



 POLITECNICO DI MILANO



Bluetooth Ranging su Android

Diego Tuzi – Marco Visin – Laerte Saliai

Elaborato SCNI 2011-2012

Prof. Monica Nicoli - Dip. di Elettronica e Informazione

Il progetto **Bluetooth Ranging** si propone di ottenere delle informazioni sul posizionamento di uno smartphone utilizzando come unica fonte di informazioni la potenza del segnale ricevuto da uno o più dispositivi.

I fattori abilitanti di questo progetto sono lo standard di trasmissione **Bluetooth** e il sistema operativo per dispositivi mobili **Android**.

Nella prima parte di questo documento si fornisce una introduzione degli elementi chiave del progetto con particolare attenzione alle funzioni coinvolte.

In seguito si presenta l'applicazione sviluppata e una analisi sulle performance mettendo in evidenza i limiti dei metodi implementati.





- **Parte 1**
 - Bluetooth
 - Android
 - Protocollo «BluetoothRanging»
- **Parte 2**
 - Formulazione del problema di localizzazione
 - Distanza real-time
 - Lateralization 2D
- **Parte 3**
 - Bluetooth Ranging (App per Android)
- **Parte 4**
 - Analisi delle performance
 - Sviluppi futuri
 - Considerazioni finali
- **Parte 5**
 - Materiale, Riferimenti e Riconoscimenti



Bluetooth, Android e “Protocollo BluetoothRanging”



Bluetooth è una tecnologia wireless utilizzata da dispositivi fissi e mobili per lo scambio di dati su brevi distanze, mediante la creazione di reti personali con un alto livello di sicurezza.

Caratteristiche:

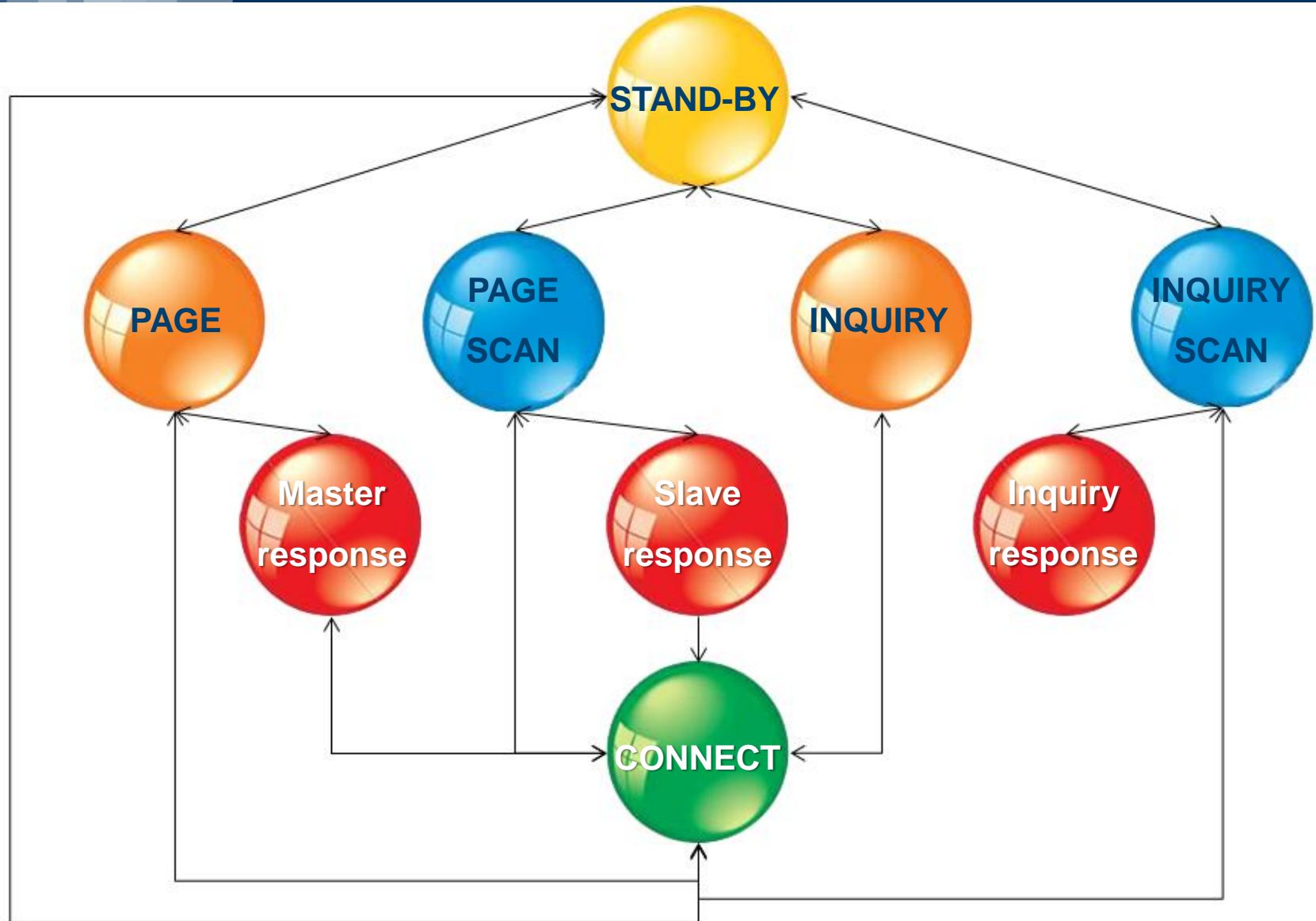
- Tecnologia radio
- Basso costo
- Raggio ridotto (10-20 m)
- Bassa complessità
- Piccole dimensioni
- Trasmissioni radio nella banda ISM 2.4 GHz
- Nato da un consorzio di costruttori
- Solo i primi due livelli sono standardizzati da IEEE 802.15.1



Nell'ambito del progetto gli aspetti di maggiore rilevanza sono le procedure di **Inquiry** e l'**RSSI**.



