

POLITECNICO DI MILANO
Facoltà di Ingegneria
Corso di Laurea in Ingegneria delle Telecomunicazioni



Adattamento della potenza in trasmissione per sistemi wireless a basso consumo

Relatore: Prof. Umberto SPAGNOLINI

Correlatore: Ing. Stefano SAVAZZI

Tesi di Laurea di:

Paolo BESTAGINI

Matr. 676285



Introduzione

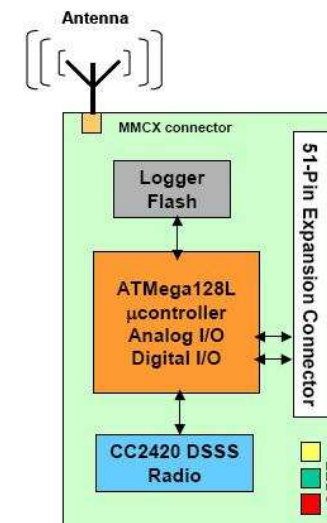
- I MICAz sono apparecchiature della Crossbow che, se equipaggiati con sensori, possono essere distribuiti su un'area per la creazione di una rete ad hoc.
- Lo scopo di questo lavoro è trovare un metodo per adattare la potenza in trasmissione alle condizioni del canale e prolungare così la durata delle batterie.
- Per raggiungere lo scopo si devono raccogliere dati sperimentali per definire il path-loss e le prestazioni dei MICAz.

MICAz



- Protocollo ZigBee per trasmissioni radio (2.4 GHz, 250 kbit/s).
- Interfacciabili al PC tramite porta seriale, USB, o ethernet.
- Alimentabili con 2 batterie AA (1.5 V ciascuna).
- Sistema operativo TinyOS.
- Sleep mode per risparmio batterie.
- Programmabili in nesC.
- Energia assorbita in ricezione maggiore di quella assorbita in trasmissione

- Microcontrollore: ATmega128L.
- Radio: CC2420.
- Connettore di espansione a 51 pin per sensori.

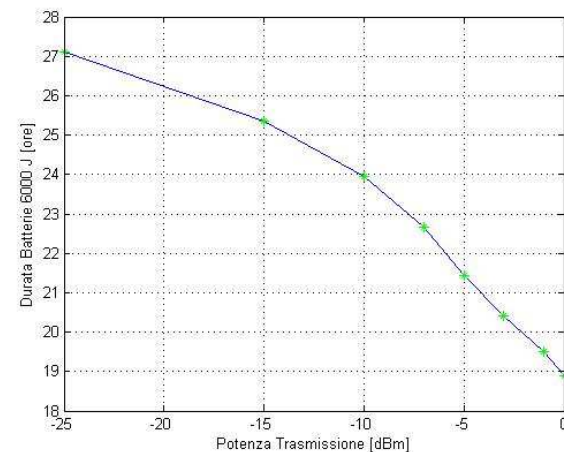
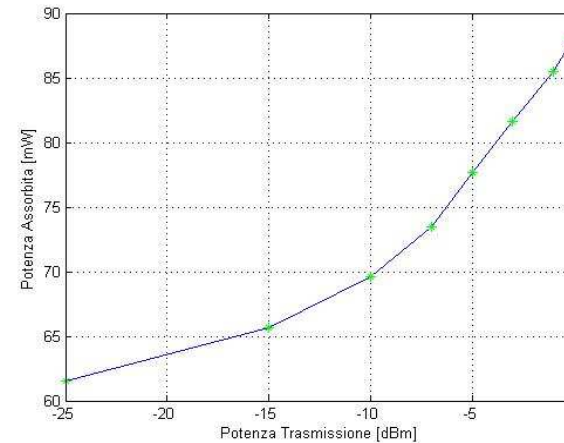


Potenza Assorbita

		Potenza in trasmissione P_T [dBm]	Corrente assorbita I [mA]
CC2420	Full operation (Rx)	/	18.8
	Full operation (Tx)	0	17.4
		-1	16.5
		-3	15.2
		-5	13.9
		-7	12.5
		-10	11.2
		-15	9.9
		-25	8.5
	ATMega128L	Sleep	/
Full operation		/	12
	P_A , Potenza Assorbita [mW]	Giorni/ore (batteria 6000J=555 mAh)	
Ricezione	90	18.5185 ore	
Sleep	0.033	2104.375 giorni	

