

# POLITECNICO DI MILANO

Facoltà di Ingegneria dell'Informazione  
Corso di Laurea in Ingegneria delle Telecomunicazioni  
Dipartimento di Elettronica e Informazione



## **Sviluppo e valutazione delle prestazioni di algoritmi di instradamento per Delay Tolerant Networks**

Relatore : Ing. Matteo Cesana

Correlatore : Ing. Ilario Filippini

Tesi di laurea di:

Riccardo BOTTI

Matr. 770322

Anno accademico 2011/2012



*A tutte le persone care  
che mi proteggono dal cielo...*

# Indice

Sommario .....	1
Introduzione.....	2
Capitolo 1 – Reti opportunistiche (DTN).....	4
1.1 - Definizione .....	4
1.2 - Scenari applicativi.....	4
1.3 - Fattori caratterizzanti .....	7
Capitolo 2 – Instradamento .....	9
2.1 - Metriche di routing .....	9
2.2 - Scenari di routing particolari.....	9
2.2.1 - Applicazione Publish / Subscribe.....	10
2.2.2 - Applicazione Content Sharing .....	16
2.2.3 - Applicazione Wildlife Tracking.....	21
2.2.4 - Applicazione PROPHET (invio di messaggi).....	27
Capitolo 3 – Routing sociale .....	31
3.1 - Introduzione.....	31
3.2 - Profili applicativi per l'instradamento .....	32
3.2.1 - Facebook.....	32
3.2.2 - Foursquare .....	34
3.3 - Tassonomia delle comunicazioni e funzioni di costo associate .....	36
3.3.1 - Comunicazione Unicast.....	36
3.3.2 - Comunicazione Multicast.....	38
3.3.3 - Comunicazione Geocast.....	39
Capitolo 4 - Valutazione delle prestazioni .....	41
4.1 - OMNeT++ 4.2.2 .....	41
4.2 - Scenario e parametri .....	43
4.3 - Analisi prestazioni per Protocollo Fisico.....	44
4.3.1 - Protocollo Fisico Unicast.....	44
4.3.2 - Protocollo Fisico Multicast .....	49
4.3.3 - Protocollo Fisico Geocast .....	56
4.4 - Analisi prestazioni per Protocollo Applicativo .....	65
4.4.1 - Protocollo Applicativo Unicast .....	65
4.4.2 - Protocollo Applicativo Multicast .....	71
4.4.3 - Protocollo Applicativo Geocast .....	78

4.5 - Analisi prestazioni per Protocollo Ibrido .....	85
4.5.1 - Protocollo Ibrido Unicast.....	85
4.5.2 - Protocollo Ibrido Multicast .....	93
4.5.3 - Protocollo Ibrido Geocast .....	103
4.6 - Confronto protocolli.....	111
4.6.1 - Protocolli Unicast.....	112
4.6.1 - Protocolli Multicast .....	114
4.6.1 - Protocolli Geocast .....	115
Conclusioni.....	118
Bibliografia.....	120

# Elenco delle figure

Figura 1.1 : esempio di Interplanetary DTN.....	4
Figura 1.2: esempio di VANET.....	5
Figura 1.3: esempio di DTN su fondale oceanico.....	6
Figura 1.4: esempio di rete per comunicazione tra villaggi rurali / città.....	6
Figura 2.1: pseudo-codice relativo al meccanismo di routing Social Cast.....	12
Figura 2.2: mostra il delivery rate all'aumentare del numero di copie $\gamma$ .....	13
Figura 2.3: mostra la latenza media all'aumentare del numero di copie $\gamma$ .....	13
Figura 2.4: mostra il traffico di rete all'aumentare del numero di copie $\gamma$ .....	14
Figura 2.5: mostra il delivery rate all'aumentare del TTL (numero di hops).....	14
Figura 2.6: mostra il traffico di rete all'aumentare del TTL (numero di hops).....	15
Figura 2.7: mostra il delivery rate all'aumentare del numero degli interessi.....	15
Figura 2.8: mostra il traffico di rete all'aumentare del numero degli interessi.....	16
Figura 2.9: mostra l'occupazione del buffer di tutti i nodi.....	16
Figura 2.10: mostra la sharing efficiency al variare del numero di possessori dei contenuti e del tempo di vita della query.....	19
Figura 2.11: mostra la latenza media al variare del numero di possessori dei contenuti.....	20
Figura 2.12: mostra il discovery ratio al variare del numero di possessori dei contenuti e dal tempo di vita della query.....	20
Figura 2.13: mostra un'analisi della CPU del dispositivo a seconda dell'utilizzo dell'applicazione.....	21
Figura 2.14: mostra il consumo di energia del dispositivo a seconda della tecnologia utilizzata (Wi-Fi , GPS, Accelerometro).....	21
Figura 2.15: pseudo-codice del Flooding protocol.....	22
Figura 2.16: pseudo-codice dell' History-Based protocol.....	23
Figura 2.17: mostra la % media del numero di nodi cui un nodo viene a contatto direttamente al variare del range del suo trasmettitore a corto raggio e della velocità.....	24
Figura 2.18: mostra la % media del numero di nodi cui un nodo viene a contatto indirettamente al variare del range del suo trasmettitore a corto raggio e della velocità.....	24
Figura 2.19: mostra la % di pacchetti raccolti dalla Base station all'aumentare del range di copertura di un nodo con vincolo sulla banda e infinita memoria.....	25
Figura 2.20: mostra la % di pacchetti raccolti dalla Base station all'aumentare del range di copertura di un nodo con infinita banda e vincolo sulla memoria.....	25

Figura 2.21: mostra la % di pacchetti raccolti dalla Base station all'aumentare del range di copertura di un nodo con infinita banda e infinita memoria.....	26
Figura 2.22: mostra il consumo energetico medio di ogni nodo al variare del range di copertura e a seconda del protocollo utilizzato.....	26
Figura 2.23: mostra i grafici ottenuti simulando lo scenario 1 utilizzando il protocollo di routing PROPHET e quello Epidemic.....	29
Figura 2.24: mostra la probabilità con cui un nodo si muove all'interno di questo scenario.....	29
Figura 2.25: mostra i grafici ottenuti simulando lo scenario 2 utilizzando il protocollo di routing PROPHET e quello Epidemic.....	30
Figura 3.1: esempio di trasmissione Unicast.....	37
Figura 3.2: esempio di trasmissione Multicast.....	38
Figura 3.3: esempio di trasmissione Geocast.....	39
Figura 4.1: mostra la divisione della rete e la numerazione delle aree.....	56

# Elenco dei Grafici

Grafico 4.1: mostra l'overhead della rete al variare della frequenza dei ping.....	45
Grafico 4.2: mostra il numero di copie medie per messaggio al variare della frequenza dei ping.....	46
Grafico 4.3: mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare della frequenza dei ping.....	46
Grafico 4.4: mostra il numero di ping medi ricevuti per nodo prima dell'invio dei messaggi e il numero di ping medi ricevuti per nodo in totale al variare della frequenza dei ping.....	47
Grafico 4.5: mostra l'overhead della rete al variare del raggio di copertura dei nodi.....	47
Grafico 4.6: mostra il numero di copie medie per messaggio al variare del raggio di copertura dei nodi.....	48
Grafico 4.7: mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare del raggio di copertura dei nodi.....	48
Grafico 4.8: : mostra il numero di ping medi ricevuti per nodo prima dell'invio dei messaggi e il numero di ping medi ricevuti per nodo in totale al variare del raggio di copertura.....	49
Grafico 4.9: mostra l'overhead della rete al variare della frequenza dei ping.....	50
Grafico 4.10: mostra il numero di copie medie per messaggio al variare della frequenza dei ping.....	50
Grafico 4.11: mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare della frequenza dei ping.....	51
Grafico 4.12: mostra il numero di ping medi ricevuti per nodo prima dell'invio dei messaggi e il numero di ping medi ricevuti per nodo in totale al variare della frequenza dei ping.....	51
Grafico 4.13: mostra l'overhead della rete al variare del raggio di copertura dei nodi.....	52
Grafico 4.14: mostra il numero di copie medie per messaggio al variare del raggio di copertura dei nodi.....	52
Grafico 4.15: mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare del raggio di copertura dei nodi.....	53
Grafico 4.16: mostra il numero di ping medi ricevuti per nodo prima dell'invio dei messaggi e il numero di ping medi ricevuti per nodo in totale al variare del raggio di copertura.....	53
Grafico 4.17: mostra l'overhead della rete al variare del numero dei destinatari dei messaggi.....	54
Grafico 4.18: mostra il numero di copie medie per messaggio al variare del numero dei destinatari dei messaggi.....	54
Grafico 4.19: mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare del numero dei destinatari dei messaggi.....	55
Grafico 4.20: mostra il numero di ping medi ricevuti per nodo prima dell'invio dei messaggi e il numero di ping medi ricevuti per nodo in totale al variare del numero dei destinatari dei messaggi.....	55
Grafico 4.21: mostra l'overhead della rete al variare della frequenza dei ping.....	57
Grafico 4.22: mostra il numero di copie medie per messaggio al variare della frequenza dei ping.....	57
Grafico 4.23: mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare della frequenza dei ping.....	58



Grafico 4.24: mostra il numero di ping medi ricevuti per nodo prima dell'invio dei messaggi e il numero di ping medi ricevuti per nodo in totale al variare della frequenza dei ping.....	59
Grafico 4.25: mostra l'overhead della rete al variare del raggio di copertura dei nodi.....	59
Grafico 4.26: mostra il numero di copie medie per messaggio al variare del raggio di copertura dei nodi.....	60
Grafico 4.27: mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare del raggio di copertura dei nodi.....	60
Grafico 4.28: mostra il numero di ping medi ricevuti per nodo prima dell'invio dei messaggi e il numero di ping medi ricevuti per nodo in totale al variare del raggio di copertura.....	61
Grafico 4.29: mostra l'overhead della rete al variare della frequenza dei ping con tutti i messaggi diretti verso l'Area 1.....	61
Grafico 4.30: mostra il numero di copie medie per messaggio al variare della frequenza dei ping con tutti i messaggi diretti verso l'Area 1.....	62
Grafico 4.31: mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare della frequenza dei ping con tutti i messaggi diretti verso l'Area 1.....	62
Grafico 4.32: mostra l'overhead della rete al variare del raggio di copertura dei nodi con tutti i messaggi diretti verso l'Area 1.....	63
Grafico 4.33: mostra il numero di copie medie per messaggio al variare del raggio di copertura dei nodi con tutti i messaggi diretti verso l'Area 1.....	64
Grafico 4.34: mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare del raggio di copertura dei nodi con tutti i messaggi diretti verso l'Area 1.....	64
Grafico 4.35: mostra l'overhead della rete al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri].....	66
Grafico 4.36: mostra il numero di copie medie per messaggio al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri].....	66
Grafico 4.37: mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri].....	67
Grafico 4.38: mostra l'overhead della rete al variare del parametro $\alpha$ con raggio di copertura fisso a 60 [metri].....	67
Grafico 4.39: mostra il numero di copie medie per messaggio al variare del parametro $\alpha$ con raggio di copertura fisso a 60 [metri].....	68
Grafico 4.40: mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare del parametro $\alpha$ con raggio di copertura fisso a 60 [metri].....	68
Grafico 4.41: mostra l'overhead della rete al variare del parametro $\alpha$ con raggio di copertura fisso a 100 [metri].....	69
Grafico 4.42: mostra il numero di copie medie per messaggio al variare del parametro $\alpha$ con raggio di copertura fisso a 100 [metri].....	69
Grafico 4.43: mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare del parametro $\alpha$ con raggio di copertura fisso a 100 [metri].....	70

Grafico 4.44: mostra l'overhead della rete al variare del parametro $\alpha$ con tutti i raggi di copertura espressi in [metri].....	70
Grafico 4.45: mostra la durata totale della simulazione al variare del parametro $\alpha$ con raggio di copertura fisso a 100 [metri].....	71
Grafico 4.46: mostra l'overhead della rete al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri] sia nel caso di 3 destinatari sia nel caso di 6 destinatari.....	72
Grafico 4.47: mostra il numero di copie medie per messaggio al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri] sia nel caso di 3 destinatari sia nel caso di 6 destinatari.....	73
Grafico 4.48: mostra la durata totale della simulazione al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri] sia nel caso di 3 destinatari sia nel caso di 6 destinatari.....	73
Grafico 4.49: mostra il tempo di consegna medio di ogni messaggio al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri] sia nel caso di 3 destinatari sia nel caso di 6 destinatari.....	74
Grafico 4.50: mostra l'overhead della rete al variare del parametro $\alpha$ con raggio di copertura fisso a 60 [metri] sia nel caso di 3 destinatari sia nel caso di 6 destinatari.....	74
Grafico 4.51: mostra il numero di copie medie per messaggio al variare del parametro $\alpha$ con raggio di copertura fisso a 60 [metri] sia nel caso di 3 destinatari sia nel caso di 6 destinatari.....	75
Grafico 4.52: mostra la durata totale della simulazione al variare del parametro $\alpha$ con raggio di copertura fisso a 60 [metri] sia nel caso di 3 destinatari sia nel caso di 6 destinatari.....	75
Grafico 4.53: mostra il tempo di consegna medio di ogni messaggio al variare del parametro $\alpha$ con raggio di copertura fisso a 60 [metri] sia nel caso di 3 destinatari sia nel caso di 6 destinatari.....	76
Grafico 4.54a: mostra l'overhead della rete al variare del parametro $\alpha$ con tutti i raggi di copertura espressi in [metri] nel caso a 3 destinatari.....	76
Grafico 4.54b: mostra l'overhead della rete al variare del parametro $\alpha$ con tutti i raggi di copertura espressi in [metri] nel caso a 6 destinatari.....	77
Grafico 4.55a: mostra la durata totale della simulazione al variare del parametro $\alpha$ con tutti i raggi di copertura espressi in [metri] nel caso a 3 destinatari.....	77
Grafico 4.55b: mostra la durata totale della simulazione al variare del parametro $\alpha$ con tutti i raggi di copertura espressi in [metri] nel caso a 6 destinatari.....	78
Grafico 4.56: mostra l'overhead della rete al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri].....	79
Grafico 4.57: mostra il numero di copie medie per messaggio al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri].....	80
Grafico 4.58: mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri].....	80
Grafico 4.59: mostra l'overhead della rete al variare del parametro $\alpha$ con raggio di copertura fisso a 60 [metri].....	81
Grafico 4.60: mostra il numero di copie medie per messaggio al variare del parametro $\alpha$ con raggio di copertura fisso a 60 [metri].....	81
Grafico 4.61: mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare del parametro $\alpha$ con raggio di copertura fisso a 60 [metri].....	82
Grafico 4.62: mostra l'overhead della rete al variare del parametro $\alpha$ con tutti i raggi di copertura espressi in [metri].....	82

Grafico 4.63: mostra la durata totale della simulazione al variare del parametro $\alpha$ con tutti i raggi di copertura espressi in [metri].....	83
Grafico 4.64a: mostra il numero di Check-In medi di un nodo lineare per area per $\alpha=0.2$ .....	83
Grafico 4.64b: mostra il numero di Check-In medi di un nodo lineare per area per $\alpha=0.4$ .....	84
Grafico 4.64c: mostra il numero di Check-In medi di un nodo lineare per area per $\alpha=0.6$ .....	84
Grafico 4.64d: mostra il numero di Check-In medi di un nodo lineare per area per $\alpha=0.8$ .....	84
Grafico 4.64e: mostra il numero di Check-In medi di un nodo lineare per area per $\alpha=1$ .....	85
Grafico 4.65: mostra l'overhead della rete al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.75$ , $Q = 0.25$ .....	86
Grafico 4.66: mostra il numero di copie medie per messaggio al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.75$ , $Q = 0.25$ .....	86
Grafico 4.67: mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.75$ , $Q = 0.25$ .....	87
Grafico 4.68: mostra la divisione del traffico di tutti i messaggi della rete al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.75$ , $Q = 0.25$ .....	87
Grafico 4.69: mostra l'overhead della rete al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.25$ , $Q = 0.75$ .....	88
Grafico 4.70: mostra il numero di copie medie per messaggio al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.25$ , $Q = 0.75$ .....	89
Grafico 4.71: mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.25$ , $Q = 0.75$ .....	89
Grafico 4.72: mostra la divisione del traffico di tutti i messaggi della rete al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.25$ , $Q = 0.75$ .....	90
Grafico 4.73: mostra l'overhead della rete al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.50$ , $Q = 0.50$ .....	91
Grafico 4.74: mostra il numero di copie medie per messaggio al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.50$ , $Q = 0.50$ .....	91
Grafico 4.75: mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.50$ , $Q = 0.50$ .....	92
Grafico 4.76: mostra la divisione del traffico di tutti i messaggi della rete al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.50$ , $Q = 0.50$ .....	92
Grafico 4.77: mostra l'overhead della rete al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.75$ , $Q = 0.25$ sia a 3 destinatari sia a 6 destinatari.....	94
Grafico 4.78: mostra il numero di copie medie per messaggio al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.75$ , $Q = 0.25$ sia a 3 destinatari sia a 6 destinatari.....	94
Grafico 4.79: mostra la durata totale della simulazione al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.75$ , $Q = 0.25$ sia a 3 destinatari sia a 6 destinatari.....	95
Grafico 4.80: mostra il tempo di consegna medio di ogni messaggio al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.75$ , $Q = 0.25$ sia a 3 destinatari sia a 6 destinatari.....	95

Grafico 4.81a: mostra la divisione del traffico di tutti i messaggi della rete al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.75$ , $Q = 0.25$ nel caso a 3 destinatari.....	96
Grafico 4.81b: mostra la divisione del traffico di tutti i messaggi della rete al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.75$ , $Q = 0.25$ nel caso a 6 destinatari.....	96
Grafico 4.82: mostra l'overhead della rete al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.25$ , $Q = 0.75$ sia a 3 destinatari sia a 6 destinatari.....	97
Grafico 4.83: mostra il numero di copie medie per messaggio al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.25$ , $Q = 0.75$ sia a 3 destinatari sia a 6 destinatari.....	98
Grafico 4.84: mostra la durata totale della simulazione al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.25$ , $Q = 0.75$ sia a 3 destinatari sia a 6 destinatari.....	98
Grafico 4.85: mostra il tempo di consegna medio di ogni messaggio al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.25$ , $Q = 0.75$ sia a 3 destinatari sia a 6 destinatari.....	99
Grafico 4.86a: mostra la divisione del traffico di tutti i messaggi della rete al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.25$ , $Q = 0.75$ nel caso a 3 destinatari.....	99
Grafico 4.86b: mostra la divisione del traffico di tutti i messaggi della rete al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.25$ , $Q = 0.75$ nel caso a 6 destinatari.....	99
Grafico 4.87: mostra l'overhead della rete al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.50$ , $Q = 0.50$ sia a 3 destinatari sia a 6 destinatari.....	100
Grafico 4.88: mostra il numero di copie medie per messaggio al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.50$ , $Q = 0.50$ sia a 3 destinatari sia a 6 destinatari.....	101
Grafico 4.89: mostra la durata totale della simulazione al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.50$ , $Q = 0.50$ sia a 3 destinatari sia a 6 destinatari.....	101
Grafico 4.90: mostra il tempo di consegna medio di ogni messaggio al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.50$ , $Q = 0.50$ sia a 3 destinatari sia a 6 destinatari.....	102
Grafico 4.91a: mostra la divisione del traffico di tutti i messaggi della rete al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.50$ , $Q = 0.50$ nel caso a 3 destinatari.....	102
Grafico 4.91b: mostra la divisione del traffico di tutti i messaggi della rete al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.50$ , $Q = 0.50$ nel caso a 6 destinatari.....	103
Grafico 4.92: mostra l'overhead della rete al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.75$ , $Q = 0.25$ .....	104
Grafico 4.93: mostra il numero di copie medie per messaggio al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.75$ , $Q = 0.25$ .....	105
Grafico 4.94: mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.75$ , $Q = 0.25$ .....	105
Grafico 4.95: mostra la divisione del traffico di tutti i messaggi della rete al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.75$ , $Q = 0.25$ .....	106
Grafico 4.96: mostra l'overhead della rete al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.25$ , $Q = 0.75$ .....	107
Grafico 4.97: mostra il numero di copie medie per messaggio al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.25$ , $Q = 0.75$ .....	107
Grafico 4.98: mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.25$ , $Q = 0.75$ .....	108

Grafico 4.99: mostra la divisione del traffico di tutti i messaggi della rete al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.25$ , $Q = 0.75$ .....	108
Grafico 4.100: mostra l'overhead della rete al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.50$ , $Q = 0.50$ .....	109
Grafico 4.101: mostra il numero di copie medie per messaggio al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.50$ , $Q = 0.50$ .....	110
Grafico 4.102: mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.50$ , $Q = 0.50$ .....	110
Grafico 4.103: mostra la divisione del traffico di tutti i messaggi della rete al variare del parametro $\alpha$ con raggio di copertura fisso a 30 [metri], $P = 0.50$ , $Q = 0.50$ .....	111
Grafico 4.104: mostra l'overhead di rete nel caso dei 3 protocolli Unicast dove ogni nodo invia un messaggio.....	112
Grafico 4.105: mostra la durata di simulazione nel caso dei 3 protocolli Unicast dove ogni nodo invia un messaggio.....	113
Grafico 4.106: mostra l'overhead di rete nel caso dei 3 protocolli Multicast dove ogni nodo invia un messaggio con 6 Destinatari diversi.....	114
Grafico 4.107: mostra la durata di simulazione nel caso dei 3 protocolli Multicast dove ogni nodo invia un messaggio con 6 Destinatari diversi.....	115
Grafico 4.108: mostra l'overhead di rete nel caso dei 3 protocolli Geocast dove ogni nodo invia un messaggio.....	116
Grafico 4.109: mostra la durata di simulazione nel caso dei 3 protocolli Geocast dove ogni nodo invia un messaggio.....	116

## **Sommario**

Con l'aumento dei dispositivi mobili quali smartphones, lo scambio di contenuti tra utenti sta diventando sempre maggiore; si cerca così di trovare dei meccanismi di routing che permettano lo scambio di informazioni tra utenti nel modo più efficiente possibile, sia in termini di tempo sia in termini di traffico. Infatti in queste reti, chiamate DTN (Delay Tolerant Network), la connettività tra tutti i dispositivi non è sempre garantita a causa della mobilità dei nodi e dunque spetta all'ingegnere trovare meccanismi di routing efficienti e che permettano una completa e corretta trasmissione dei file. Perciò in questo elaborato sono state analizzate diverse metriche di routing: alcune basate sui contatti fisici tra i nodi, altre sulle informazioni/interazioni sociali dei nodi stessi al fine di inviare correttamente i messaggi da una sorgente a un destinatario. Sono state fatte diverse simulazioni e sono stati mostrati i risultati al fine di avere un quadro d'insieme dei vari protocolli.

## Introduzione

Lo scopo dell'ingegnere e di chi si occupa delle comunicazioni è quello di cercare di ottimizzare il proprio lavoro cercando di ottenere comunicazioni sempre efficienti; in particolare, lo studio presente in questo elaborato si è concentrato su una specifica tipologia di reti di comunicazione: le DTN (Delay Tolerant Networks). Si tratta di reti tolleranti ai ritardi dove non sempre esiste un percorso diretto tra il nodo sorgente e il nodo destinatario.

Sono stati fatti diversi studi su tali reti, sviluppando le più svariate applicazioni: da quelle automobilistiche per segnalare eventuali incidenti/code a quelle satellitari per le comunicazioni a distanze molto elevate.

Il principale scopo delle DTN consiste nel far arrivare a destinazione tutti i pacchetti della rete; dunque il principale problema è proprio quello di creare un meccanismo di instradamento. Questo meccanismo di routing prevede la creazione di diverse copie dei vari messaggi da inoltrare ad altri nodi della rete che contribuiscono a far arrivare il messaggio al destinatario. Tale meccanismo dovrà tener conto di alcuni fattori come l'overhead, al fine di non saturare la rete creando un numero molto alto di copie, e il ritardo medio dei pacchetti cercando di averlo il minore possibile.

Lo scopo di questo elaborato di tesi è quello di sviluppare particolari protocolli di routing nelle reti DTN nei vari casi di comunicazioni Unicast, Multicast e Geocast; sono state create metriche di routing fisiche basate sul numero dei ping ricevuti dai nodi, metriche di routing applicative basate sulle amicizie di Facebook e sul numero di Check-In effettuati con Foursquare dai nodi; infine sono state unite le due metriche (fisiche e applicative) creando un protocollo ibrido che sfruttasse ogni informazione dei nodi della rete.

Per analizzare le prestazioni di tali protocolli è stato necessario utilizzare un simulatore ad eventi discreti scritto in C++: OMNeT++; attraverso tale simulatore è stato implementato, e in seguito simulato, ogni protocollo di routing analizzando i dati più importanti come il tempo di simulazione, l'overhead di rete, il ritardo medio e il numero medio di copie per messaggio.

La tesi è suddivisa in 4 capitoli: nel primo capitolo viene fatta un'introduzione alle DTN portando la definizione, alcuni esempi generali del loro utilizzo e i fattori caratterizzanti come il tempo e l'overhead; nel secondo capitolo viene trattato l'argomento del routing in tali reti; inoltre vengono presentati diversi scenari di DTN:

publish/subscribe, content sharing, wildlife tracking, etc; di ogni scenario viene mostrato il protocollo di routing utilizzato con i relativi risultati; nel terzo capitolo vengono riportate le metriche di routing per i protocolli che si andranno a simulare: a seconda del tipo di comunicazione utilizzato (Unicast, Multicast e Geocast) ci saranno determinate funzioni obiettivo per l'instradamento dei messaggi; nel quarto capitolo verranno mostrati i risultati, sottoforma di grafici, relativi alle simulazioni dei protocolli implementati; verrà inoltre fatto un confronto fra i risultati ottenuti mediante metriche diverse (ad esempio i risultati ottenuti con protocollo Fisico verranno confrontati con quelli del protocollo Applicativo e poi Ibrido). Alla fine del Capitolo 4 vi saranno due pagine relative alla conclusioni di tale elaborato.



# Capitolo 1 – Reti opportunistiche (DTN)

## 1.1 - Definizione

Per DTN si intende un'architettura di rete di telecomunicazioni che si propone come obiettivo la comunicazione tra reti indipendenti, mutuamente incompatibili e non appartenenti a Internet (definita in questo contesto come una rete globale continuamente interconnessa); si tratta di architetture in continuo sviluppo al fine di ottimizzare le comunicazioni tra i vari nodi che le compongono.

## 1.2 - Scenari applicativi

Gli scenari applicativi cui si prestano tali reti sono molteplici e tra queste possiamo sicuramente citare:

- Reti interplanetarie (vedi Figura 1.1): si tratta di reti di dimensioni molto grandi che possono raggiungere anche i 20 milioni di chilometri; si utilizzano queste architetture per la comunicazione tra la Terra e i dispositivi nello spazio (ad esempio i satelliti, le stazioni spaziali, le sonde interplanetarie) o per la comunicazione diretta tra i dispositivi stessi con lo scopo di inviare immagini dello Spazio o scambiarsi informazioni particolari (ad esempio eventuali guasti, errori, etc.); lo sviluppo di tali reti è gestito principalmente dalla NASA.

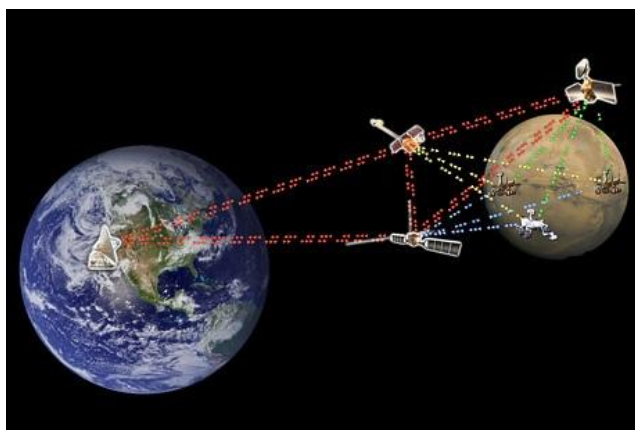
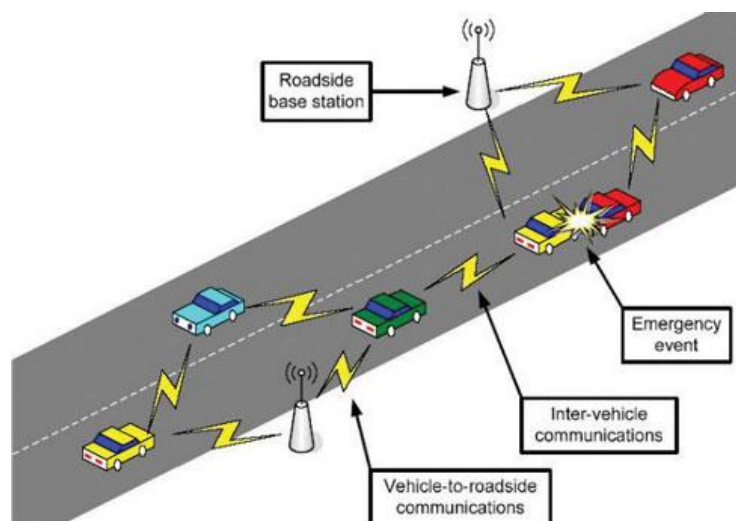


Figura 1.1: esempio di Interplanetary DTN

- Reti VANET (Vehicular Ad-hoc Network - vedi Figura 1.2) [1] [2]: si tratta di un particolare tipo di rete MANET (Mobile Ad-hoc Network), formata da veicoli dotati di dispositivi con tecnologie di comunicazione wireless a medio raggio (es. schede wireless IEEE 802.11 a/b/g); la comunicazione è limitata a veicoli posti nello stesso raggio di trasmissione, oppure può coinvolgere veicoli posti a più hop di distanza mediante la presenza di infrastrutture fisse oppure di router mobili (i veicoli stessi); tali reti possono essere utili ad esempio al fine di informare il conducente alla guida del veicolo di un eventuale incidente/rallentamento sulla strada che sta percorrendo e quindi di utilizzare un percorso alternativo.



