questa soluzione parziale, per discriminare i valori della triangolazione, controllando che tutti siano della stessa zona di appartenenza o almeno appartenenti a zone adiacenti.

Le prestazioni di questo algoritmo ibrido, sono riportate nella figura 4.19. Le osservazioni sono le seguenti:

- Presenta un'accuratezza di 2,6 metri e una precisione nel 47% dei casi. La bassa precisione è data perché in certi ambienti domina la aleatorietà dei valori di stima anziché la restituzione di un valore fisso, e questa aleatorietà è strettamente legata alle imprecisioni del modello di propagazione proposto, e perché in questo caso il terminale non sempre mantiene una linea di vista diretta con gli AP, come si ipotizza nel caso delle misurazioni dei valori di riferimento.
- Invece, per quanto riguarda alle zone di piccole dimensioni (ad esempio i corridoi), grazie allo scarto tra le due procedure, triangolazione e NNSS, si ottiene un'alta precisione nella stima, e l'utilizzazione di

meccanismi di discriminazione potrebbe essere sfruttata ulteriormente per aumentare le prestazioni del sistema a implementare.

- Si continua a lavorare con una pila di valori istantanei (massimo 3), che ricicla i valori di RSSI prelevati in precedenza, lo quale permette avere a disposizione un comportamento del segnale più ampio. Questo a sua volta, permette una convergenza più veloce della stima.

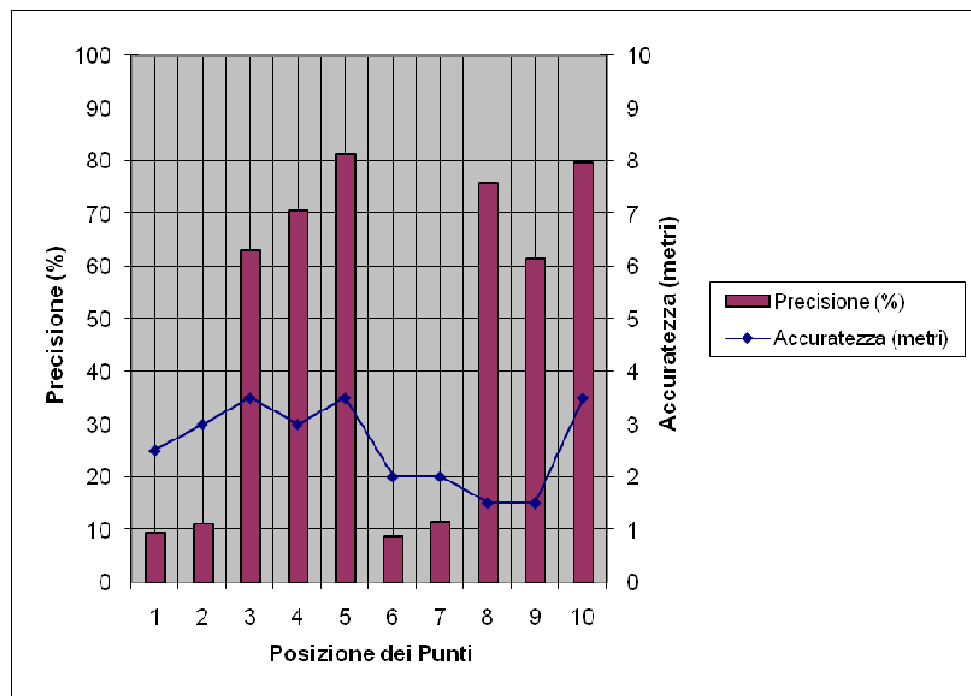


Figura 4.19: Prestazioni algoritmo ibrido modificato.

Per dimostrare l'effetto, provocato dalla propagazione del segnale, nella figura 4.20, si può apprezzare come varia la potenza ricevuta da un singolo AP, al crescere della distanza.

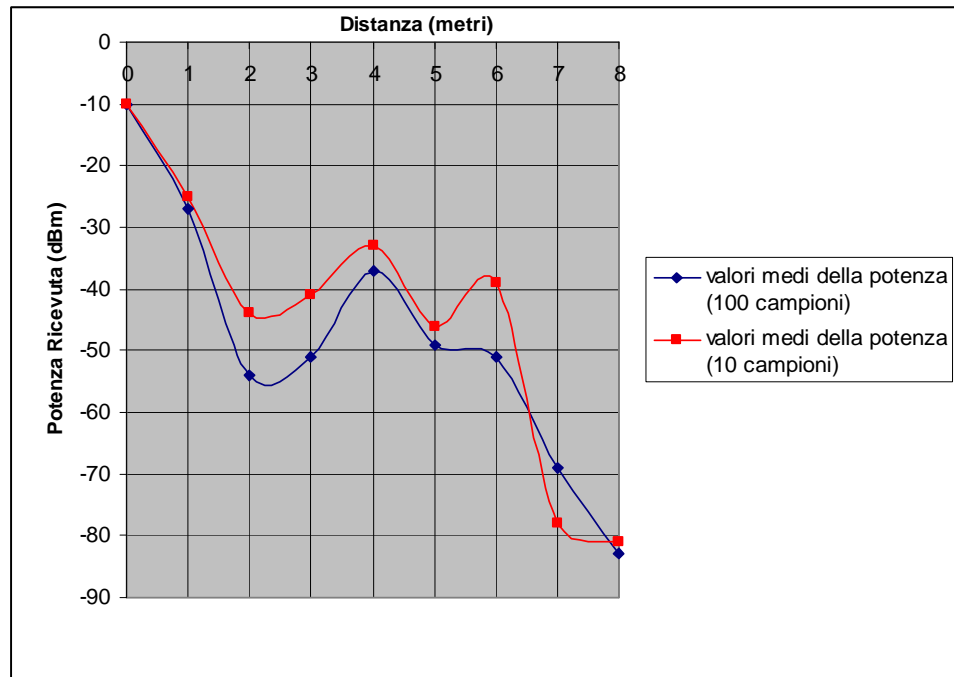


Figura 4.20 Potenza ricevuta per il terminale da un singolo AP al variare la distanza.

Per ottenere questa grafica, sono utilizzati i valori rilevati di RSSI da un singolo AP, da parte del terminale, in linea di vista, e a diverse distanze, utilizzando lo stesso programma di campionatura e prelevando i valori salvati sul database per determinare la media della potenza ricevuto per un AP qualsiasi in funzione dei RSSI rispettivi. Comunque, la grafica dimostra la poca affidabilità dei modelli di propagazione indoor, perché il segnale presenta un comportamento irregolare al variare della distanza, e questa non linearità crea dei problemi nella stima della posizione in funzione dei valori di riferimento utilizzati nella equazione che descrive il modello impiegato.

4.2.5 Implementazione dei Metodi di Filtraggio

Tra gli algoritmi di filtraggio esposti nella sezione 3.5, quella più adatta e più semplice per essere implementata a livello software, è quella del *particle filter*. Tuttavia, andando ad osservare la formula associata ai pesi, la quale viene espressa in termini di una funzione di distribuzione di probabilità del tipo Gaussiana, dipende assolutamente dalle conoscenze delle coordinate giuste del terminale in tempo reale, creando la necessità di stabilire un metodo iniziale che permette di ottenere queste coordinate. Tra questi metodi si possono includere il metodo NNSS oppure l'approccio probabilistico, dei quali si conoscono le prestazioni, ottenute nei test precedenti.

Un approccio un po' diverso per generare le particelle, ma che comunque utilizza la distribuzione Gaussiana, è stato definito in [8], e si basa nella utilizzazione del modello di propagazione che stabilisce le perdite di percorso in ambiente indoor descritto nella sezione 1.2, per determinare la posizione relativa delle particelle. Comunque, come si ha visto nella sezione 4.2.4, il modello di propagazione non è molto affidabile, lo quale potrebbe anche compromettere le prestazioni del *particle filter*.

Insomma, l'effettività del *particle filter* dipende di quanto si approssima la posizione stimata a quella reale, ragione per la quale si consiglia di utilizzarlo solo nei casi in cui si conosce qualche informazione esatta (o almeno con un alto grado di precisione) sulla posizione del terminale (ad esempio la distribuzione iniziale delle particelle). Questo potrebbe essere possibile, ad esempio, impiegando sensori di prossimità o qualche dispositivo di rilevamento del genere. La infrastruttura a disposizione per questo sistema di localizzazione Wi-Fi, non consente l'utilizzo di questi dispositivi, per cui il *particle filter* non verrà implementato a livello software.

CAPITOLO V

IMPLEMENTAZIONE FINALE

5.1 Analisi del Problema

Nella sezione 3.6 si spiegava in linee generali come funzionano le reti neurali e come le sue caratteristiche aggiuntive tale l'evoluzione e l'apprendimento potrebbero essere utili per gestire un problema di localizzazione Wi-Fi, perché precisamente la stima del posizionamento di un terminale non è un problema statico, è piuttosto un problema dinamico perché non si conoscono a priori i diversi percorsi che potrebbero seguire i terminali, e quindi il suo comportamento è imprevedibile. Inoltre, si deve ricordare che il segnale radio varia nel tempo, dovuto alle caratteristiche del canale e dell'ambiente di propagazione, e questa situazione pone dei limiti nelle prestazioni dei metodi di localizzazione. Comunque, esistono diversi approcci di rete neurale per affrontare il problema di localizzazione di utenti mobili in un ambiente provvisto di una rete Wi-Fi: il clustering e il vector quantization sono alcuni esempi (per chiarimenti di queste definizioni si rimanda alla sezione 3.6).

Comunque ci sono grosse limitazioni per sviluppare una rete neurale evolutiva e queste sono discusse e analizzate durante la fase sperimentale:

- Innanzitutto, il modello di propagazione, che descrive le perdite di percorso del segnale in ambiente indoor, non è del tutto affidabile data la sua natura empirica, quindi è difficile conoscendo a priori certi valori di potenza e di distanza di riferimento, stimare la posizione esatta di un terminale.

Infatti, il fattore più influente è la riflessione del segnale, per la presenza di muri e mobili, provocando fluttuazioni nei livelli di potenza ricevuti, i quali sono fondamentali per determinare la posizione del terminale. Per questo è difficile, trovare neuroni con vettore di peso (valori di RSSI) simili, senno per un margine di tolleranza intorno ai ± 1 dB nel caso migliore, per una stessa posizione e per un intervallo di 2 secondi tra un valore e quello successivo. Questo, margine di tolleranza sarà proporzionale alla distanza tra l'AP di cui si è rilevato il valore RSSI e il terminale.

- Se l'algoritmo che stima la posizione, non ha buone prestazioni, non è possibile aggiornare e sostituire i neuroni nel tempo, perché non si può affermare che i valori restituiti rappresentino la vera posizione del terminale, a meno che si lavori con un'accuratezza molto alta.

- Il database com'è stato progettato inizialmente, non permette variazioni in fase online dei neuroni, che in questo caso rappresentano i punti fissi, contenendo l'informazione delle coordinate e dei valori RSSI per ogni AP.

Soluzioni a questi problemi potrebbero assicurare un aggiornamento affidabile della rete neurale. Infatti, una soluzione raggiungibile, sarebbe dotare l'infrastruttura già descritta di sensori Wi-Fi (a costi bassi), che permettano di rilevare grossi cambiamenti nei valori RSSI in una posizione ben determinata, ma comunque, questo aumenterebbe la complessità e i costi del sistema. Ovviamente, se non si è in grado di aggiornare i neuroni, non si può raggiungere una evoluzione effettiva, perché i neuroni saranno sempre gli stessi.