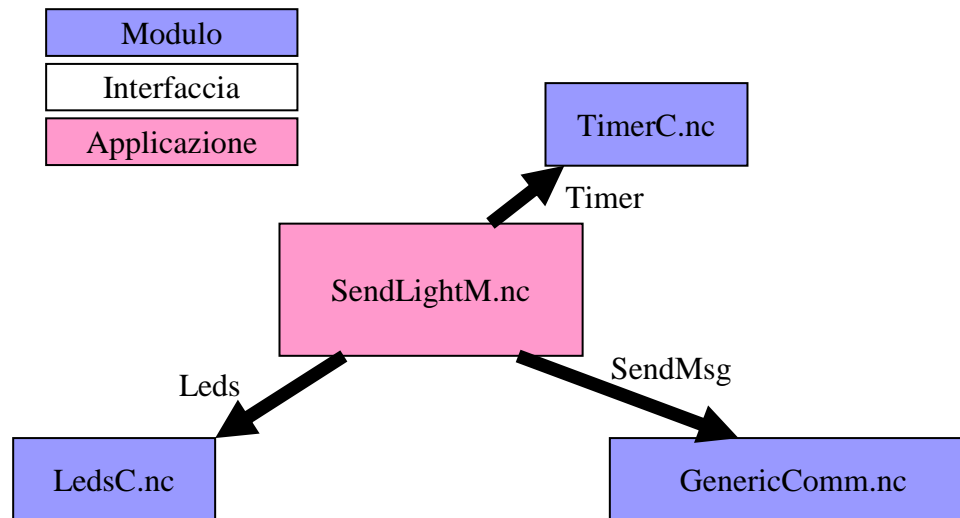
La potenza assorbita in ricezione è superiore alla potenza assorbita in trasmissione !!

$$P_{A,Full TX} = (I_{cc2420,Full TX} + I_{ATMega128L,Full TX}) \cdot TensioneBatteria$$



TinyOS e nesC

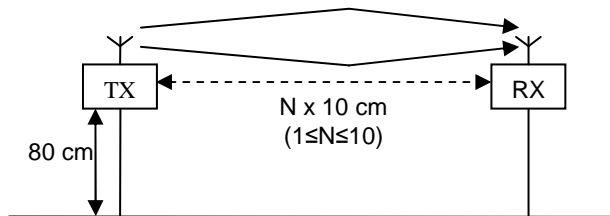
- TinyOS gestisce l'esecuzione delle applicazioni con metodologia FIFO:
 - Gli interrupt hardware sono gestiti immediatamente
 - Le applicazioni sono messe in coda
- Il nesC è il linguaggio in cui sono scritti i moduli per TinyOS:
 - Derivato del C
 - Permette una programmazione modulare
 - Definisce i moduli e le interfacce che li connettono
- Tipicamente un'applicazione usa le **interfacce** che i **moduli** forniscono:



Analisi del Path-loss

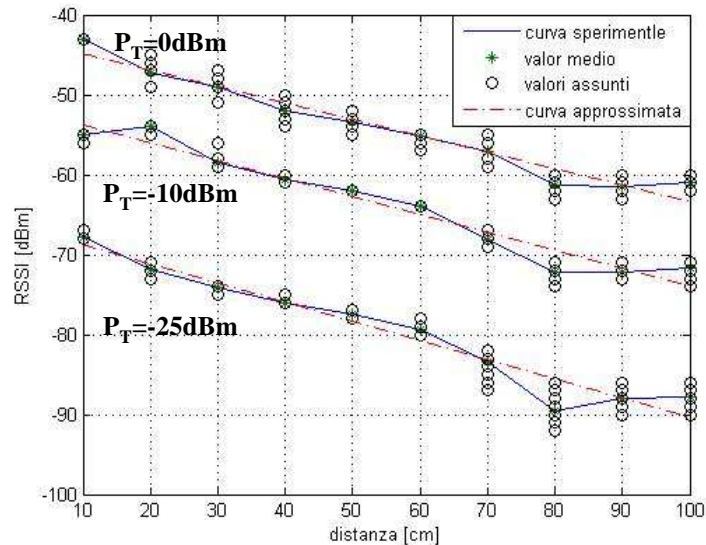
- RSSI medio valutato sulla trasmissione di 100 pacchetti

□ Trasmissioni LOS.