**Figura 1.2: esempio di VANET**

- Reti sociali short-range: si tratta di reti formate da diversi nodi che rappresentano i singoli soggetti / persone; infatti ogni nodo ha a disposizione un dispositivo (es. smartphone) che permette la comunicazione con altri dispositivi entro un piccolo raggio di copertura; di solito tali reti hanno un'area abbastanza ridotta (di circa 4-5 chilometri quadrati) ma possono essere utilizzate anche in ambienti più grandi. In tali reti i dispositivi si scambiano informazioni come email, foto e video, oppure possono far venire a conoscenza un determinato gruppo di persone (sempre identificate da un dispositivo) di un particolare evento (es. un concerto o una partita) o prodotto facendone pubblicità. La comunicazione può avvenire direttamente (se i dispositivi sono abbastanza vicini) o passare attraverso altri nodi (se il destinatario non “rientra” nel raggio di copertura della sorgente).

- Reti per monitoraggio ambientale [3][4]: si tratta di reti costituite da diversi nodi e che possono raggiungere dimensioni molto elevate (anche migliaia di chilometri); si usano spesso per monitorare le correnti marine sui fondali oceanici (vedi Figura 1.3) ma vi è anche un impiego in zone deserte a scopo sia ambientale (misurare i dati ambientali) sia militare.

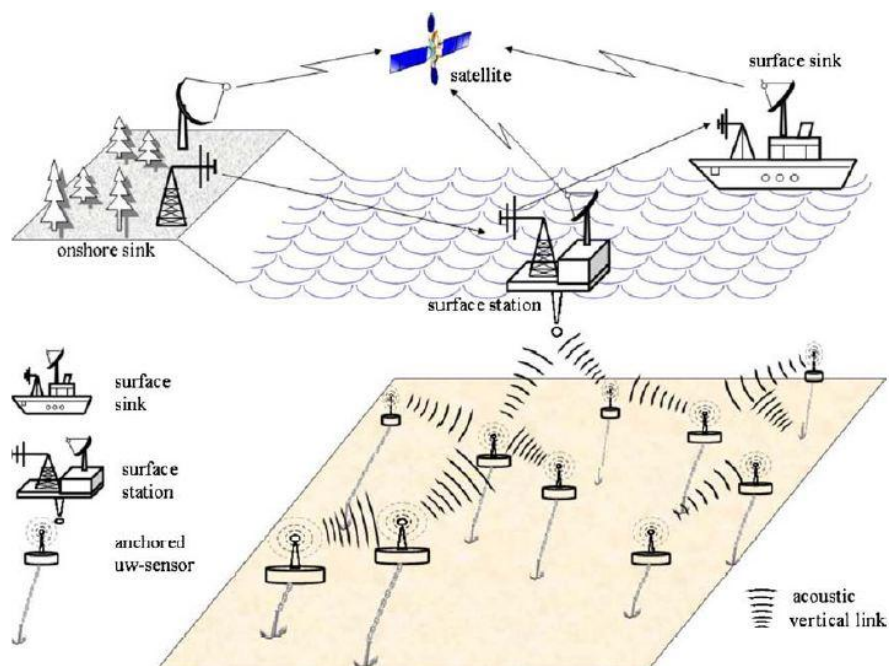


Figura 1.3: esempio di DTN su fondale oceanico

- Reti per comunicazione tra villaggi rurali e città (vedi Figura 1.4)[5]: reti costruite soprattutto in zone povere (ad es. Africa) al fine di far comunicare villaggi distanti senza una connessione Internet con le città.

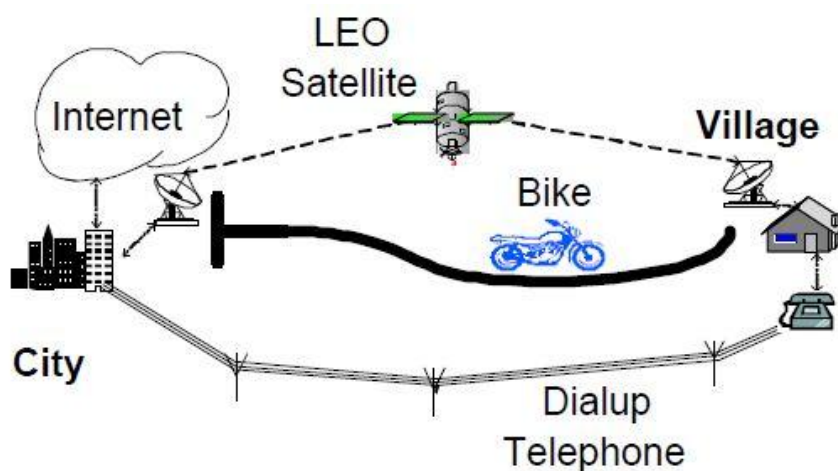


Figura 1.4: esempio di rete per comunicazione tra villaggi rurali / città

### 1.3 - Fattori caratterizzanti

Esistono diversi fattori che possono influenzare le prestazioni all'interno delle DTN e che un progettista deve considerare:

- **Mobilità:** ogni nodo della rete rappresenta un oggetto (es. satellite) o un soggetto (es. persona o animale) e quindi ha una propria “vita”; con il passare del tempo ogni nodo compie un preciso movimento dovuto ai suoi particolari interessi, bisogni o compiti; in tali reti, quindi, la mobilità ha un'importanza fondamentale in quanto in un preciso istante due nodi possono essere vicini e liberi di comunicare mentre nell'istante successivo possono essere distanti a seguito dei loro pattern comportamentali o ruoli specifici all'interno della rete stessa. Esisteranno dunque nodi che per caratteristiche analoghe (es. appartenere allo stesso club, frequentare la stessa università etc.) avranno un modello di mobilità molto simile e quindi la possibilità di comunicare tra di loro in qualsiasi momento della giornata. Anche se due nodi hanno un modello di mobilità diverso possono comunque incrociarsi, seppure per brevi periodi di tempo, durante il corso della giornata e, nel caso volessero scambiarsi file di piccole dimensioni, potrebbero utilizzare la comunicazione diretta oppure altri nodi con pattern comportamentali simili al destinatario del messaggio. Riassumendo, la mobilità dei nodi inserisce in tali reti un forte elemento di incertezza della comunicazione point-to-point in quanto non esiste sempre un canale diretto tra sorgente e destinazione se questi hanno compiti/ruoli diversi all'interno della rete.
- **Routing/instradamento:** questo aspetto deriva proprio dalla mobilità dei nodi; come già detto, nelle DTN non esiste sempre un collegamento diretto tra mittente e destinatario e dunque il pacchetto, a volte, dovrà attraversare altri nodi prima di riuscire ad arrivare a destinazione (comunicazione multi-hop). Spetterà dunque al progettista di rete il compito di trovare delle funzioni obiettivo che ogni nodo dovrà valutare prima di inviare un messaggio al fine di farlo arrivare alla corretta destinazione.
- **Trasmissione dati:** ogni utente, attraverso il proprio terminale, ha a disposizione diversi metodi per inviare un file:

- i. può utilizzare una connessione Bluetooth che permette una comunicazione a corto raggio (10 m) a velocità moderata (fino a 723,1 kbit/sec) con basso consumo energetico;
  - ii. può utilizzare una connessione WI-FI che offre una copertura e una velocità superiori rispetto al Bluetooth (100 m a diversi Mb/sec) ma con un consumo energetico maggiore;
  - iii. può utilizzare ZigBee che permette di avere un raggio di copertura medio (20 m) e una velocità pari a 250 kb/s.
- Consumo energetico: ogni nodo ha un dispositivo (es. smartphone) che permette l'invio/ricezione dei messaggi e l' "ascolto" di altri dispositivi entro un determinato raggio; tutte queste azioni portano ad un esaurimento progressivo della batteria ed è compito del progettista far sì che ciò non influenzi le prestazioni del nodo e quindi della rete in generale.
  - Buffer di memoria: ogni nodo può trasportare messaggi (immagini, email, dati etc.) ma il progettista deve sempre tenere in considerazione l'occupazione della memoria del dispositivo; infatti in tali reti si potranno utilizzare tecniche per la gestione del buffer come la possibilità di inserire un TTL al messaggio in modo tale che se entro un determinato tempo l'informazione non è arrivata a destinazione allora il nodo procederà all'eliminazione del messaggio stesso.
  - Terminali cooperativi / non cooperativi: ogni nodo può decidere se trasportare messaggi oppure no verso gli altri nodi; in tali reti è di fondamentale importanza, al fine di ottenere una comunicazione efficiente, che ogni nodo sia cooperativo, in modo che possa trasportare qualsiasi pacchetto ricevuto verso altri utenti della rete stessa.
  - Velocità terminali: ogni nodo viaggia a una propria velocità; ad esempio se un nodo rappresenta una persona, e tale persona è a piedi, avrà una velocità media di 1-2 [m/s]; se tale persona invece fosse in bicicletta avrebbe una velocità pari a 15 [m/s] e si muoverebbe dunque in maniera più rapida, trasportando i pacchetti più velocemente; il progettista deve tener presente anche questo fattore nel formulare le metriche di routing della DTN considerata.

## **Capitolo 2 – Instradamento**

Lo scopo del progettista è garantire che, in tali reti, ogni pacchetto arrivi alla destinazione desiderata, in modo diretto o attraverso altri nodi; quello che si deve tenere in considerazione sono le diverse metriche di routing che possono incidere sulle prestazioni, sia in termini di latenza di consegna del pacchetto sia in termini di overhead di rete. Una DTN è tanto efficiente quanto più bassi sono la latenza e l'overhead di rete; dunque trovare le metriche di routing adatte è un compito spesso difficile e in continua evoluzione a causa dei forti dinamismi delle reti stesse (si ricordi la componente fondamentale: la mobilità).

### **2.1 - Metriche di routing**

Per metriche di routing si intendono quali aspetti un progettista tiene in considerazione per decidere se un nodo dovrà inoltrare o no il file considerato; non esistono standard per tali metriche in quanto dipendono molto dalla specifica DTN che si sta considerando.

Esistono metriche che si basano sul numero di contatti fisici ricevuti, sui tag e sul TTL (Time To Live) del messaggio, sui gruppi di appartenenza di un nodo (es. club) e tanti altri; molte metriche sono composte anche da fattori di probabilità in quanto in una DTN la mobilità dei nodi non è sempre lineare e possono capitare “imprevisti” che potrebbero far deviare il percorso di un nodo e quindi cambiare la morfologia della rete stessa. Se un nodo vede rispettata la funzione obiettivo impostata dal progettista, allora invierà il messaggio al nodo vicino considerato. Ogni nodo è costituito da un terminale in grado di svolgere calcoli per verificare la funzione obiettivo.

### **2.2 - Scenari di routing particolari**

Come detto in precedenza, ogni DTN ha un proprio scopo e dunque metriche di routing diverse a seconda della funzione dei nodi stessi. Nel corso degli anni sono stati studiati diversi casi particolari di DTN di cui ora riporterò qualche esempio.

### 2.2.1 - Applicazione Publish / Subscribe

Lo scenario analizzato per questo studio [6] è un'area di 16 km<sup>2</sup> dove tutti i nodi sono cooperativi e hanno una loro mobilità che impedisce un continuo / diretto collegamento tra tutti i nodi della rete. La metà dei nodi presenti hanno la funzione di "editore", ovvero inviano un messaggio contraddistinto da un preciso tag e con un preciso TTL basato sul numero di hops, al fine di evitare la saturazione della rete; ogni nodo è abbonato ad uno o più interessi e tali interessi sono identificati da un tag; se c'è una corrispondenza tra il tag dell'interesse e il tag del messaggio allora tale messaggio verrà indirizzato verso quel preciso nodo. Come regola di associazione è stato imposto che nodi con interessi comuni abbiano anche comportamenti (identificati dalla mobilità) simili e dunque vengano in contatto più spesso rispetto ad altri. Ogni nodo  $n$  ha una propria funzione di utilità  $U_i$  con  $i \in A$  dove  $A$  è l'insieme degli interessi; tale funzione esprime quanto il nodo  $n$  sia un buon portatore per l'interesse  $i$  a seconda dei suoi schemi di movimento e la sua collocazione rispetto agli altri nodi. Per formulare la funzione di utilità bisogna tenere a mente alcuni fattori:

- la probabilità di un utente di essere collocato nella stessa zona di un utente con lo stesso interesse;
- il Change Degree of Connectivity (CDoC) ovvero se un utente incontra tante persone allora avrà maggiore probabilità di inviare il messaggio e dunque tanto più il CDoC è elevato tanto più il nodo considerato cambierà l'insieme dei suoi vicini.

L'importante è che la futura funzione di utilità venga stimata correttamente e per fare ciò si usano tecniche basate sul filtro di Kalman (efficiente filtro ricorsivo che valuta lo stato di un sistema dinamico a partire da una serie di misure soggette a rumore) che sfruttano il fatto che i percorsi di utenti con il medesimo interesse non sono casuali ma seguono precisi schemi basati sulla loro rete sociale; tale filtro prende in input i valori al tempo " $t$ " e da come output una stima di tali valori al tempo " $t + \tau$ ".

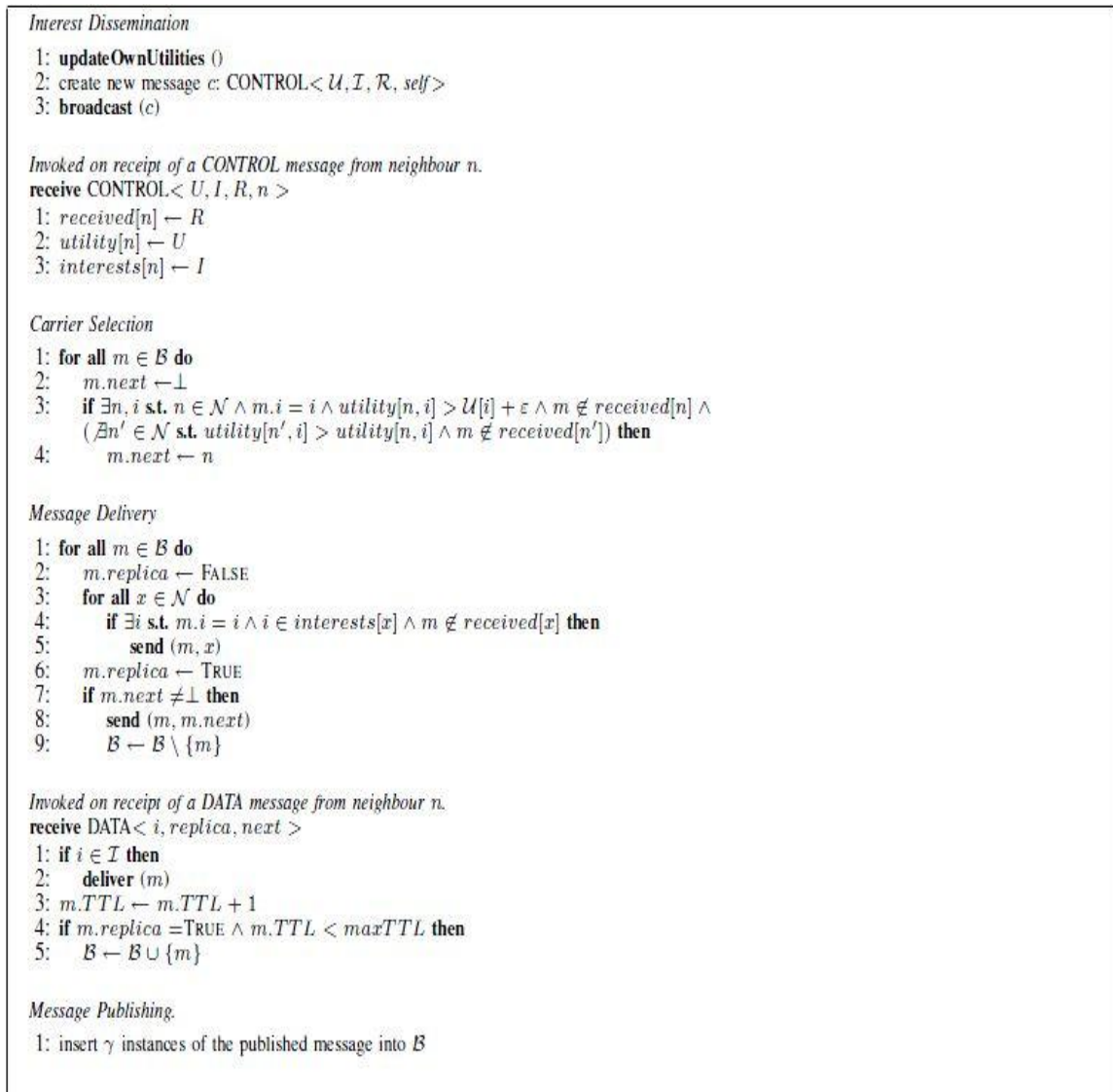
Il meccanismo di routing utilizzato in questo scenario è chiamato SocialCast ed è composto da 3 fasi:

1. Interest Dissemination: ogni nodo invia a tutti i suoi vicini la lista dei propri interessi con i rispettivi valori di utilità (calcolati in base alla posizione dei nodi appena prima di tale fase).

2. Carrier Selection: l'utilità del nodo locale,  $U_i$ , è calcolata per tutti gli interessi  $i$ ; tale utilità viene comparata con la più alta dei suoi vicini,  $U_{n,i}$ . Se  $U_{n,i} > U_i$  allora il nodo  $n$  è un miglior portatore rispetto al nodo locale, altrimenti quest'ultimo, che possiede il messaggio, rimarrà il miglior portatore.
3. Message Dissemination: durante tale fase ogni nodo aggiorna il proprio contenuto del buffer e i messaggi (insieme alle loro  $\gamma$  copie) vengono inviati ai nodi interessati o ai migliori portatori; se un nodo porta un messaggio con tag  $i$  ed ha dei vicini con interesse  $i$ , allora invierà subito loro tale messaggio; la funzione di Buffer Insertion (inserimento del nuovo messaggio nel buffer) viene eseguita solo dai migliori portatori e, se un nodo locale incontra un altro nodo che è miglior portatore per quel messaggio, allora il nodo locale invierà il messaggio a tale nodo e lo rimuoverà dal proprio buffer.

Tale meccanismo di routing può essere rappresentato anche sottoforma di pseudo-codice (vedi Figura 2.1):





**Figura 2.1:** pseudo-codice relativo al meccanismo di routing Social Cast

Nella parte di simulazione si assume che la rete sia composta da comunità formate in base agli interessi e che i nodi di bordo di ogni comunità possono comunicare con altri nodi di bordo di ogni comunità con una probabilità  $p$ . Si ottiene in questo modo una matrice quadrata che rappresenta le interazioni sociali tra i vari nodi; la rete viene dunque divisa in quadrati e in ognuno di essi viene posta una comunità; dopodiché ogni host si muoverà in linea retta all'interno del quadrato secondo un proprio obiettivo; una volta raggiunto il proprio obiettivo, il nodo ne seleziona un altro a seconda del gruppo di host in cui si trova e si muoverà verso di esso.

Vengono così analizzate le prestazioni di tale rete, sia con il protocollo Social Cast sia con un protocollo di routing senza predizione, misurando alcuni valori come:

- delivery rate

- latenza media
- network traffic

Facendo variare alcuni parametri come:

- numero di copie  $\gamma$  (vedi Figura 2.2 - 2.3 - 2.4):

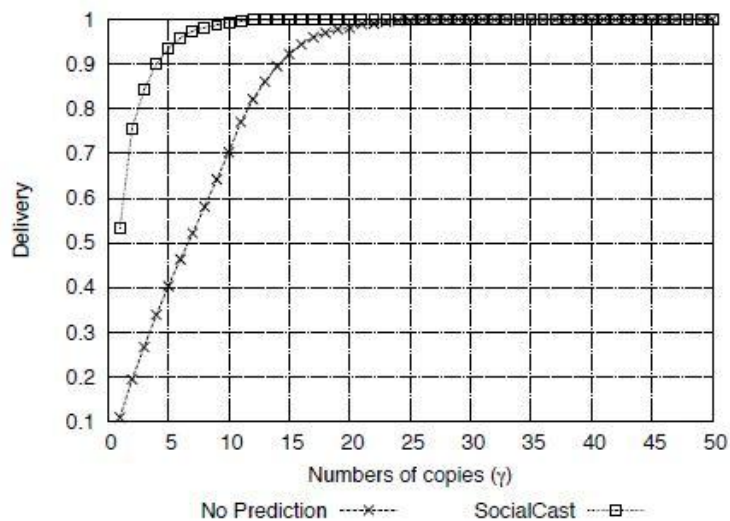


Figura 2.2: mostra il delivery rate all'aumentare del numero di copie  $\gamma$

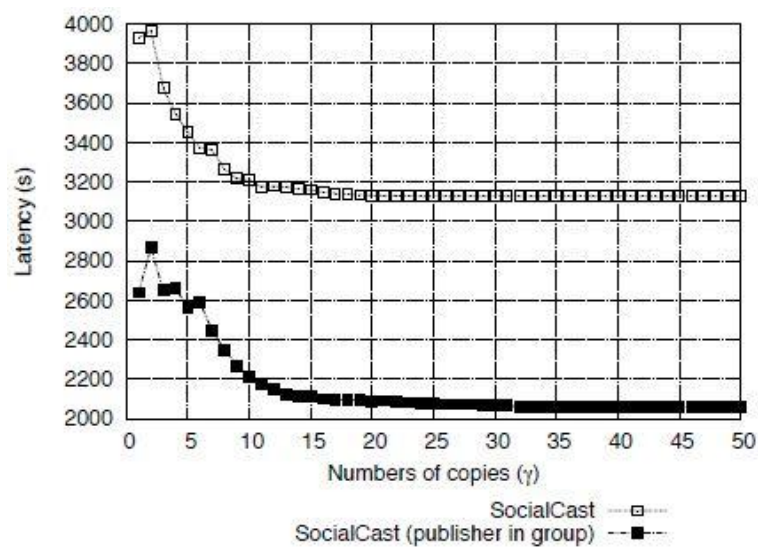


Figura 2.3: mostra la latenza media all'aumentare del numero di copie  $\gamma$

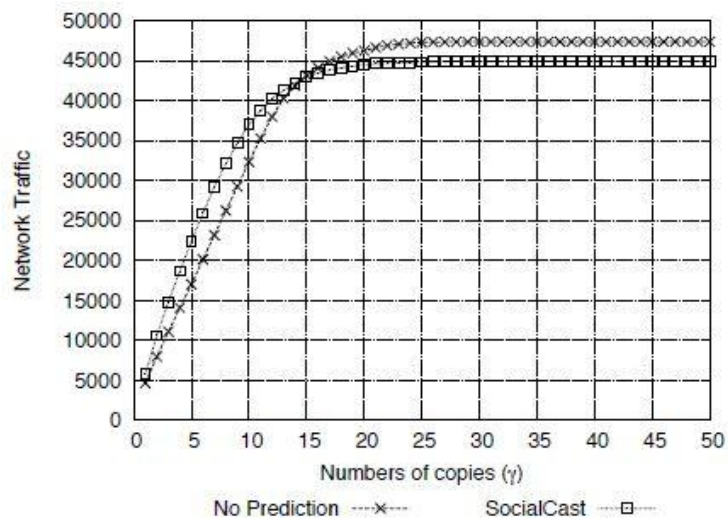


Figura 2.4: mostra il traffico di rete all'aumentare del numero di copie  $\gamma$

- TTL (Time To Live) (vedi Figura 2.5 - 2.6):

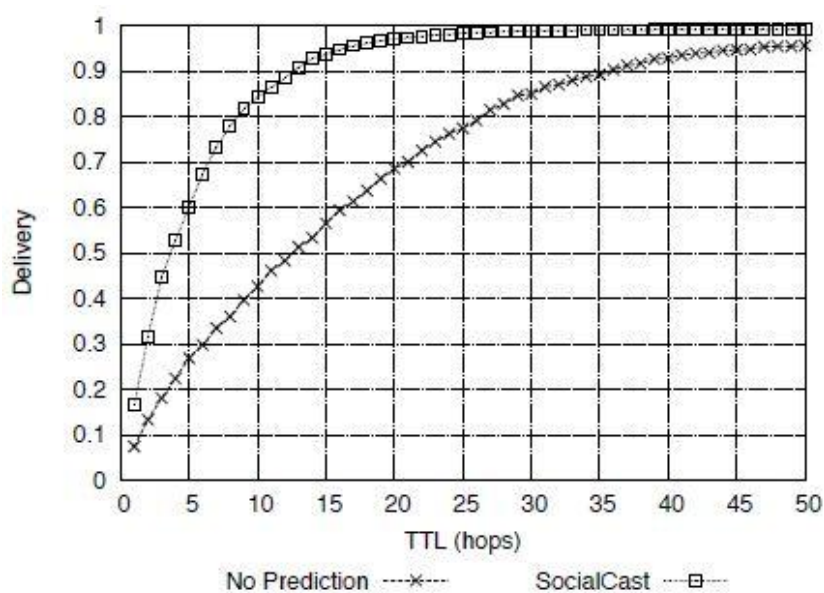


Figura 2.5: mostra il delivery rate all'aumentare del TTL (numero di hops)

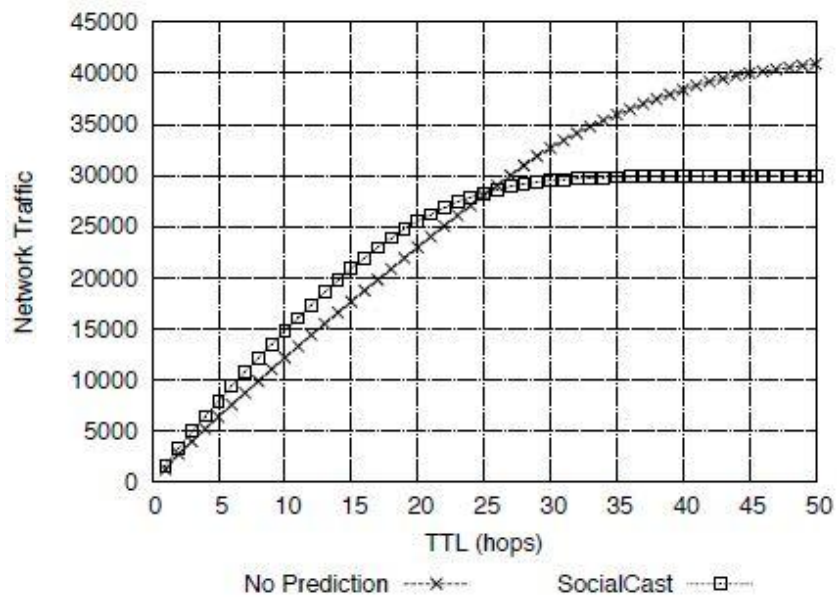


Figura 2.6: mostra il traffico di rete all'aumentare del TTL (numero di hops)

- Numero degli interessi (vedi Figura 2.7 - 2.8):