A questo punto, si decide utilizzare l'approccio neurale, soltanto per potere gestire la rete, e se in futuro, si trovano soluzioni appropriate ai problemi esposti, in

modo tale poter aggiungere le caratteristiche di evoluzione, non bisognerà fare grossi cambiamenti data la struttura a strati e modulari della rete neurale.

Il modello utilizzato è una semplificazione del modello di Sanchez [\[11\]](#), il quale a sua volta è un caso particolare di una rete neurale competitiva, che affronta meglio il problema della localizzazione Wi-Fi in ambiente indoor. L'apprendimento sarà non supervisionato perché la rete evolverà autonomamente i neuroni in ingresso, creando una clusterizzazione dettata solamente dai propri parametri, senza un addestramento guidato.

Il modello proposto da Sanchez e allievi è stato sviluppato per affrontare il problema dell'apprendimento e classificazione di pattern appartenenti a classi che non sono linearmente separabili e la cui distribuzione varia nel tempo, così come per pattern appartenenti ad ambienti stazionari e nonstazionari. La rete impara a classificare pattern in un modo supervisionato utilizzando per questo compito un insieme di neuroni nascosti che vengono creati e collegati fra loro a seconda del problema di classificazione in questione.

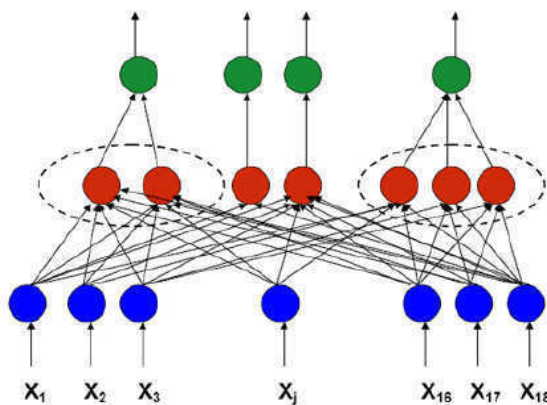


Figura 5.1 Architettura di una rete neurale evolutiva come quella proposta da Sanchez.

5.2 Realizzazione della Rete Neurale

La rete neurale a implementare (basata sul modello di Sanchez) avrà tre tipi di neuroni che si differenziano dalla tipologia di strato a cui essi appartengono. Per semplicità, è possibile distinguere principalmente tre strati: ingresso, nascosto (intermedio) e uscita. Ogni neurone ha delle caratteristiche comuni ma alcune differenze che dipendono dallo strato di appartenenza. In questo senso, si deve differenziare la realizzazione pratica della rete, nuovamente facendo distinzione in due fasi: quella *offline*, che corrisponde alla campionatura e l'addestramento della rete, e quella *online*, che corrisponde alla stima della posizione.

La fase *offline* praticamente non ha nessuna differenza rispetto a quella descritta nella sezione 4.2, durante i test dei metodi di localizzazione trovati nella letteratura, invece la fase *online* è quella che dipenderà della gestione della rete neurale. Alcune delle caratteristiche specifiche che avrà questa rete neurale sono:

- I neuroni, sono rappresentati dai punti fissi, nel caso dello strato nascosto, mentre che sono rappresentati dalle coordinate cartesiane, nel caso dello strato di uscita. Invece, i neuroni in ingresso, sono rappresentati dai valori correnti di RSSI ricevuti dal terminale per ogni AP visibile per la rete in quel istante di tempo. A questi valori in ingresso possono essere chiamati pattern d'ingresso o semplicemente pattern.
- Si potrebbe pensare di associare a ogni neurone un peso. I neuroni dello strato d'ingresso non hanno pesi. In tanto, i neuroni dello strato nascosto avranno un peso, secondo i valori di RSSI a essi vincolati durante la fase *offline*. I neuroni di uscita avranno un peso legato alla funzione dello strato nascosto. I neuroni oltre ai pesi dovranno possedere anche dei valori di uscita che rappresentano il potenziale del neurone stesso, che sarà l'ingresso per altri neuroni.

- I neuroni saranno elementi ignoranti, perciò sono definite le loro funzioni a livello di strato. Sarà la rete che gestirà il loro comportamento secondo lo strato di appartenenza. Questo tornerà utile in un eventuale cambiamento del loro comportamento (ad esempio un aggiornamento), per cui basterà aggiungere solo le funzioni e metodi necessari sulla rete già sviluppata, in modo tale di adattarsi alle nuove variazioni, e questo nuovamente va legato alla necessità di definire funzioni di strato. Per far sì che i neuroni eseguano le proprie funzioni si metteranno a disposizione metodi che possano distinguere il tipo di funzione e di strato; questi saranno richiamabili dai singoli neuroni per effettuare le sinapsi.

- Un altro punto importante sarà come la rete dovrà attendere il pattern d'ingresso. Questi possono essere inseriti in qualsiasi lunghezza e tipologia. Ad esempio per il sistema di localizzazione che si intende sviluppare il pattern avrà una lunghezza pari al numero di AP rilevati e con valori numerici che rappresentano le potenze registrate nei singoli punti campionati. E' anche importante controllare la coerenza nei valori contenuti del pattern, altrimenti l'informazione non potrà essere processata per gli strati successivi, e quindi la stima della posizione non avrà nessun senso.

- La rete sarà arricchita da un identificativo per la possibilità di avere più reti neurali che fanno localizzazione in uno stesso sistema, magari per ambienti diversi. Ad esempio, per distinguere tra diverse piani di uno stesso palazzo, sarà associato un codice identificativo della mappa di appartenenza a cui corrispondono un determinato insieme di neuroni.

- Quando la localizzazione avrà luogo ogni neurone sarà richiamato a eseguire le proprie operazioni definite nella rete. Il metodo che eseguirà questo procedimento sarà tipico di questa rete, e non farà altro che

richiamare la funzione standard di funzionamento della rete e prelevare dalla rete i risultati che interessano (uscite dell'ultimo strato).

5.2.1 Definizione delle funzioni di strato^[11]

Le funzioni di strato per questa rete neurale, sfrutteranno alcuni dei metodi di localizzazione analizzati nel capitolo 4, in particolare sono utilizzate quelle che portano migliori prestazioni, come sono il metodo NNSS e il metodo del centroide.

Il metodo NNSS, come presenta un comportamento statico in quanto alla stima della posizione, verrà utilizzato per determinare i neuroni nascosti più vicini al pattern d'ingresso, e in questo modo si potrà anche conoscere la ubicazione geografica del pattern in funzione dei cluster (zone) a cui appartengono questi neuroni. La discriminazione tra le zone, avviene mediante un peso determinato dai neuroni vincitori nello strato nascosto.

Invece il metodo del centroide, come presenta un comportamento dinamico in quanto alla stima della posizione, risulta adeguato per determinare la posizione che rappresenta meglio al pattern d'ingresso in funzione dei neuroni di uscita e dei pesi ottenuti dallo strato nascosto.

Allora, per cominciare, si fa riferimento alle notazioni espresse nella sezione 3.6, rispetto alle grandezze della rete neurale:

- Il vettore $X = \{X_1, X_2, \dots, X_N\}$ rappresenta il vettore degli ingressi, cioè il pattern, mentre che N rappresenta il numero di AP visibili per quel pattern.

- A rappresenta la funzione di attivazione dello strato.
- $F(A)$ rappresenta la funzione di propagazione dello strato.

Il primo strato avrà funzione di attivazione e propagazione identità, perciò entrambe le funzioni dovranno dare in uscita il valore d'ingresso (cioè il pattern).
 Quanto detto si può esprimere in termini matematici come:

$$A^{\text{ingresso}} = X$$

$$F^{\text{ingresso}}(A^{\text{ingresso}}) = A^{\text{ingresso}}$$

Ogni neurone del primo strato ha ora dei valori di uscita da consegnare allo strato successivo.

Lo strato nascosto avrà una funzione di attivazione con il compito di misurare la distanza fra il pattern d'ingresso e il vettore dei pesi dei neuroni nascosti. Questa distanza è calcolata mediante la differenza euclidea utilizzata nel metodo NNSS. Successivamente, questa distanza servirà alla funzione di propagazione per calcolare l'inverso della distanza più 1, così il neurone più vicino al pattern d'ingresso sarà quello con funzione di attivazione minima, comportando una funzione di propagazione massima. Quanto detto si può esprimere in termini matematici come:

$$A^{\text{nascosto}} = d(X, S) = \sqrt{(S_1 - X_1)^2 + (S_2 - X_2)^2 + \dots + (S_N - X_N)^2}$$

$$F^{\text{nascosto}}(A^{\text{nascosto}}) = \frac{1}{d(X, S) + 1}$$

Dove $S = \{S_1, S_2, \dots, S_N\}$ è il vettore dei pesi dei neuroni nascosti e N la lunghezza del vettore, che in questo caso dipenderà del numero di AP visibili al neurone nascosto.

Dalla fase di test, si tiene che la distanza euclidea per determinare il neurone nascosto più vicino al pattern d'ingresso, non dà come risultato una soluzione affidabile dal punto di vista puntuale, poiché potrebbero presentarsi tanti neuroni con differenza minima che si allontanano della vera posizione del terminale, per questo si introduce una soglia (threshold), che permetterà decidere all'amministratore della rete, quanti neuroni con differenza minima dentro lo strato nascosto potranno definire le zone a cui appartiene il pattern di ingresso.

Allora, in base al sottoinsieme di neuroni nascosti che rappresentano il comportamento del pattern d'ingresso (ordinati in una lista crescente secondo la differenza ottenuta e determinata dalla funzione di propagazione), è possibile ottenere i neuroni con i rispettivi cluster di appartenenza, e poi mediante una semplice regola statistica, determinare quale sarebbe il cluster con più probabilità (peso maggiore) di contenere al pattern d'ingresso. La regola statistica è definita come:

$$\text{Peso}_{\text{cluster}} = \frac{\text{numero volte che appare il cluster nella lista}}{\text{posizine cluster la prima volta che appare in lista} * \text{threshold}}$$

In poche parole, questa regola stabilisce il peso di un cluster dipende di due fattori: a) del numero di volte che si trova il cluster nella lista, dove l'ordine massimo della lista è determinato dalla soglia e b) della posizione nella lista del cluster quando è trovato per prima volta, posizione determinata dalla differenza calcolata. Da questa maniera più volte compare il cluster nella lista, è più probabile che appartenga il pattern a quel cluster, inoltre più piccola la differenza ottenuta più probabile che il pattern appartenga a quel cluster. Questa procedura del tutto euristica serve a determinare i cluster più vicini al pattern d'ingresso, informazione che sarà passata

allo strato di uscita, rappresentata da una lista contenente i cluster coinvolti con i rispettivi pesi.

Per lo strato di uscita la funzione di attivazione classificherà i neuroni di uscita in funzione dei pesi ottenuti dallo strato nascosto (solo saranno attivati quelli che corrispondono alle zone adiacenti al pattern d'ingresso) e in questo modo la funzione di propagazione sarà realizzata da una somma ponderata determinata dalla stessa equazione associata al centroide, quindi verrà propagata la stima della posizione del pattern d'ingresso, espressa in termini di un paio di coordinate (x,y).

Per collegare ogni neurone ricevuto dallo strato nascosto con il cluster di appartenenza, a questa somma se aggiunge il peso del cluster rispettivo, ottenuto anche dello strato nascosto.

$$A^{uscita} = LIST_N (Neuroni / Cluster, Peso_{Cluster})$$

$$F^{uscita}(A^{ingresso}) = \frac{\sum_{i=1}^N P_{Neuroni / Cluster}(i) \bar{X}}{\sum_{i=1}^N P_{Neuroni / Cluster}(i)} x Peso_{Cluster}$$

La scelta nella regola per stabilire i pesi associati ai cluster nello strato nascosto, è pensata in tal modo, di garantire parità di condizioni nella gara tra neuroni che appartengono ai cluster che possiedono più neuroni nascosti collegati (come succederebbe nei casi in cui il numero di neuroni collegati sono tanti a causa di una maggiore quantità di rilevamenti in fase di campionatura), oppure ai cluster con il neurone nascosto più vicino e che magari è l'unico nel suo cluster. Comunque, non è l'unica regola possibile e potrebbe essere modificata in funzione delle prestazioni dell'algoritmo finale.

