

Università degli studi di Catania
L.M. Ing. Informatica
Reti per l'automazione industriale
Alessandro Di Stefano - Salvatore Mulè - Alessandro Sangiorgi

Network simulation e meccanismo di acknowledgment/ritrasmissione per una rete di tre nodi su protocollo MAC TDMA al di sopra del livello fisico di 802.15.4

La presente relazione tratta lo sviluppo di un modello di simulazione di un protocollo MAC wireless che operi sul livello fisico del protocollo IEEE 802.15.4, basato su un approccio di tipo TDMA in cui ad ogni nodo sono assegnati uno o più slot, e che inoltre integra ack e ritrasmissione.

Tale modello si interfaccia con il livello fisico di IEEE 802.15.4 basato sul framework INET.

Lo scopo di implementazione, nel modello semplificato del protocollo MAC, è stato implementare un meccanismo di base per l'acknowledgment e la ritrasmissione.

802.15.4

Lo standard IEEE 802.15.4 è stato definito appositamente per occuparsi del livello fisico e del livello MAC di reti a bit rate non elevato e basso consumo energetico. Al di sopra del livello MAC, diversi standard (ZigBee, 6LoWPAN, WirelessHart, Bluetooth) operano, fino al layer applicativo, per permettere la trasmissione di dati e informazioni utili nel panorama consumer e nell'ambito dell'automazione industriale/building automation.

Le caratteristiche principali sono:

- data-rate nei range 20Kbps, 40Kbps e 250Kbps
- 2 modalità d'indirizzamento: a 16 e a 64 bit. L'indirizzo a 64 bit- permette la comunicazione con tutte le reti 802 (l'indirizzo MAC di livello 2 ha 64 bit). L'indirizzo a 16 bit é usato internamente.
- CSMA/CA (collision avoidance) per l'accesso al canale.
- La banda a 2.4GHz è suddivisa in 16 canali; quella a 900MHz in 30 canali e quella a 800MHz su un unico canale.

Lo standard prevede la classificazione di due tipologie di dispositivi:

- Full Function Device (FFD)
- Reduced Function Device (RFD)

Generalmente si usa una topologia a stella in cui l'FFD (che inizializza la rete) ne diventa coordinatore (PAN Coordinator). In questo caso gli RFD possono collegarsi esclusivamente agli FFD.

Per i casi d'uso previsti, risulta interessante nell'implementazione dell'IEEE-802.15.4 che i nodi a funzionalità ridotta possono collegarsi a più nodi e possono fare da bridge tra nodi con funzionalità completa.

Lo standard prevede anche una topologia di rete peer-to-peer, utilizzata nelle WSN in cui è necessario il multi-hop.

Livello Fisico

Le bande di frequenza sono 3:

- 868.0–868.6 MHz: Europa ;
- 902–928 MHz: Nord America ;
- 2400–2483.5 MHz: Resto del mondo.

Il livello fisico svolge diversi compiti:

- Packet generation
- Packet reception
- Data transmission/reception
- Power Management
- Radio transceiver ON/OFF
- Channel energy detection
- Link quality indication

Il pacchetto del livello fisico è costituito da 6 byte che rappresentano il Preamble, Packet Delimiter, PHY Header. A questi vanno aggiunti da 0 a 127 byte di dati.

MAC-Layer

Il livello MAC consente la trasmissione di frame MAC tramite l'utilizzo del canale fisico.

Esso inoltre controlla la validità delle frame e gestisce le associazioni dei nodi.

Le attività principali svolte dal MAC Layer sono:

- Beacon generation (if coordinator)
- Beacon synchronization
- CSMA-CA
- Error correction
- Link reliability
- Security

Lo standard definisce 4 tipi di frame:

- Data Frame
- Beacon Frame
- Acknowledgment Frame
- MAC Command Frame

Il tipo di traffico supportato dal livello MAC può essere sia periodico, e quindi dati provenienti da sensori che periodicamente inviano il parametro misurato, che aperiodico.

Nel lavoro sviluppato non è stato implementato il livello MAC come da specifiche ufficiali dell'IEEE.

TDMA

La gestione della trasmissione della frame può essere implementata in diversi modi.

Lo standard 802.15.4 prevede l'utilizzo del CSMA/CA, tuttavia si possono presentare situazioni in cui questo approccio potrebbe essere meno efficiente rispetto ad altri.

Bisogna ricordare che non esiste un approccio migliore di un altro: tutto dipende dal contesto applicativo.

Il TDMA è una tecnica di multiplexing numerica in cui la condivisione del canale è realizzata mediante ripartizione del tempo di accesso allo stesso da parte degli utenti.

Ogni nodo in sostanza può trasmettere il proprio dato nella time-slot che gli è stata dedicata.

Sebbene risulti abbastanza efficiente, questo approccio richiede che tutti i nodi siano perfettamente sincronizzati (global clock).