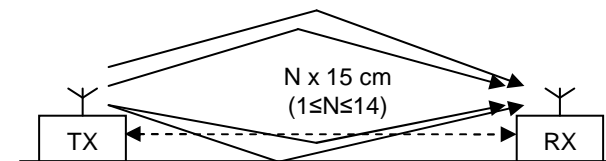


TX Power	Path-Loss
-25 dBm	-23.97 dB/m
-10 dBm	-22.4 dB/m
0 dBm	-20.57 dB/m

Modello del path-loss approssimato

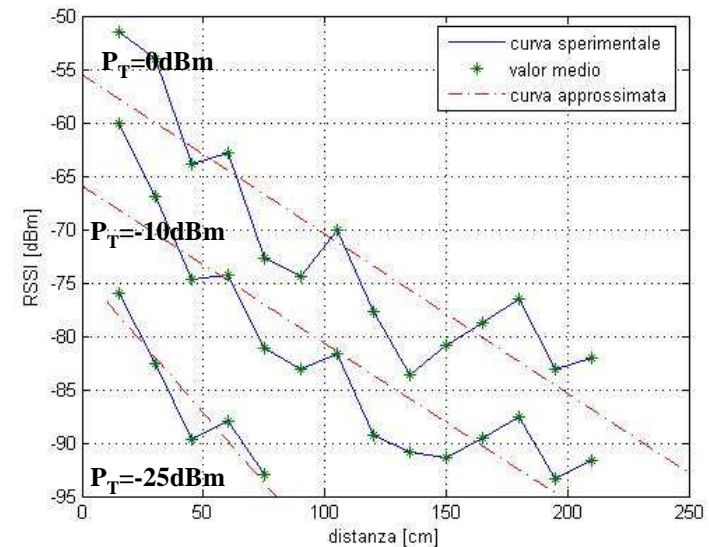


□ Trasmissioni quasi-LOS.



TX Power	Path-Loss
-25 dBm	-26.17 dB/m
-10 dBm	-14.75 dB/m
0 dBm	-14.91 dB/m

Modello del path-loss approssima male la curva
Perdita di pacchetti per RSSI < -90 dBm