Per quanto riguarda alle probabilità $p(i)$, vengono determinate considerando che i valori RSSI dei neuroni nascosto e i valori RSSI del pattern d'ingresso, presentano comportamento probabilistico simile a una funzione distribuzione Gaussiana (vedere sezione 3.4). Inoltre, l'insieme \bar{X} contiene la lista con le coordinate associate ai neuroni proveniente dallo strato nascosto.

Una volta determinato il centroide, la risposta della rete neurale, sarà quella delle coordinate stimate del terminale.

5.2.2 Diagramma di funzionamento della rete neurale

Ora è possibile, schematizzare di maniera sintetizzata, il funzionamento della rete neurale che verrà eseguita mediante un'applicazione sviluppata in ambiente Visual Studio, sotto linguaggio C#. Nella figura 5.1, si illustra, come sono implementate le funzioni di attivazione e di propagazione (A e F rispettivamente) della rete neurale. La nomenclatura utilizzata è la seguente:

- X rappresenta il pattern d'ingresso alla rete, costituito dai valori RSSI rilevato per ogni AP visibile in quel momento dal terminale, identificato per il suo rispettivo indirizzo MAC.

- S rappresenta una lista contenente i neuroni nascosti, e presenta una struttura simile al pattern d'ingresso, con la differenza che questo ultimo solo rappresenta un valore isolato, invece S rappresenta tutti i punti campionati in fase offline, rappresentati dai valori RSSI e dagli indirizzi MAC.

- C rappresenta una lista contenente i neuroni di uscita, che dovrebbero

coincidere con quelli nascosti, con la differenza che questa lista contiene l'informazione delle coordinate dei neuroni (x,y), ottenuta dallo stesso file XML, salvato in database, risultato della fase di campionatura.

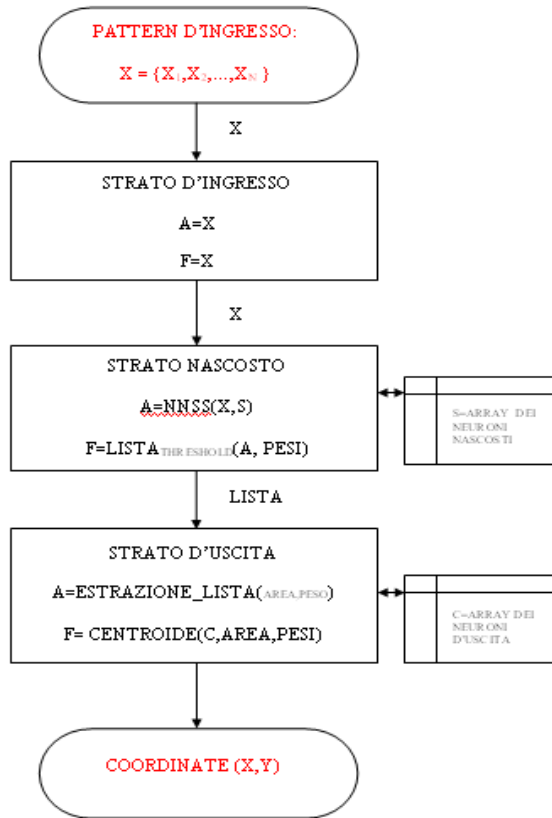


Figura 5.2 Diagramma di funzionamento della rete neurale.

L'implementazione via software della rete neurale, è basata nel diagramma precedente, cioè, si creano metodi che eseguono le funzioni di ogni strato, così ci sono 3 metodi (InputLayer(), HiddenLayer() e Output Layer()), uno per ogni strato, e che sono invocati per un metodo principale (NetworkManagment()), incaricato di ricevere il pattern d'ingresso e controllare la gestione dei neuroni dentro della rete.

La figura 5.2, mostra l'interazione tra i neuroni della rete neurale a implementare.

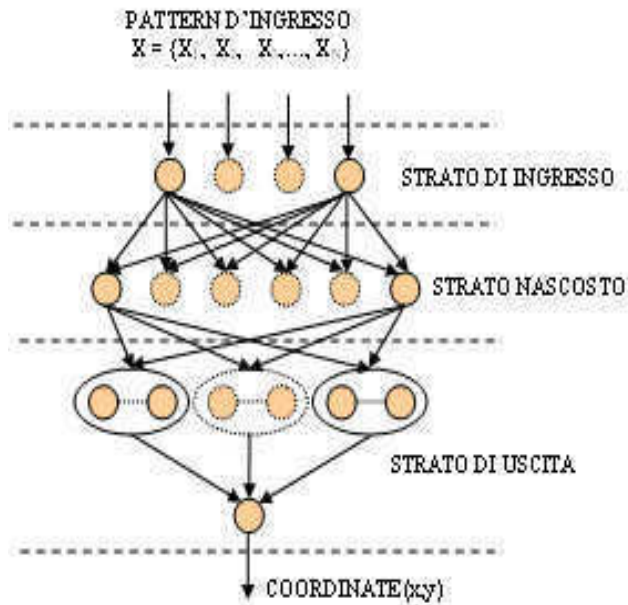


Figura 5.3: Modello di Rete Neurale per il sistema di localizzazione

5.3 Test della Rete Neurale

Innanzitutto sono stabilite le risorse utilizzate in fase di test, per la valutazione delle prestazioni della rete neurale, orientate alla localizzazione Wi-Fi in un ambiente chiuso (indoor) di tipo ufficio:

- Un server remoto, con Windows Server 2003 O.S., il quale gestisce il database con informazione riguardante alla fase di campionatura, stima della posizione tramite l'algoritmo che gestisce la rete neurale, etc.
- Un desktop o portatile per lo sviluppo software dell'algoritmo che

gestisce la rete neurale, con Windows XP Professional SP2 O.S., e con la piattaforma Microsoft .NET installata, compatibile con l'ambiente di sviluppo utilizzato Microsoft Visual Studio 2005. Tramite questo computer sarà possibile valutare e modificare la rete neurale. Questo era collegato via Ethernet al server remoto.

- Un router Wi-Fi, collegato al server remoto, per stabilire il collegamento wireless tra il dispositivo terminale e la rete apposita per gestire il posizionamento. E' stato utilizzato lo stesso modello di router impiegato durante i test descritti nel capitolo 4: SparkLAN WX-6615GT, che funziona sia come router, sia come AP.

- Gli AP impiegati per la rete di localizzazione, sono gli stessi 5 utilizzati durante i test descritti nel capitolo 4: NETGEAR WG-102. Ubicati negli stessi siti.

- I dispositivi terminali utilizzati sono gli stessi modelli di palmari impiegati durante i test descritti nel capitolo 4: HP iPAQ hx2700, che operano su Windows Mobile O.S.

Nuovamente, il funzionamento del sistema di localizzazione, è implementato in 2 fasi: fase offline o campionatura e fase online o posizionamento, dove sono eseguiti i compiti della rete neurale. Per la fase offline, non ci sono novità in quanto alla forma di campionare i punti, rispetto alla metodologia descritta nella sezione 4.2. L'ambiente in studio continua ad essere la sede a Roma di Thera S.r.l. (vedere la mappa nella figura 4.8). I campioni sono stati presi sugli stessi 22 punti, utilizzando un palmare con l'applicazione di campionatura installata, misurando 100 campioni per ciascuna posizione scelta, considerando tutti i versi (nord, sud, est, ovest). Questi poi sono ponderati in media via software. Per la fase online invece, utilizzando lo stesso palmare impiegato durante la campionatura, andando sull'opzione "Posizione"

si può valutare il comportamento della rete neurale rispetto al posizionamento, ottenuto mediante la esecuzione delle funzioni di strato, applicate ad ogni pattern processato dal terminale. Le prestazioni che sono nuovamente considerate, sono l'accuratezza e la precisione sulla stima della posizione, e sono mostrate nella figura 5.3.

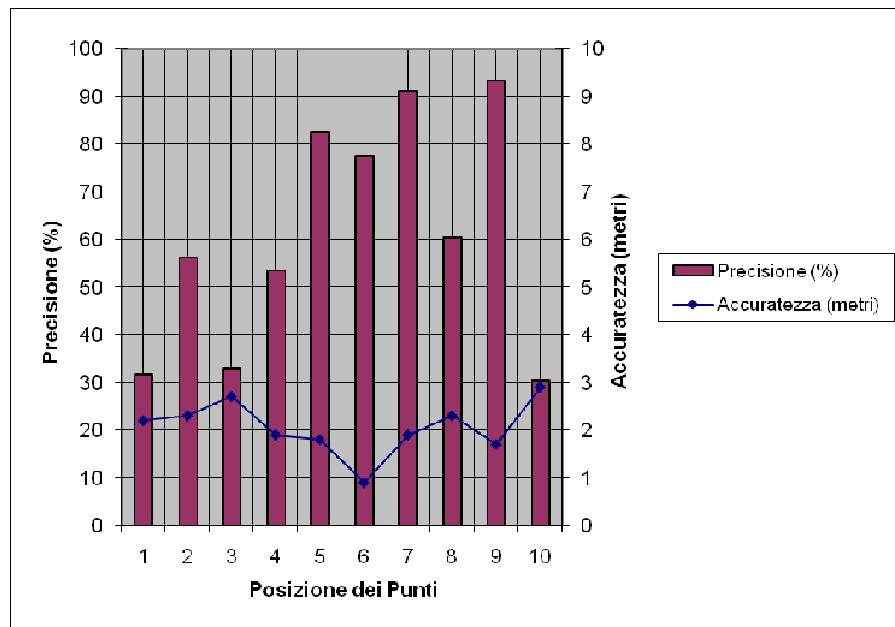


Figura 5.4: Prestazioni della Rete Neurale.

Le osservazioni rispetto a questa implementazione sono le seguenti:

- Le prestazioni medie sono: una precisione del 60% e un'accuratezza di 2.06 metri, risultati che si assomigliano molto a quelli ottenuti con il metodo del centroide.
- La stima in questo caso, è il risultato di un processo di ponderazione, quindi la posizione del terminale è determinata in modo dinamico, rendendo la rete neurale adatta ai problemi di spostamento del terminale a velocità

bassa (considerando che la velocità massima di spostamento di un utente dentro un ambiente chiuso sarebbe intorno ai 5 m/s).

- Nel caso delle zone molto strette ma lunghe (tipo i corridoi) si presentano gli stessi problemi di precisione avuti con i metodi analizzati nel capitolo 4, problemi che potrebbero essere risolti con una migliore pianificazione sull'ubicazione dei punti in fase di campionatura, la quantità di questi e la distanza tra di loro.

- Anche per il caso delle zone non coperte direttamente per un AP, dove i segnali si sovrappongono e il pattern d'ingresso presenta un comportamento molto instabile rispetto al valore attuale e quello successivo, la stima potrebbe allontanarsi del valore reale con un'accuratezza molto superiore a quella media. Questo potrebbe essere migliorato, spostando gli AP verso l'alto, o almeno in funzione dell'altezza media di una persona, per ridurre il problema di riflessione, fattore che crea maggiore instabilità del segnale in ambienti chiusi. Un'altra modifica a livello software, che permetterebbe la gestione dei problemi dell'instabilità del pattern d'ingresso, mira a controllare la qualità del pattern ricevuto dai terminali, e processarli mediante approcci statistici, con lo scopo di evitare risultati indesiderabili nel risultato della stima.