Come detto in precedenza non esiste un approccio migliore di un altro, tuttavia possiamo servirci di parametri prestazionali per capire quale tra i possibili approcci potrebbe essere la scelta migliore.

Implementazione

Come descritto sopra, l'implementazione è stata realizzata operando, a livello fisico, attraverso le librerie del framework INET.

Sono stati previsti tre nodi che inviano messaggi tra di loro (ogni nodo invia un messaggio ad un indirizzo casuale diverso dal proprio).

TDMA è stato implementato dando minore attenzione al problema di sincronizzazione e sfruttando la funzione *simtime()* per scandire il tempo tra le trasmissioni ad intervalli parametrici definiti nella fase di avvio della stessa.

Il meccanismo di acknowledgement e di ritrasmissione, anch'esso basilare, permette di stabilire se un determinato pacchetto è stato consegnato correttamente ed entro un tempo utile (*ACKTimeout*), anch'esso definibile nella fase di avvio della simulazione.

Non appena, come da TDMA, il nodo riceve la disponibilità ad inviare pacchetti, si verifica che siano presenti nella coda appropriata (*retrQueue*) degli eventuali pacchetti da reinviare ed eventualmente si procede all'invio del primo disponibile dalla coda (pop).

Ogni qualvolta un nodo riceve un pacchetto (non ack) viene, nello stesso slot del sender, inviato l'ack.

Se, invece, il pacchetto ricevuto è un Ack legato ad un precedente pacchetto trasmesso, si aggiunge un oggetto *cNamedObject* ad un *cArray*.

Per ogni pacchetto inviato viene generato un messaggio, schedato a *simtime() + ACKTimeout*; non appena, il messaggio viene processato, si verifica all'interno del *cArray* descritto precedentemente se è presente l'oggetto chiamato con l'identificativo del pacchetto (*ackId*) che ha generato il messaggio: nel caso in cui è presente tale elemento, lo stesso viene rimosso, ed il pacchetto viene considerato consegnato con ack ricevuto; viceversa, il pacchetto viene considerato non ricevuto e viene ritrasmesso (l'incremento di un counter apposito per singolo nodo) se e solo se il counter di ritrasmissioni risulta minore delle massime ritrasmissioni possibili (configurabili in fase di simulazione).

Simulazione

Nella fase di simulazione, si è deciso di raccogliere le statistiche, per ogni nodo, raggruppando pacchetti inviati, ack ricevuti, packetLoss e end2end delay a livello Mac (dalla ricezione del pacchetto dal livello applicativo alla consegna dell'ack).

Configurazione dei parametri

SlotTime: la lunghezza di un intero TDMAFrame contenente dati è 48B; un ack invece ha dimensione 15B; considerato che all'interno dello stesso slot un nodo invia un pacchetto dati e ipoteticamente riceve un ack, con un bitrate 250kbps risulta necessario un tempo di 2.016 ms; considerato, qualitativamente, il tempo di propagazione nel canale si è deciso di approssimare lo SlotTime a 2.1ms.

App.Period: impostiamo manualmente il periodo di generazione pacchetti dell'app a un tempo uguale al totale degli slot * lo slotTime + 2ms. È comunque possibile utilizzare altri parametri al lancio della simulazione.

MaxRetransmissions: configurabile in fase di lancio.

numNodes: numero dei nodi da costruire per eseguire la simulazione (richiesto in fase di lancio)

totSlot: configurabile in fase di lancio della simulazione, permettendo, nel caso in cui il totale degli slot sia maggiore del numero dei nodi di distribuire più slot in ordine dai primi nodi (è necessario che totSlot sia non inferiore a numNodes).

Statistiche

Le statistiche considerate, entrambe a livello MAC, sono:

packetLoss: percentuale di pacchetti persi media nel tempo; viene considerato perso un pacchetto che, nonostante le *MaxRetransmissions* ritrasmissioni, non ha ricevuto ack.