Bluetooth: diagramma a stati





Bluetooth: procedure di Inquiry

I principali stati in cui si può trovare un dispositivo Bluetooth sono quello di STANDBY e CONNECTION. In aggiunta ci sono altri sottostati tra cui inquiry, inquiry scan e inquiry response. Il dispositivo che inizia la procedura di inquiry si definisce *master* mentre il dispositivo che esegue la procedura di inquiry scan e inquiry response viene definito *slave*.

Inquiry	La procedura di inquiry si usa per trovare l'identità dei dispositivi Bluetooth nel range di utilizzo. Il dispositivo in ricerca (master) raccoglie gli indirizzi e i clocks delle unità che rispondono con i messaggi di inquiry.
Inquiry scan	In questo stato, i dispositivi Bluetooth sono in ascolto per le inquiry di altri dispositivi.
Inquiry response	Gli slave rispondono con un particolare pacchetto (FHS packet) che contiene informazioni sullo slave tra cui l'RSSI.

Le procedure di inquiry permettono al dispositivo master di creare la lista dei dispositivi presenti nel range di utilizzo.



Bluetooth: Received Signal Strength Indicator (RSSI)

8

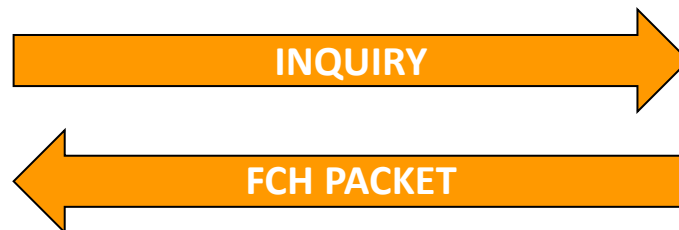
L'RSSI indica la forza del segnale ricevuto in dBm. L'RSSI può essere calcolato dal dispositivo master usando l'evento «*Inquiry Result with RSSI*» che può essere avviato durante il processo di Inquiry. Il parametro RSSI si misura durante la ricezione del pacchetto FHS trasmesso dallo slave. L'evento può essere chiamato solo durante la procedura di inquiry.

Caratteristiche:

- Intervallo valori da -127 a +20 dBm
- Evento per la rilevazione dell'RSSI non supportato da tutti i dispositivi
- L'accuratezza dovrebbe essere +/- 6 dB
- L'accuratezza dipende dall'hardware bluetooth del dispositivo



MASTER



SLAVE



Android è una piattaforma software e un sistema operativo per dispositivi mobili basato su kernel linux. E' stato sviluppato inizialmente da Google e successivamente dalla Open Handset Alliance (consorzio di molte compagnie).

Caratteristiche:

- Licenza Apache Software Licence versione 2
- Alto livello di personalizzazione
- SDK e strumenti di sviluppo
- Kernel Linux per i driver delle periferiche, la gestione della memoria, gestione dei processi e networking
- Linguaggio di programmazione Java (supportato ufficialmente) e C/C++
- Supporta i più diffusi standard di comunicazione wireless (Bluetooth, WiFi, GSM, EDGE, 3G, LTE)



Nell'ambito del progetto gli aspetti di maggiore rilevanza sono quelli relativi alla programmazione, in particolare le **API Bluetooth** per comunicare con l'hardware bluetooth.



Android: API Bluetooth

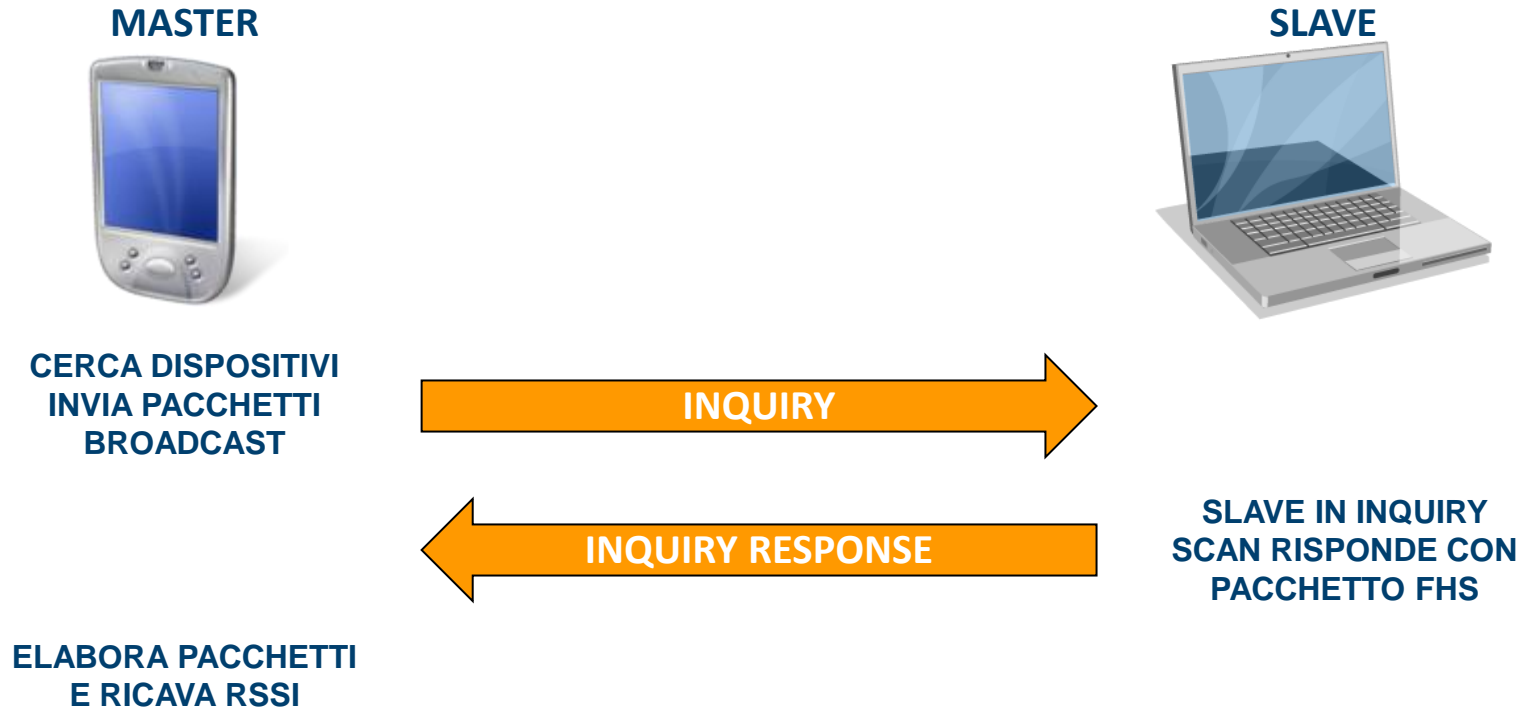
La piattaforma android include il supporto per lo stack di rete Bluetooth, il quale permette ai dispositivi di scambiare dati in modalità wireless. Il framework della applicazione fornisce l'accesso alle funzionalità del Bluetooth tramite le API Bluetooth di Android.