Implementando il meccanismo di "pila" a livello software, è stato possibile accumulare l'informazione ricevuta dai terminali (interessano in questo caso solo i valori RSSI) durante la stima del posizionamento, per poi determinare la qualità del segnale in funzione dei valori ricevuti in un certo intervallo. Perciò si propone, studiare la varianza dei valori accumulati nella pila, per avere un modo di verificare quanto variano i valori RSSI ricevuti in ogni istante di tempo. A questo punto, si può modificare lo strato d'ingresso, in modo tale di valutare la varianza dei valori RSSI immagazzinati nella pila.

Prove preliminari, dimostrano che il valore medio della varianza, permette determinare se il terminale si sta spostando, oppure si sta fermo. In poche parole la regola è che per bassi valori di varianza il terminale potrebbe considerare fermo, per cui i pattern d'ingresso non variano molto uno dall'altro, mentre che quando comincia a salire dopo un certo valore, si potrebbe affermare che il terminale si sta spostando. Non è una regola precisa, perché è possibile una fluttuazioni veloce dei valori RSSI, a causa dei disturbi propri dell'ambiente, anche se il terminale non si muove, e quindi la stima potrebbe essere sbagliata. Siccome questo analisi è molto complicato, solo si prenderanno in considerazioni due situazioni: una varianza media bassa, indicherà che il terminale non si muove, e una varianza media alta, indicherà che il terminale si sposta. Con questi criteri, è possibile evitare ricalcolare la stima se si ipotizza che il terminale è fermo, oppure determinare una nuova posizione se si ipotizza che il terminale si sta muovendo.

La pila utilizzata a livello software, soltanto accumulerà un certo numero di pattern d'ingresso (impostato a 3), e poi ogni volta che si ha un nuovo pattern, il valore più vecchio è cancellato, così si mantiene un istoriale dei valori RSSI del segnale. Dopodiché è calcolata la varianza di tutti i pattern utilizzando la seguente formula:

$$\sigma^2 = \frac{1}{n} \sum_{i=1}^n (x_i - \mu)^2$$

Dove x_i rappresenta un singolo valori RSSI e μ la media di tutti i n valori RSSI appartenenti allo stesso AP, che si trovano nella pila. Ovviamente, per questo calcolo si deve considerare lo stesso terminale, altrimenti potrebbero combinarsi i valori RSSI dai diversi terminali, creando problemi nella stima.

Una modifica messa in pratica a livello hardware è quella di spostare gli AP più in alto e di maniera regolare sull'ambiente creando condizioni di parità e uniformità rispetto ai segnali rilevati dai terminali. Questo riduce anche i problemi di riflessione e di propagazione del segnale.

Un'ultima modifica che si potrebbe provare, si riferisce alla distribuzione dei punti campionati. La distribuzione deve essere fatta, considerando l'ambiente e le dimensioni delle diverse zone. Questo per evitare campionare zone diverse ma con punti così vicini di creare problemi di decisione nella stima della posizione.

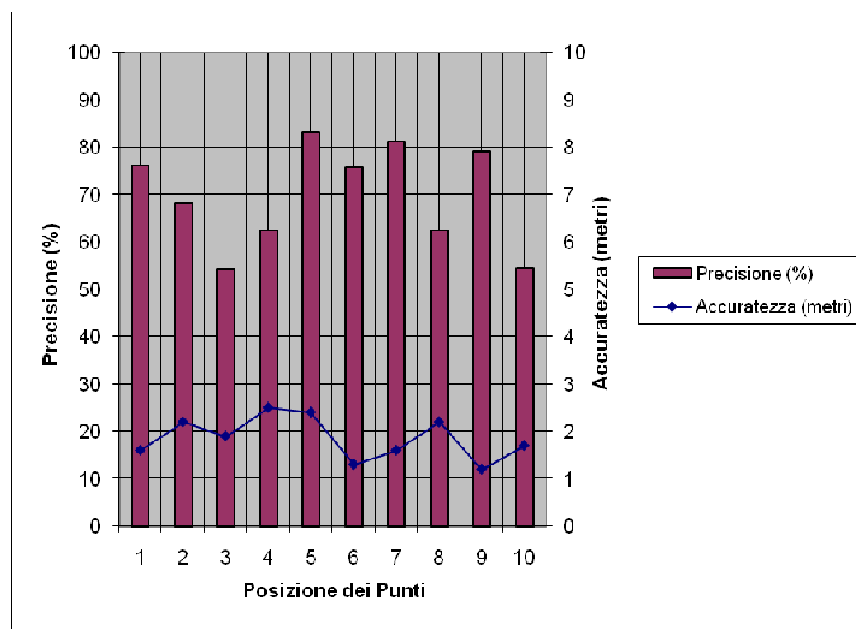


Figura 5.5 Prestazioni della Rete Neurale dopo le modifiche.

I risultati di tutte queste modifiche, possono essere verificati nella figura 5.4, che mostra le prestazioni per questa versione della rete neurale modificata. Insomma, la precisione e l'accuratezza aumentano (70% e 1.86 metri), ma comunque i problemi descritti nella versione originale della rete neurale non spariscono, anzi sono ridotti,

per cui il problema di localizzazione Wi-Fi in ambienti indoor, è fortemente costretto all'instabilità del pattern d'ingresso in quanto ai valori RSSI rilevati dai terminali variano fortemente nel tempo. Ora, saranno presentati comportamenti della rete neurale fronte a certe variazioni sia a livello software sia a livello hardware, per stabilire come potrebbero queste modifiche influire sulle prestazioni di un sistema di localizzazione Wi-Fi. Innanzitutto, si vuole mostrare l'influenza del numero di campioni per ogni punto sulla precisione e l'accuratezza del sistema. Per far ciò sono stati ricampionati soltanto i punti di alcune delle zone del ambiente di studio con diversi valori del "Numero di campionature" dell'applicazione Campionatura. Sono state scelte poche zone per il grosso lavoro che comporta il campionamento in un ambiente di lavoro di tipo ufficio. I valori settati di campionature sono {10,30,50,70,90} e le zone di riferimento erano quelle più accessibili {sala 1, sala 4}. Sono stati ricampionati gli stessi punti per ciascuna zona con un intervallo di 1 ogni secondo tra ogni rilevamento (il minimo che permette l'applicazione di campionatura). I risultati sono riportati nella figura 5.5.

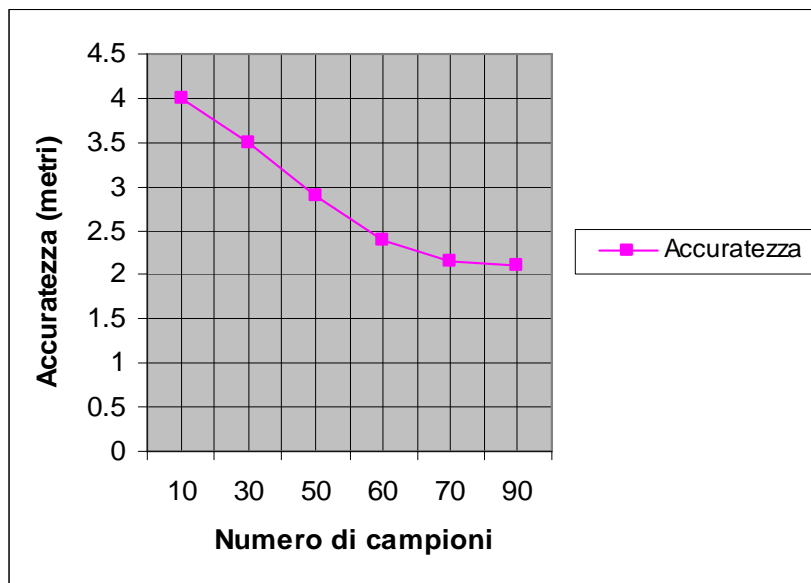


Figura 5.6: Effetti sulla stima al variare il numero di campioni.

Dalla figura, si osserva naturalmente che all'aumentare il numero di campioni lordi per ciascun punto, meglio saranno le prestazioni del sistema di localizzazione, specificamente si ottiene maggiore stabilità nella stima della posizione. Questo perché si considerano le variazioni del segnale radio in un intervallo di tempo molto lungo, avendo un comportamento più ampio del segnale. Comunque, si aspetta una convergenza rispetto al valore dell'accuratezza intorno ai 2 metri.

Un altro fattore che potrebbe influenzare le prestazioni del sistema di localizzazione, è il numero di AP impiegati e la sua disposizione nell'ambiente. Sulla disposizione si è già accennato che l'altezza degli stessi è molto importante per ridurre la riflessione e per questo era consigliabile metterli il più alto possibile. In quanto alla densità degli AP, non basta con aumentare la quantità di questi, anche la copertura è molto importante. In termini d'irradiazione si preferiscono antenne direzionali che quelle omnidirezionali, perché queste ultime non possono essere manipolate e quindi si creano problemi di sovrapposizione che come si è visto nel capitolo 4 creano problemi di stabilità del segnale e soprattutto i metodi di localizzazione potrebbero restituire valori errati perché si difficoltà l'individuazione dei segnali che vengono leggermente affetti per la presenza degli altri. Per analizzare questo effetto, occorre disattivare gli AP necessari e ricampionare (100 campioni per punto, 1 campione ogni 2 secondi) le zone in studio (sale 3 e 4), sugli stessi punti. Nella figura si mostra l'influenza sulle prestazioni del sistema di localizzazione Wi-Fi, al variare il numero di AP della rete.

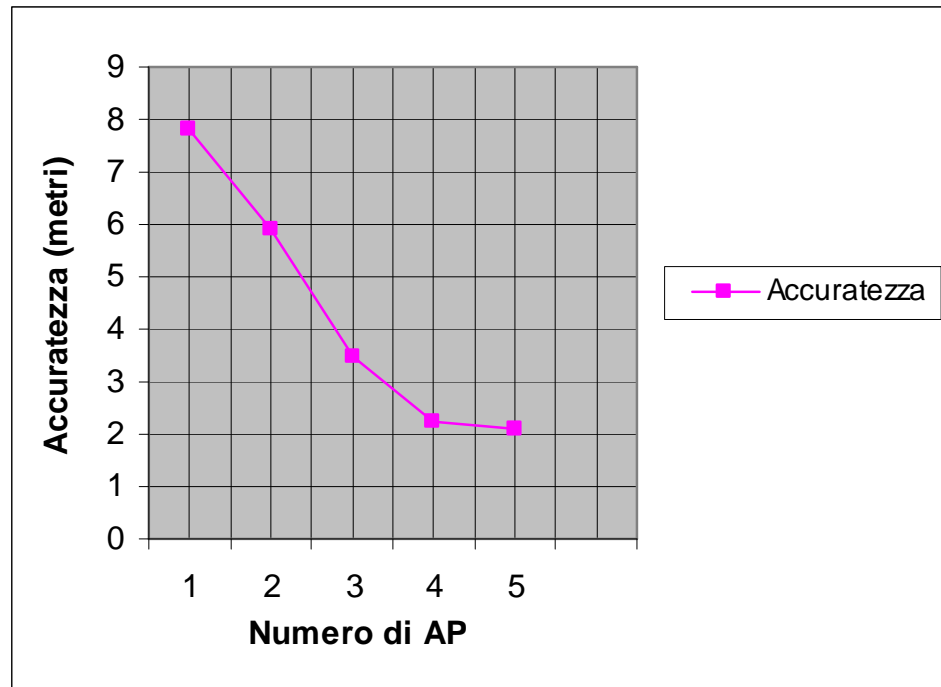


Figura 5.7: Effetti sulla stima al variare il numero di AP.