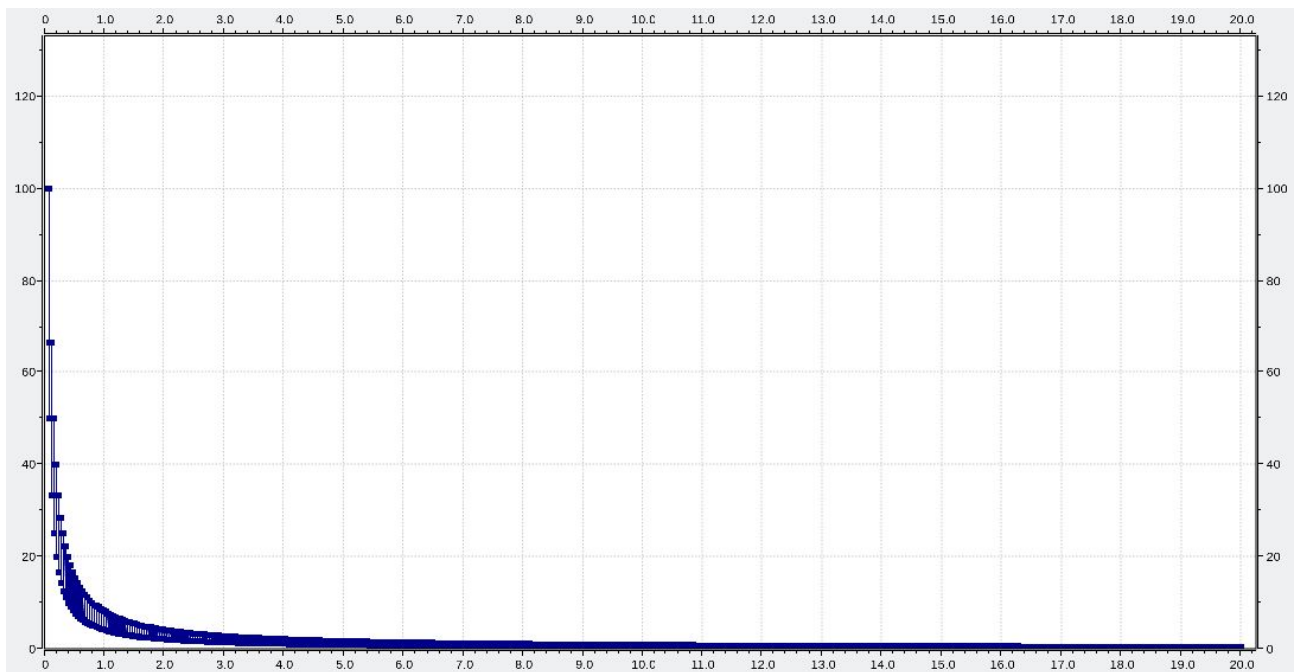
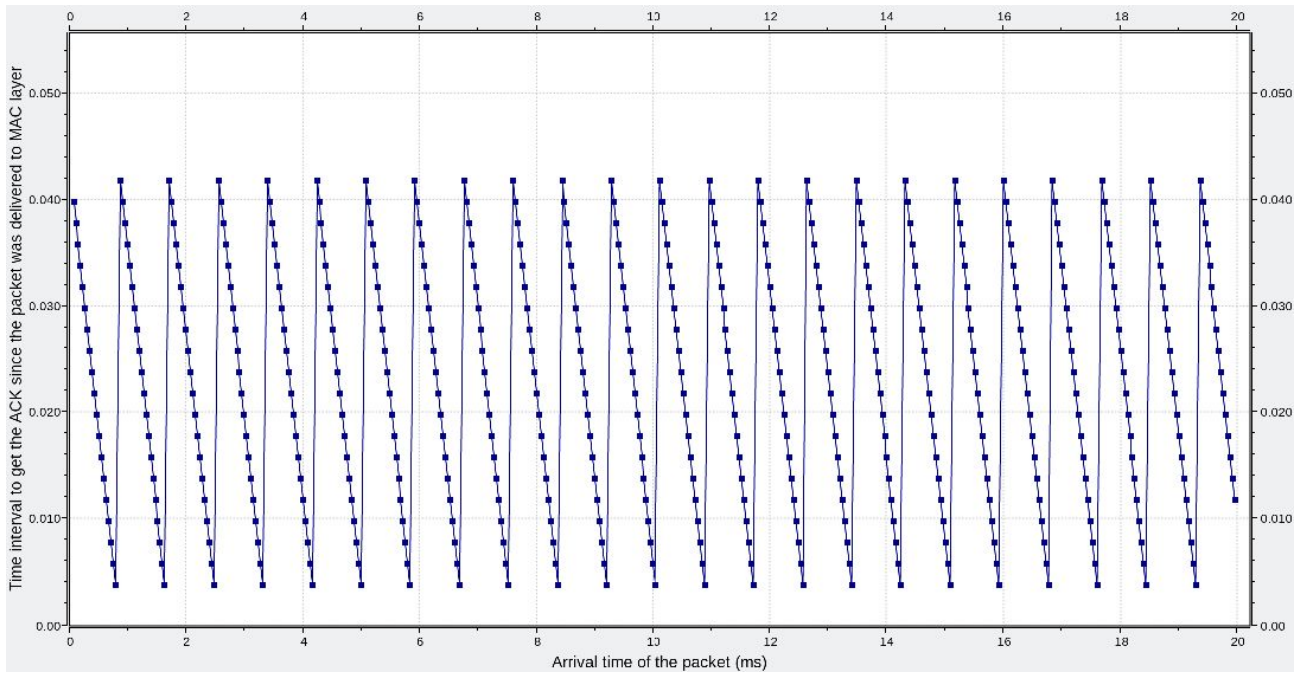
end2endDelay: intervallo di tempo dalla ricezione dal livello applicativo del pacchetto dati, alla ricezione dell'ack nel livello MAC; i pacchetti persi non vengono considerati; nei grafici si trova il tempo di registrazione della statistica sull'asse x (coincidente con il tempo di ricezione del pacchetto precedentemente inviato), sull'asse y l'intervallo di tempo differenza tra il tempo di arrivo e il tempo di partenza del pacchetto.