Tutte le API Bluetooth sono disponibili nel package **android.bluetooth**. Una delle principali classi utilizzate all'interno del codice è **BluetoothAdapter**, che rappresenta il punto di ingresso per tutte le interazioni con il Bluetooth. Per avviare la procedura di inquiry si usa il metodo **startDiscovery()** della classe BluetoothAdapter.

Una ulteriore classe di interesse è **BluetoothDevice** che rappresenta un dispositivo bluetooth remoto. Tale classe permette di ottenere informazioni sul dispositivo che rappresenta.

Quando la procedura di inquiry termina i dispositivi rilevati sono contenuti all'interno dell'oggetto che istanzia la classe BluetoothAdapter. Tali dispositivi possono essere associati ad un oggetto della classe BluetoothDevice da cui è possibile ottenere le informazioni sul dispositivo.

Se il dispositivo master supporta l'evento Inquiry Result with RSSI allora il valore di RSSI rilevato è all'interno di una costante denominata **EXTRA_RSSI** della classe BluetoothDevice.



La procedura descritta rappresenta il blocco base per le funzionalità implementate nel progetto.



Cenni teorici sulla radio-localizzazione



Il più semplice problema di localizzazione può essere descritto da:

- Un terminale master in posizione incognita u ;
- N stazioni di riferimento (slaves) in posizione nota s_i ;
- Canale radio descritto dalla funzione h_i ;
- Un termine stocastico che tiene conto degli errori di misura e del rumore presente sull' i -esimo canale radio n_i .

Nelle ipotesi precedenti, l' i -esimo parametro misurato dal terminale master è,

$$y_i = h_i(u, s_i) + n_i \quad i = 1, \dots, N.$$

L'obiettivo della radio-localizzazione è stimare la posizione u del target con l'informazione misurata y_i .

Il parametro misurato può essere di diversa natura, per lo sviluppo del progetto si è scelto di utilizzare come parametro il valore di potenza del segnale ricevuto dalle stazioni di riferimento.



Teoria: ranging tra due dispositivi

Il valore medio di RSSI che viene rilevato dalla interfaccia radio Bluetooth può essere utilizzato per stimare la distanza tra due dispositivi.

In una propagazione ideale in spazio libero, senza ostruzioni tra ricevitore e trasmettitore, la potenza media ricevuta a distanza d dal trasmettitore rispetta l'equazione di seguito,

$$\bar{P}_r = \bar{P}_{ref} \left(\frac{d_{ref}}{d} \right)^2 .$$

Ove \bar{P}_{ref} è la potenza media ad una distanza di riferimento d_{ref} . Si osserva che in tale caso la potenza ricevuta decresce con il quadrato della distanza.

Nella propagazione reale la potenza decade proporzionalmente con d^{n_p} (modello di Okomura-Hata), dove n_p è l'esponente di **path-loss** (tipicamente 2-4),

$$\bar{P}_r = \bar{P}_{ref} \left(\frac{d_{ref}}{d} \right)^{n_p} .$$

Inoltre per una migliore modellazione del canale dovrebbe essere tenuti in conto lo **shadowing** e il **multipath**. Per il progetto le prove sono state fatte in ambiente aperto e con i terminali in visibilità in modo da poter trascurare tali effetti.



Teoria: lateration in 2D (1)

Un dispositivo master può ricavare la propria posizione conoscendo la posizione di più dispositivi slave e le misure di RSSI da tali dispositivi.

Per risolvere un tale problema di localizzazione si può adottare una tecnica geometrica che consiste in una laterazione circolare.

La tecniche geometriche:

- Forniscono un approccio intuitivo per la stima della posizione;
- Non possono gestire intersezioni multiple generate da un ambiente di misura rumoroso;
- In caso di intersezioni multiple non si hanno informazioni su quale sia il punto esatto;
- Al crescere del numero di stazioni di riferimento, il numero delle intersezioni può aumentare.