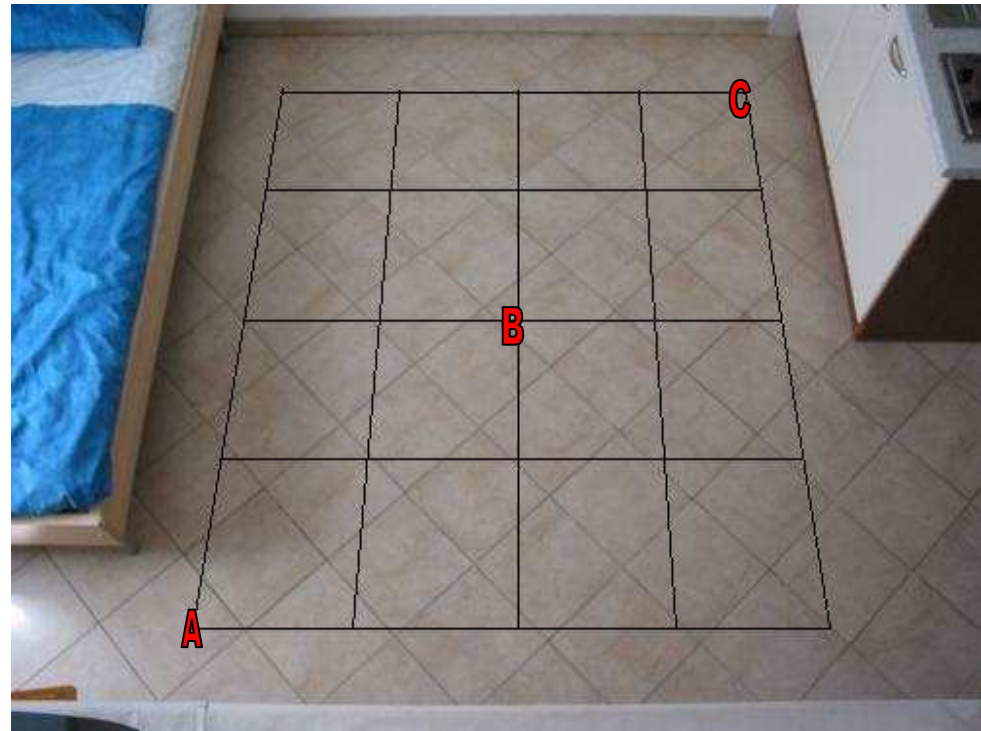
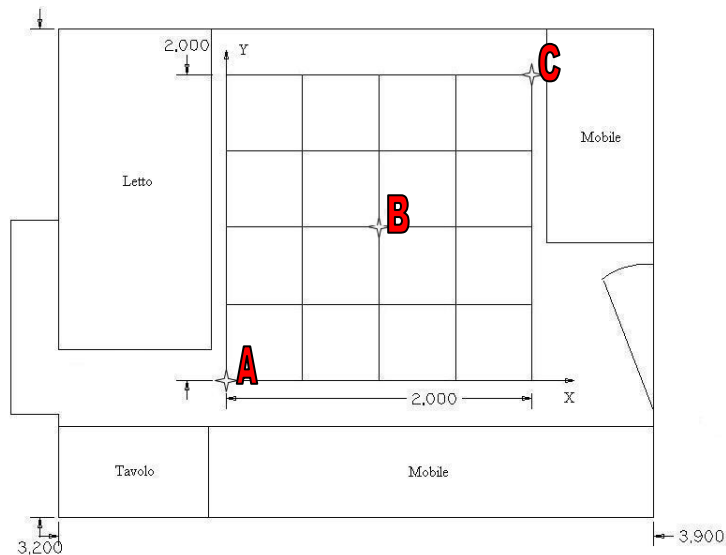


Analisi del Path-loss

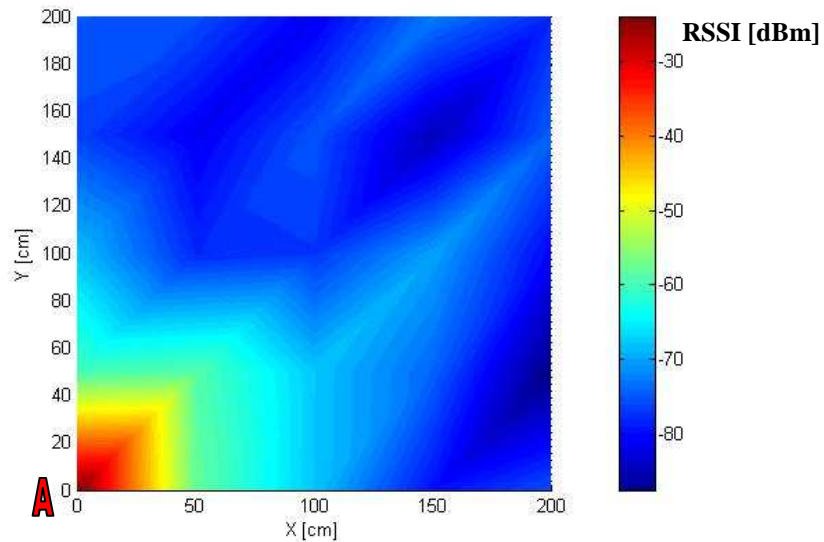
■ Trasmissioni quasi-LOS (2D)

■ Metodo:

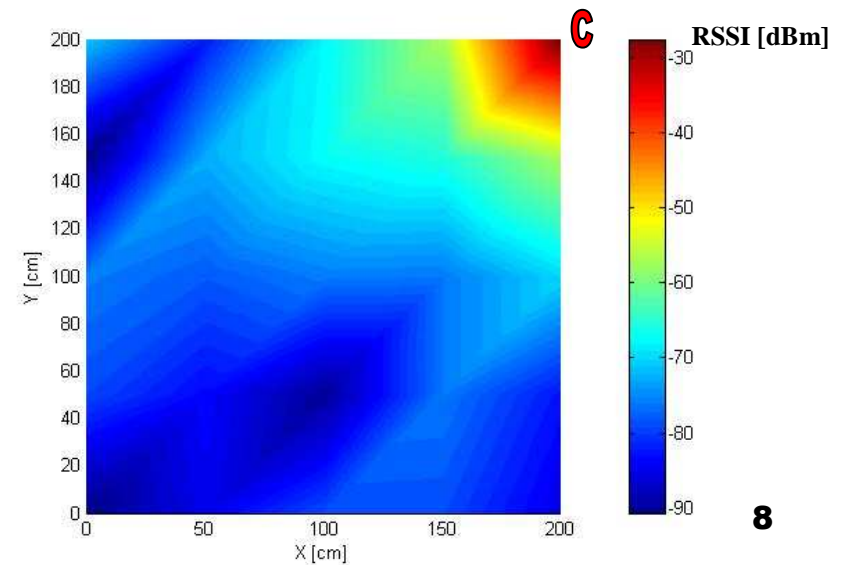
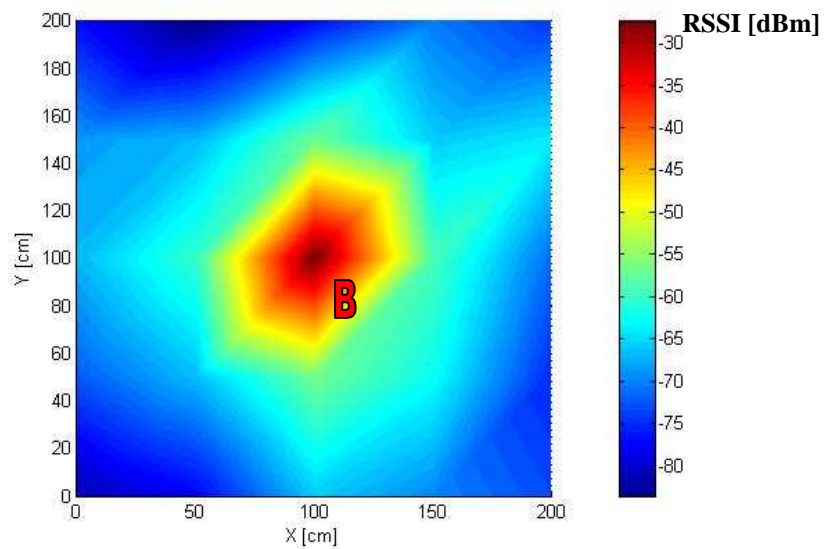
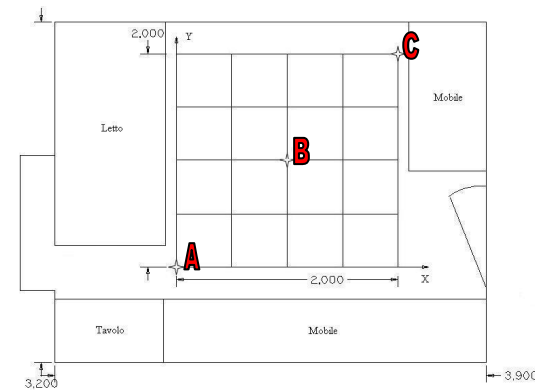
- Altezza: a terra
- RSSI valutato in media su 100 pacchetti
- Spostamenti sulla griglia 2x2 m
- Ricevitore in A, B, e C
- Usate tre potenze in trasmissione:
 - -25 dBm
 - -10 dBm
 - 0 dBm