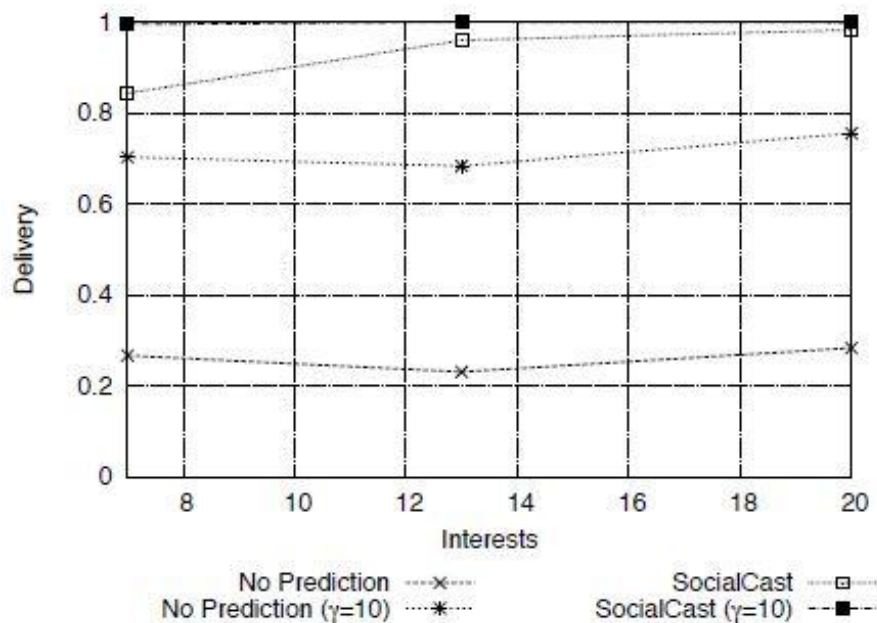


Figura 2.7: mostra il delivery rate all'aumentare del numero degli interessi

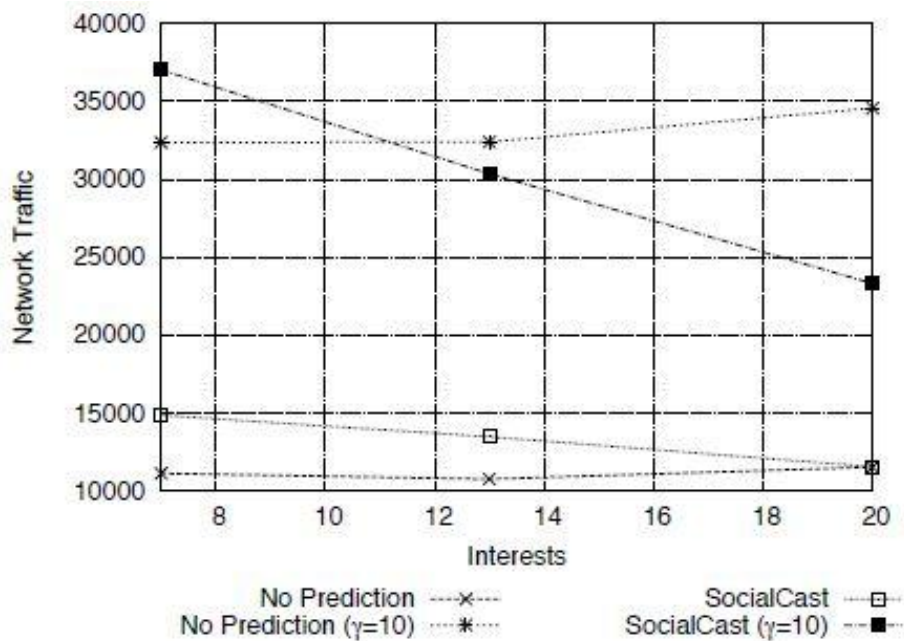


Figura 2.8: mostra il traffico di rete all'aumentare del numero degli interessi

- Buffer size (vedi Figura 2.9):

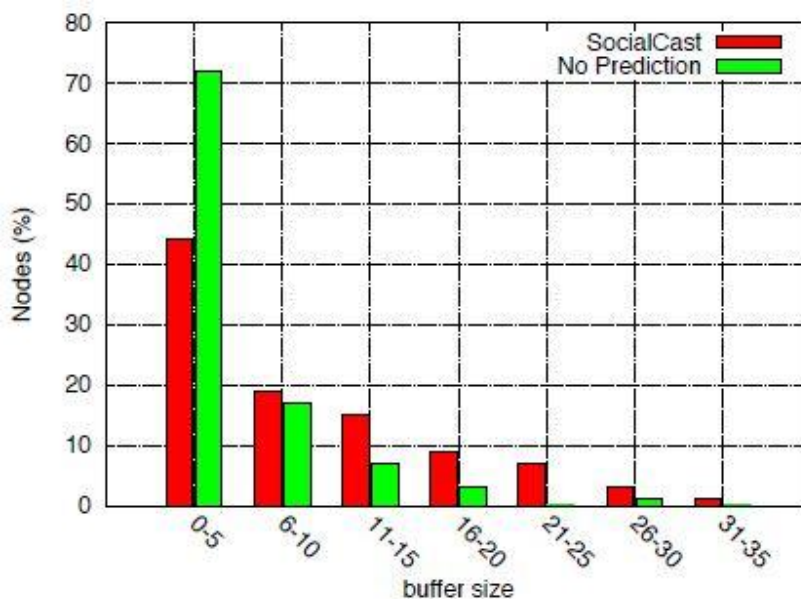


Figura 2.9: mostra l'occupazione del buffer di tutti i nodi

### 2.2.2 - Applicazione Content Sharing

Lo scenario analizzato per tale applicazione [7] è un'area di 25 km<sup>2</sup> dove tutti i nodi sono cooperativi; tale applicazione serve per far arrivare ad un utente un determinato contenuto dopo che questo lo ha richiesto attraverso una query. Si genera così il Content Sharing Process o DPD (Discover-Predict-Deliver) che è formato da 2 fasi:

1. Content Discovery phase: un utente fa una richiesta tramite un'applicazione del proprio smartphone e, se tale contenuto non è presente nella memoria locale, viene formulato un messaggio di query che verrà diffuso all'interno della rete; ogni messaggio di query inviato contiene il tempo di creazione della query ( $t_{qc}$ ), il tempo di vita ( $T$ ), l'ID del nodo che l'ha generata e le sue informazioni di mobilità. Ogni nodo trasmette il messaggio in modo indipendente e per limitare l'invio delle query si possono stabilire due condizioni da rispettare contemporaneamente:

I. Split query lifetime:  $\begin{cases} t - t_{qc} < (1 - \alpha)T & \text{query inviata} \\ t - t_{qc} > (1 - \alpha)T & \text{query non inviata} \end{cases}$  dove  $\alpha$  è il rapporto tra il tempo di ricerca del contenuto e il tempo di consegna/invio del contenuto;

II. Query distance limit:  $\begin{cases} |l_{f,t} - l_{q,t}| < H & \text{query inviata} \\ |l_{f,t} - l_{q,t}| > H & \text{query non inviata} \end{cases}$  dove  $l_{f,t}$  è la localizzazione del nodo  $f$  (possessore del contenuto) al tempo  $t$ ,  $l_{q,t}$  è la localizzazione del nodo  $q$  (quello che ha creato la query) al tempo  $t$ ,  $H$  è la distanza massima tra i due nodi per scambiarsi il contenuto.

Una volta che il contenuto è stato trovato, si può passare alla seconda fase.

2. Content Delivery phase: ogni nodo invia in broadcast un messaggio di beacon per informare gli altri nodi della rete della propria esistenza; in tale messaggio è presente anche una lista delle queries e una lista dei contenuti che il nodo sta trasportando; sia le queries sia i contenuti includono l'informazione di mobilità del nodo che ha generato le queries stesse. Al fine di limitare il numero di copie del contenuto si usa il protocollo di routing chiamato Spray And Focus, composto da due fasi:

I. Spray phase: per ogni messaggio originato dal nodo sorgente, ne saranno inviate nella rete  $L$  copie.

II. Focus phase: un nodo  $i$  passa una copia del messaggio con destinazione  $d$  ad un nodo  $j$  se:  $U_j(d) > U_i(d)$ ; tale funzione di utilità viene calcolata in questo modo:  $U_j(d) = \sum_{m=t}^{t+k\delta} U_m$  dove

$$U_m = \begin{cases} \frac{t}{m} & \text{se } |l_{i,m} - l_{d,m}| \leq H \\ 0 & \text{altrimenti} \end{cases}$$

,  $t$  è il tempo corrente,  $k = \lceil T/\delta \rceil$  è il numero di istanze di tempo rimanenti prima che la query “muoia”,  $T$  il tempo di vita rimanente della query.

Il Content Sharing è possibile solo se 2 smartphones rimangono nello stesso luogo per un determinato periodo di tempo in modo da “sentirsi” e poter scambiare il file: questi luoghi prendono il nome di “meaningful places” (o luoghi significativi).

Per ogni nodo un Activity Manager monitora il vettore dell’accelerazione sui tre assi, trova i movimenti dell’utente e fornisce come output due possibili valori: Moving o Stationary; nella pratica dunque in input si ha un vettore delle accelerazioni  $\vec{v}_t$  al tempo  $t$  e l’Activity Manager monitora tale vettore; siano:

$\mu(\vec{v}_{t-W}, \dots, \vec{v}_t)$  la media dell’accelerazione di una finestra temporale di durata  $W$ ,

$\sigma(\vec{v}_{t-W}, \dots, \vec{v}_t)$  la varianza dell’accelerazione di una finestra temporale di durata  $W$ .

Se  $\sigma(\vec{v}_{t-W}, \dots, \vec{v}_t)$  è maggiore di una certa soglia allora si rivela l’attività di movimento.

Inoltre viene utilizzato un particolare algoritmo di “peak detection” che permette di evidenziare il possibile nodo in movimento che afferma:

se  $\vec{v}_t \geq \mu(\vec{v}_{t-W}, \dots, \vec{v}_t) + \sigma(\vec{v}_{t-W}, \dots, \vec{v}_t)$  allora  $\vec{v}_t$  viene preso come picco (peak).

Esistono alcuni problemi come le vibrazioni e gli ondeggiamenti del dispositivo e per far fronte a ciò si effettua una comparazione del *max axis* e del *forward axis* del picco corrente con quelli del tempo precedente. In questo modo è possibile venire a conoscenza della “stabilità di orientamento” di un utente in movimento.

Dei tre assi dell’accelerazione, il *max axis* è quello perpendicolare al terreno e si calcola:

*max axis* =  $\arg \max \{v_{x,t}, v_{y,t}, v_{z,t}\}$  dove:

$$v_{x,t} = \mu(|v_{x,t-W}|, \dots, |v_{x,t}|),$$

$$v_{y,t} = \mu(|v_{y,t-W}|, \dots, |v_{y,t}|),$$

$$v_{z,t} = \mu(|v_{z,t-W}|, \dots, |v_{z,t}|).$$

Il valore del *max axis* viene preso come il valore di cardinalità perché il movimento di una persona non può generare un’accelerazione maggiore di quella della gravità.

Il *forward axis* è l’asse che più si avvicina alla direzione di movimento dell’utente e si determina come:

*forward axis* =  $\arg \max \{\widehat{v}_1, \widehat{v}_2\}$  dove:

$$\widehat{v}_1 = \sigma(v_{1,t-W}, \dots, v_{1,t}),$$

$$\widehat{v}_2 = \sigma(v_{2,t-W}, \dots, v_{2,t})$$

sono le varianze lette dall'accelerazione degli altri due assi (escluso il *max axis*).

Se questi due assi rimangono invariati si deduce che l'utente non ha cambiato direzione / orientamento rispetto al tempo precedente.

Quando un user rimane oltre una soglia di tempo  $\delta$  in un luogo (e lo si capisce attraverso l'Activity Manager che analizza il vettore delle accelerazioni), allora tale luogo prende il nome di "meaningful place"; quando viene trovato un nuovo meaningful place " $l_{new}$ ", si pone il problema della classificazione dei vari livelli (ovvero ci si potrebbe trovare in un palazzo a più piani); se abbiamo dunque  $l_{new}$  e  $l_k$  si applica in questo caso un metodo preciso, il *BSSID*: se nel nostro caso si prende  $A_k$  come il numero di APs trovati in  $l_k$  e con  $A_{new}$  il numero di APs trovati in  $l_{new}$ , si ha:

$$BSSID = \frac{A_k \cap A_{new}}{A_k + A_{new}};$$

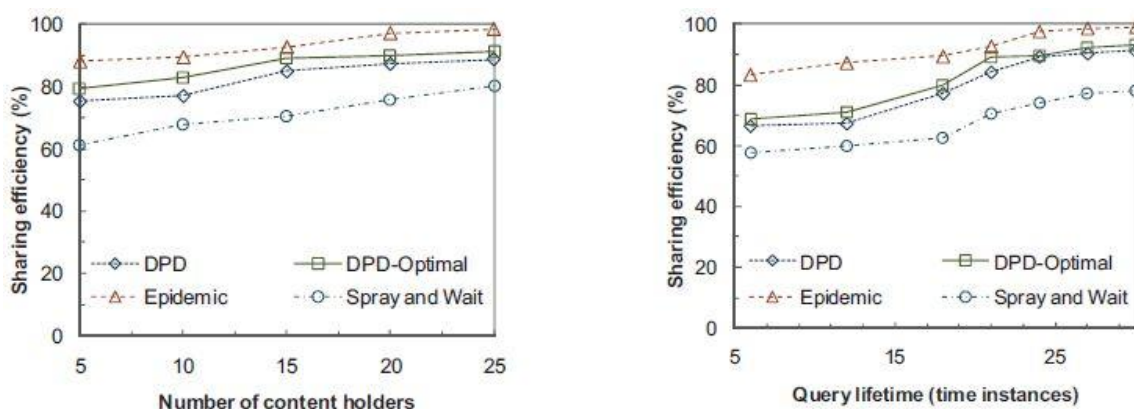
se  $BSSID < 0.6$  allora i due posti sono considerati diversi.

Dopo viene eseguita una parte di simulazione utilizzando diversi algoritmi di routing, tra cui l'epidemic routing e il Content Sharing (o DPD), dove vengono misurati alcuni valori principali come:

- sharing efficiency (vedi Figura 2.10)
- latenza media (vedi Figura 2.11)
- discovery ratio (vedi Figura 2.12)

Facendo variare alcuni parametri come:

- numero di possessori dei contenuti
- tempo di vita della query ( $T$ )



**Figura 2.10:** mostra la sharing efficiency al variare del numero di possessori dei contenuti e del tempo di vita della query



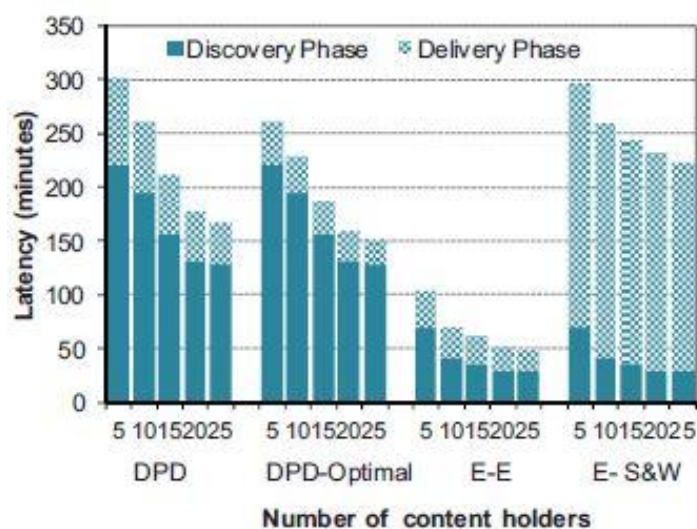


Figura 2.11: mostra la latenza media al variare del numero di possessori dei contenuti

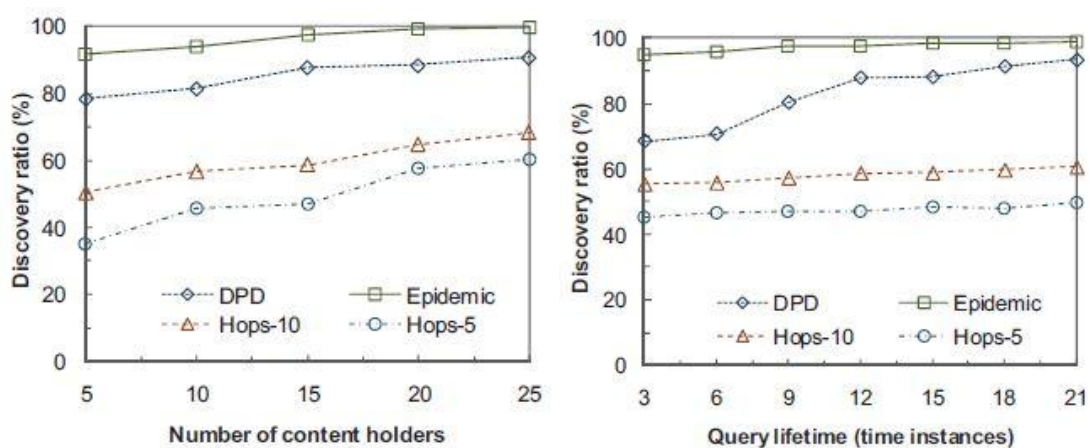


Figura 2.12: mostra il discovery ratio al variare del numero di possessori dei contenuti e dal tempo di vita della query

Tale applicazione (chiamata owapFinder) è stata successivamente installata su un dispositivo android ed è stata fatta un'analisi sull'overhead e sul consumo di batteria del dispositivo (vedi Figura 2.13 - 2.14):

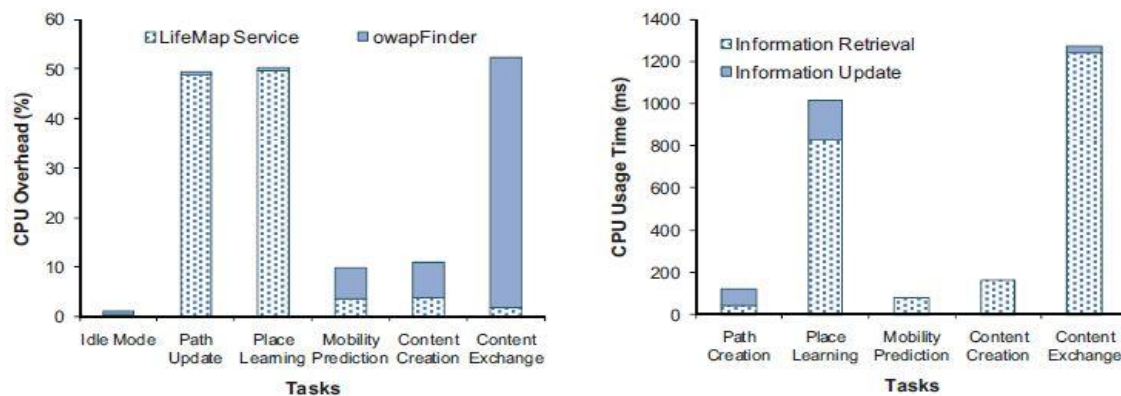


Figura 2.13: mostra un'analisi della CPU del dispositivo a seconda dell'utilizzo dell'applicazione

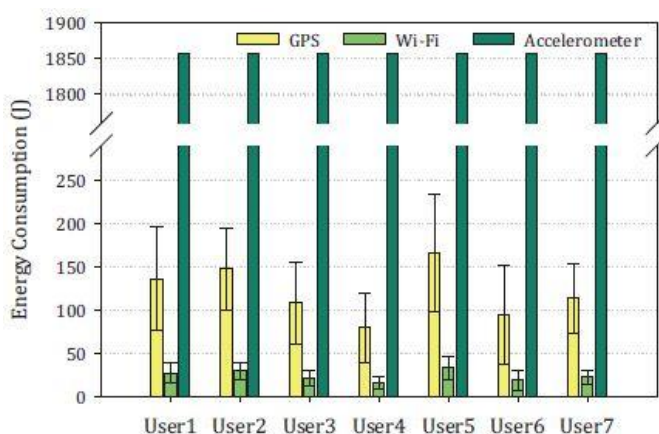


Figura 2.14: mostra il consumo di energia del dispositivo a seconda della tecnologia utilizzata (Wi-Fi, GPS, Accelerometro)

### 2.2.3 - Applicazione Wildlife Tracking

Lo scenario analizzato per questo studio [8] è un'area molto più vasta rispetto ai due casi precedenti e arriva fino a 400 km<sup>2</sup>; esattamente si è cercato di monitorare il comportamento degli animali (nel caso zebre) nella savana. Tale applicazione prevede che ogni nodo sia dotato di un collare a sua volta formato da un rivelatore GPS, da una memoria flash, da una piccola CPU, da un trasmettitore a piccolo raggio e da uno a lungo raggio. In tale rete, come nelle altre, la mobilità dei nodi assume un ruolo importantissimo e viene classificata in tre tipologie:

1. pascolo;
2. camminata;
3. corsa;

In questo scenario sono inserite in modo random diverse sorgenti d'acqua e si fanno alcune assunzioni come che le zebre possano trovare facilmente tali sorgenti e che non dormano mai e dunque che il loro stato di mobilità sia sempre o pascolo o camminata o

corsa. Inoltre, per 3 ore al giorno, una base station mobile sarà attiva e sorvolerà alcune zone di quest'area.

Ricapitolando, lo scenario è composto da nodi mobili (collari) che raccolgono dati e poi li passano ad una base station mobile oppure ad altri collari; è proprio per tale motivo che ogni nodo è dotato di un trasmettitore a corto raggio (comunicazione tra nodi) e di uno a lungo raggio (comunicazione nodo - base station). Lo scopo finale è dunque che la base station riceva i dati di tutti i nodi presenti nella rete e questo può essere fatto solo nelle 3 ore in cui è attiva.

Vengono proposti perciò due tipi di protocolli di routing dell'informazione:

- Flooding protocol: ogni nodo invia i dati a tutti i suoi vicini appena li scopre; in tal modo la base station non deve venire a contatto con tutti i nodi per avere le informazioni e perciò basta che trovi quei nodi che hanno ricevuto il maggior numero di dati da altri nodi; seppure tale protocollo permetta una percentuale di consegna pari al 100 %, il consumo delle risorse di rete come la banda, la capacità di memoria e l'energia sarebbe elevato.

Viene presentato anche lo pseudo-codice di tale protocollo (vedi Figura 2.15):

```

1.  At each scan for neighbors,
2.    if node is within range of the base station,
3.      send data to base station;
4.      delete this data, since it has successfully
        reached the base station;
5.    else
6.      send data to all neighbors;

```

**Figura 2.15: pseudo-codice del Flooding protocol**

- History-Based protocol: si assume che i nodi che incontrano più volte la base station ottengono una priorità maggiore; tale protocollo si basa dunque su una gerarchia a diversi livelli, dove il nodo al livello più alto è quello che ha incontrato più volte la base station e dunque sarà il nodo destinato a ricevere più informazioni; infatti ogni nodo ricorda il proprio livello di gerarchia e ogni volta che incontra dei vicini, “chiede” il loro livello; dopodiché invierà i dati che ha collezionato a quel nodo con il più alto livello della gerarchia. Quando un nodo entra nel range della base station, allora incrementa il proprio livello di gerarchia di 1: viceversa, se un nodo non rientra nel range della base station

dopo D scansioni, decrementa il proprio livello di gerarchia di 1; ovviamente all'inizio tutti i nodi hanno lo stesso livello.

Viene presentato anche lo pseudo-codice di tale protocollo (vedi Figura 2.16):

```

1.  At each scan for neighbors,
2.    if node is within range of the base station,
3.      send data to base station;
4.      delete this data, since it has successfully
        reached the base station;
5.      increment hierarchy level;
6.  else
7.    check hierarchy levels of neighbors;
8.    send data to neighbor with highest level,
        breaking ties randomly;
9.    decay hierarchy level after D scans;

```

**Figura 2.16: pseudo-codice dell' History-Based protocol**

Dopodiché si procede con la simulazione di tale scenario dove vengono posti 50 nodi (rappresentanti ovvero le zebre / i collari) e 10 sorgenti d'acqua; si presuppone inoltre che non esistano montagne o colline che possano disturbare le comunicazioni dell'intera area e che le zebre non corrano il rischio di rimanere uccise a causa di interazioni con altri animali. Tale simulazione viene eseguita per un mese intero e durante la giornata, a distanza di due ore e per una durata complessiva di 30 minuti, vengono svolte le seguenti 4 fasi:

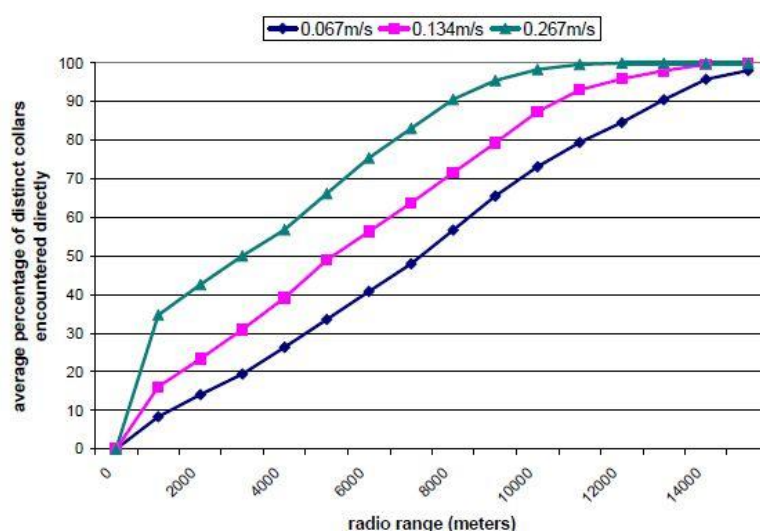
- I. Peer discovery: ogni nodo usa il trasmettitore a corto raggio per cercare i vicini;
- II. Base discovery: ogni nodo usa il trasmettitore a lungo raggio per trovare la base station;
- III. Peer transfer: avviene il trasferimento dati tra un peer e i suoi vicini a seconda del protocollo considerato;
- IV. Base transfer: se la base station viene trovata, allora tutti i dati presenti in memoria vengono inviati; nel caso in cui la memoria di un nodo fosse piena, allora vengono eliminati i dati più vecchi appartenenti agli altri nodi; una volta che i dati sono ricevuti dalla Base station, vengono inseriti in una "delete list".

Da questa simulazione si sono ricavati diversi grafici riguardanti la connettività tra i nodi della rete e la percentuale di dati raccolti dalla Base station.

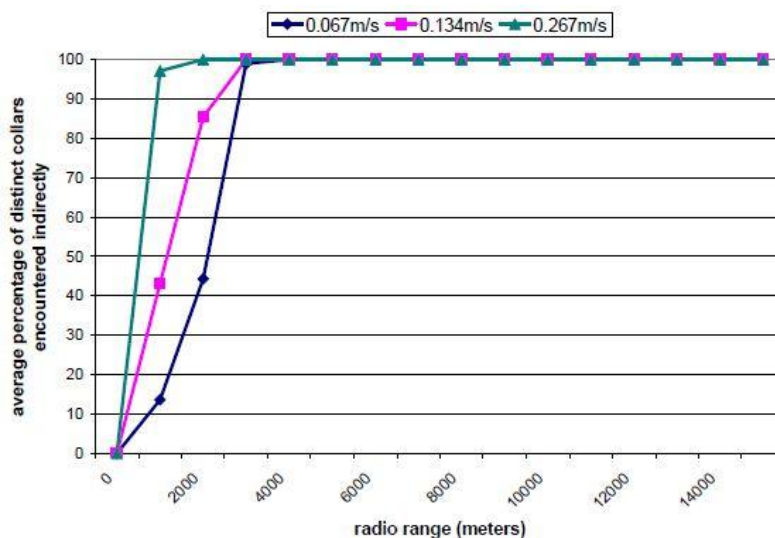
Prima di mostrare i grafici è interessante notare che esistono due tipi di connettività:

- Connettività diretta: dato un cerchio di raggio  $r$  dove al centro risiede il nodo, tutti i nodi all'interno di tale cerchio hanno un cammino diretto verso il nodo considerato;
- Connettività indiretta: sono i nodi che possono essere raggiunti attraverso cammini multi-hop, ovvero passando attraverso altri nodi.

Vengono perciò riportate due figure (vedi Figura 2.17 - 2.18) che mostrano al variare del raggio di copertura di un nodo e della velocità alla quale si muove, a seconda che sia al pascolo (0.067 [m/s]) o stia camminando (0.134 [m/s]) oppure correndo (0.267 [m/s]) con che % di nodi entra a contatto direttamente o indirettamente:



**Figura 2.17:** mostra la % media del numero di nodi cui un nodo viene a contatto direttamente al variare del range del suo trasmettitore a corto raggio e della velocità

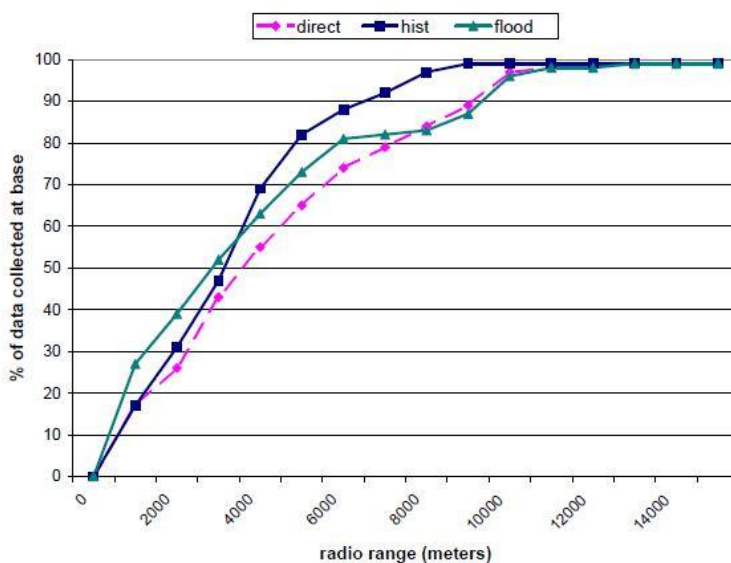


**Figura 2.18:** mostra la % media del numero di nodi cui un nodo viene a contatto indirettamente al variare del range del suo trasmettitore a corto raggio e della velocità

E' facile notare che tanto maggiore sarà la velocità del nodo, tanto più velocemente riuscirà a venire a contatto con tutti gli altri nodi.