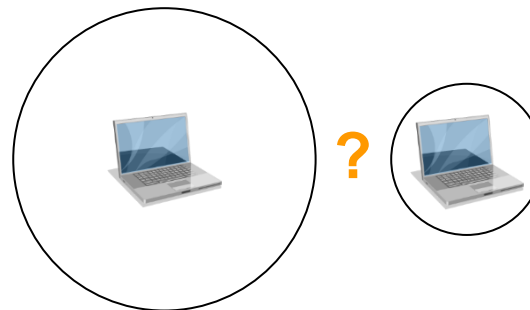
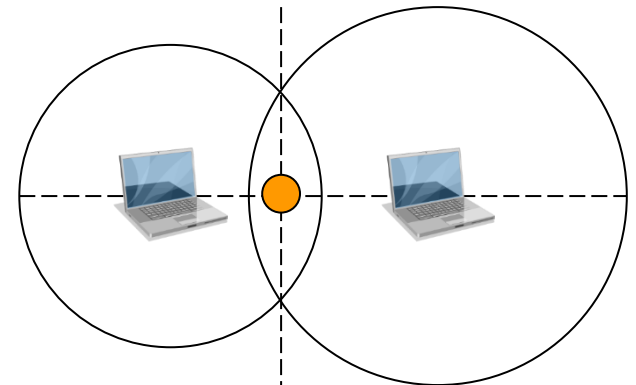
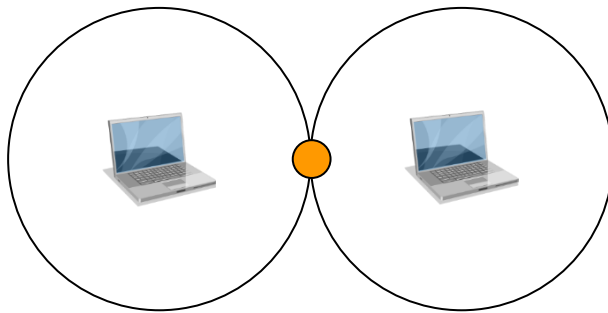


Teoria: lateration in 2D (2)

Il primo passo di una laterazione circolare consiste nel posizionare su ogni dispositivo una circonferenza di raggio pari alla distanza rilevata.

Si possono verificare tre situazioni, le due circonferenze:

- si intersecano in un solo punto (CASO 1);
- si intersecano in due punti (CASO 2);
- a causa di errori nella stima, non si intersecano (CASO 3).





Teoria: lateration in 2D (3)

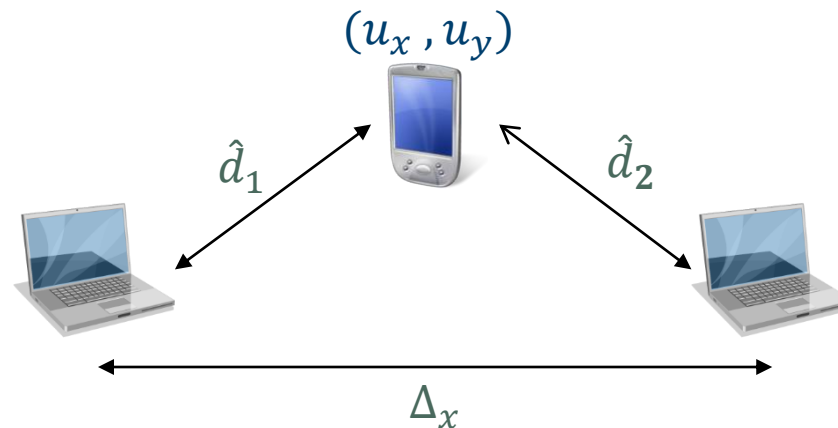
Per il CASO 1 e CASO 2, di slide precedente, la posizione del dispositivo è calcolata come segue

$$u_x = \frac{\hat{d}_1^2 - \hat{d}_2^2 + \Delta_x^2}{2\Delta_x}.$$

Per il CASO 3, ovvero quando $\hat{d}_1 + \hat{d}_2 < \Delta_x$, la posizione del dispositivo master viene stimata come segue

$$u_x = \frac{(\hat{d}_1 + c_1)^2 - (\hat{d}_2 + c_2)^2 + \Delta_x^2}{2\Delta_x},$$

con $c_1 = \left(1 - \frac{P_1}{P_1 + P_2}\right) \cdot (\Delta_x - \hat{d}_1 - \hat{d}_2)$, $c_2 = \left(1 - \frac{P_2}{P_1 + P_2}\right) \cdot (\Delta_x - \hat{d}_1 - \hat{d}_2)$ ove P_1 e P_2 rappresentano la potenza in dBm dei terminali slave 1 e slave 2, rilevata dal dispositivo master.





BluetoothRanging app per Android



L'applicazione per Android **Bluetooth Ranging** permette di effettuare le seguenti funzioni:

- **Distanza real-time:** calcola la distanza in real time tra il dispositivo master e un dispositivo slave.
- **Lateration 2D:** stima la posizione del dispositivo master conoscendo la distanza da due dispositivi slave.
- **Test da distanza nota:** funzionalità di sviluppo che permette di utilizzare il dispositivo come un sensore.

Tutte le funzionalità richiedono una calibrazione, che consiste nello stimare l'RSSI ad un distanza nota.





Bluetooth Ranging: distanza real-time

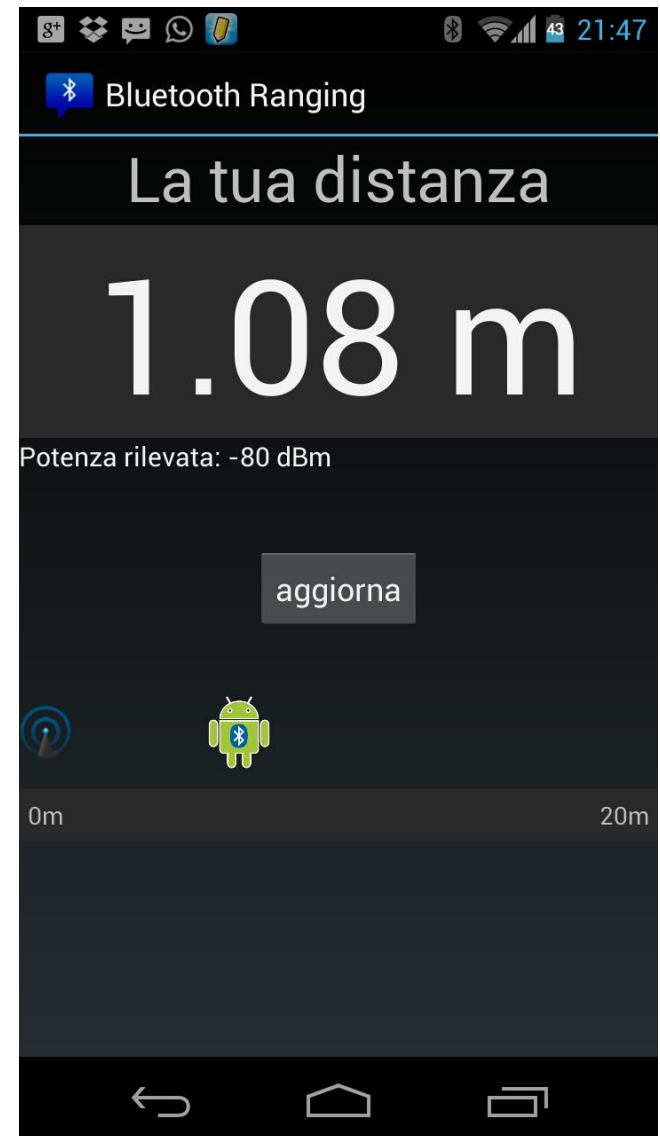
La funzione **distanza in real-time** permette di rilevare la distanza del dispositivo master (quello su cui gira la applicazione), da un dispositivo slave nel range di copertura del Bluetooth.