Alcuni esempi di test

Configurazione 1

App.period: 44.1ms; numNodes: 20; totSlot: 20; MaxRetransmission: 1

Tempo di simulazione: 20s;

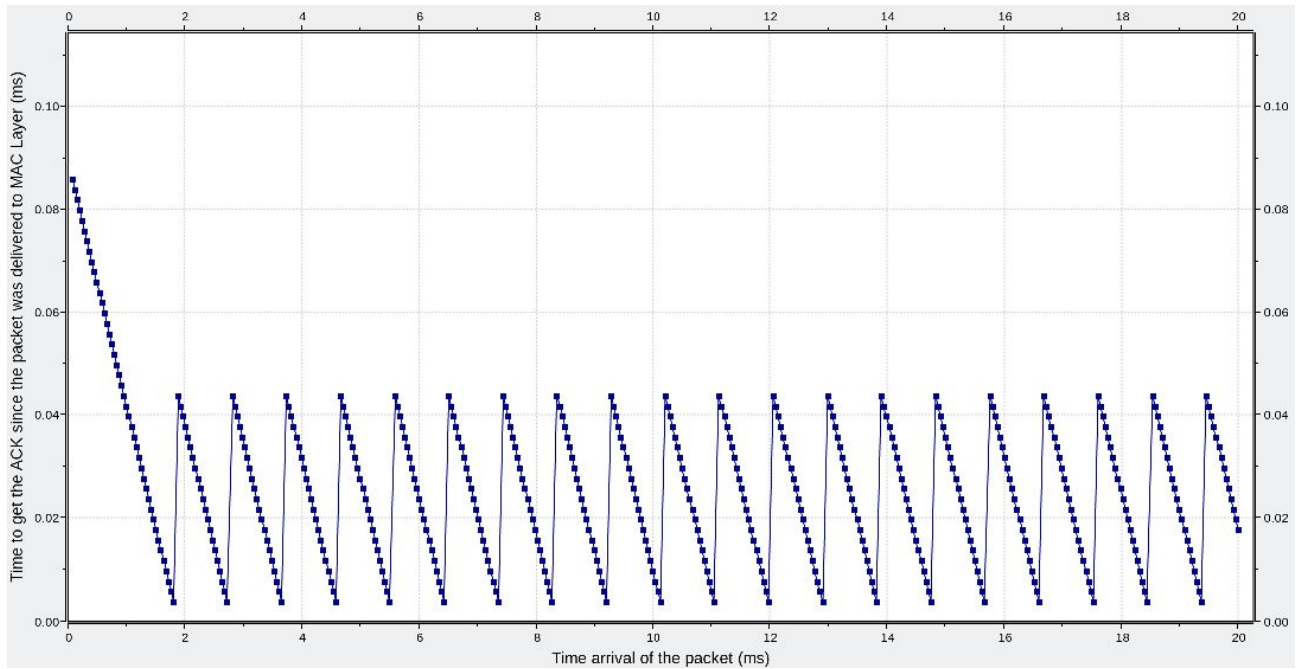


PacketLoss Minimo al 2% e minore e2e delay, mediamente stabile.

Configurazione 2

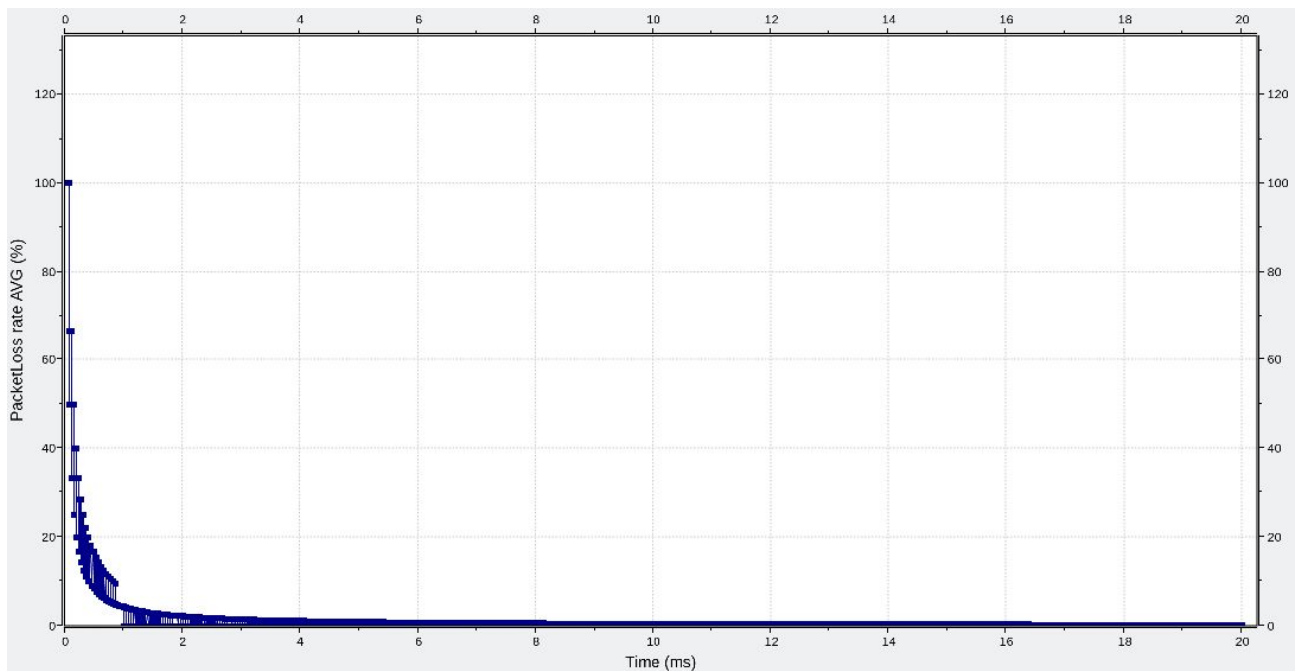
App.period: 44.1ms; numNodes: 20; totSlot: 20; MaxRetransmission: 2