Analisi del Path-loss

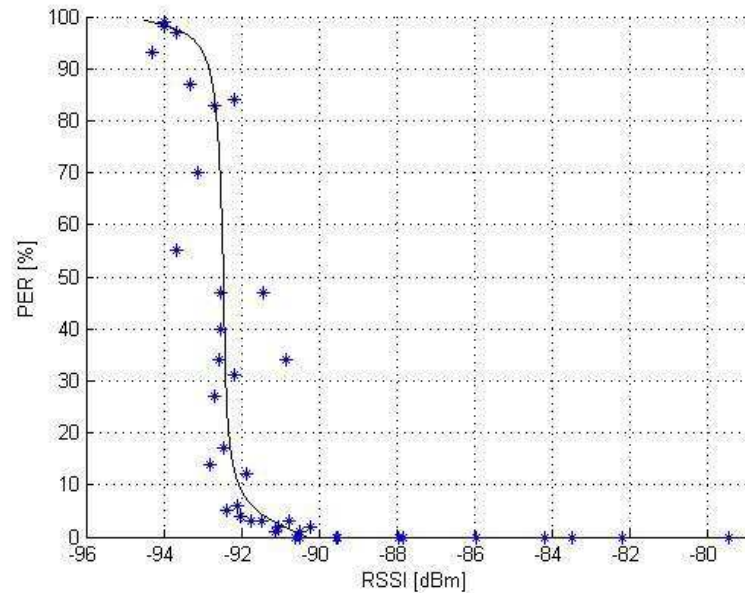


- Trasmissioni quasi-LOS (2D)
- Potenza in trasmissione 0 dBm



Analisi delle prestazioni

Packet Error Rate e Fuori Servizio



■ Packet Error Rate:

- Dagli esperimenti si nota perdita di pacchetti per RSSI vicini a -90 dBm

■ Fuori servizio:

- $P[\text{fuori-servizio}] = P[\text{RSSI} < -90 \text{ dBm}]$

■ Range:

- Alcuni esempi:
 - Indoor corridoio: >11m
 - Indoor stanza: 3 – 7 m (a seconda di dimensioni stanza)

RSSI > -90 dBm

→

PER → 0%

-94 dBm < **RSSI** < -90 dBm

→

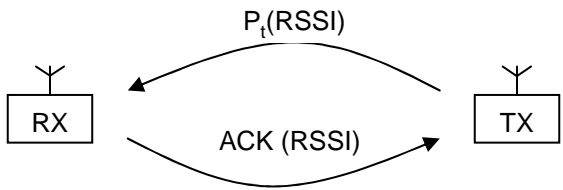
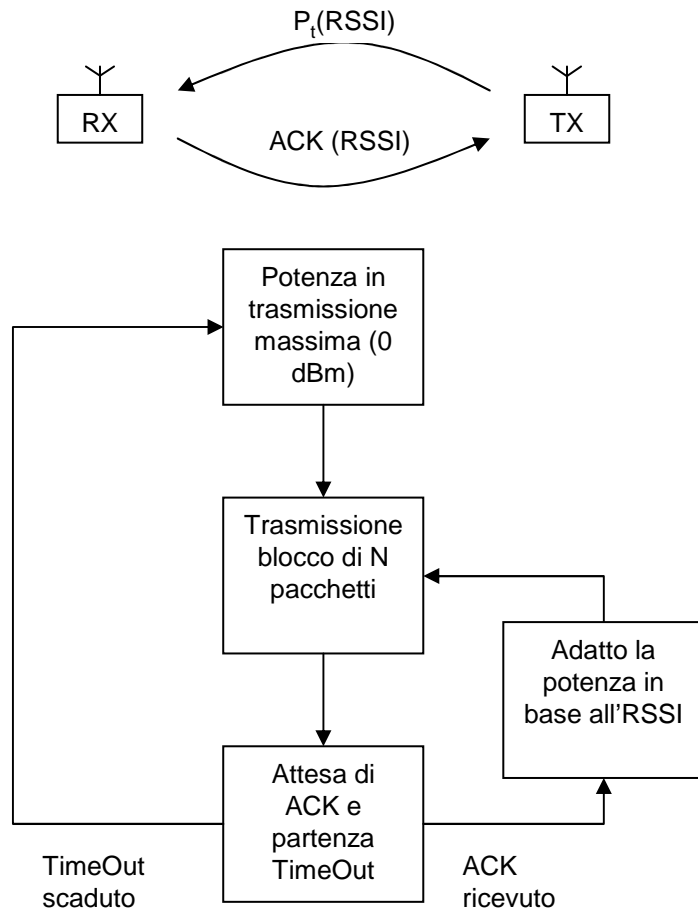
0% < **PER** < 100%