Tale procedura è subordinata ad una procedura di calibrazione che stima la RSSI ad una distanza nota.

Il valore di output è dato da una media di dieci misurazioni. L'interfaccia grafica permette di visualizzare la potenza rilevata di tutte le misure.

E' presente inoltre una animazione grafica che rappresenta la distanza del dispositivo master (robo di android) dal dispositivo slave.

La funzione implementa il modello di Okomura-Hata con $n_p = 2$, illustrato in precedenza.





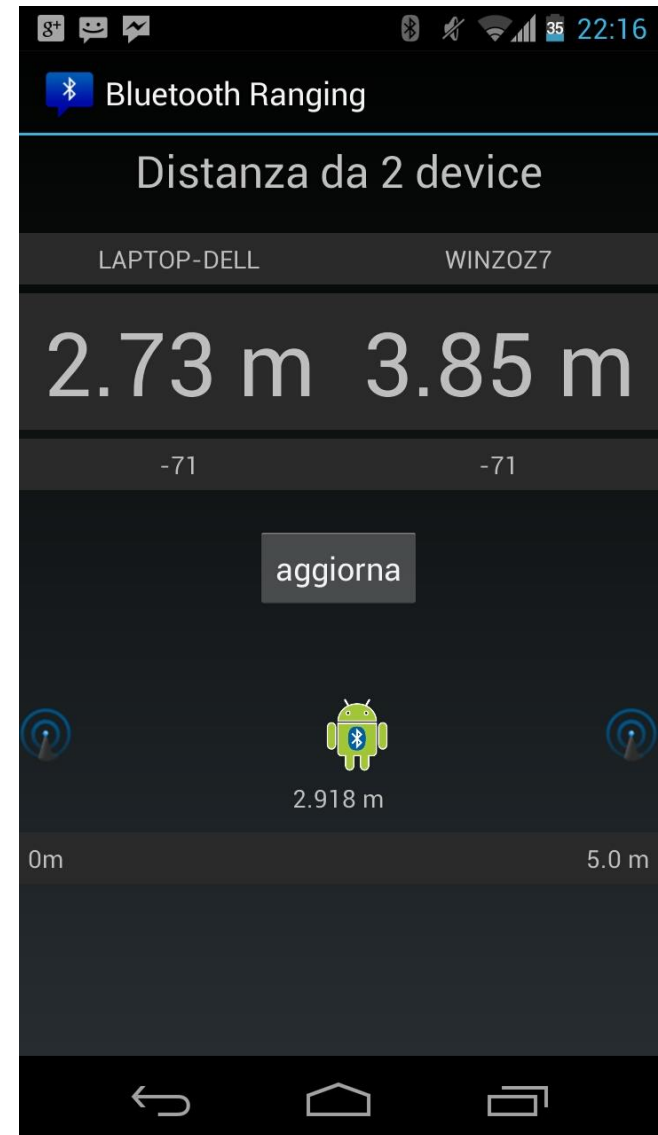
La funzione **Lateration 2D** permette di stimare la posizione del dispositivo master conoscendo la distanza tra due dispositivi slave e stimando la distanza da master a slave mediante misure di potenza.

La funzione è subordinata a due procedure di calibrazione una con ogni slave.

I valori di output sono valori medi utilizzando dieci misurazioni. L'interfaccia grafica permette di visualizzare i nomi dei dispositivi a cui si è associati e le relative misure di potenza istantanee.

Come per la funzione distanza in real-time vi è una animazione grafica che permette di visualizzare la posizione stimata del dispositivo master (robo di android).

La funzione implementa la tecnica geometrica descritta precedentemente.





La funzione **Test da distanza nota** è una funzione di sviluppo che permette di utilizzare lo smartphone come un sensore.

Ogni valore di RSSI misurato viene scritto all'interno di un file di testo.

L'interfaccia grafica è la stessa della funzione distanza real-time ma in aggiunta dispone di un contatore delle misure rilevate.

Come tutte le funzioni è subordinata alla procedura di calibrazione.

Tale funzionalità è stata utilizzata per raccogliere i dati con cui si sono analizzate le prestazioni dei metodi implementati.





E' possibile scaricare la applicazione direttamente dal market Android, al link:

<https://play.google.com/store/apps/details?id=net.kernelxp.BluetoothRanging&hl=it>

The screenshot shows the Google Play Store interface for the 'Bluetooth Ranging' app. The app is by 'Kernel-XP' and has a 4.5-star rating. The main banner features the Bluetooth logo and the Android robot. Below the banner, there's a green box stating compatibility with Samsung GT-I9003. The 'Descrizione' section explains it's an application for calculating distances using Bluetooth technology. The 'Screenshot applicazione' section shows four preview images of the app's interface, displaying distance measurements like '1.08 m' and '3.07 m'. On the right, there's a sidebar with 'INFORMAZIONI SULL'APPLICAZIONE' including version (1.0), update date (23 luglio 2012), and category (Istruzione). The bottom left shows other apps by the developer, 'Orario Polimi' and 'Android SMS', both with 4.5-star ratings.