Dalla figura 5.6 si osserva, più sono gli AP, meglio l'individuazione dei segnali che permettono di stimare la posizione del terminale in funzione dei valori RSSI rilevati. Tuttavia vale, quanto è detto prima della copertura radio, per cui sempre bisogna collocare gli AP, in funzione del tipo di antenne e il suo patrone di radiazione, ed anche in funzione delle dimensioni dell'area a coprire. Dalle rilevazioni ottenute durante i test, si ha che i valori RSSI stabili e soddisfacenti per stimare la posizione, saranno superiori ai -80 dBm, oltre a questo valore la stima diventa instabile, e bisogna aumentare il numero di AP per assicurare una buona copertura, perché la sovrapposizione dei patroni di radiazioni potrebbe causare interferenze e fluttuazioni nei segnali rilevati, e quindi un numero grande di AP, non sempre sarà la migliore soluzione. Inoltre, utilizzare un numero elevato di AP, aumenta i costi di progettazione della rete neurale, quindi questo pone anche un limite sui numeri di AP a impiegare. Allora la scelta giusta, è che si deve cercare un

compromesso tra i costi e la precisione del sistema, questa ultima determinata dal tipo di servizio che si vuole offrire all'utente. La latenza di questo sistema è di circa 2 secondi, tempo che tiene conto del collegamento del terminale al server remoto in modo wireless, le operazioni via software della rete neurale, e la restituzione delle coordinate sulla mappa. La maggior parte del tempo si perde nei collegamenti con il server e nell'elaborazione dell'informazione ottenuta in fase di campionatura. Tipicamente, i test sono stati effettuati con un solo dispositivo palmare, perché il segnale elaborato per lo strato d'ingresso della rete neurale, utilizza una pila che accumula i valori precedenti, regolando le fluttuazioni del segnale e di certe condizioni che potrebbero influire sulla stima. Quindi questa elaborazione considera solo un terminale, e non è in grado di creare tante pile per ogni terminale, è per questo che per studiare il comportamento della rete con più di una richiesta di localizzazione, si torna alla rete neurale originale (senza la pila), e così è possibile verificare il comportamento del sistema. Le prove mostrano che i terminali mobili, sono in grado di collegarsi al server di maniera indipendenti dagli altri senza problemi, quindi è possibile ricevere tante richieste di localizzazione, anche se con un tempo di attesa di circa 1 secondo. Nel caso in cui si trovavano terminali nella stessa posizione, le stime delle posizioni restituite dalla rete erano simili.

CAPITOLO VI

CASO DI STUDIO

Nel mese di dicembre 2007, sono state fatte una serie di prove dell'intero sistema di localizzazione Wi-Fi, nel Museo Civico di Albano, nella località di Albano Laziale, con lo scopo di verificare le prestazioni del sistema in un ambiente diverso a quello di un ufficio.

L'accuratezza raggiunta per l'implementazione mostrata nel capitolo 5, non è adeguata per conoscere la posizione esatta dei beni esposti nelle diverse sale del museo, per questo che si decide di applicare una localizzazione geografica, anziché una localizzazione puntuale com'è stato fatto finora. In questo contesto con la localizzazione geografica si intende alla determinazione univoca delle sale del museo.

Le risorse hardware e software utilizzate per questa implementazione sono:

- Un server remoto, con Windows Server 2003 O.S., il quale gestisce il database con l'informazione riguardante la fase di campionatura, la stima della posizione tramite l'esecuzione della rete neurale, etc. Questo server è provvisto di una scheda Wi-Fi, in modo tale di potersi collegare via wireless con gli AP e i terminali mobili. Inoltre, utilizza la tecnologia push, che permette gestire le applicazioni in modo remoto tramite web service sulla piattaforma .NET, per cui viene installato l'IIS.

- Un portatile per lo sviluppo e controllo del software che gestisce la rete neurale, con Windows XP Professional SP2 e con la piattaforma Microsoft

.NET installata compatibile con l'ambiente di sviluppo impiegato Microsoft Visual Studio 2005. Anche questo era provvisto di una scheda Wi-Fi.

- Gli AP impiegati per definire la rete di localizzazione, sono gli stessi utilizzati nei test descritti nei capitoli 4 e 5: NETGEAR WG-102. Per l'ubicazione si rimanda alla sezione Mappe del Museo Civico di Albano dell'appendice. In questo caso sono impiegati 10 AP per coprire tutti i piani del museo.

- I dispositivi terminali utilizzati, sono gli stessi modelli di palmare impiegati nei test descritti nei capitoli 4 e 5: HP iPAQ hx2700, che operano con Windows Mobile O.S. Su questi palmari fu installata l'applicazione di campionatura.

Per quanto riguarda al campionamento, non sono considerati gli uffici amministrativi, quindi sono stati campionati soltanto le 22 sale del museo, con 2 punti distanti almeno 2 metri per ogni sala, misurando in tutti i sensi rispetto allo stesso punto. Il numero di campioni è impostato in 50. Tuttavia, in questo campionamento a differenza dei precedenti, si deve fare attenzione ad associare ogni punto al piano rispettivo, mediante un identificativo che permetta discriminare tra le diverse mappe (per ciascun piano) . In questo modo la rete neurale, sarà in grado di stimare la posizione del terminale, sul piano giusto. Questo identificativo si ottiene direttamente durante la campionatura, poiché il campionamento si fa in funzione della mappa, quindi i punti sono associati alla mappa corrispondente. Alla fine, la posizione non è più rappresentata da un sistema bidimensionale, perché esiste una terza coordinata che corrisponde al piano in cui si trova il terminale. Tutta questa informazione è accessibile dal file in formato XML elaborato dal server, con tutta l'informazione di campionatura.

Per definire con maggiore precisione la localizzazione geografica, si ha

utilizzato l'applicazione Map sviluppata per l'azienda Thera S.r.l., che serve a disegnare e gestire le mappe. Con questa applicazione, è possibile delimitare le zone, in modo tale che il server dopo di stimare via software la posizione del terminale, cerca la sala alla quale appartiene il punto ottenuto, a partire dell'associazione salvata sul server, e finalmente restituisce all'utente la sala cui si trova in quel momento. Gli AP sono ubicati in maniera aleatori, in modo tale di coprire i 3 piani del museo. La disposizione scelta degli AP è come segue:

- 5 AP per il piano seminterrato, configurati in topologia, stella, per eseguire la funzione di *repeater*. C'è un coordinatore centrale che si collega in modo bridge punto-multipunto al resto degli AP. Il resto si collega in modo bridge punto-punto al coordinatore centrale. In questo modo si crea una copertura che raggiunge tutto il piano. Per far ciò, è necessario configurare i diversi AP con il proprio software di gestione.

- 5 AP per primo piano, anche questi configurati in topologia stella, com'è spiegato prima.

- Nessun AP nel piano rialzato, perché soltanto ci sono 3 sale da coprire, e secondo le misurazioni rilevate su queste sale, il segnale degli AP degli piani raggiungeva quelle sale, con un livello di potenza minore ai -80 dBm, nella maggioranza dei casi, lo quale è soddisfacente perché in questi livelli non si presentano grosse variazioni del segnale radio.

Tutti i dispositivi Wi-Fi furono indirizzati via IP al server remoto, in modo manuale, per evitare conflitti con gli indirizzi IP si questi sono spenti. Di questa maniera, tutti i dispositivi che formano parte della rete, sono collegati tra di loro alla stessa rete, ed accessibili da qualsiasi punto.

Per eseguire il posizionamento, il server sempre in attesa di un beacon con l'informazione dei valori RSSI rilevati dal terminale, esegue la stima attraverso i meccanismi specificati dalla rete neurale, implementata a livello software, mediante una serie di applicazioni accessibili da remoto mediante il web service disposto sul server remoto. Una volta ottenuta la posizione, il server cerca la sala associata a quel punto, e poi questa viene informazione è mostrata sul palmare.

I risultati ottenuti sono favorevoli, perché si parte della localizzazione geografica e non quella puntuale, quindi con una grande precisione la stima ottenuta restituisce la sala giusta in cui si trova il terminale e soprattutto nel piano corrispondente. Inoltre, sebbene la localizzazione è geografica, si deve far notare che con un numero di campioni più basso di quello utilizzato durante i test effettuati e descritti nei capitoli 4 e 5, i risultati ottenuti mostrano maggiore stabilità e precisione. Questo è possibile grazie alle caratteristiche fisiche dell'ambiente, dove le zone sono più regolari, le divisioni fisiche tali come muri, porte, mobili, sono più robuste. Inoltre l'altezza media degli AP (sopra i 2 metri) è idonea per ridurre gli effetti della riflessione sui segnali che rilevano i terminali.

La connessione in modalità *repeater* tra gli AP periferici e quello centrale, è una funzionalità molto utile in ambiente di grosse dimensioni come succede in questo caso, per permettere sempre al terminale di collegarsi via wireless con l'AP che abbia il segnale più forte in quel momento, diminuendo la latenza complessiva del sistema. Questo è un fattore importante, perché l'utente si aspetta una risposta immediata dell'applicazione.

CONCLUSIONI

L'obiettivo principale di questa tesi è stato quello di progettare e testare un sistema di localizzazione Wi-Fi in ambiente indoor basato in una delle tecniche di posizionamento proposte in letteratura, più frequentemente utilizzata in questi sistemi, e denominata RSSI. Successivamente sono stati scelti un insieme di metodi di localizzazione, anche trovati in letteratura e utilizzati in altri sistemi, compatibili con la tecnica RSSI. Questo studio, ha permesso un'ulteriore implementazione mediante algoritmi noti e modificati secondo le condizioni del ambiente in esame, con lo scopo di studiare il loro comportamento fronte al problema di localizzazione in ambienti chiusi. Il risultato finale è utilizzare questo sistema dentro di un prodotto denominato Dimmi Museo, commercializzato per l'azienda Thera S.r.l., basato in una guida museale. Il metodo di localizzazione, in sintesi ed eseguito tramite una rete neurale a strati. Questi strati sono dotati di funzioni che permettono manipolare il segnale di ingresso per determinare la posizione di un terminale mobile dotato di schede Wi-Fi, e collegato a una rete, anche questa Wi-Fi. Le prestazioni raggiunte mediante questa rete neurale, sono legate al performance dei due metodi che hanno presentato la miglior risposta in fase di test. Questi metodi sono il NNSS e il metodo del centroide. Le prestazioni finali del sistema di localizzazione sono rappresentate in termini di accuratezza e precisione, con valori rispettivi di 1,8 metri e 70%. Questi valori possono essere confrontati con altri sistemi proprietari, come quelli accennati nella sezione 6 dell'appendice, i quali mostrano prestazioni molto vicine a quelle ottenute via sperimentale. Comunque, queste prestazioni non possono essere generalizzate per tutti i casi, data la complessità del problema di localizzazione indoor, poiché le misurazioni dei valori RSSI, dipendono dalla propagazione del segnale, il quale è soggetto alle fluttuazioni occasionate per le costanti variazioni negli ambienti chiusi.

Il percorso effettuato per ottenere l'applicazione finale ha permesso di venire a conoscenza delle problematiche tipiche di un ambiente indoor e di proporre nuove soluzioni da applicare direttamente al sistema o altrimenti ad evoluzioni future. A continuazione vengono segnalati i problemi e limitazioni più importanti, trovate durante lo svolgimento di questo progetto:

- La propagazione del segnale radio in ambiente indoor, è il principale problema da gestire in questi sistemi. Nessuno dei metodi studiati considera il comportamento del canale radio, il quale è variante in tempo, e per questo il segnale radio è molto sensibile alle variazioni del mezzo. Si crede, che il *performance* del sistema, deve essere orientato in potere predire il comportamento del segnale, e procurare il trattamento statistico necessario via software, per evitare che queste variazioni possano disturbare la stima della posizione.