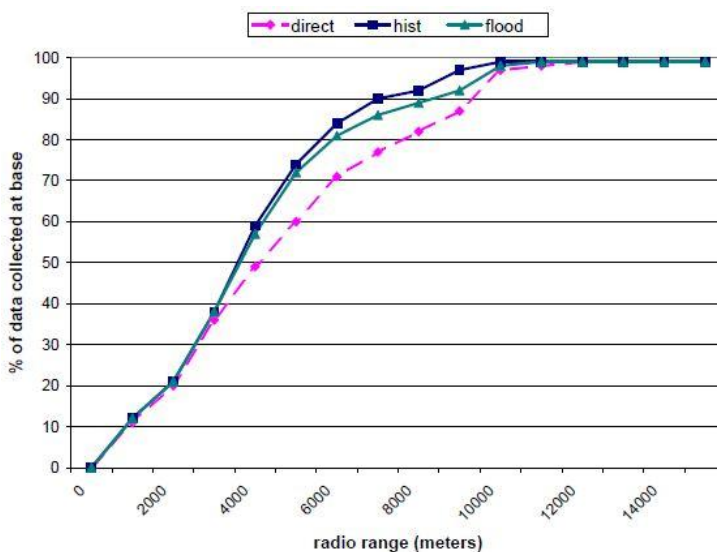
Altre figure interessanti sono quelle che mostrano la % dei pacchetti raccolti dalla Base station utilizzando i protocolli spiegati prima e imponendo alcuni vincoli come:

- vincolo sulla banda e infinita memoria (vedi Figura 2.19):



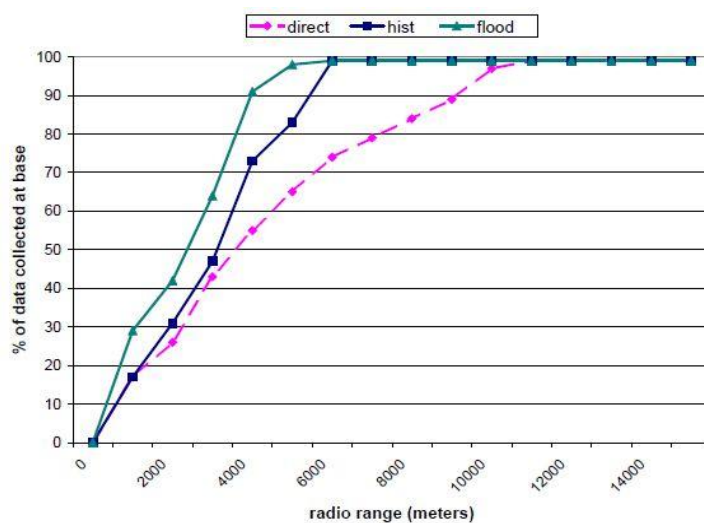
**Figura 2.19:** mostra la % di pacchetti raccolti dalla Base station all'aumentare del range di copertura di un nodo con vincolo sulla banda e infinita memoria

- infinita banda e vincolo sulla memoria (vedi Figura 2.20):



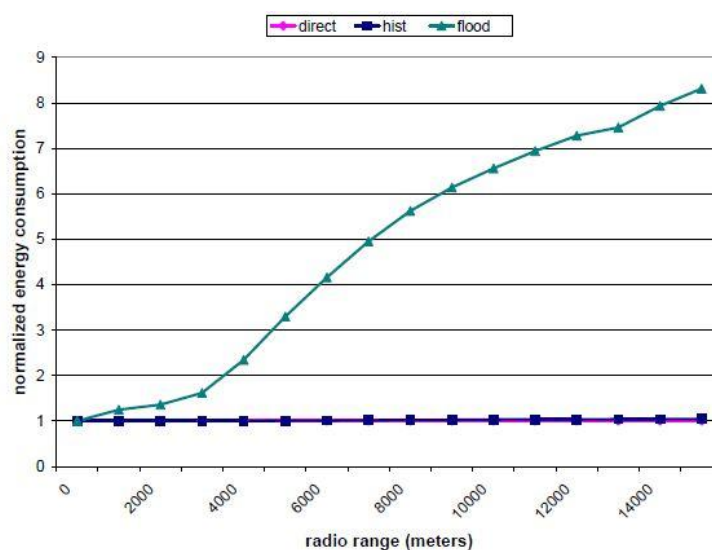
**Figura 2.20:** mostra la % di pacchetti raccolti dalla Base station all'aumentare del range di copertura di un nodo con infinita banda e vincolo sulla memoria

- infinita banda e infinita memoria (vedi Figura 2.21):



**Figura 2.21:** mostra la % di pacchetti raccolti dalla Base station all'aumentare del range di copertura di un nodo con infinita banda e infinita memoria

Si noti che attraverso il primo protocollo presentato, Flooding, si ottengono risultati migliori in termini di dati consegnati in quanto ogni nodo invia i suoi dati a tutti i suoi vicini e quindi c'è una maggior probabilità di completa ricezione dati da parte della Base station; d'altro canto è facile intuire come l'overhead di rete e il consumo energetico (vedi Figura 2.22) siano maggiori nel caso di Flooding rispetto all'History-Based in quanto nel primo tutti i nodi inviano dati senza alcuna condizione, mentre nel secondo si presta attenzione al livello gerarchico.



**Figura 2.22:** mostra il consumo energetico medio di ogni nodo al variare del range di copertura e a seconda del protocollo utilizzato

### 2.2.4 - Applicazione PROPHET (invio di messaggi)

Lo scopo principale di questa applicazione [9] è di sfruttare la delay tolerant network per inviare messaggi in uno scenario abbastanza “ridotto” quale può essere un campus universitario. Il concetto base in questa applicazione / protocollo è quello di delivery predictability,  $P_{(a,b)}$ , per un nodo  $a$  verso un nodo  $b$ ; tale valore indica quanto il nodo  $a$  sia in grado di trasportare/inviare il messaggio a  $b$ . Infatti tale protocollo presuppone che quando due nodi si incontrano si scambiano tale informazione e aggiornano il proprio vettore locale composto da tutti i valori di delivery predictability di tutti i nodi.

Il calcolo della delivery predictability da parte di un nodo è composto da 3 fasi:

1. aggiornare costantemente la metrica, così che ogni volta che i nodi si incontrano allora la delivery predictability tra loro aumenta; se prendiamo  $P_{init} \in [0,1]$  come una costante iniziale, si ha:

$$P_{(a,b)} = P_{(a,b)old} + (1 - P_{(a,b)old}) * P_{init}$$

Se due nodi non si incontrano spesso allora la probabilità di essere dei buoni portatori di messaggi l’uno per l’altro è più bassa.

2. si aggiunge una costante di invecchiamento  $\gamma \in [0,1]$  e  $k$  che sarebbe il numero in unità di tempo che tale metrica non è stata aggiornata e si ottiene:

$$P_{(a,b)} = P_{(a,b)old} * \gamma^k$$

3. la delivery predictability gode di una proprietà transitiva secondo cui se il nodo  $a$  incontra molte volte il nodo  $b$ , che a sua volta incontra molte volte il nodo  $c$ , allora il nodo  $c$  è un nodo “buono” per inviare messaggi verso il nodo  $a$ ; si aggiunge perciò una costante  $\beta \in [0,1]$  che esprime quanto la proprietà transitiva incide sulla delivery predictability:

$$P_{(a,c)} = P_{(a,c)old} + (1 - P_{(a,c)old}) * P_{(a,b)} * P_{(b,c)} * \beta$$

Di solito inviare un messaggio è veramente semplice ed esistono diverse opzioni: si può decidere di inviare il messaggio solo al nodo dei vicini a più basso costo di trasmissione rischiando che il messaggio vada perso a causa di una scomparsa del collegamento prima esistente, oppure si può decidere di inviare il messaggio a più nodi con il rischio di saturare il traffico della rete.

Perciò il meccanismo di routing adottato in tale rete è abbastanza semplice: quando un nodo rientra nel range di copertura di un altro, l’altro invia il messaggio se e solo se la



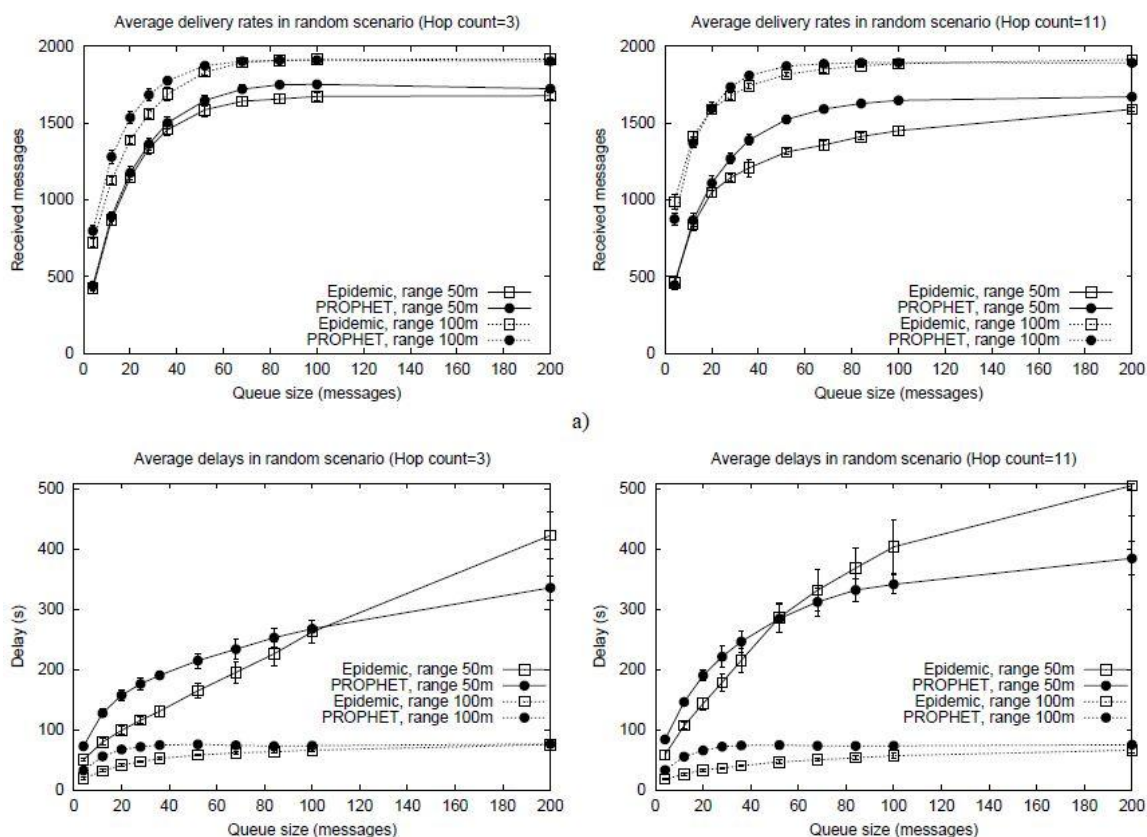
delivery predictability verso il nodo destinazione è maggiore nel nodo appena incontrato rispetto a quello che ha già il messaggio.

Dopodiché si è passati ad una simulazione analizzando in particolare modo due tipi di scenario; alcuni parametri della simulazione ( $P_{init}$ ,  $\beta$ ,  $\gamma$ ) sono stati presi fissi mentre altri (buffer size, range del nodo, hop count del messaggio) sono stati fatti variare.

Vediamo ora i due tipi di scenario e i relativi risultati ottenuti:

- Scenario 1: viene simulata un'area di 1500 [m] x 300 [m] dove sono stati inseriti 50 nodi in modo random; tali nodi si muovono ad una velocità che varia dai 0 ai 20 [m/s]; da un sottoinsieme di 45 nodi, un messaggio al secondo viene inviato da ogni nodo per 1980 secondi e la simulazione viene lasciata girare per altri 2020 secondi.

I grafici mostrati in Figura 2.23 mostrano i risultati della simulazione in termini di messaggi correttamente ricevuti e il ritardo medio a seconda del protocollo utilizzato; infatti, oltre che per il protocollo PROPHET, vengono mostrate le prestazioni in caso di utilizzo di protocollo "Epidemic", ovvero quando tutti i nodi inviano i messaggi ai loro vicini che rientrano in un range di copertura fino a quando il messaggio è arrivato alla destinazione corretta.



**Figura 2.23:** mostra i grafici ottenuti simulando lo scenario 1 utilizzando il protocollo di routing PROPHET e quello Epidemic

- Scenario 2: viene simulata un'area di 3000 [m] x 1500 [m] che viene suddivisa in 12 rettangoli, di cui 11 rappresentano le comunità e 1 il “gathering place” (o luogo di ritrovo); ogni nodo appartiene a una comunità; inoltre ogni comunità ha 5 nodi più un nodo fisso che funge da gateway per quell'area. La mobilità in questo scenario funziona che i nodi si muovono con una velocità dai 10 ai 30 [m/s], arrivano un punto, si fermano e poi ripartono; le destinazioni di un nodo sono scelte in modo che se un nodo è nella propria comunità allora si sposterà con una certa probabilità verso il gathering place, mentre se non è nella propria comunità, si sposterà con una certa probabilità verso di essa. Per capire meglio tale concetto si osservi la Figura 2.24:

From \ To	Home	Gathering place	Elsewhere
Home	-	0.8	0.2
Elsewhere	0.9	-	0.1

**Figura 2.24:** mostra la probabilità con cui un nodo si muove all'interno di questo scenario

Il traffico in tale scenario è diverso rispetto al precedente, in quanto ogni 10 secondi due gateway scelti in modo random generano un messaggio per altri

gateway o per il gathering place; 5 secondi dopo, due nodi generano un messaggio con destinazione random; dopo 3000 secondi tale meccanismo cessa e la simulazione è fatta proseguire per altri 8000 secondi per permettere ai messaggi di arrivare a destinazione.

I risultati di tale simulazione sono presenti in Figura 2.25:

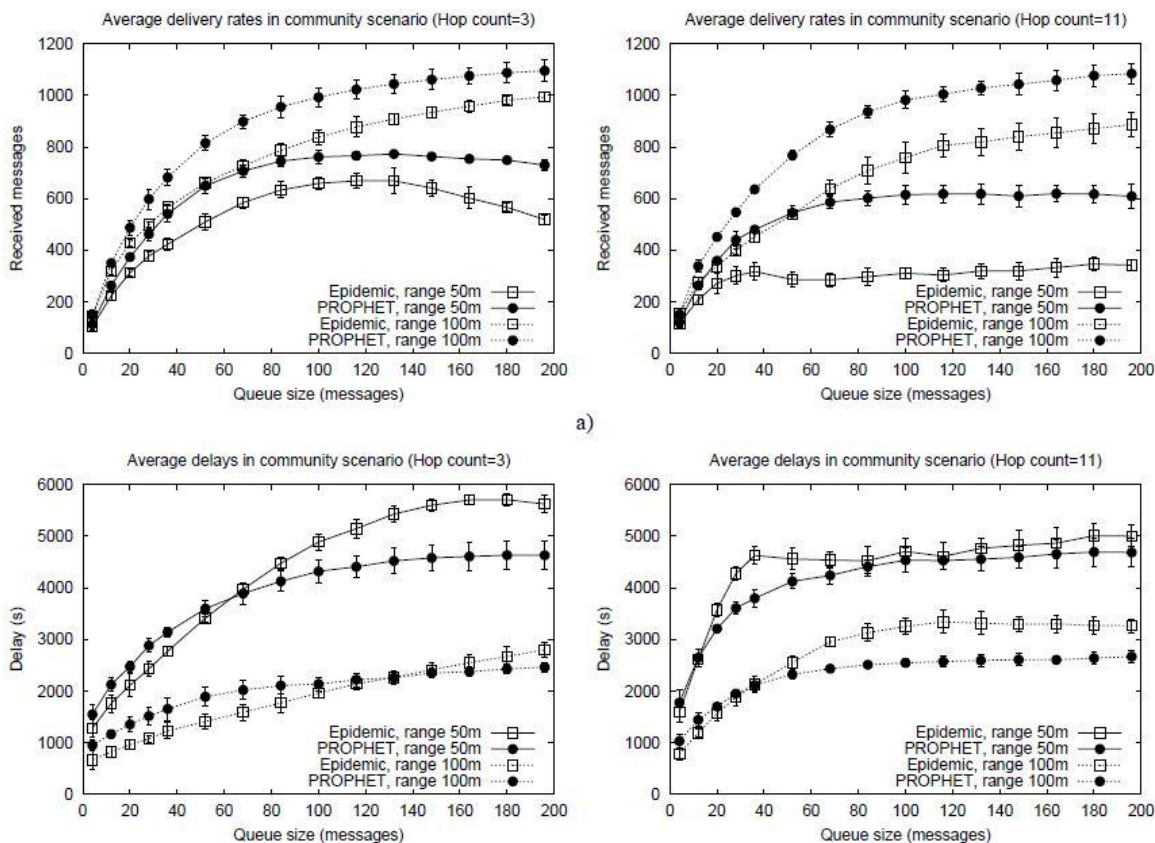


Figura 2.25: mostra i grafici ottenuti simulando lo scenario 2 utilizzando il protocollo di routing PROPHET e quello Epidemic

## Capitolo 3 – Routing sociale

### 3.1 - Introduzione

In questo capitolo si parlerà di un particolare tipo di routing, il routing sociale: si tratta di un meccanismo che per instradare i vari pacchetti da un nodo ad un altro analizza le componenti sociali del nodo stesso per poi decidere di inviare o no il pacchetto. Dunque tale modello non si basa sulle caratteristiche fisiche del nodo, come ad esempio la velocità, e nemmeno sulle caratteristiche intrinseche del messaggio, come ad esempio un determinato tag o un TTL (Time To Live), ma bensì si basa sui rapporti sociali di ogni componente della DTN considerata.

Con il continuo sviluppo di dispositivi portatili quali ad esempio smartphone (Iphone, Samsung Galaxy, etc.), si è notato anche un parallelo sviluppo di applicazioni sociali che racchiudono informazioni particolari su ogni utente, dalla data di nascita fino al numero di contatti, e sono proprio queste informazioni che vengono analizzate nei vari protocolli di routing sociali per instradare pacchetti.

Come detto, esistono diverse applicazioni che vengono utilizzate da milioni di persone nel mondo e prendono il nome di social media; tali applicazioni vengono installate nel dispositivo, occupano un piccolo spazio di memoria e attraverso un collegamento a internet (WI-FI, 3G, Bluetooth etc.) rimangono a contatto con tutta la loro rete sociale.

In tale elaborato vengono prese in considerazione solo due di queste applicazioni:

- Facebook
- Foursquare

Ne esistono diverse e con tantissimi altri utilizzi (ad es pubblicare foto, video) ma le due più interessanti ed utili in tale protocolli sociali sono appunto quelle che ora vi ho appena citato e che vi presenterò.

## **3.2 - Profili applicativi per l'instradamento**

### **3.2.1 - Facebook**

Facebook è un social network lanciato nel febbraio 2004, posseduto e gestito dalla corporation Facebook, Inc.

Il sito, fondato a Cambridge (USA) da Mark Zuckerberg e dai suoi compagni di università Eduardo Saverin, Dustin Moskovitz e Chris Hughes, era originariamente stato progettato esclusivamente per gli studenti dell'Università di Harvard, ma fu presto aperto anche agli studenti di altre scuole della zona di Boston, della Ivy League e della Stanford University. Successivamente fu aperto anche agli studenti delle scuole superiori e poi a chiunque dichiarasse più di 13 anni di età. Da allora Facebook raggiunse un enorme successo: è diventato il secondo sito più visitato al mondo, preceduto solo da Google; è disponibile in oltre 70 lingue e nel giugno 2012 conta più di 955 milioni di utenti attivi che effettuano l'accesso almeno una volta al mese, classificandosi come primo social network per numero di utenti attivi. Il nome "Facebook" prende spunto da un elenco con nome e fotografia degli studenti, che alcune università statunitensi distribuiscono all'inizio dell'anno accademico per aiutare gli scritti a socializzare tra loro.

Gli utenti possono accedere al sito previa una registrazione gratuita, durante la quale vengono richiesti dati personali come nome, cognome, data di nascita e indirizzo email. Completata la registrazione, gli utenti possono creare un profilo personale, includere altri utenti nella propria rete sociale, aggiungendoli come amici, e scambiarsi messaggi, anche via chat, incluse le notifiche automatiche quando questi aggiornano i propri profili; inoltre gli utenti possono fondare e unirsi a gruppi per condividere interessi in comune con altri utenti, organizzati secondo il luogo di lavoro, la scuola, l'università o altre caratteristiche, condividere contenuti multimediali ed utilizzare varie applicazioni presenti sul sito. Per personalizzare il proprio profilo l'utente può caricare una foto, chiamata immagine del profilo, con la quale può rendersi riconoscibile. Può inoltre fornire ulteriori informazioni, come il comune di nascita (esempio: Città natale: Roma) e quello di residenza (esempio: Vive a Milano), la scuola frequentata, il proprio datore di lavoro, l'orientamento religioso e quello politico, la propria situazione sentimentale e altri suoi hobby. Altra funzione molto utilizzata è quella di creare degli album fotografici contenenti numerose foto e la possibilità di taggare gli amici eventualmente presenti in questi album.

L'aspetto per noi più importante è che Facebook integra alcune delle suddette funzionalità sulle sue applicazioni per dispositivi mobili (smartphone), come la possibilità di visualizzare le varie informazioni degli amici (es. data di nascita, luogo di nascita, numero degli amici totali e quelli in comune etc.), di caricare contenuti come immagini e video, di ricevere e rispondere ai messaggi, di scrivere sulla bacheca degli utenti o semplicemente la possibilità di navigare sul network. Altra caratteristica importante presente sui dispositivi mobili è quella di poter far notare agli amici la propria presenza su Facebook facendo dei "check-in" in alcuni luoghi importanti (es. un bar, un cinema) limitrofi sfruttando il segnale GPS.

---

*Curiosità: tre studenti di Harvard, Cameron, Tyler Winklevoss e Divya Narendra, accusarono Zuckerberg di averli truffati, in quanto avrebbe promesso loro di aiutarli a creare un servizio di rete sociale chiamato "HarvardConnection.com" mentre era intenzionato a rubargli l'idea per creare un sito concorrente (Facebook appunto); l'accusa si tradusse successivamente in una causa legale in cui Zuckerberg fu costretto a pagare un'onerosa multa.*

### 3.2.2 - Foursquare

Foursquare è un social network basato sulla geolocalizzazione disponibile tramite web e applicazioni per dispositivi mobili.

Il servizio è stato creato da Dennis Crowley e Naveen Selvadurai; i due cominciarono a costruire la prima versione di Foursquare nell'autunno del 2008 e lo lanciarono nel Sud al Southwest Interactive di Austin, in Texas nel marzo del 2009. Crowley aveva precedentemente creato un progetto simile dal nome Dodgeball, che è stato comprato da Google nel 2005 e messo offline nel 2009 per lasciare spazio a Google Latitude. Al suo lancio nel 2009 dunque Foursquare era disponibile in maniera limitata in sole 100 aree metropolitane in tutto il mondo; nel gennaio 2010 il social network ha cambiato il modello alla base dei check-in permettendo di farlo in qualsiasi parte del mondo.

Nel giugno 2011 Foursquare raggiunge i 10 milioni di membri, una notevole crescita rispetto ai 6 milioni di inizio anno, con un incremento del 3400%.

Nell'aprile 2012 l'azienda comunica di aver raggiunto i 20 milioni di utenti registrati, di aver superato i 2 miliardi di check-in, e di aver assistito più di 750.000 commercianti con la propria piattaforma.

Foursquare è un'applicazione mobile e web che permette agli utenti registrati di condividere la propria posizione con i propri contatti; il check-in nei luoghi permette di ottenere punti necessari a scalare una classifica settimanale, basata sugli ultimi 7 giorni, della quale fanno parte i contatti della propria stessa città. I check-in possono inoltre essere condivisi, insieme ad un breve status, collegando Foursquare ai propri profili Facebook e Twitter. Gli utenti ricevono inoltre dei "badge", dei riconoscimenti per aver raggiunto certi obiettivi, eseguendo il check-in in certi luoghi, a una certa frequenza o trovandosi in una certa categoria di luoghi. Se un utente esegue il check-in in uno stesso luogo più giorni di seguito e visita un luogo più di qualsiasi altro utente nei precedenti 60 giorni ne diventa sindaco e il suo avatar è inserito nella pagina relativa al luogo fino a quando un nuovo utente non esegue più check-in del sindaco in carica; per diventare sindaco non valgono check-in multipli eseguiti nello stesso luogo durante lo stesso giorno. I proprietari di un'attività hanno la facoltà, qualora la pagina relativa al luogo non sia stata creata da loro, di reclamarla e di offrire sconti e offerte al sindaco o agli utenti che vi fanno il check-in.

Gli utenti possono inoltre creare una lista pubblica di cose da fare e scrivere dei brevi suggerimenti per gli utenti che seguono check-in nel luogo stesso o in quelli vicini.

Foursquare è disponibile con applicazioni per iOS, Android, WebOS, Windows Phone 7 e Blackberry; tuttavia il servizio è comunque accessibile tramite browser mobili attraverso i quali i luoghi devono però essere cercati manualmente e non possono essere trovati automaticamente tramite il GPS come avviene nelle applicazioni dedicate.

---

*Curiosità: in Foursquare esistono gli “Special” che sono un incentivo per gli utenti ad effettuare check-in in una determinata venue. Ci sono più di 750.000 punti vendita che offrono diverse promozioni, i principali casi di successo sono quelli che offrono uno sconto con l'acquisto, quelli che regalano qualcosa, o quelli che offrono un trattamento speciale ai clienti migliori. Gli Special sono gratuiti e valgono solo per i luoghi dichiarati. Inoltre esistono dei Superuser, ovvero uno status che viene assegnato agli utenti che effettuano check-in e aggiungono luoghi al database del sito molto frequentemente e ai quali vengono assegnati delle possibilità di amministrazione delle pagine relative ai luoghi.*



### 3.3 - Tassonomia delle comunicazioni e funzioni di costo associate

Per capire l'importanza di questi profili applicativi nell'instradare i pacchetti verso altri nodi supponiamo di avere una rete e di voler testare tre tipi di comunicazione:

1. Unicast
2. Multicast
3. Geocast

Questi tipi di comunicazione vengono testati utilizzando diverse funzioni di costo a seconda del protocollo di routing utilizzato; distinguiamo tre tipi di protocollo:

1. Fisico: utilizzano i contatti a livello fisico che un nodo riceve entro un determinato raggio;
2. Applicativo: utilizzano i profili applicativi degli utenti (Facebook, Foursquare);
3. Ibrido: utilizzano un mix dei primi due protocolli

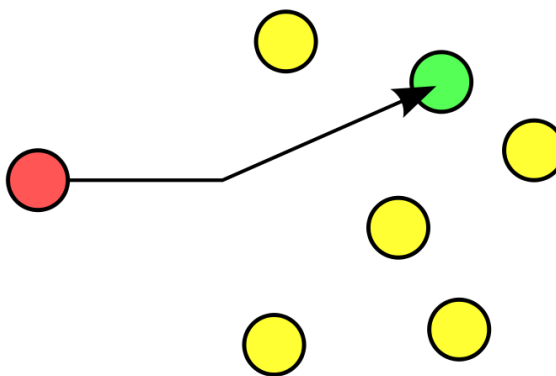
Prima di entrare nel dettaglio dei tre tipi di comunicazione e delle varie funzioni di costo a seconda del protocollo utilizzato, bisogna specificare l'esistenza di alcune funzioni importanti:

- $f(i)$  : dato  $i \in A$  dove  $i$  è un generico nodo e  $A$  l'insieme dei nodi, la funzione  $f(i)$  restituisce un valore che rappresenta il numero di nodi che hanno pingato il nodo  $i$  per un certo tempo  $t$  entro un raggio  $R$ ;
- $g(i)$  : dato  $i \in A$  dove  $i$  è un generico nodo e  $A$  l'insieme dei nodi, la funzione  $g(i)$  restituisce un valore che rappresenta il numero di amici totale del nodo  $i$ ;
- $h(i, ID)$  : dato  $i \in A$  dove  $i$  è un generico nodo e  $A$  l'insieme dei nodi, la funzione  $h(i, ID)$  restituisce un valore che rappresenta il numero di check in del nodo  $i$  in un luogo con un preciso  $ID$ ;

Vediamo ora i tre casi di comunicazione elencati prima.

#### 3.3.1 - Comunicazione Unicast

Con il termine Unicast si intende la trasmissione di un pacchetto che ha come destinatario un singolo nodo (vedi Figura 3.1).



**Figura 3.1: esempio di trasmissione Unicast**

A seconda del protocollo considerato (fisico, applicativo, ibrido) vengono presentate diverse funzioni di costo per la trasmissione del pacchetto, vediamole:

1. Fisico: dati  $i, j \in A$  dove  $i, j$  sono nodi generici e  $A$  è l'insieme dei nodi, per la trasmissione del messaggio dal nodo  $i$  al nodo  $j$  si deve veder rispettata la seguente condizione:

$$\begin{cases} f(i) > f(j) \text{ allora } i \text{ invia il messaggio a } j \\ f(i) < f(j) \text{ allora } i \text{ non invia il messaggio a } j \end{cases}$$

dove  $f(i)$  è la funzione descritta al sotto-capitolo precedente che conta il numero di contatti che pingano il nodo per un tempo  $T$ .