Bluetooth Ranging
Kernel-XP

INSTALLA

Questa applicazione è compatibile con il tuo 3 Italia Samsung GT-I9003.

Altro dallo sviluppatore

Orario Polimi
KERNEL-XP
★★★★★ (20)
Gratis

Android SMS
KERNEL-XP
★★★★★ (20)
Gratis

Mostra altro >

Descrizione

Applicazione per calcolare le distanze utilizzando la tecnologia BlueTooth!

Visita il sito web dello sviluppatore > Invia un'email allo sviluppatore >

Screenshot applicazione

Informazioni sull'applicazione

VALUTAZIONE: ★★★★★

DATA AGGIORNAMENTO: 23 luglio 2012

VERSIONE ATTUALE: 1.0

VERSIONE ANDROID: 2.0.1 e superiori

CATEGORIA: Istruzione

DIMENSIONI: 306k

PREZZO: Gratis

CLASSIFICAZIONE CONTENUTI: Per tutti



Analisi delle Performance



Analisi delle performance: script matlab

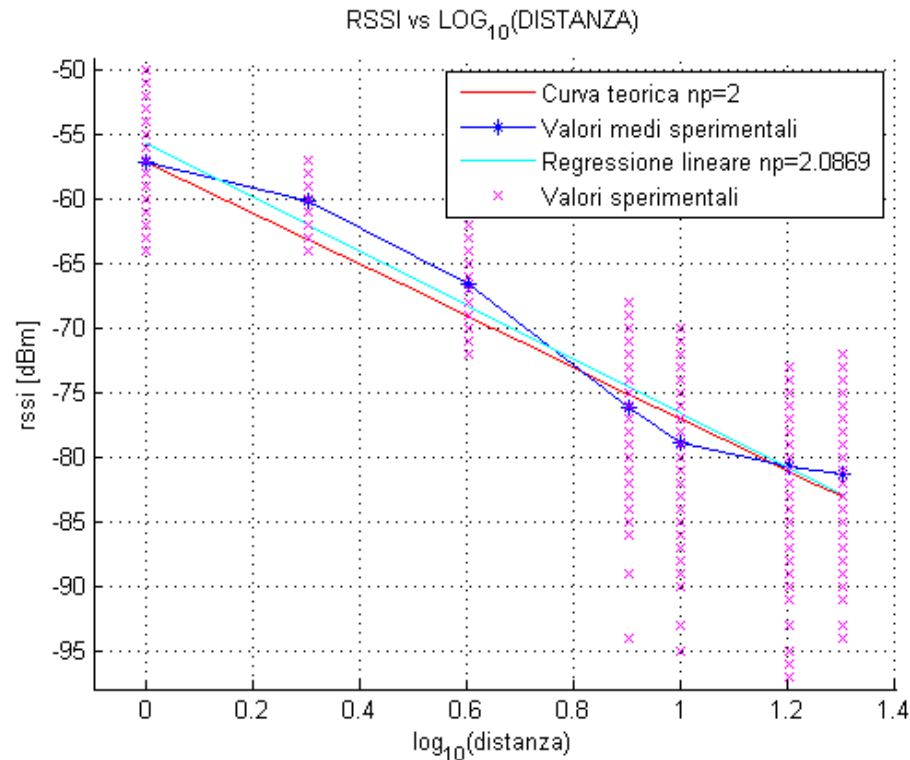
Per analizzare i dati raccolti dal terminale utente, mediante la funzione «test da distanza nota», il progetto comprende uno script matlab che accetta come input tali dati, li elabora e fornisce in output:

- Grafico RSSI vs distanza;
- Grafico distribuzioni di probabilità per ogni valore di distanza a cui è stato effettuato il test;
- Grafico varianza e deviazione standard in base alla distanza;
- Tabella con valori medi, varianza e deviazione standard alle varie distanze test.

I grafici delle slide successive sono stati generati con lo script, a cui sono stati dati in input dei test a distanza 1m, 2m, 4m, 8m, 10m, 16m e 20m.

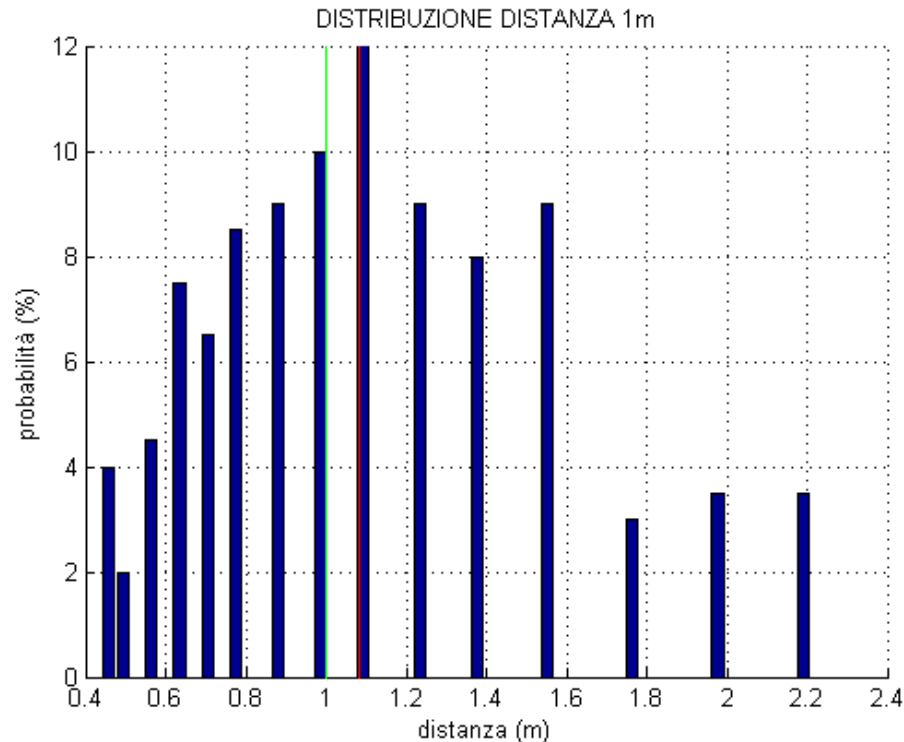
I test sono stati effettuati in ambiente aperto, con due terminali in visibilità e senza ostacoli sul cammino diretto in modo da poter ridurre l'effetto dei fenomeni di shadowing e multipath.