- L'altezza delle antenne degli AP deve essere la più alta possibile. Inoltre, la posizione dove verranno inseriti e la quantità deve essere studiata in funzione del raggio di copertura e il patrone di radiazione delle antenne, con lo scopo di coprire perfettamente le aree dove si vuole offrire il servizio di localizzazione.

- La campionatura può essere decisiva nelle prestazioni finali del sistema di localizzazione. Questa è effettiva sempre che non variano le condizioni iniziali dei mobili presenti nell'ambiente in studio. L'intervallo di campionatura anche influisce sulle prestazioni finali, più alto, meno si conosce il comportamento del segnale radio. Un ultimo fattore, si riferisce alla risoluzione considerata durante la scelta dei diversi punti di campionamenti sulla mappa, quindi non si può accertare sulla posizione esatta dei punti, a meno di un certo margine di tolleranza, a causa degli errori nella manipolazione della mappa e dei fattori di scala.

Un problema che dovrebbe essere studiato e risolto ulteriormente, è la risposta del sistema in ambienti caratterizzati di una grande affluenza di utenti, oppure che variano costantemente, poiché questo degrada i livelli di potenza del segnale ricevuto. Due approcci sono stati suggeriti nella letteratura: fare diverse campionature *offline* e poi utilizzare quella più idonea alla situazione in esame, oppure sfruttare i diversi approcci delle reti neurali, per creare un aggiornamento ed evoluzione automatico e periodico, che tenga conto delle variazioni del segnale radio. Esistono anche altri fattori che determinano le prestazioni di un sistema di localizzazione sono la velocità di spostamento del terminale e il tempo di risposta del sistema. In questo progetto, si ha riscontrato una forte dipendenza alla velocità di spostamento del terminale, cioè del utente che lo porta, dovuta alle fluttuazioni veloci del segnale, situazione che permette affermare, che più veloce si sposta l'utente, più grande è la variazione dei valori RSSI rilevati, producendo instabilità nella stima della posizione. In quanto al tempo di risposta, il sistema di localizzazione, per il caso dei palmari, presenta un ritardo di incirca 2 secondi, tempo impiegato maggiormente nella campionatura dei valori RSSI e nell'elaborazione dei dati sul server. Per il caso della guida museale, questi due comportamenti, non creano dei problemi perché si parte dall'ipotesi che un'utente in giro per un museo, è un soggetto che si sposta lentamente tra le diverse opere e sale del museo. Anche se, il problema dell'implementazione di un sistema di localizzazione Wi-Fi, in ambiente indoor, è stato gestito senza problemi con le prestazioni raggiunte, tuttavia può essere perfezionato, man mano, si trovino nuove soluzioni ai problemi esposti, e manipolando il software su cui è stato sviluppato il sistema. Il sistema di localizzazione progettato potrebbe divenire la base di un sistema più ampio applicabile e altre situazione e contesti (fiere, ospedali, ecc.). Infatti, come già per le reti, questi servizi sono stati concepiti con uno sguardo al futuro, ovvero, con la capacità di implementare nuove soluzioni con le minime modifiche al progetto già sviluppato.

BIBLIOGRAFIA

- [1] Küpper Axel. *Location-based Services: Fundamentals and Operation*. Wiley, 2005.

- [2] Rappaport Theodore S. *Wireless Communications: Principles and Practice*. Prentice Hall Communications Engineering and Emerging Technologies Series, 2002.

- [3] *Dispense del corso Comunicazioni Mobili 2* del prof. Roberto Cusani, Dipartimento INFOCOM, Facoltà di Ingegneria, Università degli Studi di Roma “La Sapienza”.

- [4] IEEE Standards Association: *IEEE 802.11: LAN/MAN Wireless LANS Standards*. IEEE, 1999-2007

- [5] Lyu M., Kin F. and Tilen M.: *Location-Based Multimedia Mobile Service. Final Year Project Report*. Department of Computer Science and Engineering CUHK, 2005.

- [6] Patmanathan, Vinod. *Area Localization using WLAN*. Master of Science Thesis, Stockholm, Sweden, 2006.

- [7] Youssef, M. and Agrawala, A.: *The Horus WLAN Location Determination System*. Department of Computer Science University of Maryland, 2005.

- [8] Zawar S. and Malaney R. A.: *Particles Filters and Positioning Tracking in Wi-Fi Networks*. IEEE, 2006.

- [9] Wang H., Lenz H., Szabo A., Bamberger J. And Hanebeck U.: *WLAN-Based Pedestrian Tracking Using Particle Filters and Low-Cost MEMS Sensors*. Workshop on Positioning, Navigation and Communication (WPCN), 2007.
- [10] Widyawan, Klepal M. and Pesch Dirk: *RF-BASED Location system In Harsh Environment*. IFAWC, Tel-Aviv, Israel, 2007.
- [11] Buda Claudio. *Localizzazione Indoor in WLAN attraverso Rete Neuronale Evolutiva*. Tesi del corso di Laurea Specialistica in Ingegneria dei Sistemi e delle Tecnologie dell'Informazione, Università degli studi di Bologna, 2005.
- [12] Bahl, P. and Padmanabhan, V.: *RADAR: An In-Building RF-Based User Location and Tracking System*. Microsoft Research, 2000.
- [15] Bahl P, Padmanabhan V. and Balachandran A.: *A software System for Locating Mobile Users: Design, Evaluation, and Lessons*. Microsoft Research, 2002.
- [16] Cheng, Y., Chawathe, Y., LaMarca A. and Krumm, J.: *Accuracy Characterization for Metropolitan-scale Wi-Fi Localization*. IEEE, 2005.
- [17] Bernardini Angelo. *Lezioni del Corso di Radiocomunicazioni*. Edizioni Ingegneria 2000, 2005.
- [18] Haykin Simon. *Kalman Filtering and Neural Networks*. John Wiley & Son, INC. 2001.
- [19] Sito Hewlett-Packard, <<http://www.hp.com>> [Consulta 2007]
- [20] Sito di SparkLAN, <<http://www.sparklan.com>> [Consulta 2007]

- [21] Sito di NETGEAR Italia, <www.netgear.it> [Consulta 2007]
- [22]Sito di RADAR,
<http://research.microsoft.com/~bahl/ms_projects/radardemo/demo.htm> [Consulta 2007]
- [23] Sito di Place Lab, <<http://www.placelab.org/>> [Consulta 2007]
- [24] Sito di Herecast, <<http://www.herecast.com>> [Consulta 2007]
- [25] Sito di Google Map Mobile, <<http://www.google.it/mobile/gmm/index.html>> [Consulta 2007]
- [26] Sito di Ekahau, <<http://www.ekahau.com>> [Consulta 2007]
- [27] Sito di Aeroscout, <<http://www.aeroscout.com>> [Consulta 2007]
- [28] Sito di Navizon, <<http://www.navizon.com>> [Consulta 2007]

[APPENDICE 1]

[Caratteristiche Tecniche del Palmare HP modello iPAQ serie hx2700^[19]]



System features	
See detailed specs	US QuickSpecs » html » pdf
Operating system	Windows Mobile® 5.0 for Pocket PC, Premium Edition (Pocket versions of Microsoft software are included (Outlook, Word, Excel and Internet Explorer for Pocket PC))
Processors available	Marvell PXA270 processor 624 MHz
Memory	
Memory	64 MB SDRAM
Flash memory	320 MB Flash ROM 512 MB Flash ROM
Audio, Slots, and Ports	
Ports	Standard: 1 IrDA 1 USB 1.1 Client 1 Serial RS232 1 DC power 1 cradle interface 1 adapter connection 1 AC/DC connector
Slots	1 secure digital 1 Compact flash (Support 1-bit SDIO and 4-bit SD/MMC type memory standard)
Communication features	
Wireless	Integrated Wi-Fi (802.11 b); Integrated Bluetooth wireless technology
Software	
Software installed	Outlook Mobile; Microsoft ActiveSync 3.7.1; Microsoft Reader eBooks
Software included, ROM	ActiveSync; Adobe Reader for PC; ClearVUE Suite (Word and Excel); ClearVUE Suite (PPT and PDF); Media Player Desktop

[APPENDICE 2]

[Datasheet del Router Sparklan Modello WX-6615GT^[20]]



SPECIFICATIONS

INTERFACE	
WAN	1 x 10/100Base-T RJ-45 socket
LAN	4 x 10/100Base-T Ethernet switch port (MDI/MDIX)
Wireless	IEEE 802.11b,b+g, g-only, Afterburner
LED Indicators	LED x7: Power, Wireless, WAN, LAN4, LAN3, LAN2, LAN1
Antenna	2 x 3dBi Detachable Antennas
Standards	IEEE 802.3, 802.3u, 802.11b and 802.11g
Reset Button	Reset to default setting
RADIO CHARACTERISTICS	
Frequency Bands	2.412~ 2.4385GHz ISM Band
Modulation Tech	802.11b: 11Mbps & 5.5Mbps CCK; 2Mbps: DQPSK; 1Mbps: BPSK 802.11g: OFDM
Media Access Protocol	CSMA/CA with ACK
Transmit Power	16 dBm±1@ Nominal Temp Range (Maximum)
Receive Sensitivity	11 Mbps 10-5 BER @ -80dBm; 54 Mbps 10-5 BER @ -65dBm
Data Rates	802.11b: 11Mbps, 5.5Mbps, 2Mbps, 1Mbps, auto fallback 802.11g: 54, 48, 36, 24, 18, 12, 9, 6 Mbps Turbo-G: Afterburner option (Enhanced throughput by up to 40%)
Operating Range	802.11b: 11Mbps Up to 300m (open space), 35-60m (Indoor) 802.11g: 54Mbps up to 50m (open space), 20m (Indoor) * The performance various depend on environment.
Number of Channels	802.11b: USA and Taiwan 11, China and Europe 13 802.11g: USA, Europe, and Japan - 13
SOFTWARE FEATURES	
Internet	Dynamic / Static IP, PPPoE, PPTP
Security	WEP, WPA, Enable/Disable SSID, Password Protection, VPN Pass-Through, Web Filters, DMZ, Block WAN ICMP
System	Firmware upgrade, Remote Management, Traffic Log
Advanced	DHCP Server Enable/Disable, SNMP, Wireless Distribution System (WDS), MAC Filtering, Access Control, Virtual Server, Dynamic DNS, Static/Dynamic Routing
ENVIRONMENTAL SPECIFICATIONS	
Dimension	195x160x27mm (LxWxH)
Weight	700g
Temperature	-10°C to 50°C(Operating)
Humidity	95% Non-condensing
Voltage Supply	5VDC

[APPENDICE 3]

[Datasheet degli AP NETGEAR Modello WG102^[21]]