2. Applicativo: dati  $i, j \in A$  dove  $i, j$  sono nodi generici e  $A$  è l'insieme dei nodi,  $i$  e  $j$  si scambiano la loro lista amici di Facebook,  $i$  fa il parsing della lista amici di  $j$  e se trova l'amico cui vuole inviare il messaggio, allora invierà il messaggio a  $j$  altrimenti no; ricapitolando, si presentano due casi:
  - I.  $i$  fa il parsing della lista amici di  $j$ , restituisce 1 allora  $i$  invia il messaggio a  $j$ ;
  - II.  $i$  fa il parsing della lista amici di  $j$ , restituisce 0 allora  $i$  non invia il messaggio a  $j$ ;
3. Ibrido: dati  $i, j \in A$  dove  $i, j$  sono nodi generici e  $A$  è l'insieme dei nodi, per il nodo  $i$  possessore del messaggio si presentano 4 casi:
  - I. : se  $f(i) < f(j)$  e Parsing lista amici di  $j \rightarrow 1$  allora  $i$  invia il messaggio a  $j$ ;
  - II. se  $f(i) > f(j)$  e Parsing lista amici di  $j \rightarrow 1$  allora  $i$  con probabilità  $P$  invia il messaggio a  $j$ ;

- III. se  $f(i) < f(j)$  e Parsing lista amici di  $j \rightarrow 0$  allora  $i$  invia con probabilità  $Q$  il messaggio a  $j$ ;
- IV. se  $f(i) > f(j)$  e Parsing lista amici di  $j \rightarrow 0$  allora  $i$  non invia il messaggio a  $j$ .

### 3.3.2 - Comunicazione Multicast

Con il termine Multicast si intende la trasmissione di un pacchetto che ha come destinatario un gruppo numeroso di nodi (vedi Figura 3.2).

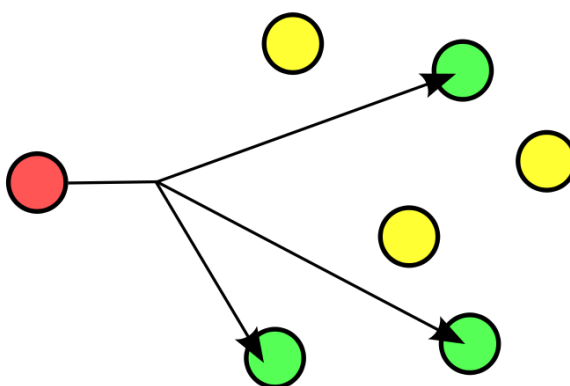


Figura 3.2: esempio di trasmissione Multicast

A seconda del protocollo considerato (fisico, applicativo, ibrido) vengono presentate diverse funzioni di costo per la trasmissione del pacchetto, vediamole:

1. Fisico: dati  $i, j \in A$  dove  $i, j$  sono nodi generici e  $A$  è l'insieme dei nodi, per la trasmissione del messaggio dal nodo  $i$  al nodo  $j$  si deve veder rispettata la seguente condizione:

$$\begin{cases} f(i) > f(j) \text{ allora } i \text{ invia il messaggio a } j \\ f(i) < f(j) \text{ allora } i \text{ non invia il messaggio a } j \end{cases}$$

dove  $f(i)$  è la funzione descritta al sotto-capitolo precedente che conta il numero di contatti che pingano il nodo per un tempo  $T$ .

2. Applicativo: dati  $i, j \in A$  dove  $i, j$  sono nodi generici e  $A$  è l'insieme dei nodi,  $i$  e  $j$  si scambiano la loro lista amici di Facebook,  $i$  fa il parsing della lista amici di  $j$  e se trova l'amico cui vuole inviare il messaggio, allora invierà il messaggio a  $j$  altrimenti no; ricapitolando, si presentano due casi:
  - I.  $i$  fa il parsing della lista amici di  $j$ , restituisce 1 allora  $i$  invia il messaggio a  $j$ ;

- II.  $i$  fa il parsing della lista amici di  $j$ , restituisce 0 allora  $i$  non invia il messaggio a  $j$ ;
3. Ibrido: dati  $i, j \in A$  dove  $i, j$  sono nodi generici e  $A$  è l'insieme dei nodi, per il nodo  $i$  possessore del messaggio si presentano 4 casi:
- I. se  $f(i) < f(j)$  e Parsing lista amici di  $j \rightarrow 1$  allora  $i$  invia il messaggio a  $j$ ;
  - II. se  $f(i) > f(j)$  e Parsing lista amici di  $j \rightarrow 1$  allora  $i$  con probabilità  $P$  invia il messaggio a  $j$ ;
  - III. se  $f(i) < f(j)$  e Parsing lista amici di  $j \rightarrow 0$  allora  $i$  con probabilità  $Q$  invia il messaggio a  $j$ ;
  - IV. se  $f(i) > f(j)$  e Parsing lista amici di  $j \rightarrow 0$  allora  $i$  non invia il messaggio a  $j$ .

### 3.3.3 - Comunicazione Geocast

Con il termine Geocast si intende la trasmissione di un pacchetto che ha come destinatario uno o più nodi che sono situati in un preciso luogo contraddistinto da un ID (es. ID Mensa = 1, ID Palestra = 2, etc.) (vedi Figura 3.3).

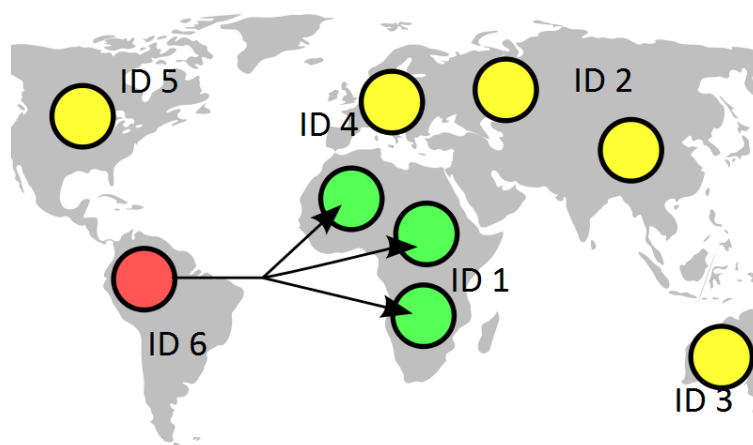


Figura 3.3: esempio di trasmissione Geocast

A seconda del protocollo considerato (fisico, applicativo, ibrido) vengono presentate diverse funzioni di costo per la trasmissione del pacchetto, vediamole:

1. Fisico: dati  $i, j \in A$  dove  $i, j$  sono nodi generici e  $A$  è l'insieme dei nodi, per la trasmissione del messaggio dal nodo  $i$  al nodo  $j$  si deve veder rispettata la seguente condizione:

$$\begin{cases} f(i) > f(j) \text{ allora } i \text{ invia il messaggio a } j \\ f(i) < f(j) \text{ allora } i \text{ non invia il messaggio a } j \end{cases}$$

dove  $f(i)$  è la funzione descritta al sotto-capitolo precedente che conta il numero di contatti che pingano il nodo per un tempo  $T$ .

2. Applicativo: dati  $i, j \in A$  dove  $i, j$  sono nodi generici e  $A$  è l'insieme dei nodi, si supponga che ogni luogo abbia un preciso ID; per la trasmissione del messaggio dal nodo  $i$  al nodo  $j$  si deve veder rispettata la seguente condizione:

$$\begin{cases} h(i, ID) > h(j, ID) \text{ allora } i \text{ invia il messaggio a } j \\ h(i, ID) < h(j, ID) \text{ allora } i \text{ non invia il messaggio a } j \end{cases}$$

dove  $h(i, ID)$  è la funzione descritta al sotto-capitolo precedente che conta il numero di check in del nodo in un luogo con un certo ID.

3. Ibrido: dati  $i, j \in A$  dove  $i, j$  sono nodi generici e  $A$  è l'insieme dei nodi, per il nodo  $i$  possessore del messaggio si presentano 4 casi:
- I. se  $f(i) < f(j)$  e  $h(i, ID) < h(j, ID)$  allora  $i$  invia il messaggio a  $j$ ;
  - II. se  $f(i) > f(j)$  e  $h(i, ID) < h(j, ID)$  allora  $i$  con probabilità  $P$  invia il messaggio a  $j$ ;
  - III. se  $f(i) < f(j)$  e  $h(i, ID) > h(j, ID)$  allora  $i$  con probabilità  $Q$  invia il messaggio a  $j$ ;
  - IV. se  $f(i) > f(j)$  e  $h(i, ID) > h(j, ID)$  allora  $i$  non invia il messaggio a  $j$ .

## Capitolo 4 - Valutazione delle prestazioni

In questo capitolo introdurremo il simulatore utilizzato (OMNeT++ 4.2.2) per valutare le prestazioni dei vari protocolli per poi osservare attraverso alcuni grafici diversi parametri misurati durante le simulazioni dei vari protocolli traendo le giuste conclusioni

### 4.1 - OMNeT++ 4.2.2

OMNet++ è un simulatore basato sul linguaggio C++, a eventi discreti, per modelli di reti di comunicazioni; nasce nel 2003, è open-source e viene principalmente utilizzato in ambito di ricerca accademica al fine di simulare protocolli. Ogni entità presente è rappresentata mediante un modulo, tutti i moduli sono organizzati gerarchicamente e comunicano mediante scambio di messaggi (che possono anche contenere strutture dati complesse) che viaggiano attraverso i canali inoltrandoli direttamente verso una destinazione precisa o lungo un percorso, usando porte e connessioni. Tale simulatore è dotato di un interfaccia grafica di alto livello che aiuta l'utente a visualizzare i messaggi scambiati e i possibili movimenti dei nodi durante le simulazioni. Il simulatore può essere installato sia su sistema operativo Windows sia su Linux; inoltre esistono diversi Framework che possono essere utilizzati su OMNeT++ e che aiutano l'utente in particolari casi da simulare. Uno di questi è INET Framework, dove sono state implementate diverse caratteristiche e diversi "comportamenti" dei nodi tra cui, ad esempio, la mobilità all'interno della rete o la procedura di handover in reti wireless.

Un modello/progetto di OMNeT++ si compone di alcuni file fondamentali:

- file **.NED**: serve per specificare la topologia di un modulo facilitando la descrizione modulare di una rete ed offre la possibilità di costruire una rete con sottomoduli innestati. I possibili componenti di un modello NED sono:
  - direttive *import*: usate per importare dichiarazioni da un altro file per la descrizione di una rete (es: *import "ethernet"*);
  - definizioni dei canali: specificano un tipo di connessione; il nome del canale può essere usato per la creazione di connessioni e gli

attributi possono essere assegnati anche nel corpo della dichiarazione del canale;

- *simple module*: blocco base per la costruzione dei moduli, viene definito dalla dichiarazione di parametri e porte (es. *parameters, gates*)
  - *compound module*: moduli composti da uno o più sotto-moduli, anch'essi possono avere porte e parametri;
- file **.MSG**: vi è la definizione dei messaggi, ovvero i vari campi che ogni nodo potrà poi leggere / scrivere;
  - file **.H/.CC**: sono i file più importanti, sono quelli che devono essere compilati e danno le istruzioni ai nodi; nei file *.H* vi sono le dichiarazioni di variabili e di funzioni che poi nel relativo *.CC* verranno implementate; un nodo per fare qualsiasi cosa (dal semplice movimento all'interazione con altri nodi etc.) deve avere una particolare funzione nel corrispettivo file *.CC* che gli dà quel compito (ad es. la funzione *handlemessage()* che viene utilizzata ogni qualvolta un nodo riceve un messaggio o la funzione *sendmess()* che serve ad inviare messaggi ai nodi vicini etc.).
  - file **.INI**: contiene informazioni di controllo della rete, ad esempio il numero di nodi, la potenza di trasmissione (da cui si ricava il raggio di copertura), il tipo di mobilità degli utenti (Lineare, Circolare, Random, etc.) e tanti altri parametri della rete considerata.

La simulazione ha una durata diversa da quella reale in quanto deve analizzare e gestire tutti i messaggi e tutti i comportamenti dei vari nodi della rete con il passare del tempo; la simulazione può essere rallentata / fermata ogni volta che si desidera al fine di controllare il corretto andamento del protocollo in questione. Terminata la simulazione, si possono far stampare i risultati sulla console per poi inserirli e analizzarli in uno dei tools a disposizione (ad esempio in Microsoft Excel). Nel seguito di questo capitolo infatti riporterò grafici ottenuti con Excel che rappresenteranno i risultati delle varie simulazioni.

## 4.2 - Scenario e parametri

Lo scenario considerato per le simulazioni è quello di un campus universitario di dimensioni  $700 \times 500 [m^2]$  dove ogni nodo si muove secondo una funzione lineare. Per realizzare ciò, ogni nodo ha un modulo chiamato 'Linear Mobility' nel quale si ricava la funzione con cui si muoverà al passare del tempo; tale funzione avrà alcuni parametri base come l'angolo di movimento, la velocità, l'eventuale accelerazione con il relativo periodo di update della velocità.

Al fine di simulare una rete il più realistica possibile si è fatto in modo di assegnare ad ogni nodo una velocità uniforme compresa tra  $2 [km/h]$  (persone che camminano) e  $35 [km/h]$  (persone che si spostano con un mezzo come macchina o moto); il dispositivo di ogni nodo avrà una capacità fissa pari a  $2 [Mbit/sec]$  con la quale invierà i messaggi (di dimensione fissa pari a  $400 [bit]$ ) e/o i ping (di dimensione fissa pari a  $56[bit]$ ). Si suppone che tutti i dispositivi siano cooperativi, ovvero che tutti possano accettare messaggi / ping e trasportarli alla destinazione finale o ad altri dispositivi nella rete. Inoltre, per questioni di comodità, si è supposto che quando un nodo raggiunge il "confine" della rete, questo "rimbalza" con lo stesso angolo con il quale si sta muovendo; dunque nessun nodo può uscire dalla rete durante il periodo di simulazione.

Tutti questi parametri vengono passati ad ogni progetto OMNeT attraverso un proprio file *.INI*. In tale file sono, poi, presenti altri valori che verranno fatti variare durante alcune simulazioni come la potenza di trasmissione (dalla quale si ricava il raggio di copertura), la frequenza di invio dei messaggi e/o dei ping, etc.

Durante le simulazioni, a seconda dei protocolli analizzati, sono stati misurati diversi valori come:

- Overhead di rete: si intende il numero totale di messaggi (e le relative copie) nella rete;
- Numero di ping medi ricevuti dai nodi prima dell'invio dei messaggi e quelli a fine simulazione;
- Durata totale di simulazione: si intende il tempo espresso in secondi affinché l'ultimo pacchetto venga ricevuto dal corretto destinatario;
- Tempo di consegna media: si intende il tempo medio di arrivo di ogni messaggio alla corretta destinazione;



- Numero copie medie per messaggio: si intende il numero di copie prodotte in media per ogni messaggio generato nella rete ed è pari all'Overhead totale della rete diviso per il numero di messaggi generati.
- Numero di Check-In medi per area: strettamente al caso Applicativo Geocast, sono stati misurati il numero di Check-In medi effettuati dai nodi in ogni singola area.

La probabilità di consegna è sempre pari al 100% in quanto abbiamo atteso il tempo che ogni messaggio arrivi alla corretta destinazione.

### 4.3 - Analisi prestazioni per Protocollo Fisico

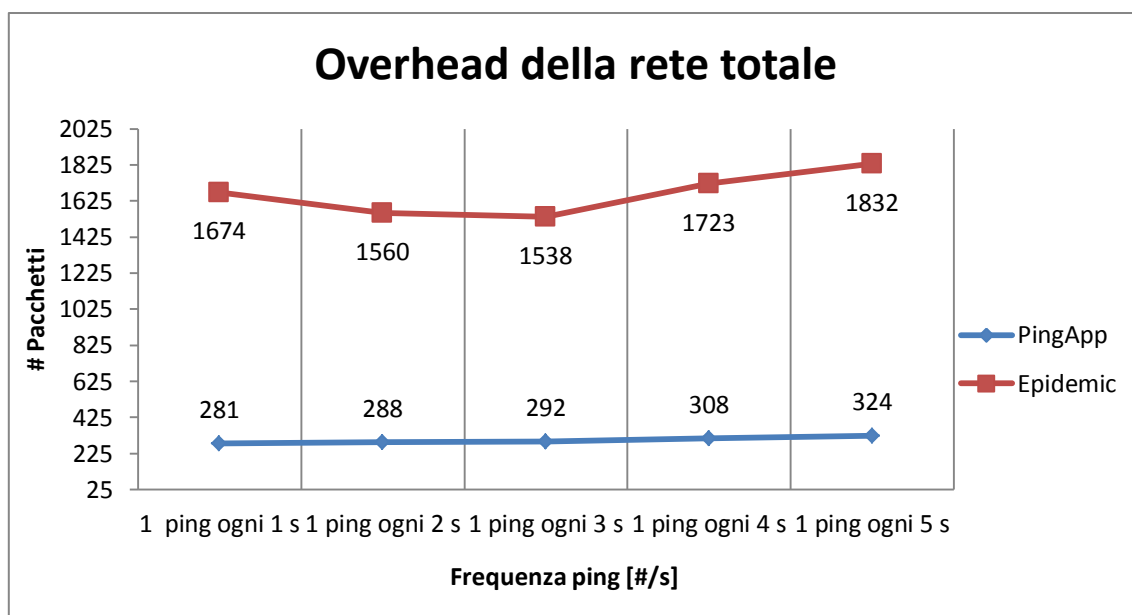
#### 4.3.1 - Protocollo Fisico Unicast

All'inizio vengono inseriti nella rete 30 nodi che si muovono ognuno in modo Lineare con un proprio angolo di movimento; i nodi vengono lasciati liberi di girare nella rete per un tempo pari a 8 [s] a scambiarsi i ping, dopodiché ogni dispositivo seleziona in modo random l'ID di un altro nodo e immette, allo stesso tempo, tale messaggio nella rete (continuando comunque a scambiarsi i ping).

Per valutare le prestazioni di tale protocollo sono state fatte due serie di simulazioni:

- nella prima serie di simulazioni viene fatta variare la frequenza dei ping mantenendo fisso il raggio di copertura di ogni nodo (a 30 [metri]);
- nella seconda serie di simulazioni viene fatto variare il raggio di copertura di ogni nodo mantenendo fissa la frequenza di invio dei ping (a 1 [ $\frac{ping}{sec}$ ]).

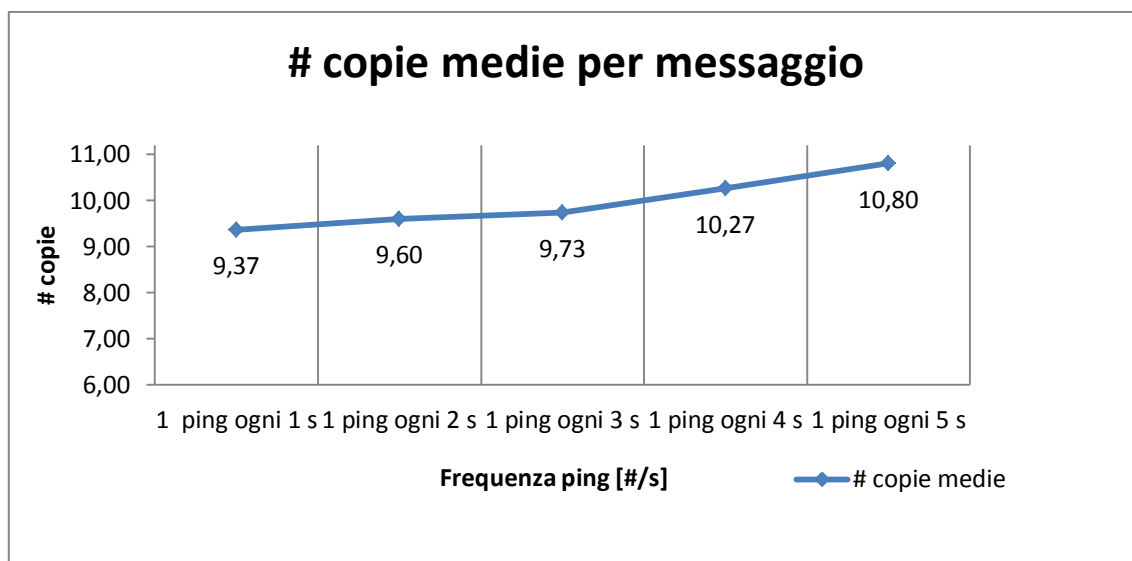
Vediamo i risultati ottenuti con la prima serie di simulazioni:



**Grafico 4.1:** mostra l'overhead della rete al variare della frequenza dei ping

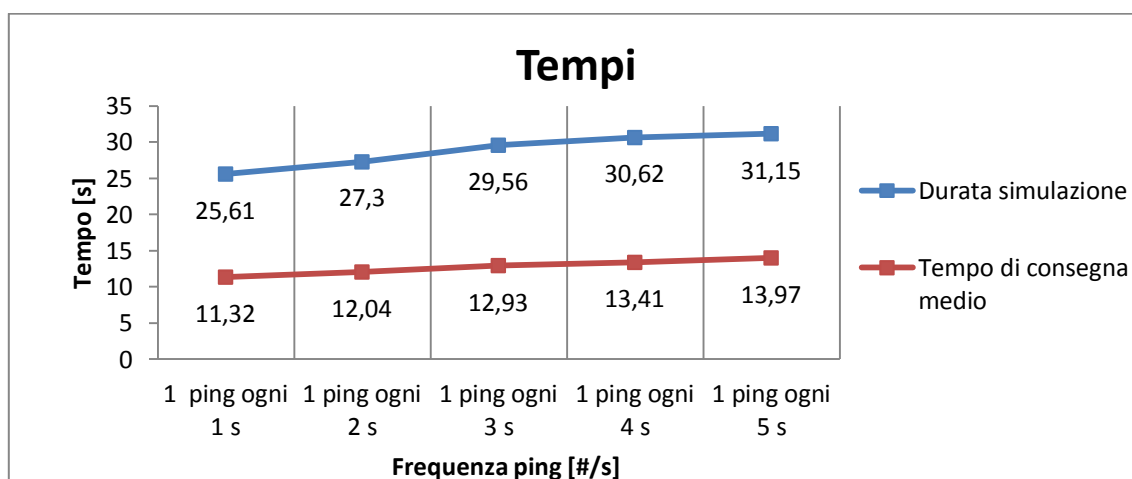
Nel Grafico 4.1 si può notare che diminuendo la frequenza di invio dei ping il traffico (in termini di messaggi) tenderà leggermente ad aumentare in quanto l'accuratezza del conteggio dei contatti è inversamente proporzionale alla frequenza di invio dei ping. Per meglio capire, si consideri il seguente esempio: prendiamo i due casi estremi di frequenza di invio dei ping ( $1 \left[ \frac{ping}{sec} \right]$  e  $1 \left[ \frac{ping}{5 sec} \right]$ ) e 4 nodi di cui 3 hanno lo stesso percorso mentre il quarto li incontra tutti insieme a cicli di 5 secondi; se la frequenza di ping è pari a  $1 \left[ \frac{ping}{5 sec} \right]$  allora dopo 20 secondi tutti i nodi avranno lo stesso numero di ping (4 nodi x 4 cicli = 16 ping per l'esattezza) risultando, ad esempio per un quinto nodo, tutti e 4 ottimi portatori per un messaggio; nel caso la frequenza dei ping sia invece di  $1 \left[ \frac{ping}{sec} \right]$ , dopo 20 secondi i 3 nodi avranno un numero di ping maggiore rispetto all'altro risultando migliori portatori. Operando così avremo un'informazione più accurata in quanto i 3 nodi potranno sempre comunicare tra di loro (dato il numero di ping maggiore) rispetto al quarto nodo che rientra invece nel range a cicli di 5 secondi. Da ciò si deduce appunto che nel primo esempio (frequenza ping =  $1 \left[ \frac{ping}{5 sec} \right]$ ) vengono fatte 4 copie del messaggio (tutti e 4 i nodi sono buoni portatori), mentre nel secondo caso (frequenza ping =  $1 \left[ \frac{ping}{sec} \right]$ ) solo 3 dato che il quarto nodo ha un numero di ping minore.

Lo stesso comportamento lo si può osservare nel Grafico 4.2 che mostra il numero di copie medio per messaggio:



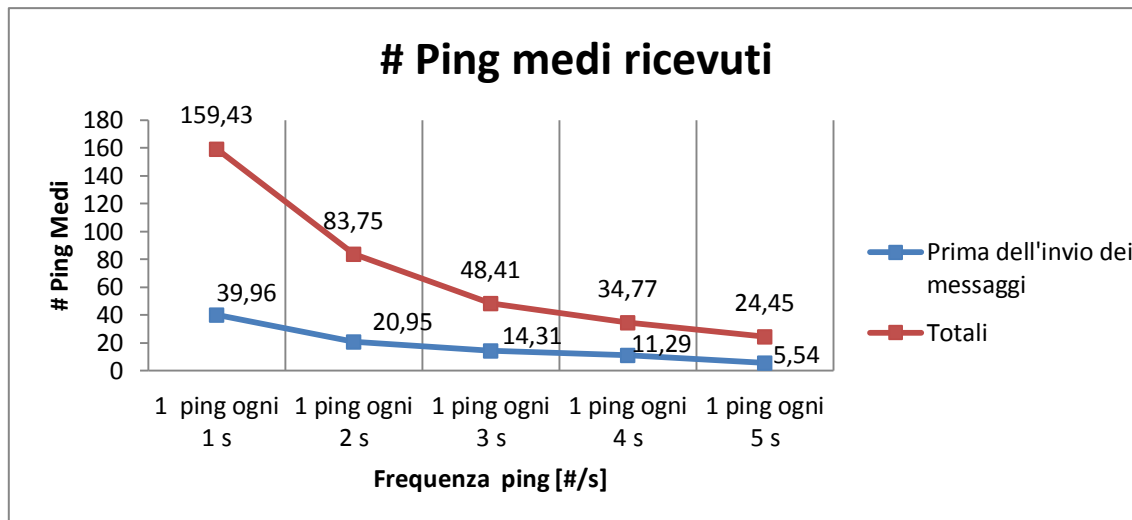
**Grafico 4.2:** mostra il numero di copie medie per messaggio al variare della frequenza dei ping

Nonostante il traffico sia leggermente maggiore, anche il tempo di simulazione e quello di consegna media (mostrati nel Grafico 4.3) sono leggermente più alti al diminuire della frequenza di invio dei ping e questo è dovuto al fatto che, nel momento d'invio dei messaggi, tutti i nodi avranno lo stesso numero di ping (visto il tempo di inizializzazione abbastanza breve pari a 8 [s]) e dunque tenderanno a non fare copie del messaggio data l' "assenza" di migliori portatori; nonostante ciò, all'aumentare del tempo i nodi tenderanno a differenziarsi maggiormente diventando migliori portatori rispetto ad altri nodi e dunque il traffico aumenterà facendo sì che i messaggi arrivino alla destinazione voluta.



**Grafico 4.3:** mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare della frequenza dei ping

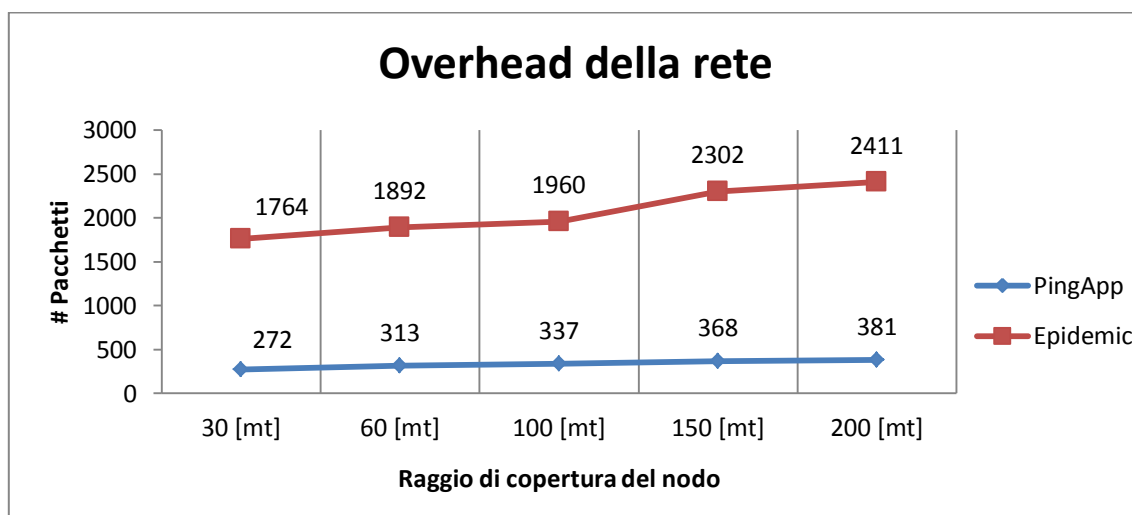
D'altro canto, il numero di ping medi ricevuti per nodo diminuisce al diminuire della frequenza d'invio dei ping stessi; si noti infatti il Grafico 4.4:



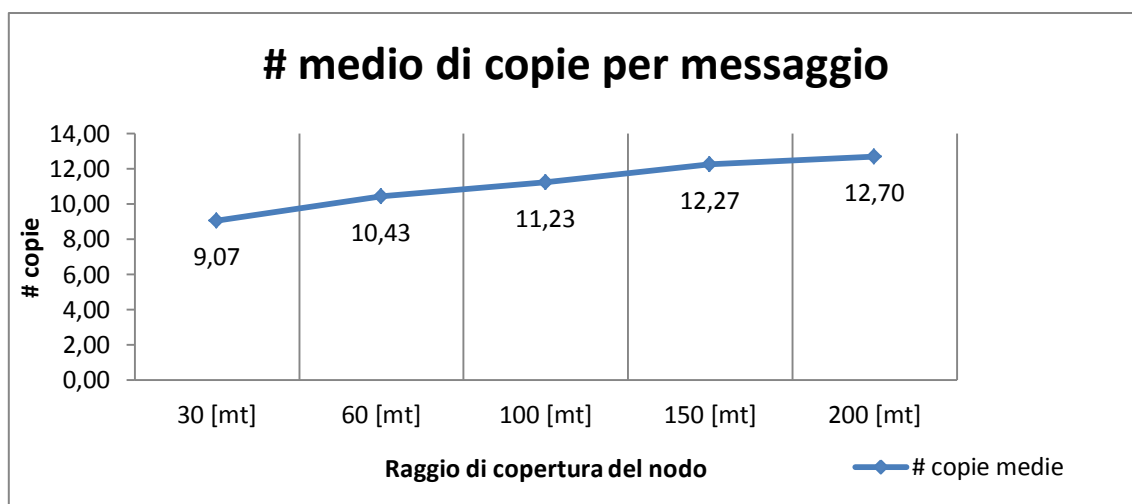
**Grafico 4.4:** mostra il numero di ping medi ricevuti per nodo prima dell'invio dei messaggi e il numero di ping medi ricevuti per nodo in totale al variare della frequenza dei ping

Nonostante la durata della simulazione aumenti, il numero medio dei ping ricevuti continuerà a diminuire al diminuire della frequenza in quanto l'aumento del tempo di simulazione non è così elevato da far aumentare anche il numero medio dei ping ricevuti.