Per la curva «Regressione lineare» nella successiva slide, si veda il file `regressione_lineare.pdf`.

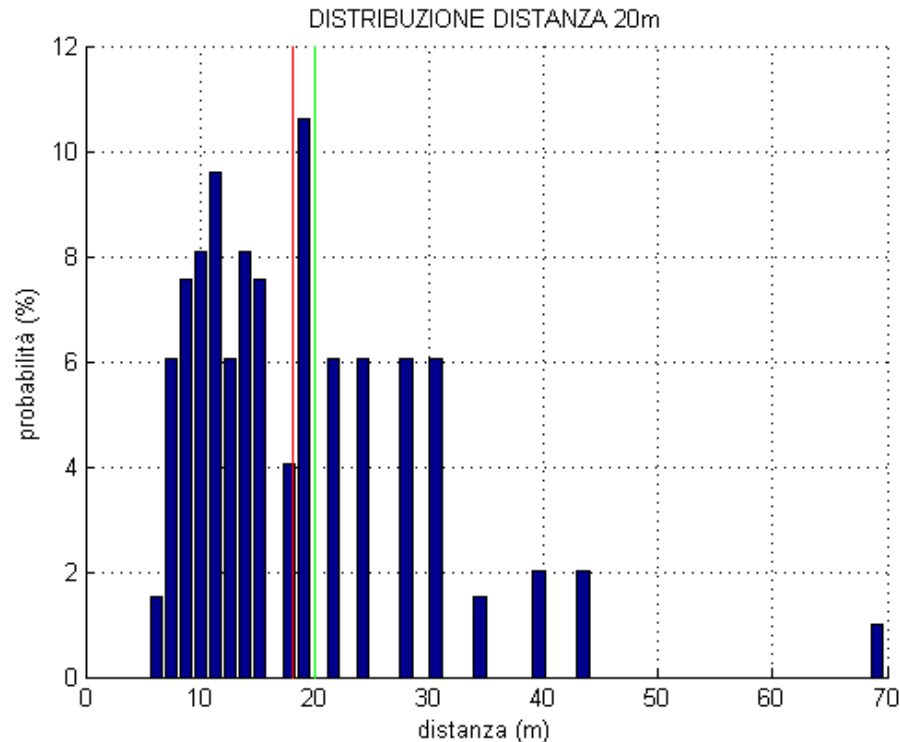


Il grafico mostra gli andamenti teorici e sperimentali delle curve di RSSI in funzione della distanza. La curva teorica è ottenuta con il modello del path-loss, ove come potenza di riferimento si è preso il valore di RSSI stimato ad 1m.

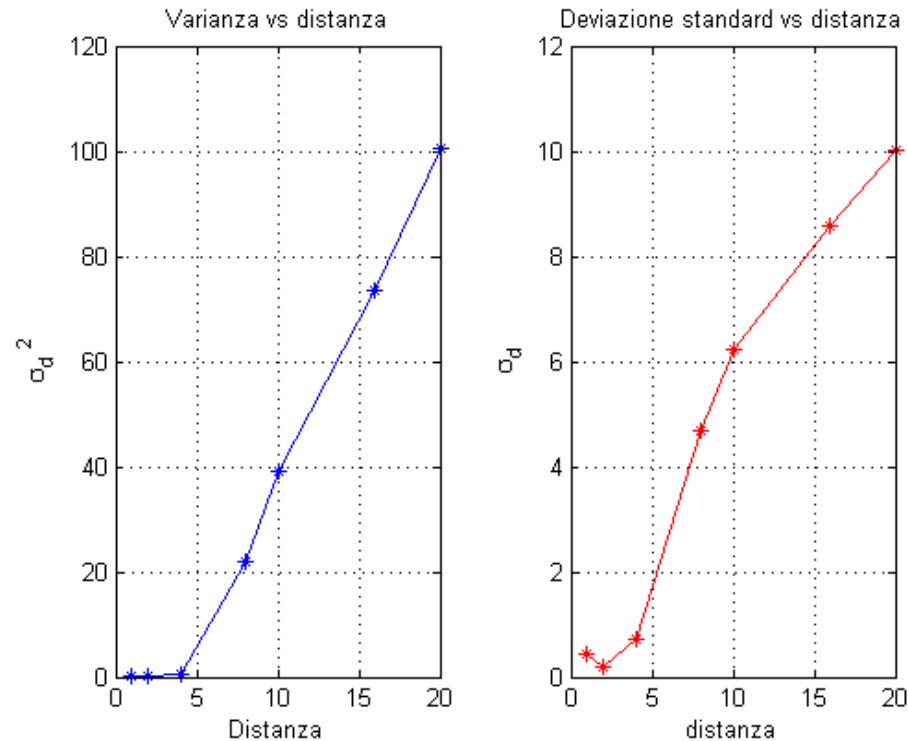
La curva sperimentale è stata effettuata facendo dei test in ambiente aperto ed ogni valore corrisponde alla media di duecento misurazioni.



Il grafico mostra l'andamento della distribuzione di probabilità in funzione della distanza. Le misure sono state effettuate ad 1m di distanza tra terminale master e terminale slave. La linea verticale verde indica la distanza vera, 1m, mentre la linea rossa indica il valore medio pari a 1.08m. La deviazione standard di tale distribuzione è pari a 0.43m.