RSSI < -94 dBm

→

PER → 100%

Adattamento di potenza



■ Algoritmo:

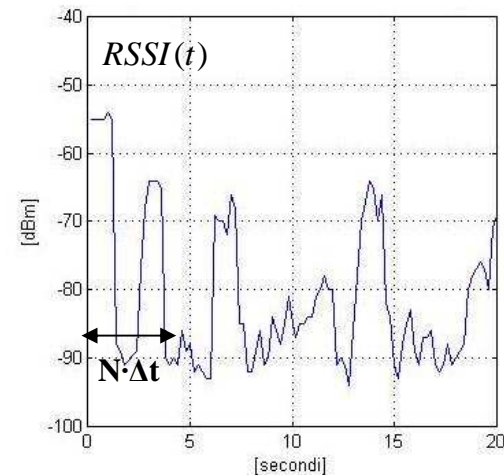
- Il trasmettitore spedisce N pacchetti (uno ogni Δt secondi).
- Se vengono ricevuti senza errori, il ricevitore spedisce un pacchetto di ACK contenente anche il valore di $RSSI$ degli N pacchetti.
- Se il trasmettitore riceve l'ACK, adatta la potenza per garantire $RSSI$ minimo di $RSSI_{TH} = -90$ dBm.
- Se il trasmettitore non riceve l'ACK entro lo scadere di un TimeOut, spedisce i successivi N pacchetti a potenza massima (0 dBm).

■ Calcolo nuova potenza:

$$\overline{RSSI} = \frac{1}{N \cdot \Delta t} \cdot \int_0^{N \cdot \Delta t} RSSI(t) \cdot dt$$

$$\Delta P = RSSI_{TH} - \overline{RSSI}$$

$$P(t+1) = P(t) + \Delta P$$



Adattamento di potenza

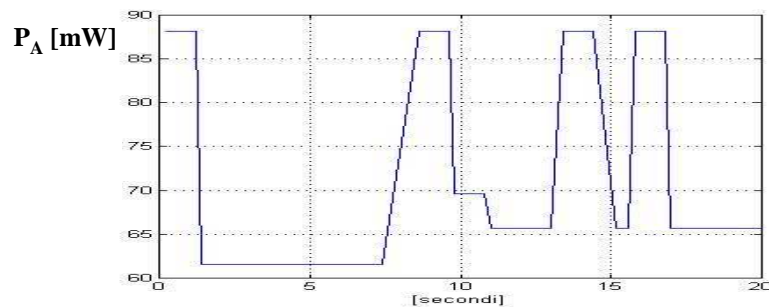
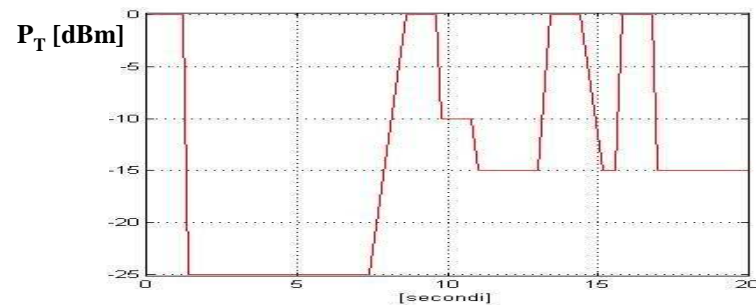
- **Metodo:**

- Trasmettitore allontanato dal ricevitore per creare diminuzione di RSSI
- Spedizione pacchetti ogni $\Delta t = 200$ ms