Tempo di simulazione: 20s;



L'end to end delay resta mediamente lo stesso della configurazione precedente.

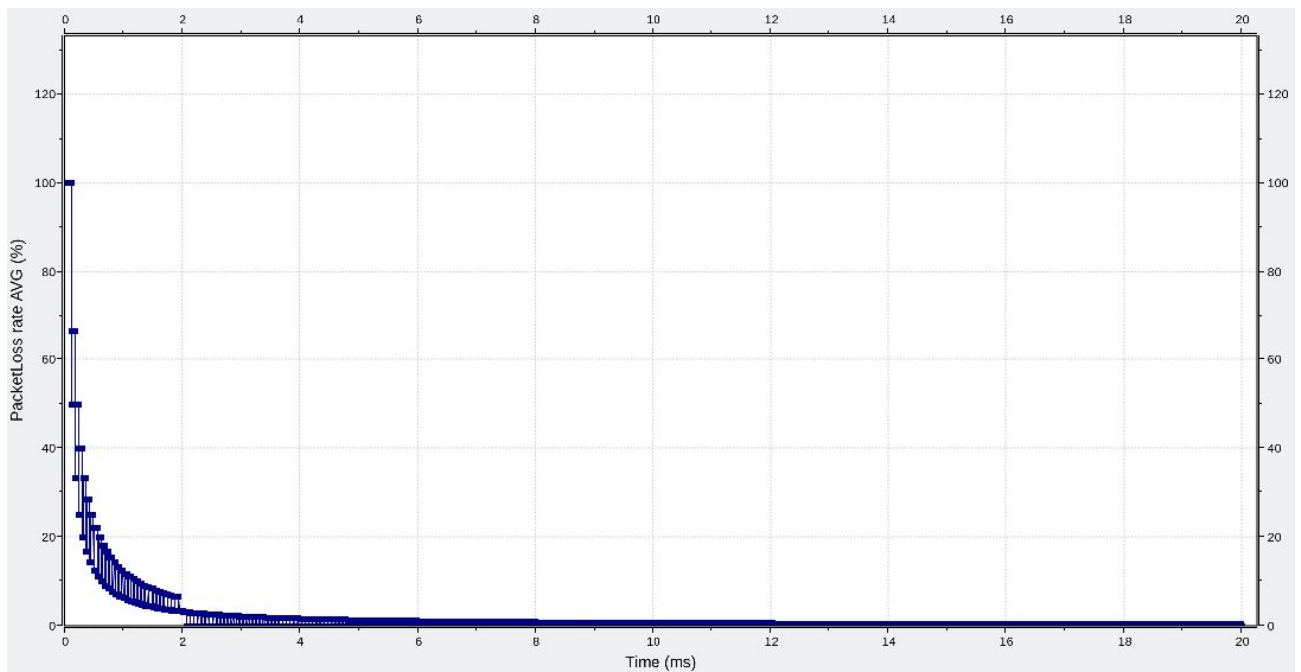
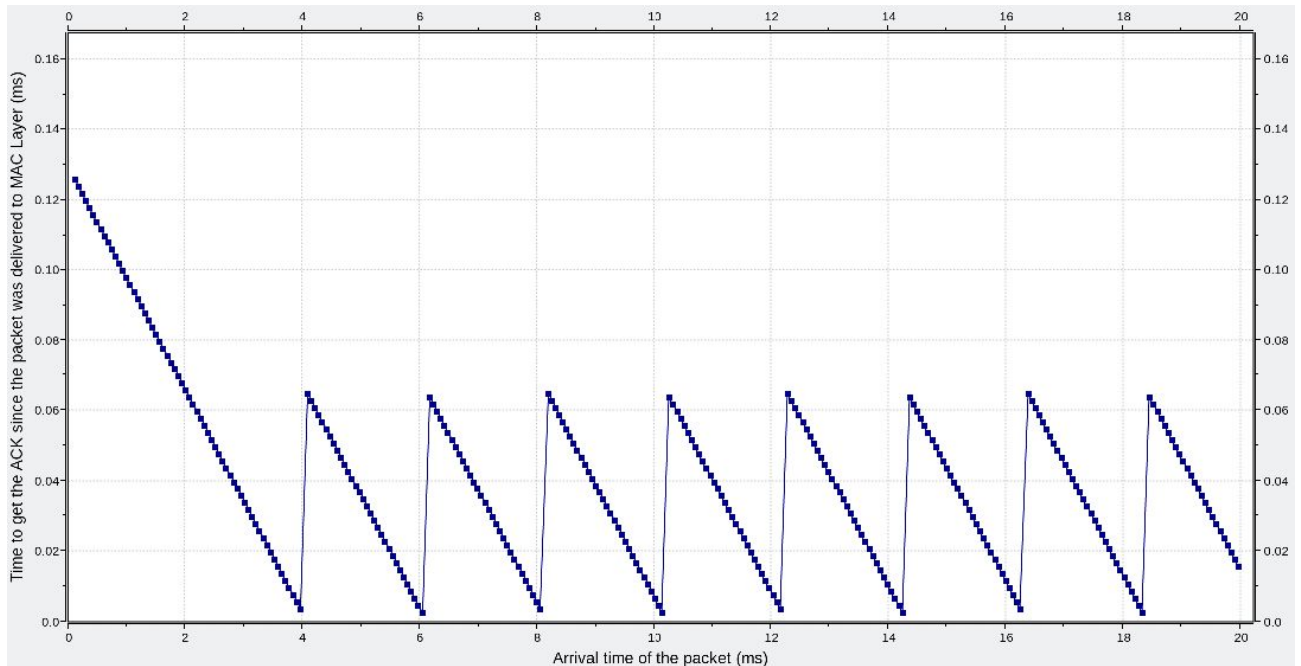
Diminuisce notevolmente il packetLoss rate medio che tende a 1 %.