Il grafico mostra l'andamento della distribuzione di probabilità in funzione della distanza. Le misure sono state effettuate ad 20m di distanza tra terminale master e terminale slave. La linea verticale verde indica la distanza vera, 20m, mentre la linea rossa indica il valore medio pari a 18.11m. La deviazione standard di tale distribuzione è pari a 10.02m.



Come visto dai grafici precedenti, la deviazione standard e quindi la varianza crescono con la distanza. Anche se, all'aumentare della distanza, la varianza della distribuzione cresce di molto, il valore medio tende a restare vicino al valore vero.



Analisi delle performance: limiti

- Qualità del ricevitore: il valore di RSSI viene rilevato con una accuratezza di $\pm 6dB$.

Esempio:

$$d_{ref} = 1m, P_{ref} = -57dBm$$

$$RSSI = -76 dBm \rightarrow RSSI \in [-82, -70]dBm$$

$$\hat{d} = 8,96m \rightarrow \hat{d} \in [4.5, 17,8]m$$

In questo esempio il valore rilevato ha un incertezza di 13m.

C'è da dire che utilizzando la procedura di calibrazione, con cui viene rilevata la potenza di riferimento, molto di questo gap viene ridotto ma l'imprecisione del dato rimane.

- Il modello utilizzato per stimare la distanza è il semplice path loss applicabile solo in caso di spazio libero. Non si tiene conto di shadowing e multipath. Tali fattori non possono essere trascurati nella maggior parte degli ambienti.
- La lentezza di acquisizione della singola misurazione rende molto difficile fornire la posizione del target in real-time.



Molti dei limiti illustrati nella slide precedente sono difficilmente superabili.

Per migliorare il funzionamento della applicazione si dovrebbe scendere al di sotto delle API Bluetooth fornite da Android, andare a lavorare direttamente con l'hardware bluetooth.

Cio' permetterebbe di sviluppare un protocollo ad-hoc, ad esempio:

- Forzare il dispositivo slave a inviare continuamente pacchetti FHS;
- Permettere al dispositivo master di rilevare il valore di RSSI anche dopo l'associazione tra dispositivo master e slave.

In entrambe le direzioni indicate, il master riuscirebbe ad ottenere molte più misure e quindi sarebbe in grado di fornire un dato medio molto più accurato e permetterebbe una stima della posizione in real-time molto più veritiera.



Considerazioni finali

La realizzazione del progetto ha comportato alcune difficoltà:

- per i componenti del gruppo di lavoro è stata una delle prime esperienze di programmazione in Android e acquisire confidenza con l'ambiente di lavoro non è stato immediato;
- Ottenere l'RSSI non si è rivelata una procedura semplice;
- Complessità del canale radio.

Per la realizzazione del progetto, si sono fatte notevoli esemplificazioni ma nonostante ciò si è riusciti ad ottenere delle prestazioni sperimentali che sono in linea con i risultati teorici.

Come descritto nella precedente slide il progetto lascia spazio per miglioramenti e l'aggiunta di ulteriori funzionalità. Si augura che BluetoothRanging possa essere un punto di partenza ulteriori progetti.



Materiale, riferimenti e riconoscimenti



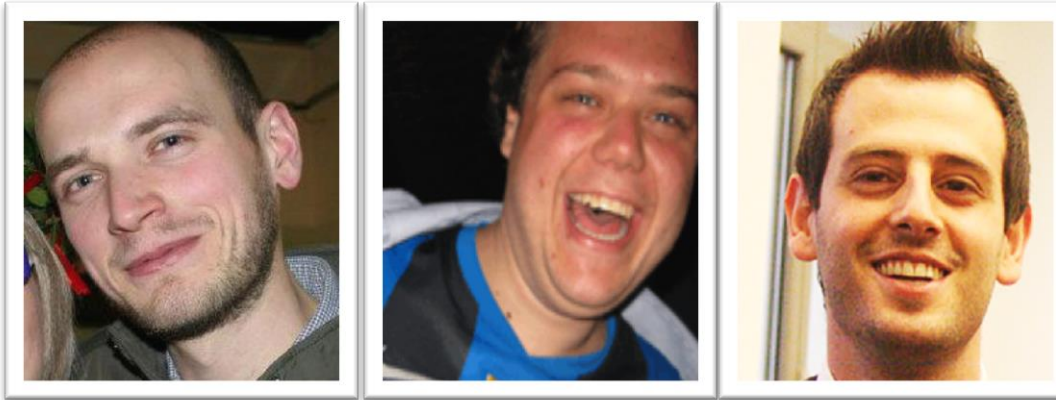
Tutti il materiale del progetto è contenuto nella cartella «progetto_BluetoothRanging_Tuzi_Visin_Saliai»:

- Cartella «codice_sorgente»: contiene il codice sorgente della applicazione bluetooth ranging;
- Cartella «doc»: contiene l'API Document del progetto (accessibile tramite il file index.html);
- Cartella «script_matlab»: contiene lo script per la generazione dei grafici;
- File regressione_lineare.pdf;
- File presentazione.pdf.



- Bluetooth:
Bluetooth specification Version 2.1 + EDR [vol 2] reperibile da <http://www.bluetooth.org/Technical/Specifications/adopted.htm>;
Slides corso Reti Wireless – Prof. Antonio Capone, Politecnico di Milano.
- Android:
API bluetooth per Android
<http://developer.android.com/guide/topics/connectivity/bluetooth.html>;
Portale Android Developer <http://developer.android.com/index.html>.
- Radio-localizzazione:
Slides corso Sistemi di Comunicazione Navigazione e Infomobilità – Prof. Monica Nicoli, Politecnico di Milano;
- Wikipedia per ricerche varie.

Bluetooth Ranging è un progetto realizzato da



Diego Tuzi

Marco Visin

Laerte Saliai

nell'ambito del corso di

Sistemi di Comunicazione e Navigazione per l'Infomobilità
tenuto dalla prof. Monica Nicoli nell'anno accademico 2011/2012.

diegotuzi@gmail.com

visin.marco@gmail.com

laerte85@gmail.com