- **Esperimenti (LOS e quasi-LOS):**

- N=6, aggiornamento ogni 1200 ms

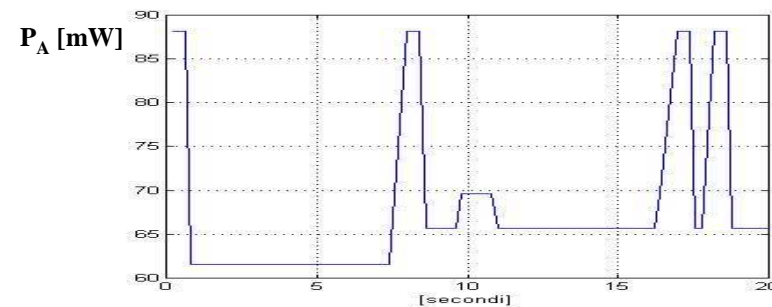
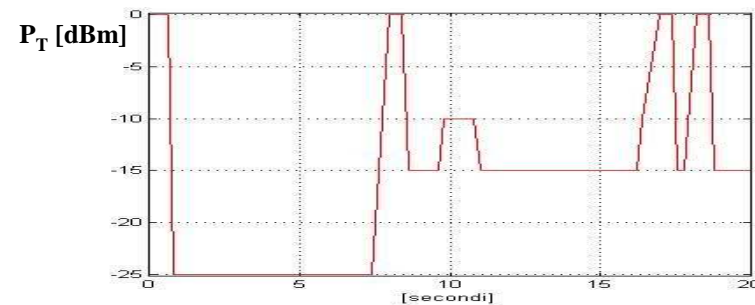
	Quasi-LOS	LOS
P_T media	-5.8 dBm	-4.5 dBm
PER	9%	7%



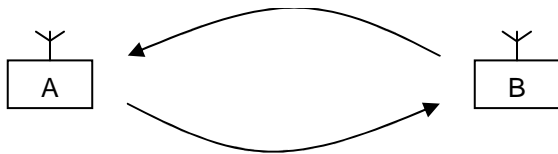
- **Esperimenti (LOS e quasi-LOS):**

- N=3, aggiornamento ogni 600 ms

	Quasi-LOS	LOS
P_T media	-8.4 dBm	-10.5 dBm
PER	5%	5%



Energia assorbita



- Considero flusso bidirezionale tra 2 MICAz
- Il 50% del tempo trasmettono e il 50% del tempo ricevono
- I feedback aggiungono pochi bit al flusso (trascurabili)
- **Confronto caso adattato e non adattato (1s trasmissione e 1s ricezione):**
 - Devo considerare le ritrasmissioni dovute agli errori: ogni pacchetto perso determina un aumento dei tempi di trasmissione e ricezione

	Non Adattato	Adattamento $\Delta t = 1.2 \text{ s}$ (N=6)		Adattamento $\Delta t = 600 \text{ ms}$ (N=3)	
		/	Quasi-LOS	LOS	Quasi-LOS
P_T media [dBm]	0	-5.8	-4.5	-8.4	-10.5
E_T in 1 s [mJ]	1	0.26	0.35	0.14	0.09
$P_{A, TX}$ [mW]	88.2	70.4	72.8	67.3	65.8
$P_{A, RX}$ [mW]	90	90	90	90	90
PER	0%	9%	7%	5%	5%
Tempo effettivo [s]	2	2.18	2.14	2.1	2.1
$E_{A, TOT}$ [mJ]	178.2	174.8	174.2	165.2	163.6
Guadagno % su E_A	/	1.9%	2.2%	7.3%	8.1%



Conclusioni

- Il modello del path-loss convenzionale non approssima bene il rapporto tra potenza in ricezione e distanza:
 - Nelle trasmissioni quasi-LOS (in cui ci sono maggiori riflessioni), il rapporto tra RSSI e distanza si discosta dall'andamento lineare approssimato, più vicino all'andamento sperimentale in trasmissioni LOS.
- Si è modellizzata la probabilità di perdita dei pacchetti attraverso una probabilità di fuori-servizio. L'RSSI può essere usato come soglia per imporre una probabilità di errore in ricezione:
 - Dal grafico che lega RSSI e PER risulta infatti che il tasso di errori è quasi nullo se la potenza in ricezione è superiore a -90 dBm.
- Si è visto che è possibile adattare la potenza in trasmissione sfruttando la lettura della potenza in ricezione, e un RSSI di soglia (-90 dBm) per imporre un certo packet error rate:
 - L'adattamento di potenza porta ad un guadagno sull'energia assorbita e di conseguenza sulla durata delle batterie.