Vediamo i risultati ottenuti con la seconda serie di simulazioni:



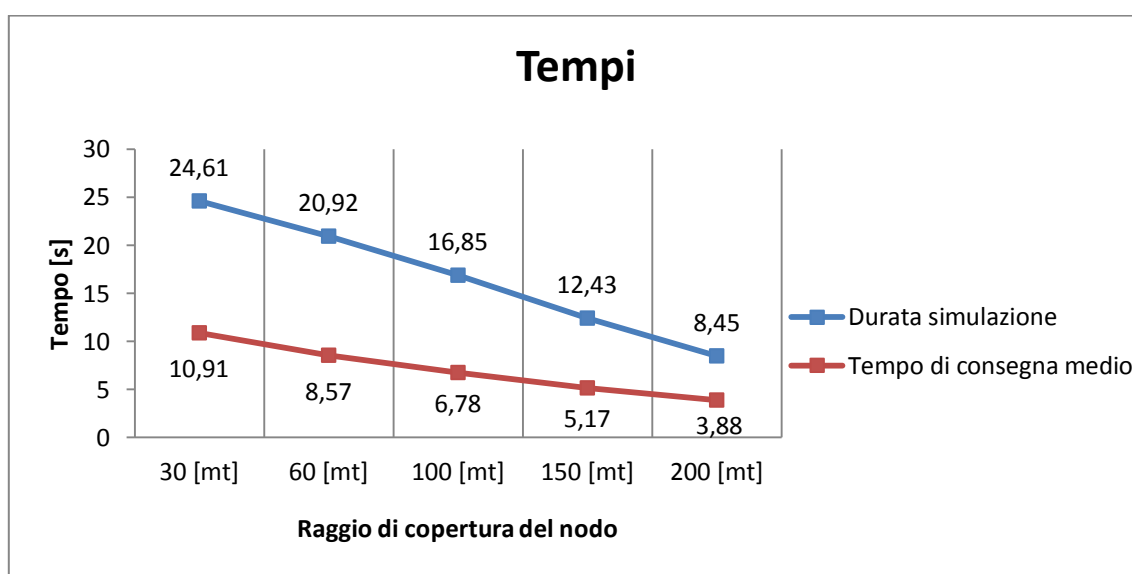
**Grafico 4.5:** mostra l'overhead della rete al variare del raggio di copertura dei nodi



**Grafico 4.6:** mostra il numero di copie medie per messaggio al variare del raggio di copertura dei nodi

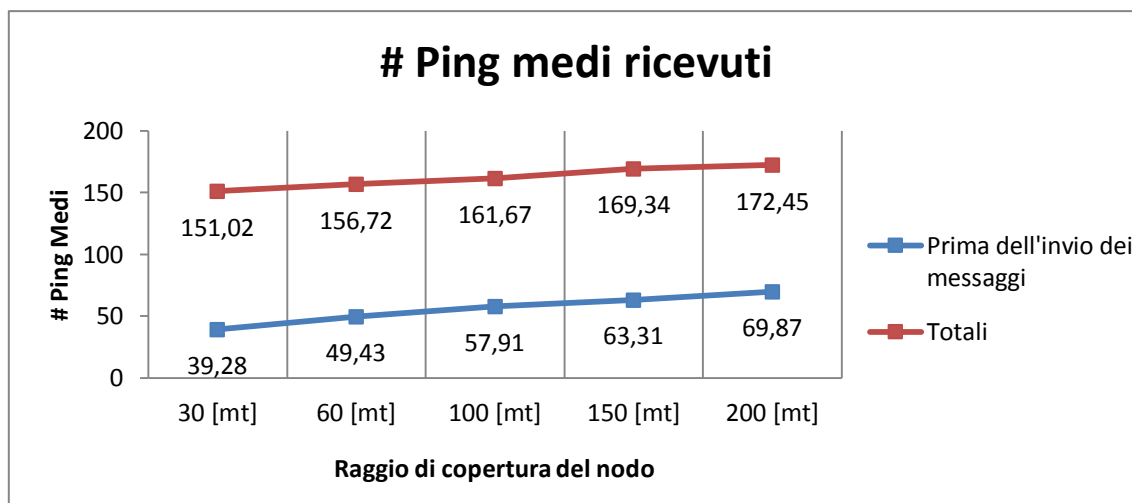
Come mostrato nei Grafici 4.5 e 4.6, all'aumentare del raggio di copertura anche l'overhead di rete e il numero di copie per messaggio tenderanno ad aumentare in quanto più nodi saranno raggiungibili e dunque più portatori potranno trasportare le copie dei vari messaggi.

D'altra parte, all'aumentare del raggio, la durata totale della simulazione e il tempo di consegna media diminuiscono in quanto molti nodi destinatari vengono raggiunti più velocemente; a dimostrazione di tale osservazione si osservi il Grafico 4.7:



**Grafico 4.7:** mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare del raggio di copertura dei nodi

Per quanto riguarda i ping ricevuti prima dell'invio dei messaggi e quelli totali si osservi il Grafico 4.8:



**Grafico 4.8:** mostra il numero di ping medi ricevuti per nodo prima dell'invio dei messaggi e il numero di ping medi ricevuti per nodo in totale al variare del raggio di copertura

Per quanto riguarda il numero medio di ping ricevuti prima dell'invio dei messaggi, si può notare un aumento del numero dei ping all'aumentare del raggio di copertura perchè più nodi rientreranno nel range di copertura di altri; invece, per il numero medio ping ricevuti in totale, questa tendenza si affievolisce in quanto, nonostante l'aumentare del raggio di copertura, vi è una forte diminuzione del tempo di simulazione che porta ad un numero leggermente superiore di ping totali.

#### 4.3.2 - Protocollo Fisico Multicast

Le modalità di inizializzazione sono identiche al caso Fisico Unicast, ovvero vengono inseriti nella rete 30 nodi che si muovono ognuno in modo Lineare con un proprio angolo di movimento; i nodi vengono lasciati liberi di girare nella rete per un tempo pari a 8 [s] per scambiarsi i ping, dopodiché ognuno di essi seleziona in modo random diversi ID dei nodi a cui inviare il messaggio; dunque in tale rete i messaggi pubblicati da ogni nodo saranno sempre 30, ma avranno più destinatari (anziché uno solo); come per il caso Fisico Unicast, i nodi anche durante l'invio dei messaggi continueranno a scambiarsi i ping.

Per valutare le prestazioni di tale protocollo sono state fatte tre serie di simulazioni:

- nella prima serie di simulazioni viene fatta variare la frequenza dei ping mantenendo fisso il raggio di copertura di ogni nodo (a 30 [metri]) e il numero dei destinatari (a 6 nodi);

- nella seconda serie di simulazioni viene fatto variare il raggio di copertura di ogni nodo mantenendo fissa la frequenza di invio dei ping (a  $1 \left[ \frac{ping}{sec} \right]$ ) e il numero dei destinatari (a 6 nodi).
- nella terza serie di simulazioni viene fatto variare il numero dei destinatari del messaggio mantenendo fisso il raggio di copertura di ogni nodo (a 30 [metri]) e la frequenza di invio dei ping (a  $1 \left[ \frac{ping}{sec} \right]$ ).

Vediamo i risultati ottenuti dalla prima serie di simulazioni:

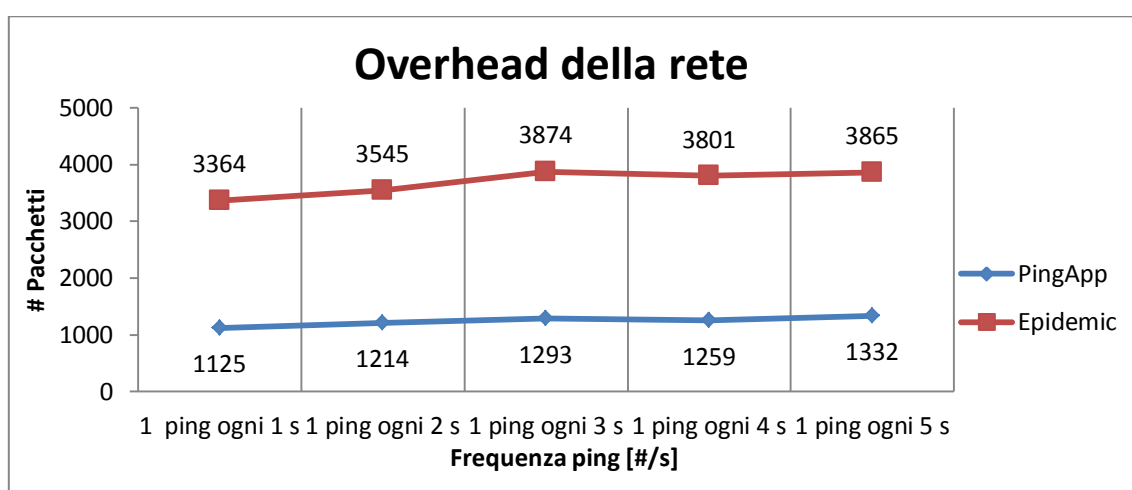


Grafico 4.9: mostra l'overhead della rete al variare della frequenza dei ping

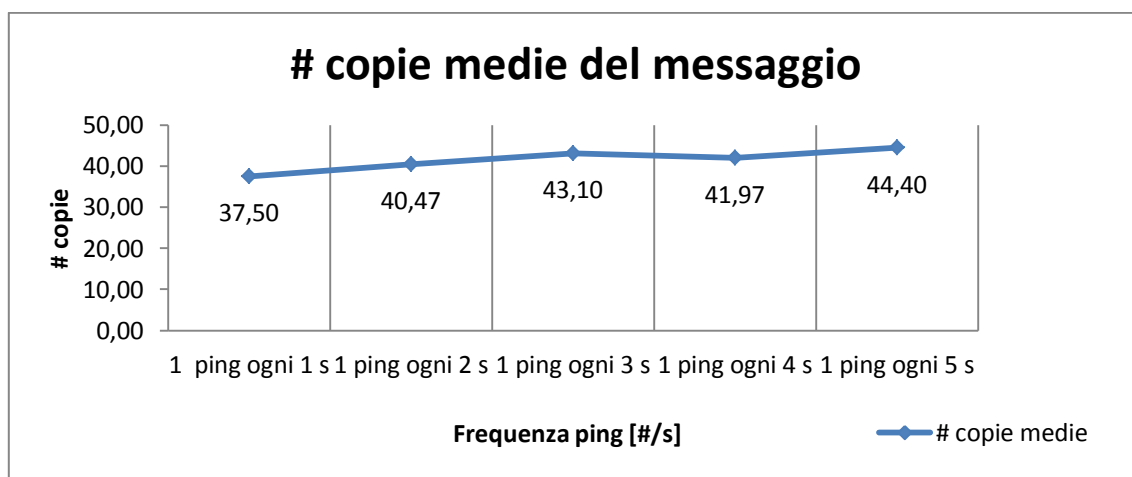
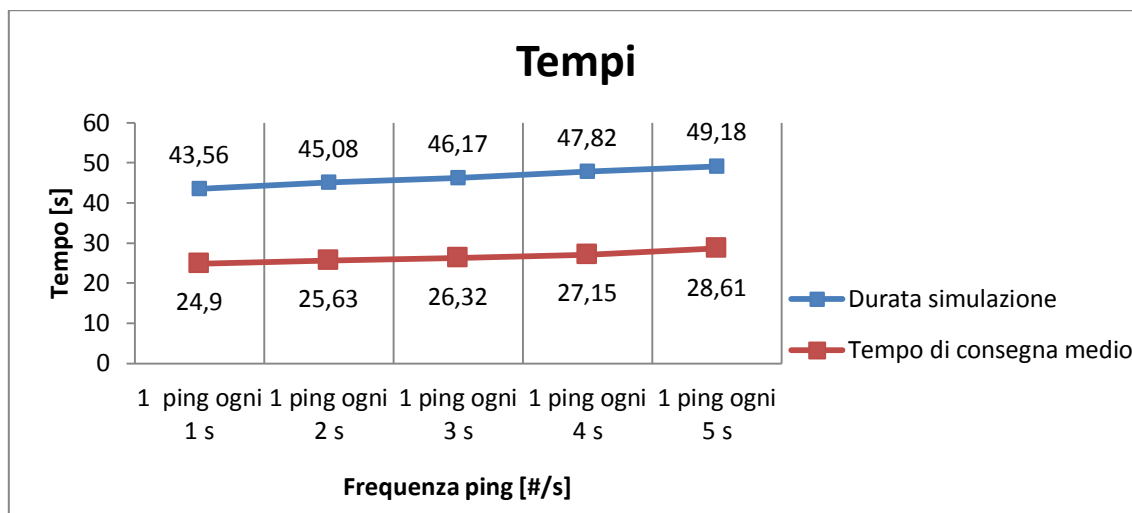


Grafico 4.10: mostra il numero di copie medie per messaggio al variare della frequenza dei ping

Il comportamento mostrato nei Grafici 4.9 e 4.10 è il medesimo di quello visto per il protocollo Fisico Unicast (Grafici 4.1 e 4.2), con la differenza che l'overhead di rete

(inteso di soli messaggi) è maggiore nel caso Fisico Multicast in quanto il numero di destinatari da raggiungere è appunto più grande (6 anziché 1).

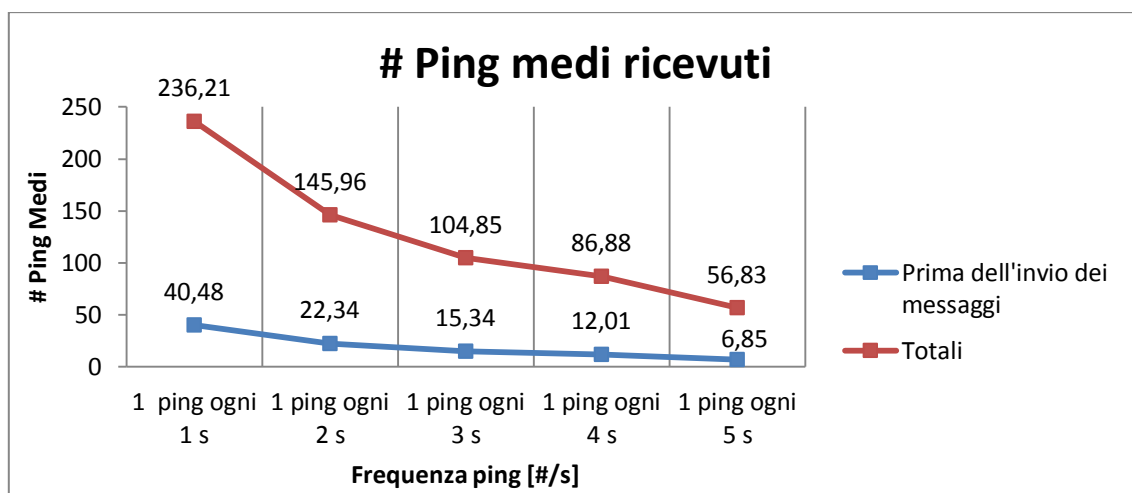
Anche dal punto di vista dei tempi di simulazione e del tempo di consegna medio, il comportamento registrato è simile al protocollo precedente; si osservi il Grafico 4.11:



**Grafico 4.11:** mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare della frequenza dei ping

Ovviamente, rispetto al caso precedente, i tempi sono maggiori in quanto i messaggi da consegnare hanno 6 destinatari anziché 1.

Per quanto riguarda i ping si osservi il Grafico 4.12:



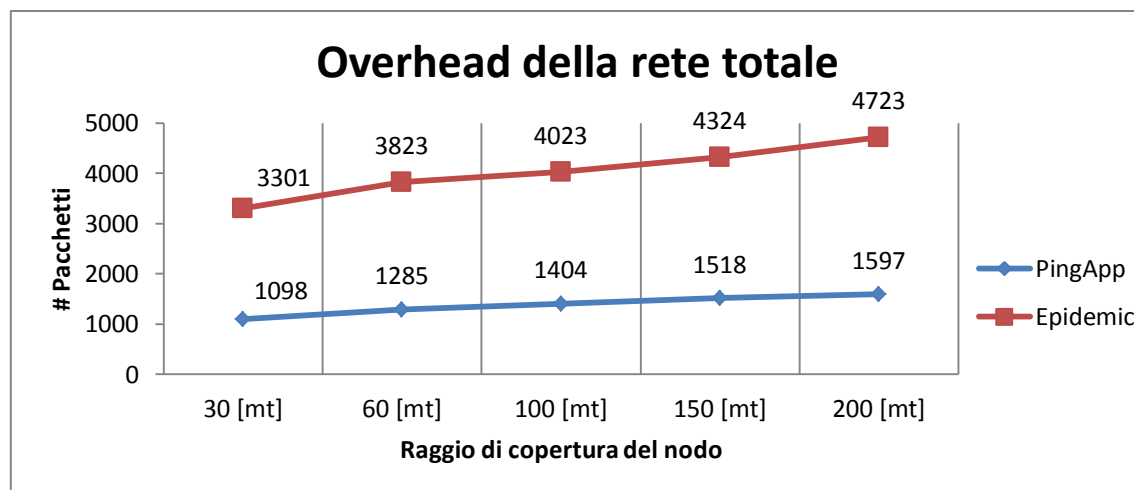
**Grafico 4.12:** mostra il numero di ping medi ricevuti per nodo prima dell'invio dei messaggi e il numero di ping medi ricevuti per nodo in totale al variare della frequenza dei ping

Il comportamento è simile al protocollo precedente; al diminuire della frequenza allora si osserva anche un diminuire del numero dei ping seppur si presenti un aumento del

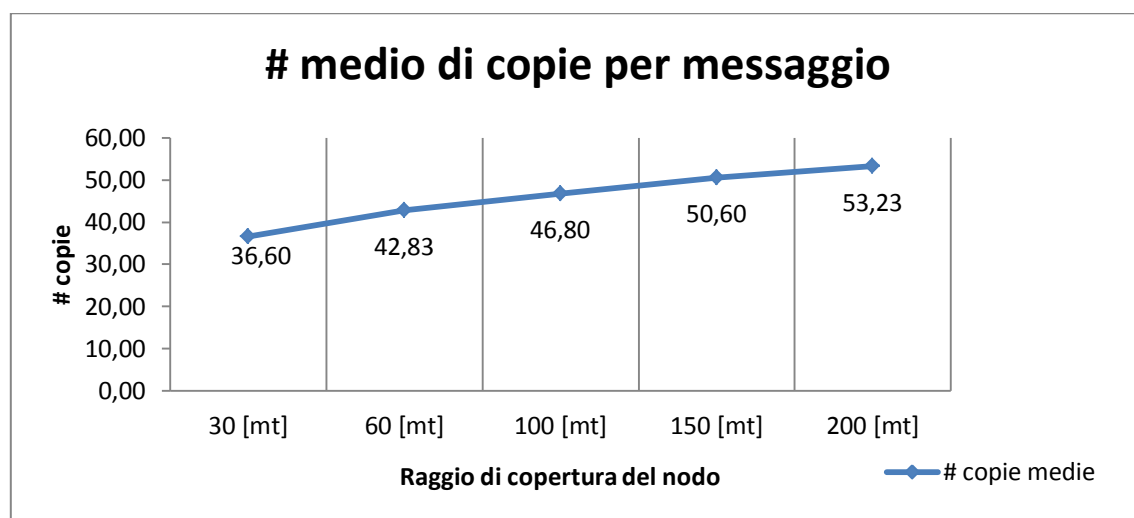


tempo totale di simulazione che non risulta sufficiente però a invertire il trend al ribasso.

Vediamo i risultato ottenuti dalla seconda serie di simulazioni:



**Grafico 4.13:** mostra l'overhead della rete al variare del raggio di copertura dei nodi



**Grafico 4.14:** mostra il numero di copie medie per messaggio al variare del raggio di copertura dei nodi

Come nel caso Fisico Unicast, all'aumentare del raggio di copertura, l'overhead di rete (vedi Grafico 4.13) e il numero di copie per messaggio (vedi Grafico 4.14) tenderanno ad aumentare in quanto più nodi saranno raggiungibili e dunque più portatori potranno trasportare le copie dei vari messaggi.

All'aumentare del raggio di copertura dei nodi, la durata totale della simulazione e il tempo di consegna media diminuiranno in quanto molti nodi destinatari verranno raggiunti più velocemente; tale comportamento si può osservare nel Grafico 4.15:

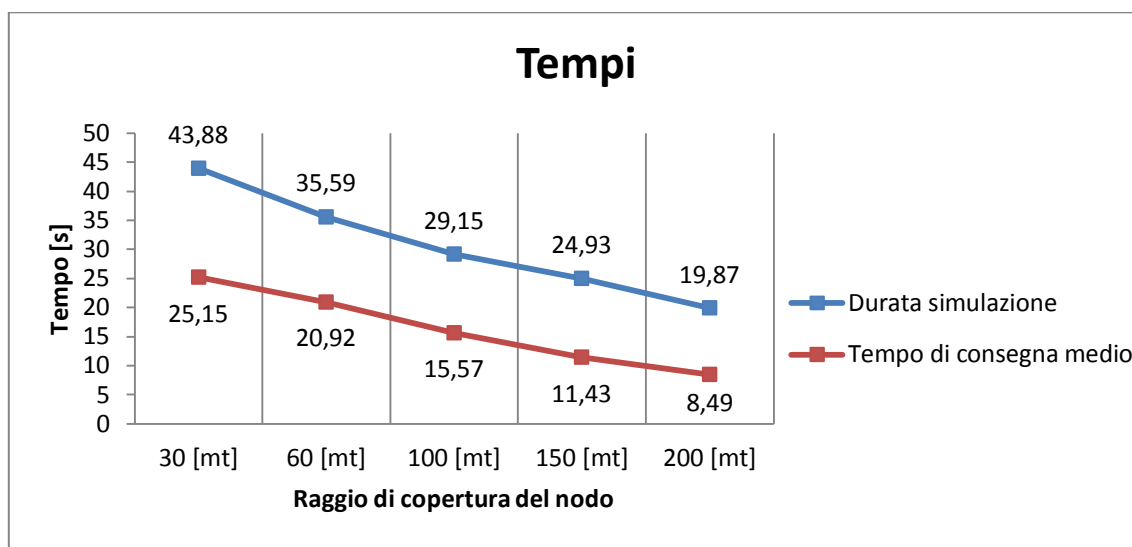


Grafico 4.15: mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare del raggio di copertura dei nodi

Nel Grafico 4.16 viene proposto l'andamento dei ping all'aumentare del raggio di copertura:

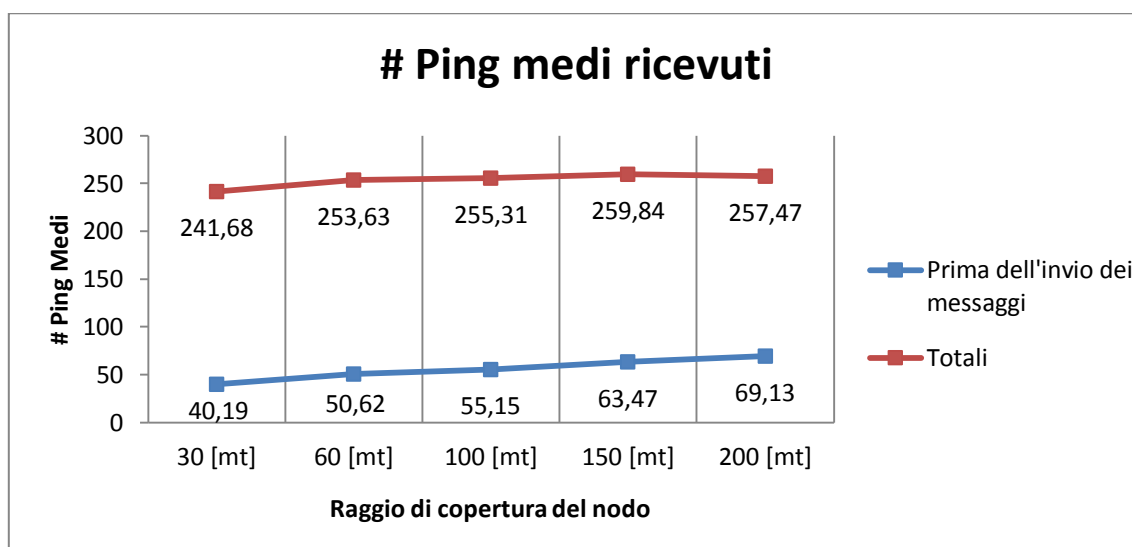


Grafico 4.16: mostra il numero di ping medi ricevuti per nodo prima dell'invio dei messaggi e il numero di ping medi ricevuti per nodo in totale al variare del raggio di copertura

Per quanto riguarda il numero medio di ping ricevuti prima dell'invio dei messaggi, si ha un aumento di tale numero al crescere del raggio di copertura in quanto più nodi

rientreranno nel range di copertura di altri; per quanto riguarda il numero medio di ping ricevuti in totale invece questa tendenza si affievolisce diventando quasi una linea retta, perchè, nonostante il raggio di copertura aumenti, vi è una forte diminuzione del tempo di simulazione che porta ad un numero leggermente superiore di ping totali.