Configurazione 3

App.period: 65.1ms; numNodes: 30; totSlot: 30; MaxRetransmission: 2

Tempo di simulazione: 20s;

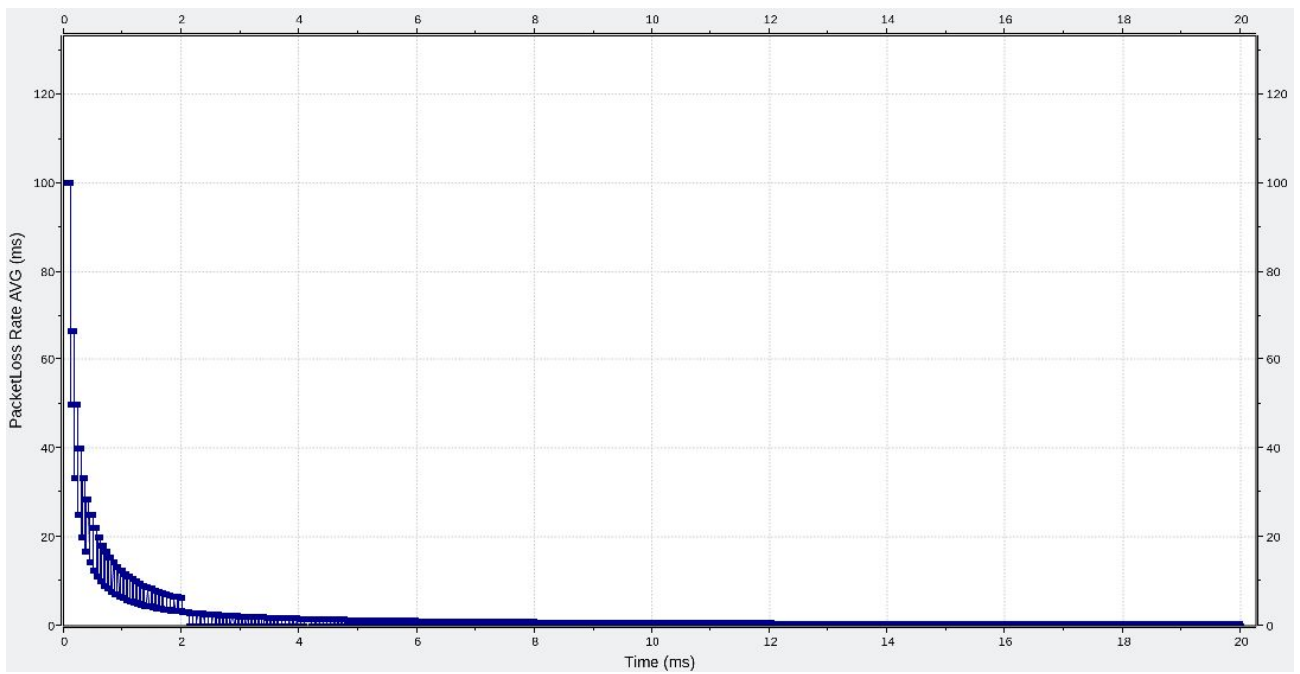
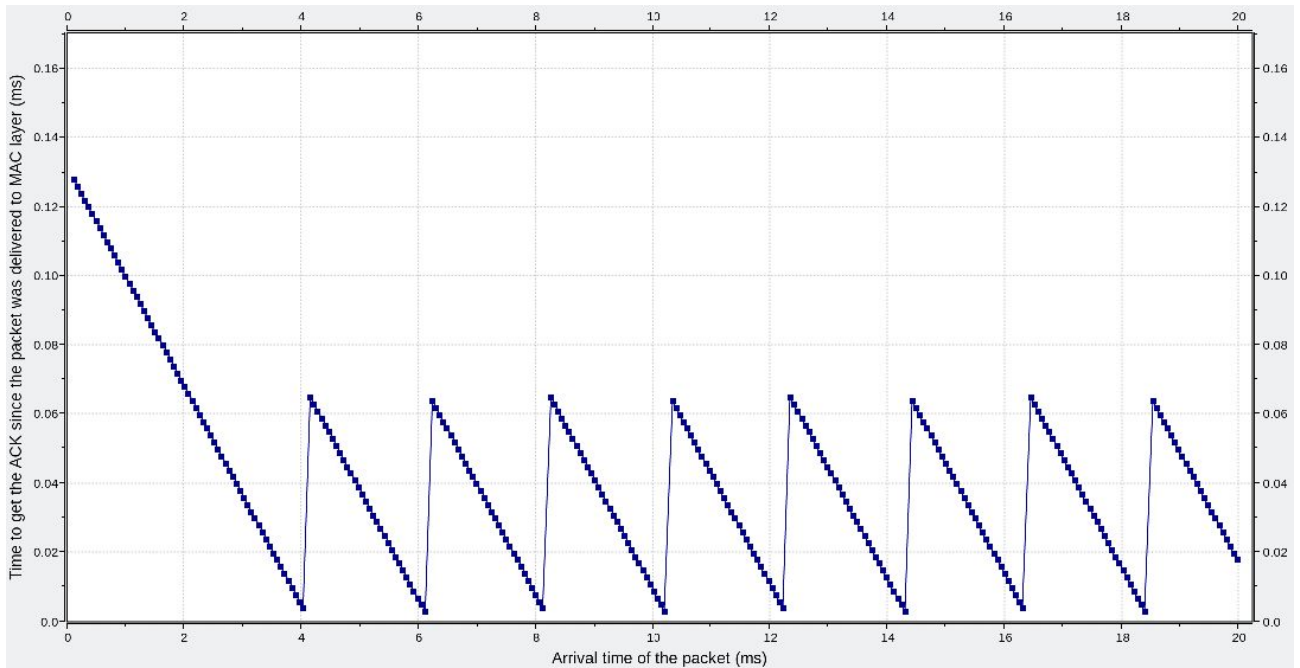


Aumentando esclusivamente il numero di nodi è aumentato il packetLoss rate che si ferma ~2.3% alla fine della simulazione. Contemporaneamente aumenta in media il tempo di delay per l'arrivo dei pacchetti pacchetti quando si ha perdita.

Configurazione 4

App.period: 65.1ms; numNodes: 30; totSlot: 30; MaxRetransmission: 3

Tempo di simulazione: 20s;