Specifiche tecniche

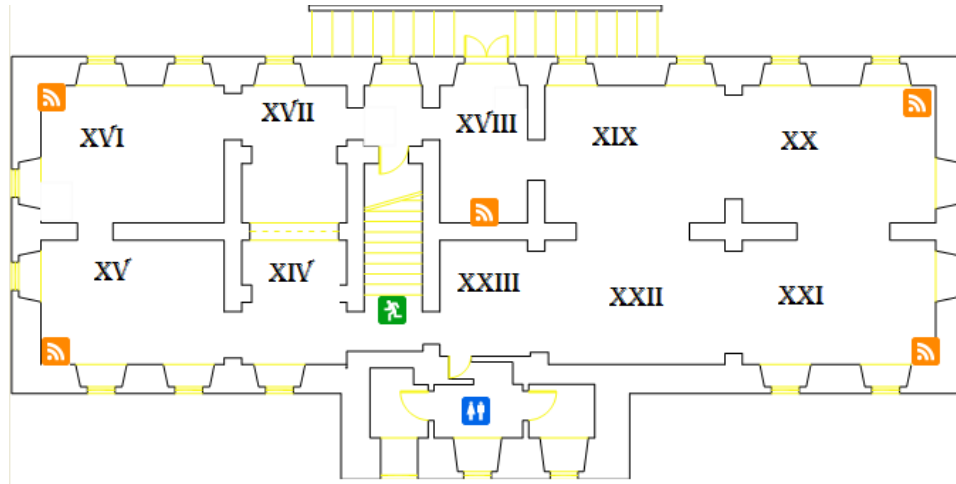
- **Wireless:**
 - Standard: IEEE 802.11g 54 Mbps
 - Antenne: una antenna 2,4 GHz 5 dBi staccabile
- **Interfacce fisiche:**
 - Porta LAN: 1 porta Ethernet 10/100BASE-T (RJ-45) con Auto Uplink™ (Auto MDI-X) e supporto Power over Ethernet (PoE) IEEE 802.3af
 - Alimentatore: Uscita 12VDC - 1,2 A; spina localizzata per i paesi di vendita
 - Quattro LED: Alimentazione, Test, 100 Mbps, Link/ACT e WLAN 802.11g
 - Connettore Antenna: 1 connettore SMA inverso
- **Network Management:**
 - Configurazione e gestione remota attraverso Web browser o software basato su SNMP
 - La gestione SNMP supporta SNMP MIB I, MIB II e MIB 802.11
- **Funzioni avanzate wireless:**
 - Modalità bridge e ripetitore
 - Potenza trasmissione regolabile da 100 mW a 0 mW
- **Potenza d'uscita massima tipica:**
 - modalità 802.11b, da 1 a 11 Mbps +19 dBm
 - modalità 802.11g, da 6 a 24 Mbps +18 dBm
 - modalità 802.11g, 36, 48 e 54 Mbps +17/16/16 dBm
- **Sensibilità in ricezione:**

- 802.11b	
2 Mbps	-93 dBm
5,5 Mbps	-91 dBm
11 Mbps	-89 dBm
- 802.11g	
6 Mbps	-91 dBm
9 Mbps	-90 dBm
12 Mbps	-89 dBm
18 Mbps	-87 dBm
24 Mbps	-84 dBm
36 Mbps	-81 dBm
48 Mbps	-77 dBm
54 Mbps	-75 dBm
108 Mbps	-73 dBm
- **Power Over Ethernet Richiesta:**
 - 4.3 W
- **Sicurezza:**
 - Wi-Fi Protected Access (WPA), predisposto per 802.11i
 - Blocco della diffusione di SSID
 - Supporto VPN pass-through
 - Filtraggio degli indirizzi MAC con access control list - fino a 256 utenti
 - Supporto 802.1x RADIUS con EAP-TLS, TTLS e PEAP
- **Conformità agli standard e alle norme di sicurezza ed emissione elettromagnetica:**
 - IEEE 802.11g (Banda di frequenza 2,4 GHz; modulazione DSSS)
 - IEEE 802.3af - Power over Ethernet (PoE)
 - FCC Part 15 Subpart B e Subpart C
 - CE EN 60950-1 (2001), EN 301 489-17 V1.2.1 (2002), EN 301 489-1 V1.4.1 (2002), EN 300 328-1 (2001-12), EN 300 328-2 (2001-12), Direttiva R&TTE 99/5/EC, CISPR22(B), AS/NZS 3548 (B)
 - Plenum Rated
- **Specifiche fisiche:**
 - Dimensioni: 14,1 x 10 x 2,7 cm (5,56" x 3,94" x 1,06")
 - Peso: 386 g (0,85 libbre)
- **Specifiche ambientali:**
 - Temperatura di esercizio: da 0° a 40° C (da 32 a 104°F)
 - Umidità di esercizio: 90% di umidità relativa senza condensa

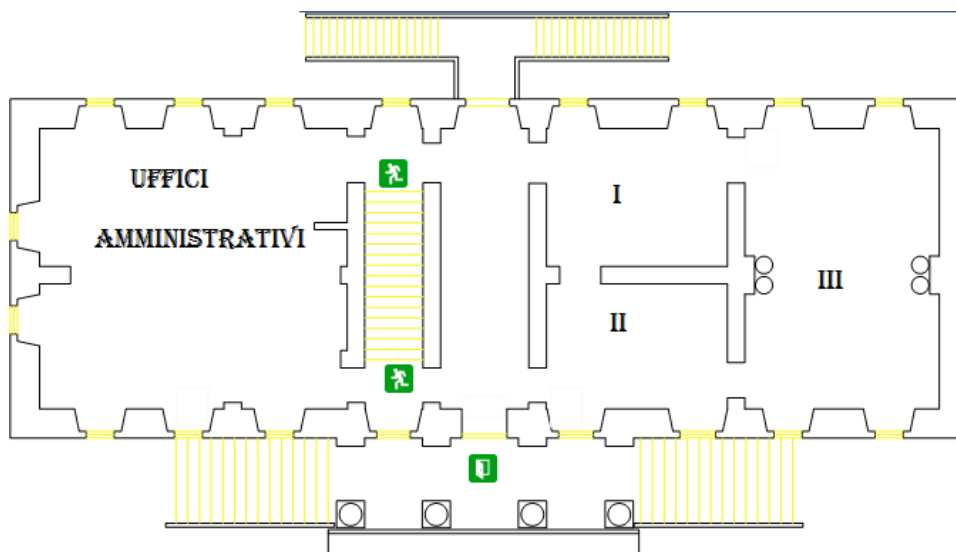
[APPENDICE 4]

[Mappe del Museo Civico di Albano]

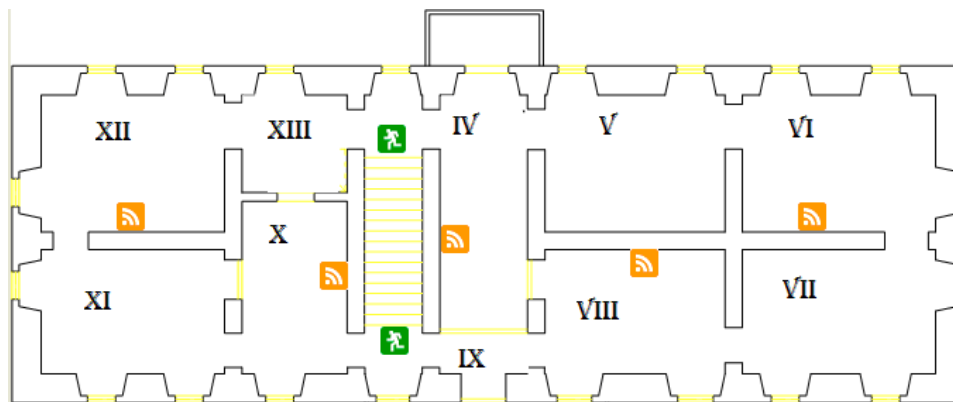
Piano Seminterrato



Piano Rialzato



Primo Piano



Leggenda

-  AP
- I,II,...** SALE
-  USCITA
-  TOILETTE
-  SCALE

[APPENDICE 5]

[Caratteristiche di altri Sistemi di localizzazione Wi-Fi applicati in ambienti indoor^[1]]

Name	Basic method	Mode			Type of signal	Measurement	Type of network
		ta	tb	nb			
Indoor positioning							
RADAR	Fingerprint.			×	Radio	RSS	WLAN
Ekahau	Fingerprint.	×			Radio	RSS	WLAN
Indoor GPS	Circ. lat.		×		Radio	Time	
RFID	Prox. sens.	×	×	×	Radio	ID	
ActiveBadge	Prox. sens.			×	Infrared	ID	
WIPS	Prox. sens.	×			Infrared	ID	WLAN
ActiveBat	Circ. lat.			×	Ultrasound	Time	418 MHz radio
Cricket	Prox. sens.		×		Ultrasound	ID + time	

Leggenda:

- Con *Mode* si intende chi gestisce il posizionamento (le misure e i calcoli); nel caso *ta* (*terminal assisted*) il terminale è assistito dalla rete, nel caso *tb* (*terminal based*) il terminale agisce da solo, e nel caso *nb* (*network based*) è la rete chi gestisce tutto.
- Con *Matching* si intende il tipo di calcolo per determinare la posizione, nel caso *det* si parla di un calcolo deterministico, invece nel caso *prob* si parla di una procedura statistica.
- Con *Measurement* si intende il tipo di parametro misurato: RSS → RSSI, Time → Tempo e ID → identificativo.
- Con *Basic Method* si intende la tecnica di posizionamento impiegata: Fingerprinting → RSSI, Circ. Lat. → Lateration Circolare, Prox. Sens. → Sensori di Prossimità.

[APPENDICE 6]

[Confronto tra diversi Sistemi di localizzazione Wi-Fi applicati in ambienti indoor^[1]]

System	Observable	Accuracy	Mode			Radio Map		Matching	
			ta	tb	nb	emp.	mod.	det.	prob.
RADAR	RSS	2.1 m/50%			×	×		×	
Ekahau	RSS	3.1–4.6 m/90%	×			×		×	
Horus	RSS	2.1 m/90%		×		×		×	
Nibble	SNR	10 m/80%	×			×		×	
WhereMops	RSS	1.5 m/50% 6.0 m/95%		×			×	×	

Leggenda:

- Con *Mode* si intende chi gestisce il posizionamento (le misure e i calcoli); nel caso *ta* (*terminal assisted*) il terminale è assistito dalla rete, nel caso *tb* (*terminal based*) il terminale agisce da solo, e nel caso *nb* (*network based*) è la rete chi gestisce tutto.
- Con *Radio Map* si intende com'è costruita la mappa; nel caso *emp* la mappa è costruita in modo empirico.
- Con *Matching* si intende il tipo di calcolo per determinare la posizione, nel caso *det* si parla di un calcolo deterministico, invece nel caso *prob* si parla di una procedura statistica.

I sistemi RADAR, Horus, WhereMops e Nibble sono sistemi ancora in fase di sperimentazione, quindi l'accuratezza riportata non corrisponde alle prestazioni in un ambiente realistico, invece Ekahau è un sistema già commercializzato. I dati sono aggiornati al 2005.