Vediamo i risultati ottenuti con la terza serie di simulazioni:

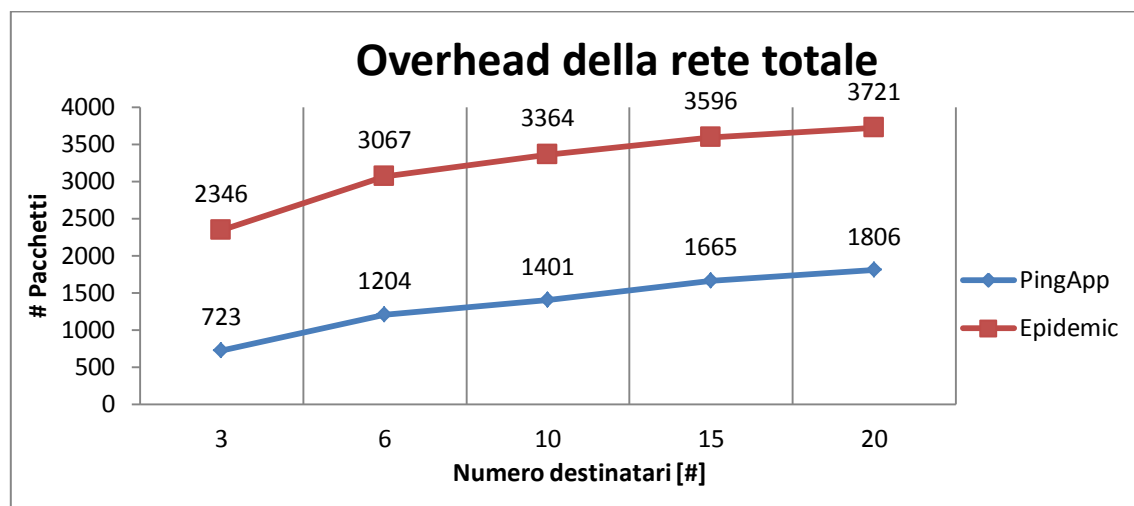


Grafico 4.17: mostra l'overhead della rete al variare del numero dei destinatari dei messaggi

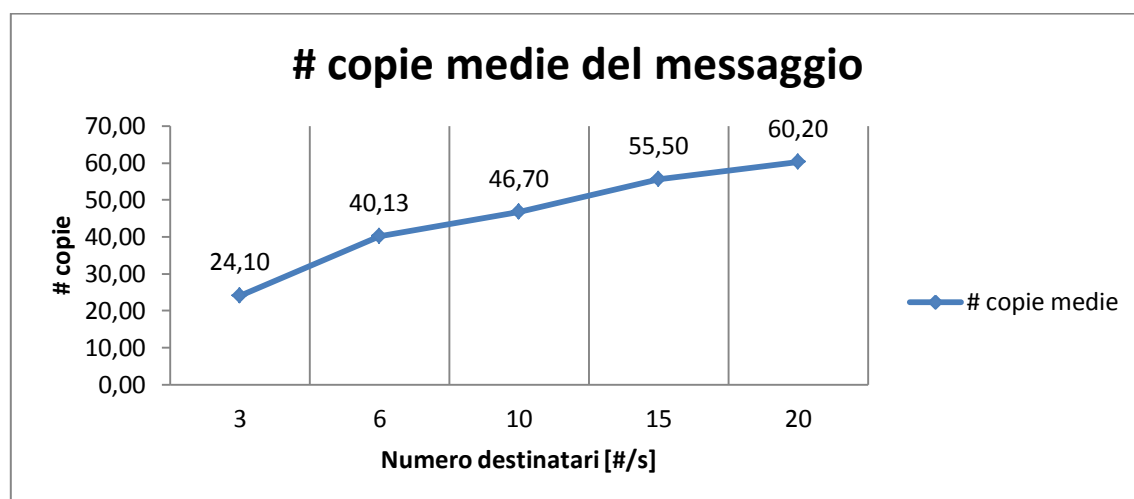
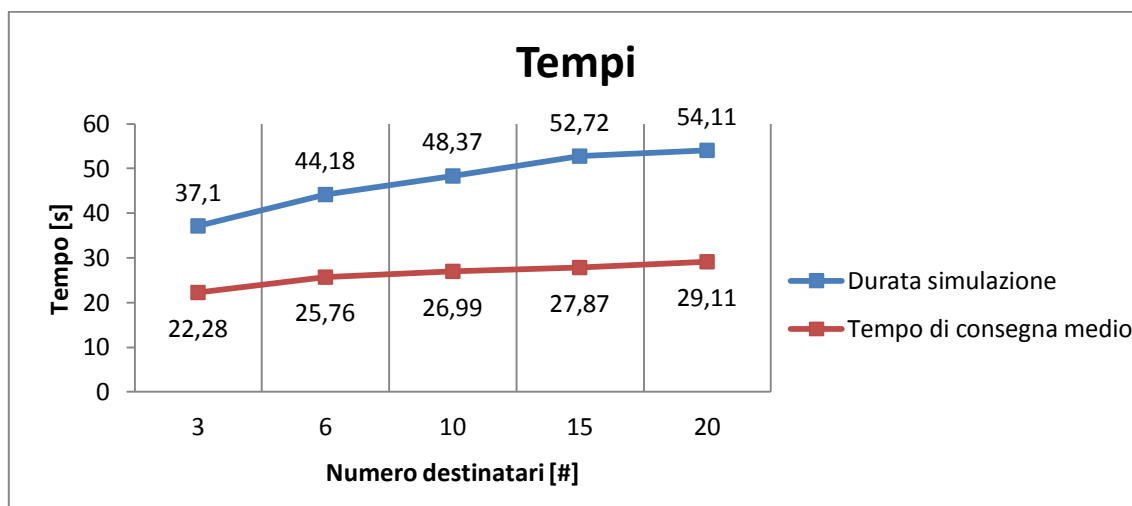


Grafico 4.18: mostra il numero di copie medie per messaggio al variare del numero dei destinatari dei messaggi

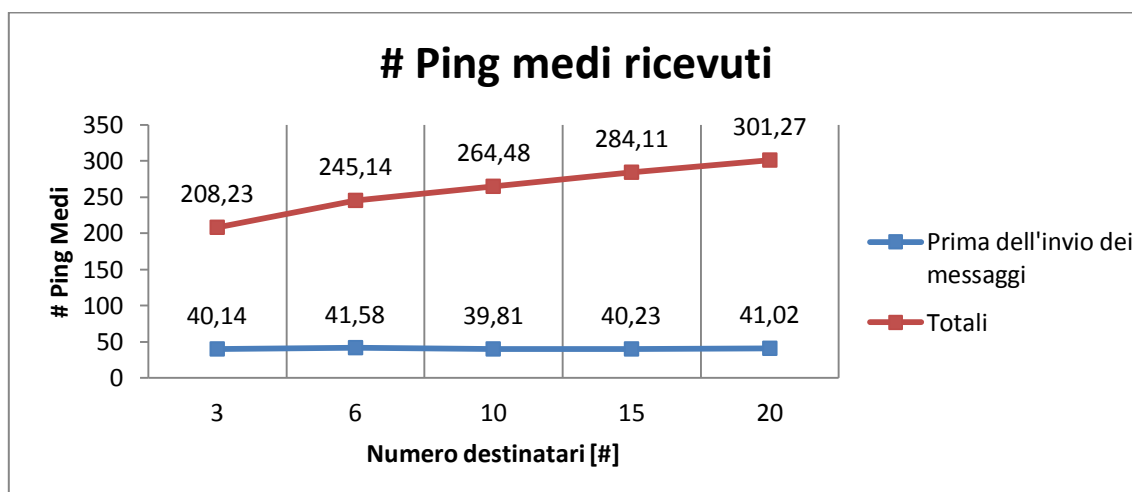
Come mostrato nei Grafici 4.17 e 4.18, si può osservare che, aumentando il numero dei destinatari di ogni messaggio anche l'overhead di rete (inteso di soli messaggi) tende ad aumentare, così come il numero di copie, in quanto più nodi dovranno essere raggiunti da più messaggi.

Come si nota nel Grafico 4.19, all'aumentare del numero dei destinatari anche il tempo di simulazione aumenta in quanto ogni messaggio dovrà raggiungere più nodi nella rete; stesso comportamento per il tempo di consegna medio che aumenta in quanto più nodi dovranno essere raggiunti e quindi si impiegherà un tempo medio maggiore per consegnare i messaggi.



**Grafico 4.19:** mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare del numero dei destinatari dei messaggi

Nel Grafico 4.20 si mostra l'andamento dei ping medi ricevuti:



**Grafico 4.20:** mostra il numero di ping medi ricevuti per nodo prima dell'invio dei messaggi e il numero di ping medi ricevuti per nodo in totale al variare del numero dei destinatari dei messaggi

Per quanto riguarda il numero di ping ricevuti prima dell'invio dei messaggi, in media tale valore è sempre uguale in quanto non vengono toccati né raggio di copertura né frequenza di invio dei ping; per quanto riguarda l'andamento dei ping totali ricevuti, invece, tale numero tende a salire e questo è dovuto al fatto della diversa durata della

simulazione: a parità di frequenza d'invio dei ping e di raggio di copertura, tanto maggiore sarà la durata di simulazione allora tanto maggiore sarà il numero di ping ricevuti in totale.

#### 4.3.3 - Protocollo Fisico Geocast

La rete viene divisa in 6 aree numerate, ottenendo un'area 2 x 3 (vedi Figura 4.1).

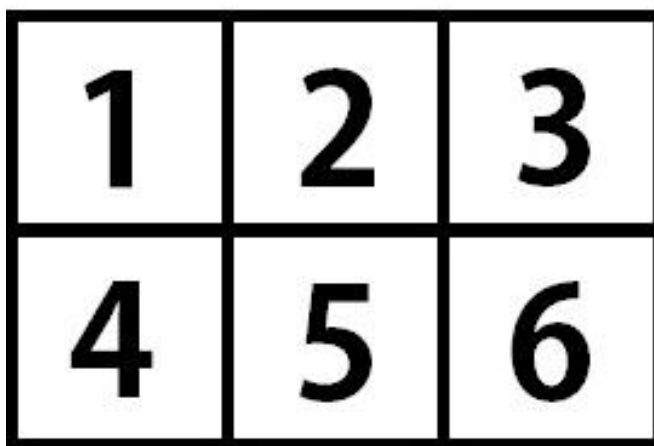


Figura 4.1: mostra la divisione della rete e la numerazione delle aree

Vengono inseriti nella rete 30 nodi a mobilità lineare e altre 6 nodi fissi (uno per ogni area della rete); questi 6 nodi non fanno altro che ricevere i messaggi destinati in quell'area; il motivo di ciò è dovuto al fatto che in qualsiasi momento durante la simulazione almeno un nodo è presente all'interno dell'area di destinazione del messaggio ottenendo così una delivery del 100%.

Per valutare le prestazioni di tale protocollo sono state fatte sei serie di simulazioni:

- nella prima serie di simulazioni viene fatta variare la frequenza dei ping mantenendo fisso il raggio di copertura di ogni nodo (a 30 [metri]) e selezionando in modo random l'area di destinazione del messaggio;
- nella seconda serie di simulazioni viene fatto variare il raggio di copertura di ogni nodo mantenendo fissa la frequenza di invio dei ping (a 1 [ $\frac{ping}{sec}$ ]) e selezionando in modo random l'area di destinazione del messaggio;
- nella terza serie di simulazioni si seleziona l'area numero 1 della rete e tutti i nodi inviano i messaggi a quell'area facendo variare la frequenza di invio dei ping e mantenendo fisso il raggio di copertura di ogni nodo (a 30 [metri]);

- nella quarta serie di simulazioni si seleziona l'area numero 1 della rete e tutti i nodi inviano i messaggi a quell'area facendo variare il raggio di copertura di ogni nodo e mantenendo fissa la frequenza di invio dei ping (a  $1 \left[\frac{ping}{sec}\right]$ );
- nella quinta serie di simulazioni si seleziona l'area numero 2 della rete e tutti i nodi inviano i messaggi a quell'area facendo variare la frequenza di invio dei ping e mantenendo fisso il raggio di copertura di ogni nodo (a  $30 [metri]$ );
- nella sesta serie di simulazioni si seleziona l'area numero 2 della rete e tutti i nodi inviano i messaggi a quell'area facendo variare il raggio di copertura di ogni nodo e mantenendo fissa la frequenza di invio dei ping (a  $1 \left[\frac{ping}{sec}\right]$ ).

Vediamo i risultati ottenuti con la prima serie di simulazioni:

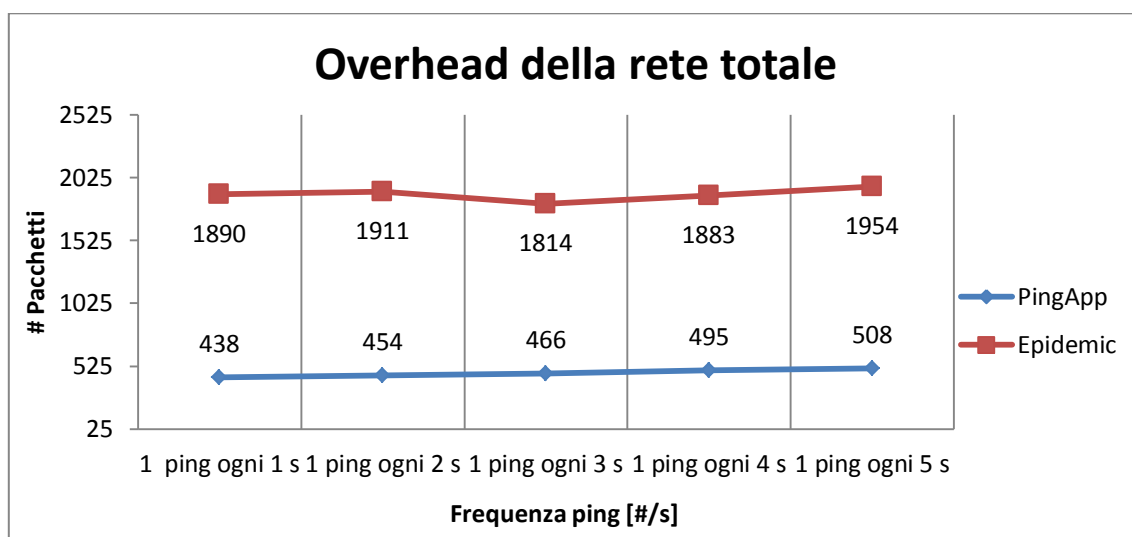


Grafico 4.21: mostra l'overhead della rete al variare della frequenza dei ping

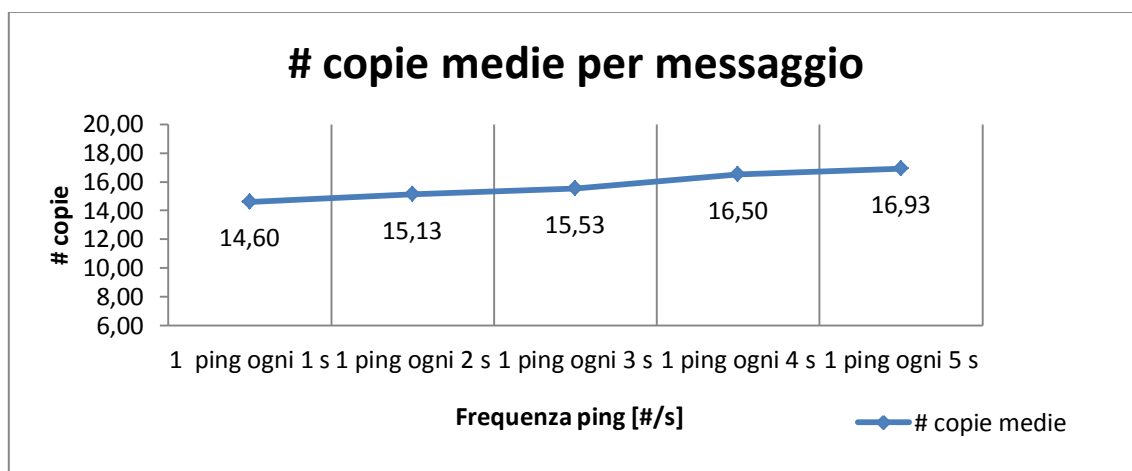
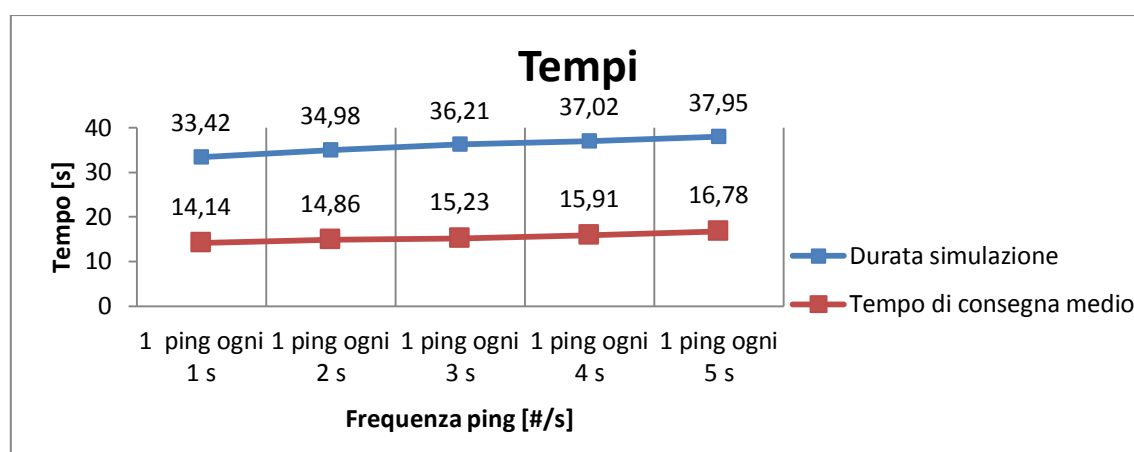


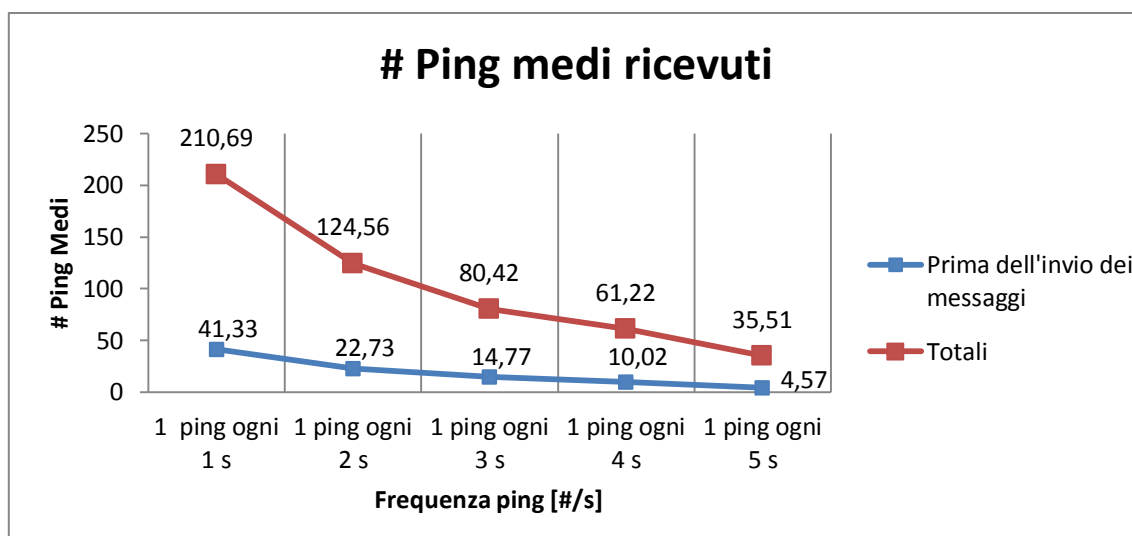
Grafico 4.22: mostra il numero di copie medie per messaggio al variare della frequenza dei ping

Come già visto per gli altri due protocolli precedenti, l'overhead di rete (vedi Grafico 4.21) e il numero di copie medie per messaggio (vedi Grafico 4.22) tenderanno ad aumentare al diminuire della frequenza di invio dei ping; il motivo è sempre dovuto alla perdita di accuratezza di quando si inviano ping con una frequenza minore; tutto questo si traduce anche in un maggior tempo di simulazione e in un maggior tempo di consegna medio (vedi Grafico 4.23) in quanto avendo un tempo di inizializzazione della rete abbastanza corto (8 [s]) al momento dell'invio dei messaggi, al diminuire della frequenza d'invio dei ping, tutti i nodi "vicini" avranno tutti lo stesso numero di ping e dunque tenderanno a consegnare il messaggio direttamente. Solo con il passare del tempo i nodi tenderanno a differenziarsi e dunque ad aumentare le copie dei propri messaggi, ottenendo così una delivery del 100%.



**Grafico 4.23:** mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare della frequenza dei ping

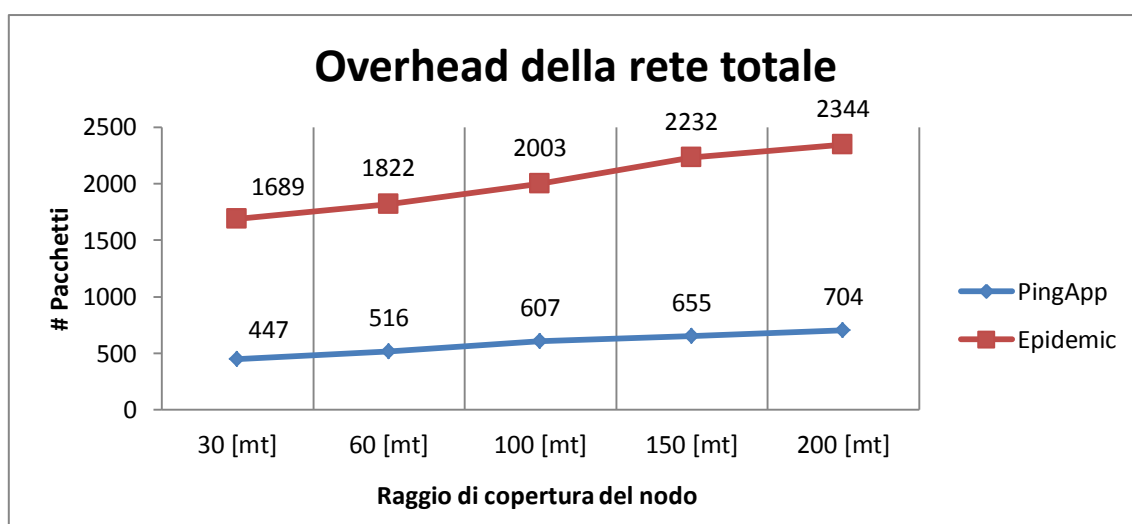
Il comportamento dal punto di vista dei ping medi ricevuti prima dell'invio dei messaggi e quelli totali lo si può osservare nel Grafico 4.24:



**Grafico 4.24:** mostra il numero di ping medi ricevuti per nodo prima dell'invio dei messaggi e il numero di ping medi ricevuti per nodo in totale al variare della frequenza dei ping

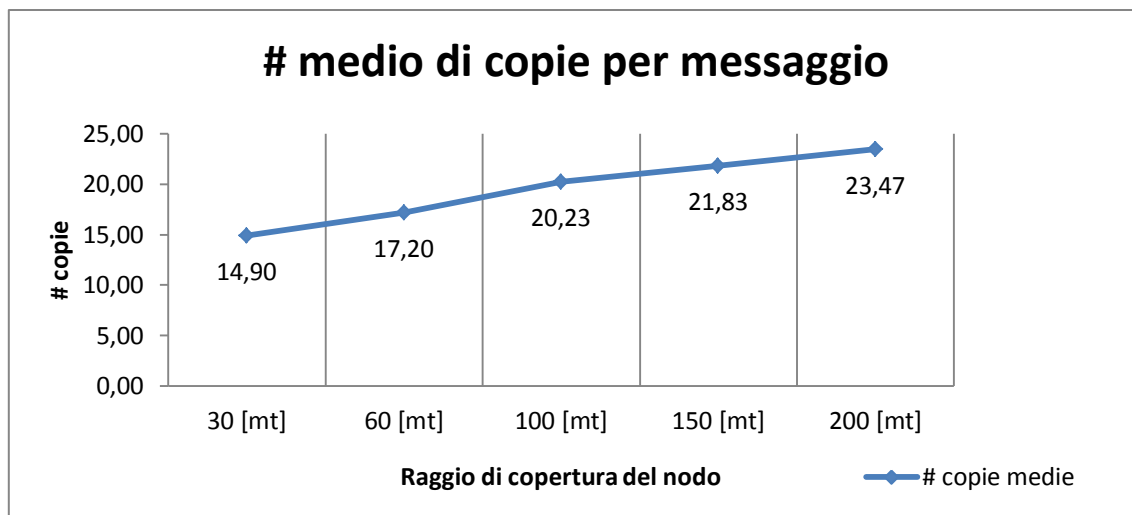
Al diminuire della frequenza, nonostante la durata della simulazione sia maggiore, i ping medi ricevuti in totale da ogni nodo diminuiscono ugualmente in quanto la durata di simulazione non è talmente lunga da compensare tale differenza.

Vediamo i risultati ottenuti con la seconda serie di simulazioni:



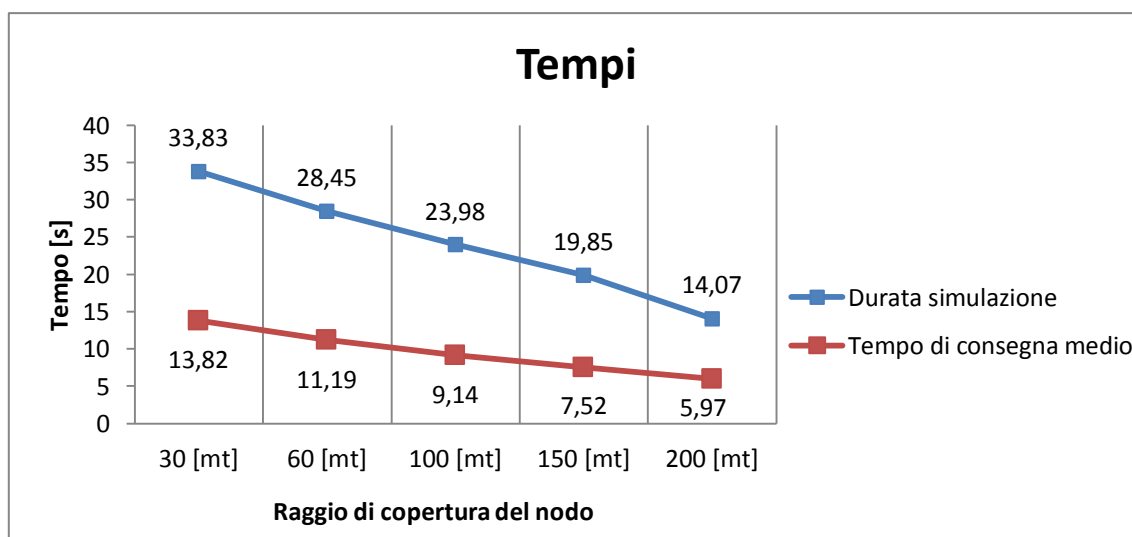
**Grafico 4.25:** mostra l'overhead della rete al variare del raggio di copertura dei nodi





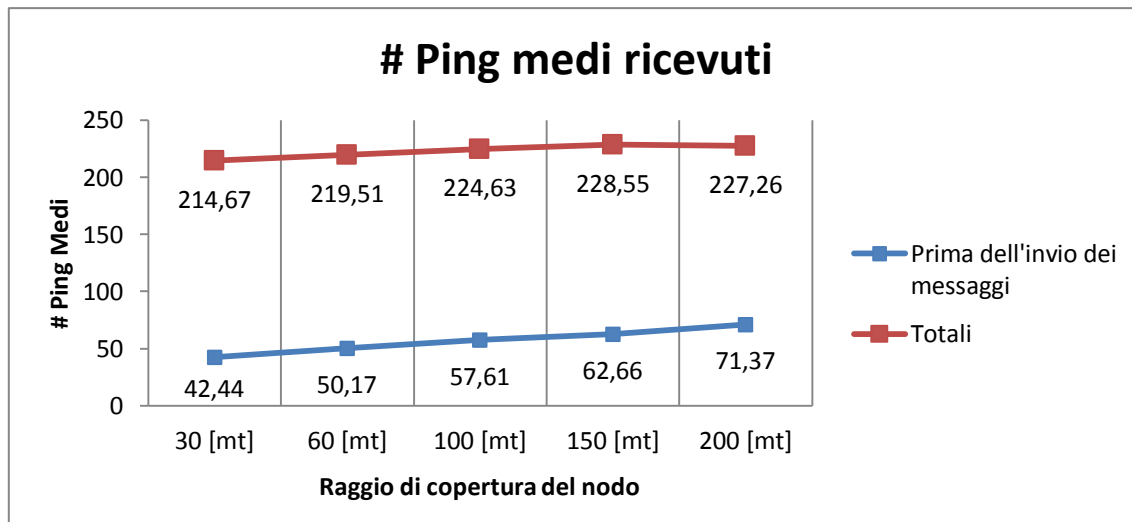
**Grafico 4.26:** mostra il numero di copie medie per messaggio al variare del raggio di copertura dei nodi

Come mostrato nei Grafici 4.25 e 4.26, all'aumentare del raggio di copertura si può notare un aumento dell'overhead di rete in quanto più nodi processeranno un messaggio e dunque creeranno più copie dello stesso. Il tempo di simulazione e il tempo di consegna medio (vedi Grafico 4.27) diminuiscono all'aumentare del raggio di copertura in quanto più nodi vengono raggiunti e dunque il messaggio arriverà a destinazione in un tempo più breve perché gestito da più nodi.



**Grafico 4.27:** mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare del raggio di copertura dei nodi

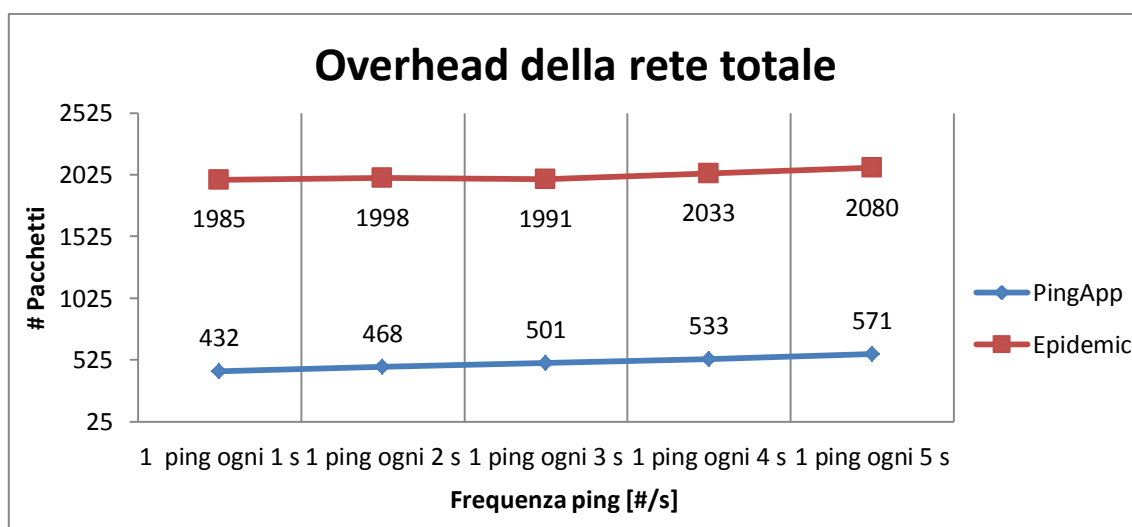
L'andamento del numero di ping medi ricevuti prima dell'invio dei messaggi e quelli totali sono rappresentati nel Grafico 4.28:



**Grafico 4.28:** mostra il numero di ping medi ricevuti per nodo prima dell'invio dei messaggi e il numero di ping medi ricevuti per nodo in totale al variare del raggio di copertura

Nonostante il tempo di simulazione diminuisca, il numero medio di ping ricevuti in totale tende leggermente ad aumentare, e questo perché con un maggior raggio di copertura vengono pingati di conseguenza più nodi e il tempo di simulazione non è sufficientemente corto da far diminuire il numero di ping medi.