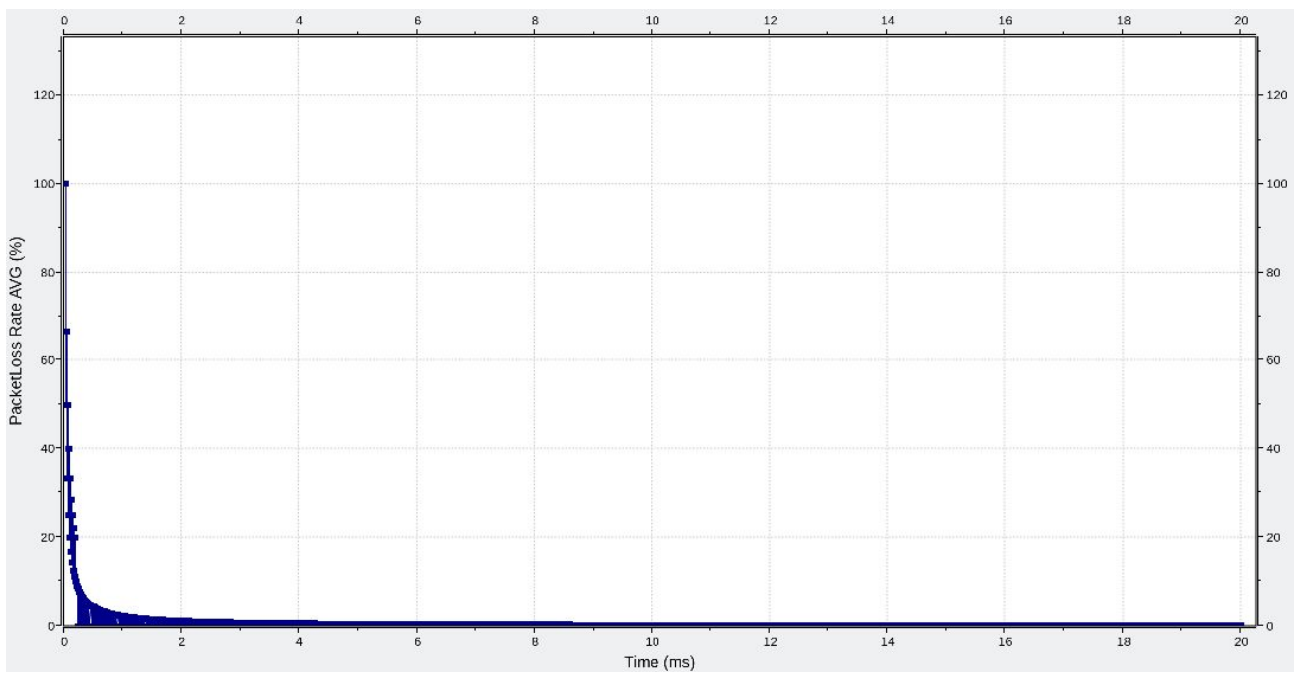
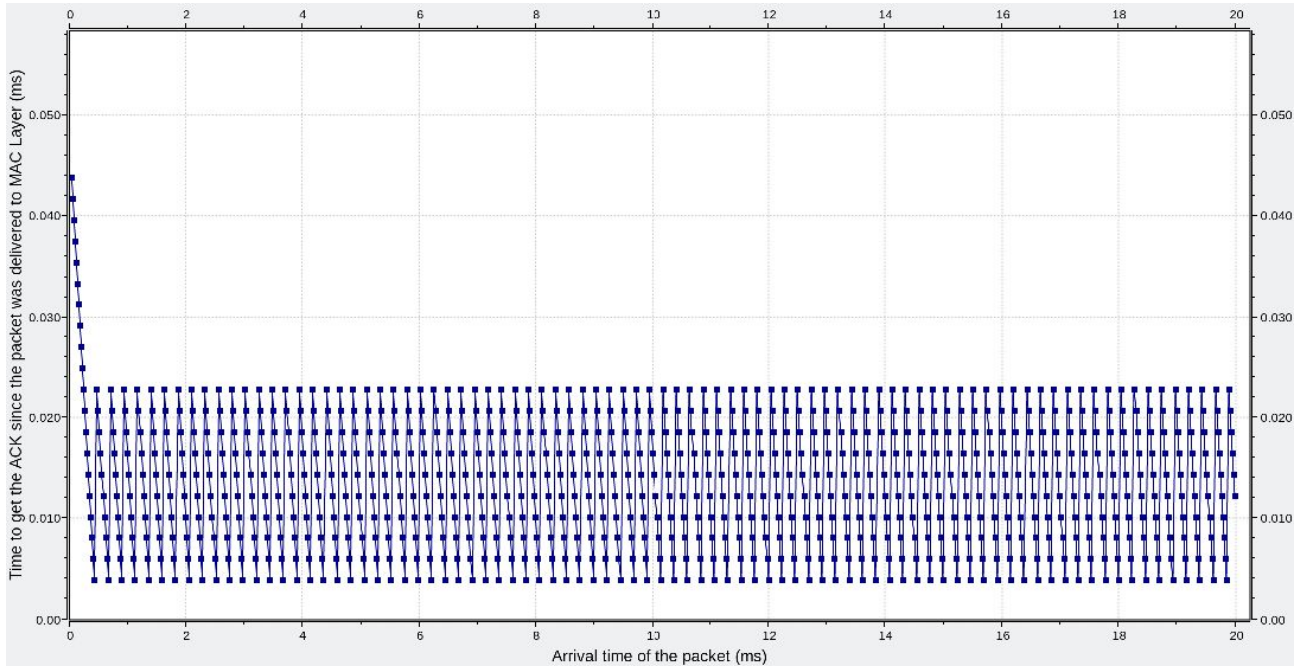
Le performance restano invariate aumentando il limite di ritrasmissioni a 3

Minimo packetLoss rate: 2.3 %

Configurazione 5

App.period: 23ms; numNodes: 10; totSlot: 10; MaxRetransmission: 2

Tempo di simulazione: 20s;



Min = 0.73 %