[APPENDICE 7]

[Codice in linguaggio C# che gestisce la rete neurale]

```
using System;
using System.Data;
using System.Configuration;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Web.UI.HtmlControls;
using System.Xml;
using System.IO;
using System.Collections;
using System.Collections.Generic;

/// <summary>
/// Descrizione di riepilogo per Positioning
/// </summary>

public class Positioning
{
    internal class SetAccessPoint : IComparable<SetAccessPoint>
    {
        public int power = 0;
        public string MAC = null;

        #region IComparable<SetPoint>

        public int CompareTo(SetAccessPoint other)
        {
            return power.CompareTo(other.power);
        }

        #endregion
    }

    internal class AccessPointStat
    {
        public string MAC = null;
        public int NumberScans = 0;
    }
}
```

```

    public double DIST = 0;
    public List<int> Power = new List<int>();
}

internal class SetArea : IComparable<SetArea>
{
    public string Zone = null;
    public double Frequence = 0;

    #region IComparable<SetArea>

    public int CompareTo(SetArea other)
    {
        return Frequence.CompareTo(other.Frequence);
    }

    #endregion
}

internal class SetCentroidPoint : IComparable<SetCentroidPoint>
{
    public int[] Coord = new int[2];
    public string Zone = null;
    public double Prob = 0;
    public int IdMap = 0;

    #region IComparable<SetCentroidPoint>

    public int CompareTo(SetCentroidPoint other)
    {
        return Prob.CompareTo(other.Prob);
    }

    #endregion
}

internal class SetPoint : IComparable<SetPoint>
{
    public double Difference = 0;
    public int pointIndex = 0;
    public string Zone = null;

    #region IComparable<SetPoint>

```

```

public int CompareTo(SetPoint other)
{
    return Difference.CompareTo(other.Difference);
}

#endregion
}

internal class ScanPoint
{
    public int TotalScans = 0;
    public string Name = null;
    public int IdMap = 0;
    public int[] Coord = new int[2];
    public List<AccessPointStat> AccessPoint = new List<AccessPointStat>();
}

private List<ScanPoint> _scanPoints = new List<ScanPoint>();

public Positioning()
{
}

public string[] MacSort()
{
    dimmiPos.Service posService = new dimmiPos.Service();
    string xmlString = null;

    try
    {
        xmlString = posService.GetCampionamenti();
        ParseXmlDocument(xmlString);
    }
    catch (Exception)
    {
        return null;
    }

    List<string> listAP = new List<string>();

    for (int i = 0; i < _scanPoints.Count; i++)
    {
        for (int j = 0; j < _scanPoints[i].AccessPoint.Count; j++)

```

```

        {
            if (listAP.Contains(_scanPoints[i].AccessPoint[j].MAC) == false)
                listAP.Add(_scanPoints[i].AccessPoint[j].MAC);
        }
    }

listAP.Sort();

string[] macSort = new string[listAP.Count];

for (int i = 0; i < macSort.Length; i++)
    macSort[i] = listAP[i];

return macSort;
}

public string Input(int[] power, string[] mac, string[] Mac)
{
    if (power == null || Mac == null)
        return null;

    int[] Power = new int[Mac.Length];

    for (int m = 0; m < Mac.Length; m++)
    {
        for (int n = 0; n < mac.Length; n++)
        {
            if (Mac[m] == mac[n])
                Power[m] = power[n];
        }

        if (Power[m] == 0)
            Power[m] = -99;
    }

    string[] stringArray = Array.ConvertAll<int, string>(Power, new Converter<int,
string>(ConvertIntToString));

    string valuePower = string.Join(" ", stringArray);

    return valuePower;
}

```

```

public int[] Where(int[] power, string[] mac)
{
    int[] retValue = null;

    if (_scanPoints.Count > 0 )
        retValue = InputLayer(power, mac);
    else
        retValue = null;

    return retValue;
}

private string ConvertIntToString(int intParameter)
{
    return intParameter.ToString();
}

private void ParseXmlDocument(string xmlData)
{
    int currentMap = 0;
    using (StringReader xmlDataStringReader = new StringReader(xmlData))
    using (XmlTextReader xmlReader = new
XmlTextReader(xmlDataStringReader))
        while (xmlReader.Read())
        {
            if (xmlReader.Name == "Map" && xmlReader.IsStartElement())
            {
                xmlReader.MoveToAttribute("Id");
                currentMap = Convert.ToInt32(xmlReader.Value);
            }

            if (xmlReader.Name == "Point" && xmlReader.IsStartElement())
            {
                ScanPoint newPoint = new ScanPoint();
                xmlReader.MoveToAttribute("Name");
                newPoint.Name = xmlReader.Value;
                newPoint.IdMap = currentMap;
                xmlReader.MoveToAttribute("X");
                newPoint.Coord[0] = Convert.ToInt32(xmlReader.Value);
                xmlReader.MoveToAttribute("Y");
                newPoint.Coord[1] = Convert.ToInt32(xmlReader.Value);
                _scanPoints.Add(newPoint);
            }

            if (xmlReader.Name == "AccessPoint" && xmlReader.IsStartElement())

```



```

        {
            xmlReader.MoveToAttribute("MAC");

            int statIndex = -1;

            for (int j = 0; j < _scanPoints[_scanPoints.Count - 1].AccessPoint.Count;
j++)
                if (_scanPoints[_scanPoints.Count - 1].AccessPoint[j].MAC ==
xmlReader.Value)
                    {
                        _scanPoints[_scanPoints.Count -
1].AccessPoint[j].NumberScans++;
                        statIndex = j;
                        break;
                    }

            if (statIndex < 0)
                {
                    AccessPointStat newAccessPoint = new AccessPointStat();
                    newAccessPoint.MAC = xmlReader.Value;

                    newAccessPoint.NumberScans++;
                    xmlReader.MoveToAttribute("DIST");
                    double tmpDouble = 0;
                    Double.TryParse(xmlReader.Value, out tmpDouble);
                    newAccessPoint.DIST = (int)tmpDouble;
                    _scanPoints[_scanPoints.Count -
1].AccessPoint.Add(newAccessPoint);
                    statIndex = _scanPoints[_scanPoints.Count - 1].AccessPoint.Count -
1;
                }

            if (statIndex >= 0)
                while (xmlReader.Read())
                    if (xmlReader.Name == "Power" && xmlReader.IsStartElement())
                        if (xmlReader.Read())
                            {
                                _scanPoints[_scanPoints.Count -
1].AccessPoint[statIndex].Power.Add(Convert.ToInt32(xmlReader.Value));
                                break;
                            }

                    _scanPoints[_scanPoints.Count - 1].TotalScans++;
                }
        }
    }
}

```

```

}

public int[] InputLayer(int[] power, string[] MAC)
{
    List<string> outputHiddenLayer = new List<string>();

    int threshold = 5;

    int[] pointIndex = new int[threshold];

    outputHiddenLayer = HiddenLayer(power, MAC, threshold);

    return OutputLayer(power, MAC, outputHiddenLayer);
}

public List<string> HiddenLayer(int[] power, string[] MAC, int threshold)
{
    List<SetPoint> listPoint = new List<SetPoint>();
    List<string> listOutput = new List<string>();

    for (int i = 0; i < _scanPoints.Count; i++) //per ogni punto campionato
    {
        double difference_points = 0;

        for (int p = 0; p < MAC.Length; p++) //per ogni mac passato in input
        {
            int accessPointIndex = -1;

            for (int j = 0; j < _scanPoints[i].AccessPoint.Count; j++) //cerca l'access
point corrispondente nel punto
                if (_scanPoints[i].AccessPoint[j].MAC == MAC[p]) //se lo trova effettua
il calcolo ed esce dal ciclo
                    {
                        accessPointIndex = j;
                        break;
                    }

            if (accessPointIndex >= 0)
            {

                int[] vectorFP = new
int[_scanPoints[i].AccessPoint[accessPointIndex].NumberScans];
                int vectorpoint = 0;

                vectorpoint = power[p];

```

```

        vectorFP =
_scanPoints[i].AccessPoint[accessPointIndex].Power.ToArray();

        double FPmean = AverageVector(vectorFP);

        difference_points += Math.Pow((FPmean - vectorpoint), 2);

    }

}

difference_points = Math.Sqrt(difference_points);

SetPoint set = new SetPoint();
set.Difference = difference_points;
set.pointIndex = i;
set.Zone = _scanPoints[i].Name;
listPoint.Add(set);

}

listPoint.Sort();

List<SetArea> listArea = new List<SetArea>();
int d = 0;

for (int q = 0; q < threshold; q++)
{

    SetArea setArea = new SetArea();

    setArea.Zone = listPoint[q].Zone;

    int count = 1;

    for (int r = q + 1; r < threshold; r++)
    {
        if (listPoint[r].Zone == listPoint[q].Zone)
        {
            listPoint.RemoveRange(r, 1);
            count++;
            threshold--;
            r--;
        }
    }
}

```

```

listPoint.RemoveRange(q, 1);
threshold--;

setArea.Frequency = count / ((d + 1.0) * 5);
d++;
q--;
listArea.Add(setArea);

}

listArea.Sort();

for (int s = 0; s < listArea.Count; s++)
{
    //if (listArea[s].Frequency >= 1 / 5.0)
    listOutput.Add(listArea[s].Zone + "_" + listArea[s].Frequency);
}

return listOutput;

}

public int[] OutputLayer(int[] power, string[] MAC, List<string> listOutput)
{
    List<SetCentroidPoint> PointCen = new List<SetCentroidPoint>();
    List<string> nameCluster = new List<string>();
    List<SetArea> listCluster = new List<SetArea>();

    for (int s = 0; s < listOutput.Count; s++)
    {
        SetArea cluster = new SetArea();
        cluster.Zone = listOutput[s].Split('_')[0];
        nameCluster.Add(listOutput[s].Split('_')[0]);
        cluster.Frequency = System.Convert.ToDouble(listOutput[s].Split('_')[1]);
        listCluster.Add(cluster);
    }

    for (int i = 0; i < _scanPoints.Count; i++) //per ogni punto campionato
    {

        if (nameCluster.Contains(_scanPoints[i].Name))
        {

```

```

double Prob_cum = 1;

for (int c = 0; c < nameCluster.Count; c++)
{
    if (listCluster[c].Zone == nameCluster[c])
    {
        Prob_cum = listCluster[c].Frequence;
    }
}

for (int p = 0; p < MAC.Length; p++) //per ogni mac passato in input
{
    int accessPointIndex = -1;

    for (int j = 0; j < _scanPoints[i].AccessPoint.Count; j++) //cerca l'access
point corrispondente nel punto
        if (_scanPoints[i].AccessPoint[j].MAC == MAC[p]) //se lo trova
effettua il calcolo ed esce dal ciclo
            {
                accessPointIndex = j;
                break;
            }

    if (accessPointIndex >= 0)
    {
        int[] vectorFP = new
int[_scanPoints[i].AccessPoint[accessPointIndex].NumberScans];
        int vectorpoint = 0;

        vectorpoint = power[p];
        vectorFP =
_scanPoints[i].AccessPoint[accessPointIndex].Power.ToArray();

        double FPmean = AverageVector(vectorFP);
        double FPdev = Deviation(vectorFP, FPmean);

        Prob_cum = (1 / (Math.Sqrt(2 * Math.PI * FPdev))) * Math.Exp(-
(vectorpoint - FPmean) * (vectorpoint - FPmean) / (2 * FPdev * FPdev)) * Prob_cum;

    }
}

SetCentroidPoint CentroidPoint = new SetCentroidPoint();
CentroidPoint.Coord = _scanPoints[i].Coord;

```

```

        CentroidPoint.Prob = Prob_cum;
        CentroidPoint.Zone = _scanPoints[i].Name;
        CentroidPoint.IdMap = _scanPoints[i].IdMap;
        PointCen.Add(CentroidPoint);
    }

}

PointCen.Sort();
PointCen.Reverse();

double cumP1 = 0;
double cumP2 = 0;
double cumS = 0;

int sensib = (int)(0.8 * PointCen.Count);

for (int k = 0; k < sensib; k++)
{
    int[] Pt = (int[])PointCen[k].Coord;

    cumP1 = Pt[0] * PointCen[k].Prob + cumP1;
    cumP2 = Pt[1] * PointCen[k].Prob + cumP2;
    cumS = PointCen[k].Prob + cumS;
}

int[] Point = new int[3];

if (cumS != 0)
{
    Point[0] = PointCen[0].IdMap;
    Point[1] = System.Convert.ToInt32(cumP1 / cumS);
    Point[2] = System.Convert.ToInt32(cumP2 / cumS);

    return Point;
}
else
    return null;
}

public double AverageVector(int[] vector)
{

```

```
double average = 0;

for (int i = 0; i < vector.Length; i++)
    average += vector[i];

return average / vector.Length;
}

public double Deviation(int[] vector, double mean)
{
    double dev = 0.0;

    for (int i = 0; i < vector.Length; i++)
    {
        dev = (vector[i] - mean) * (vector[i] - mean) + dev;
    }

    return Math.Sqrt(dev / vector.Length);
}
}
```