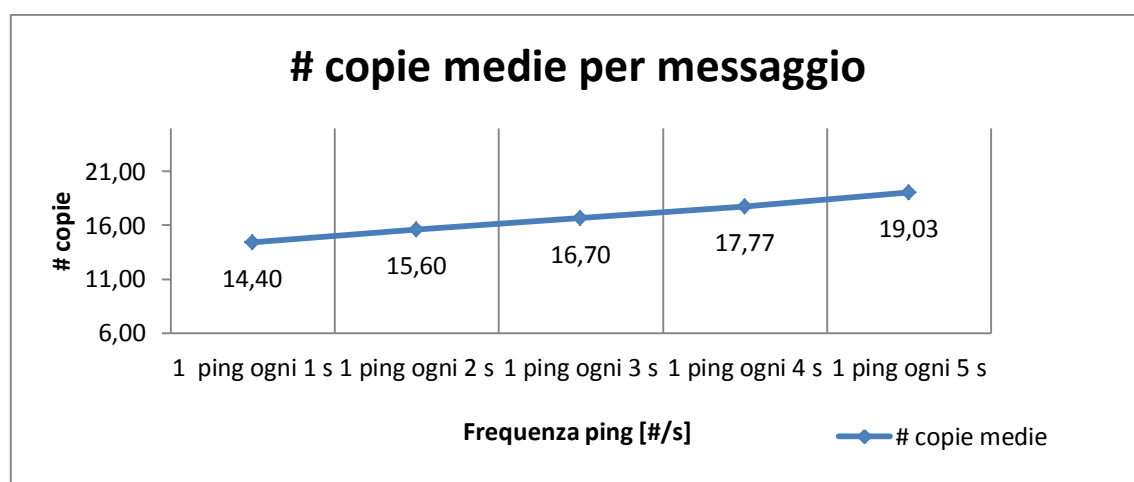
Vediamo ora i risultati delle altre 4 serie di simulazioni; nel Grafico 4.29 viene mostrato l'overhead di rete, al variare della frequenza di invio dei ping, nel caso in cui tutti i messaggi sono destinati all'Area 1 del playground:



**Grafico 4.29:** mostra l'overhead della rete al variare della frequenza dei ping con tutti i messaggi diretti verso l'Area 1

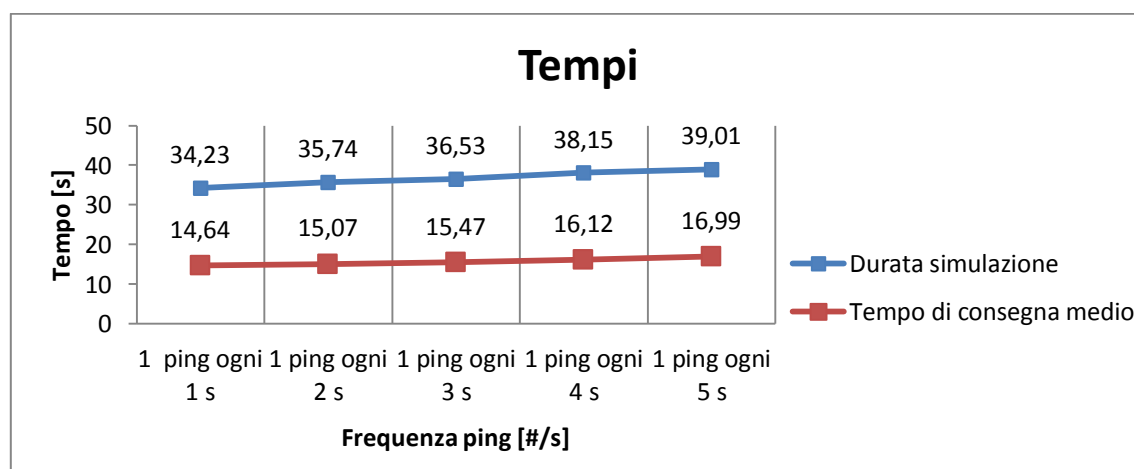
Come si può notare, nel Grafico 4.29 l'overhead tende ad aumentare al diminuire della frequenza di invio dei ping; il motivo è sempre dovuto alla perdita di accuratezza di quando si inviano ping con una frequenza minore; risultati simili sono stati trovati facendo inviare tutti i messaggi verso l'Area 2.

Stesso ragionamento vale, dunque, per il numero di copie sia nel caso in cui tutti i messaggi sono destinati all'Area 1 (vedi Grafico 4.30) del playground sia nel caso in cui tutti i messaggi sono destinati all'Area 2:



**Grafico 4.30:** mostra il numero di copie medie per messaggio al variare della frequenza dei ping con tutti i messaggi diretti verso l'Area 1

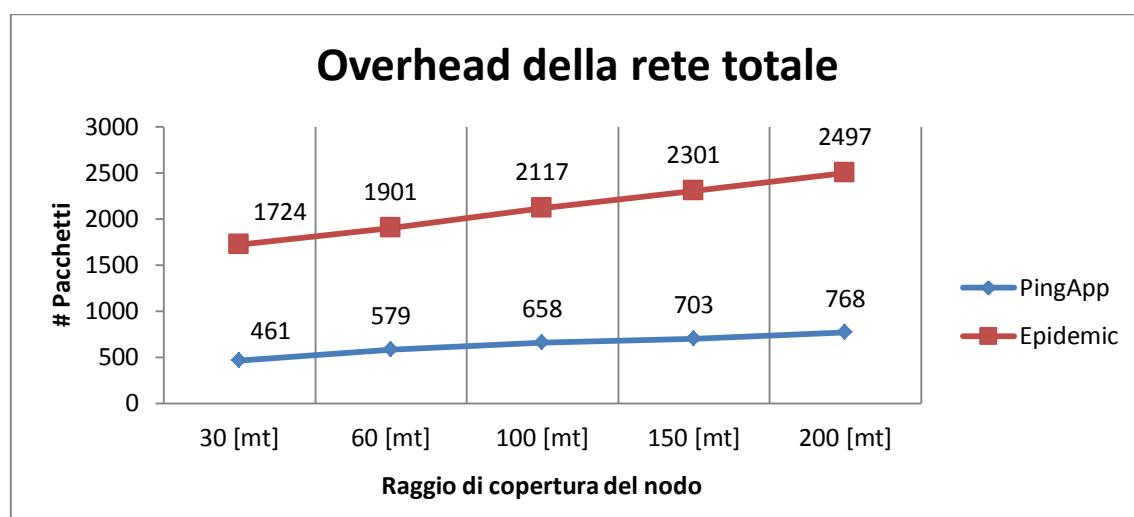
La durata di simulazione e il tempo di consegna medio (nel caso in cui tutti i messaggi sono diretti verso l'Area 1) vengono presentati nel Grafico 4.31:



**Grafico 4.31:** mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare della frequenza dei ping con tutti i messaggi diretti verso l'Area 1

Il tempo di simulazione e quello di consegna medio presentati nel Grafico 4.31 tendono ad aumentare in quanto avendo un tempo di inizializzazione della rete abbastanza breve (8 [s]) al momento dell'invio dei messaggi, al diminuire della frequenza d'invio dei ping, tutti i nodi "vicini" avranno tutti lo stesso numero di ping e dunque tenderanno a consegnare il messaggio direttamente anziché inviarlo ad altri nodi; solo con il passare del tempo i nodi tenderanno a differenziarsi e dunque ad aumentare le copie dei propri messaggi, causando dunque un tempo di simulazione e di consegna maggiori. Risultati simili sono stati trovati anche per l'Area 2 del playground.

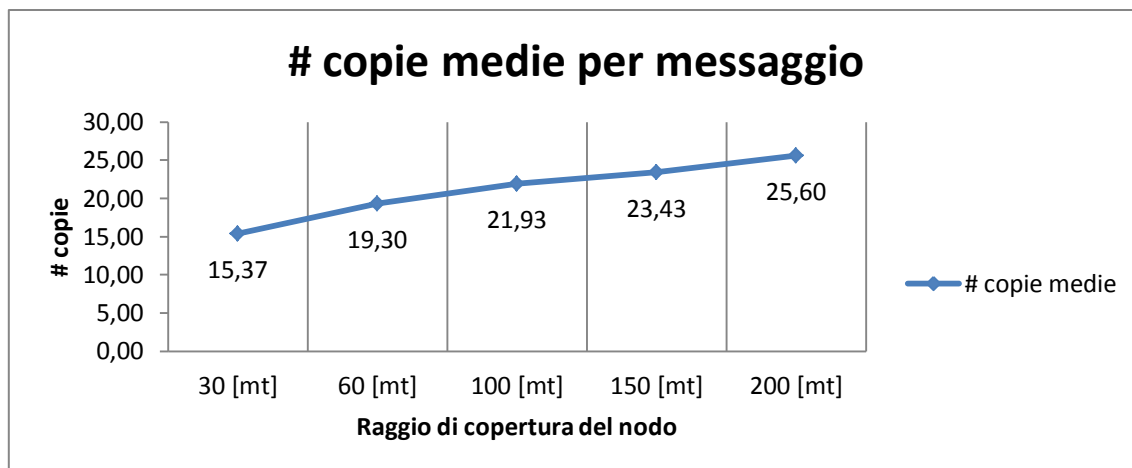
Nel Grafico 4.32 viene mostrato l'overhead di rete, al variare del raggio di copertura, nel caso in cui tutti i messaggi sono destinati all'Area 1 del playground:



**Grafico 4.32:** mostra l'overhead della rete al variare del raggio di copertura dei nodi con tutti i messaggi diretti verso l'Area 1

Come si può notare, all'aumentare del raggio di copertura, più messaggi vengono inviati in quanto più nodi potranno risultare come migliori portatori; medesimo comportamento è stato osservato per l'Area 2 del playground.

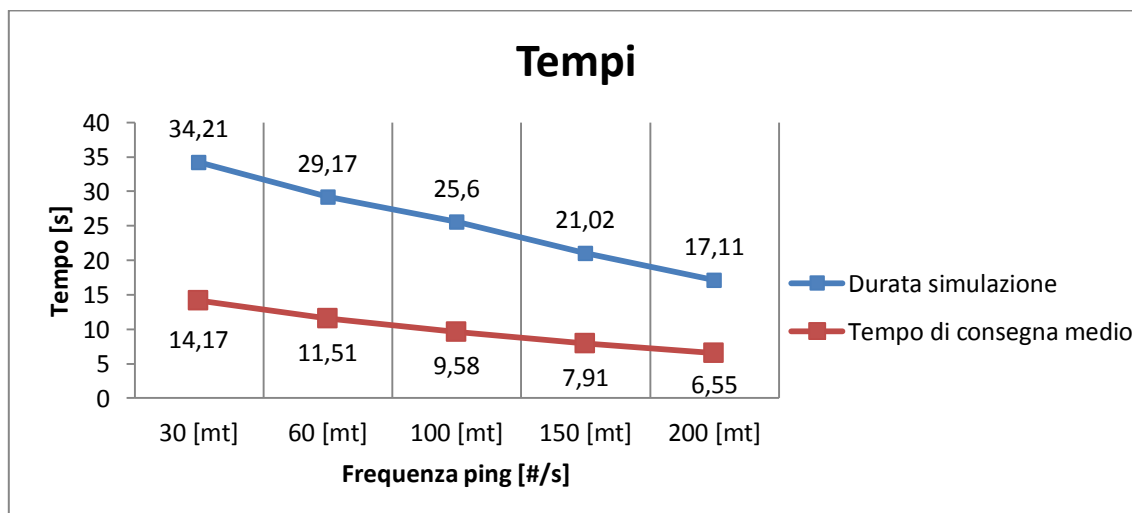
A tal proposito si osservi il Grafico 4.33 (con tutti i messaggi diretti verso l'Area 1) che mostra il numero di copie medie per messaggio:



**Grafico 4.33:** mostra il numero di copie medie per messaggio al variare del raggio di copertura dei nodi con tutti i messaggi diretti verso l'Area 1

Il numero di copie medie nel Grafico 4.33 tenderà ad aumentare per il motivo sopra citato; anche qui, è stato osservato lo stesso comportamento nel caso in cui tutti i messaggi fossero diretti verso l'Area 2 del playground.

Il tempo di simulazione e quello di consegna medio diminuiranno all'aumentare del raggio di copertura in quanto i nodi verranno raggiunti più velocemente; si osservi a tal proposito il Grafico 4.34:



**Grafico 4.34:** mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare del raggio di copertura dei nodi con tutti i messaggi diretti verso l'Area 1

Risultati simili sono stati ottenuti nel caso in cui tutti i messaggi fossero diretti verso l'Area 2 del playground.

## 4.4 - Analisi prestazioni per Protocollo Applicativo

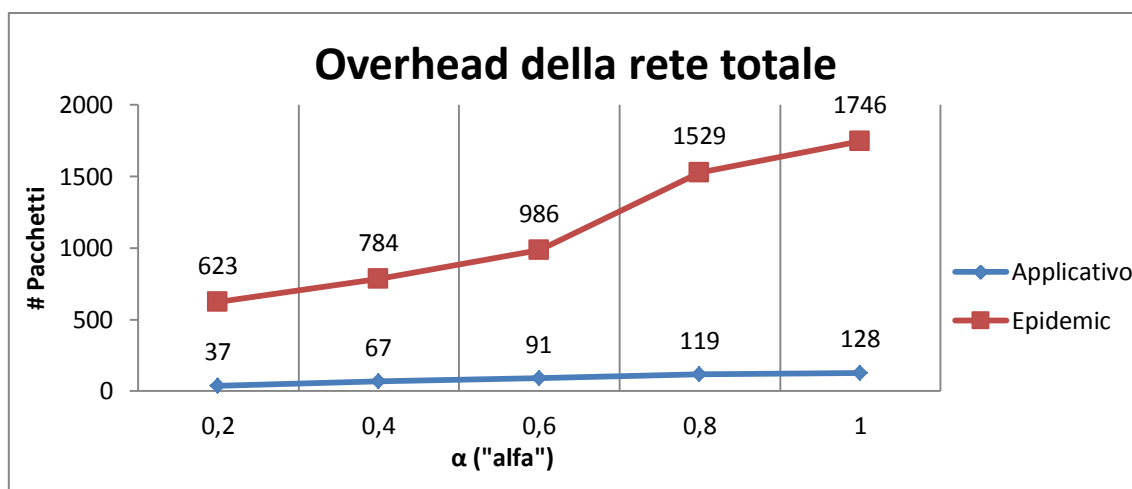
### 4.4.1 - Protocollo Applicativo Unicast

Per simulare tale protocollo sono stati inseriti nella rete 6 nodi fissi (a 2 valori precisi sull'asse delle  $y$ ) più altri 30 nodi; di questi 30 nodi, fissato un  $0 < \alpha \leq 1$ , una parte ( $30\alpha$ ) è libera di muoversi nella rete, l'altra ( $30 - 30\alpha$ ) si muove orizzontalmente allo stesso livello dei nodi fissi; esattamente metà dei nodi che si muovono orizzontalmente attraverserà i tre nodi fissi più in alto nella rete mentre l'altra metà attraverserà gli altri tre nodi fissi di destinazione; per semplicità tali nodi li chiamerò nodi "amici" in quanto se due nodi sono amici allora tenderanno ad incontrarsi maggiormente rispetto ad altri. Anche i nodi lineari che si muovono nella rete hanno una loro lista amici dove in modo random sono presenti 2 dei 6 nodi destinatari. Al momento dell'invio del messaggio, ogni nodo libero di muoversi linearmente ( $30\alpha$ ) selezionerà l'ID di uno dei 6 nodi fissi e procederà a inoltrare il pacchetto verso di lui secondo la funzione obiettivo scritta nel Capitolo 3.

Per valutare le prestazioni di tale protocollo sono state fatte 3 serie di simulazioni:

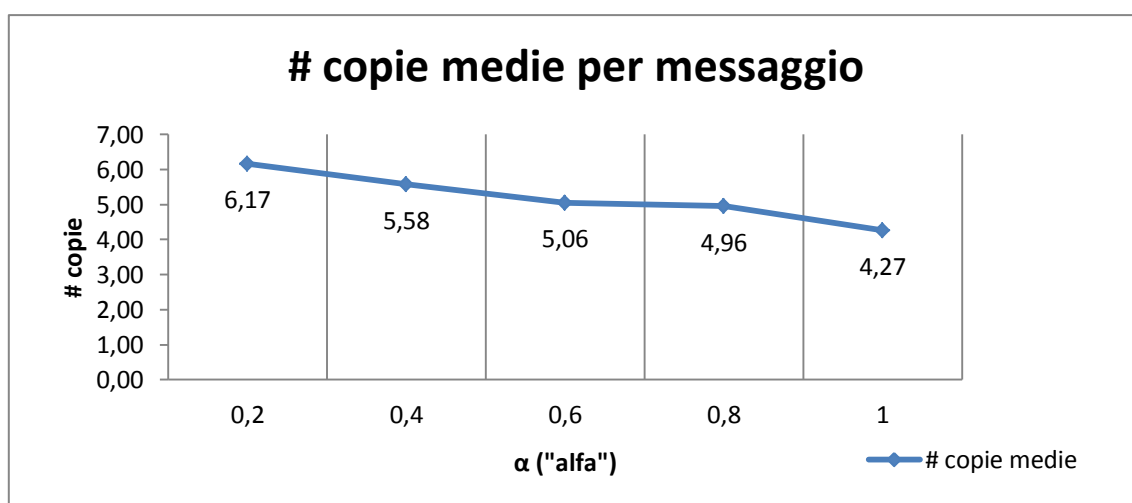
- nella prima serie di simulazioni viene fatto variare il valore di  $\alpha$  (0.2 - 0.4 - 0.6 - 0.8 - 1) tenendo fisso il raggio di copertura a 30 [*metri*];
- nella seconda serie di simulazioni viene fatto variare il valore di  $\alpha$  (0.2 - 0.4 - 0.6 - 0.8 - 1) tenendo fisso il raggio di copertura a 60 [*metri*];
- nella terza serie di simulazioni viene fatto variare il valore di  $\alpha$  (0.2 - 0.4 - 0.6 - 0.8 - 1) tenendo fisso il raggio di copertura a 100 [*metri*];

Vediamo i risultati per la prima serie di simulazioni:



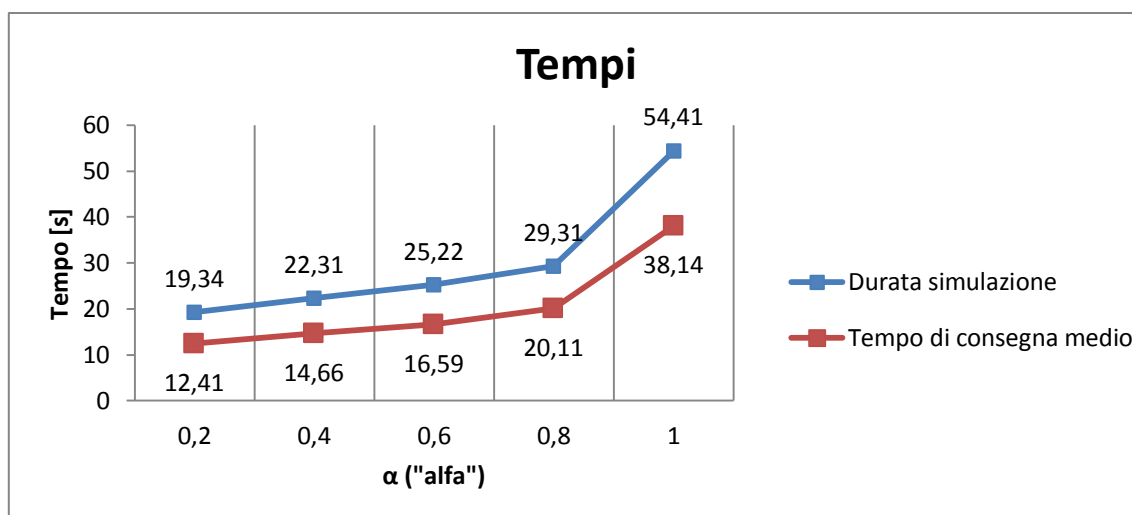
**Grafico 4.35:** mostra l'overhead della rete al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri]

Nel Grafico 4.35 viene mostrato l'overhead di rete: all'aumentare di  $\alpha$  il traffico aumenta in quanto più nodi genereranno messaggi; ad esempio con  $\alpha = 0,2$  saranno presenti in rete 6 nodi liberi ( $30 \times 0,2$ ) e dunque verranno generati solo 6 messaggi, mentre già con  $\alpha = 0,4$  verranno immessi in rete 12 messaggi diversi. D'altra parte il numero di copie medie per messaggio (vedi Grafico 4.36) tenderà a diminuire in quanto il numero di nodi "amici" ( $30 - 30\alpha$ ) decresce all'aumentare di alfa:



**Grafico 4.36:** mostra il numero di copie medie per messaggio al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri]

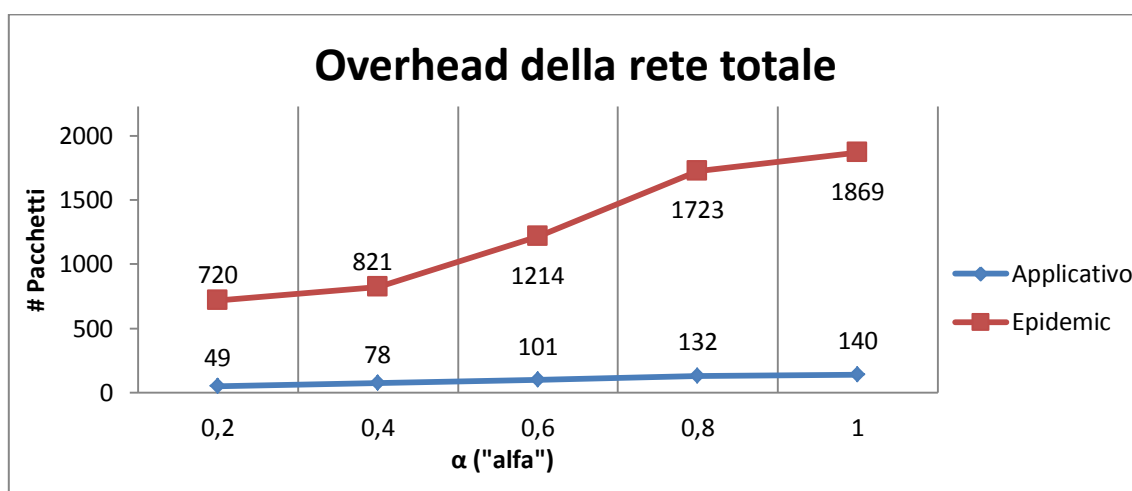
Anche il tempo di simulazione e il tempo di consegna medio (vedi Grafico 4.37), all'aumentare del parametro  $\alpha$  aumentano in quanto ci saranno meno portatori (dato che il numero di nodi "amici" decresce) che porteranno il messaggio a destinazione e dunque i tempi si allungheranno di conseguenza:



**Grafico 4.37:** mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri]

Vediamo ora i risultati della seconda serie di simulazioni:

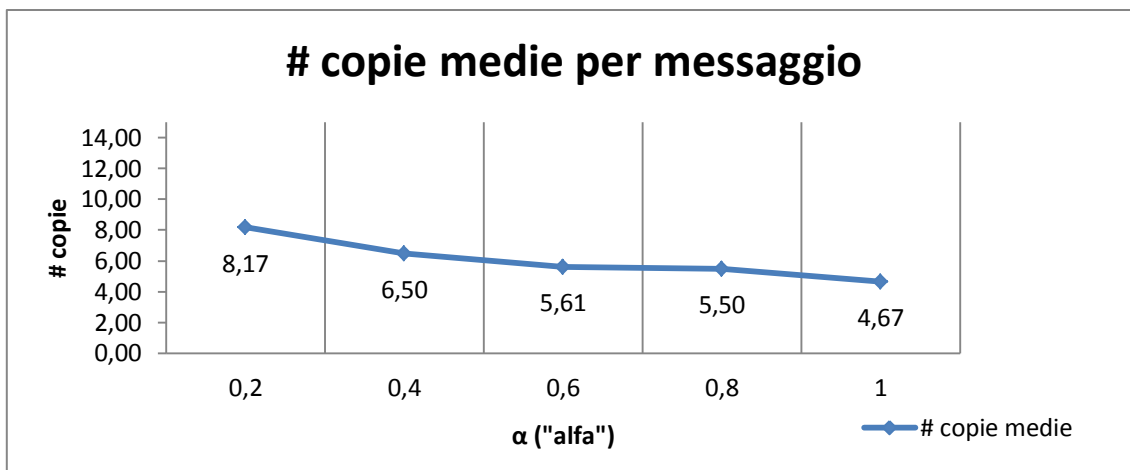
L'overhead di rete (vedi Grafico 4.38) tende ad aumentare al crescere del parametro  $\alpha$  in quanto più messaggi verranno immessi nella rete (30  $\alpha$ ):



**Grafico 4.38:** mostra l'overhead della rete al variare del parametro  $\alpha$  con raggio di copertura fisso a 60 [metri]

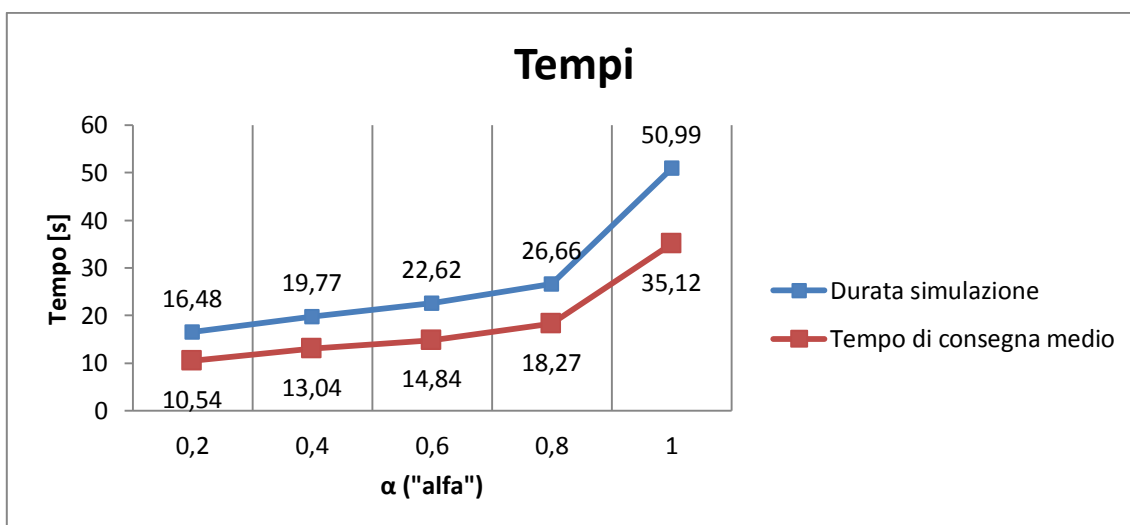
Anche in questo caso il numero di copie medio per messaggio (vedi Grafico 4.39) tenderà a diminuire in quanto il numero di nodi "amici" possibili ( $30 - 30\alpha$ ) per un messaggio diminuirà al crescere di  $\alpha$ :





**Grafico 4.39:** mostra il numero di copie medie per messaggio al variare del parametro  $\alpha$  con raggio di copertura fisso a 60 [metri]

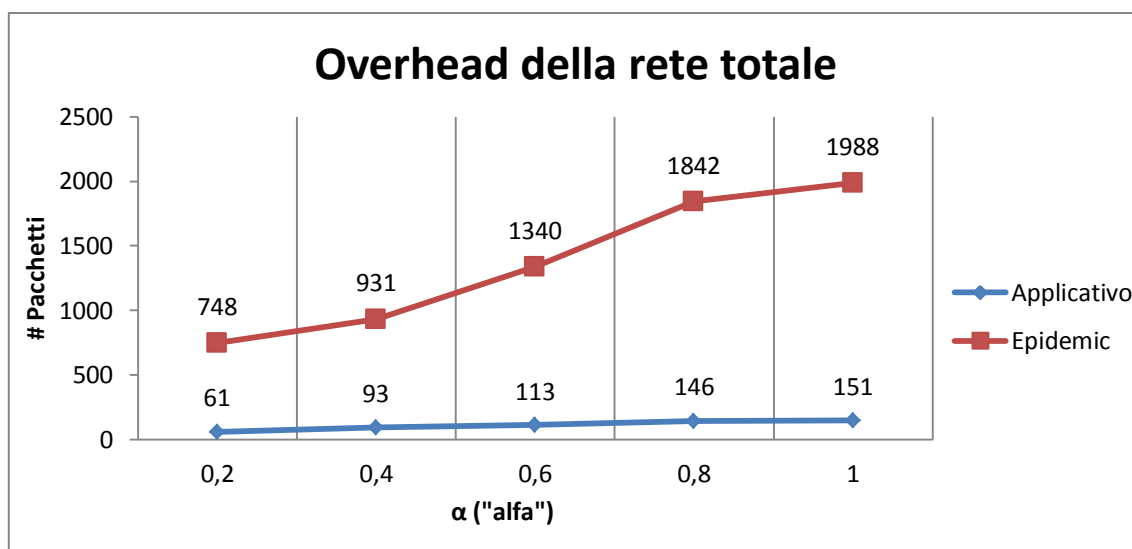
Il tempo di simulazione e il tempo di consegna medio (vedi Grafico 4.40), al crescere del parametro  $\alpha$  aumentano in quanto ci saranno meno portatori che porteranno il messaggio a destinazione causando dunque un allungamento dei tempi medi:



**Grafico 4.40:** mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare del parametro  $\alpha$  con raggio di copertura fisso a 60 [metri]

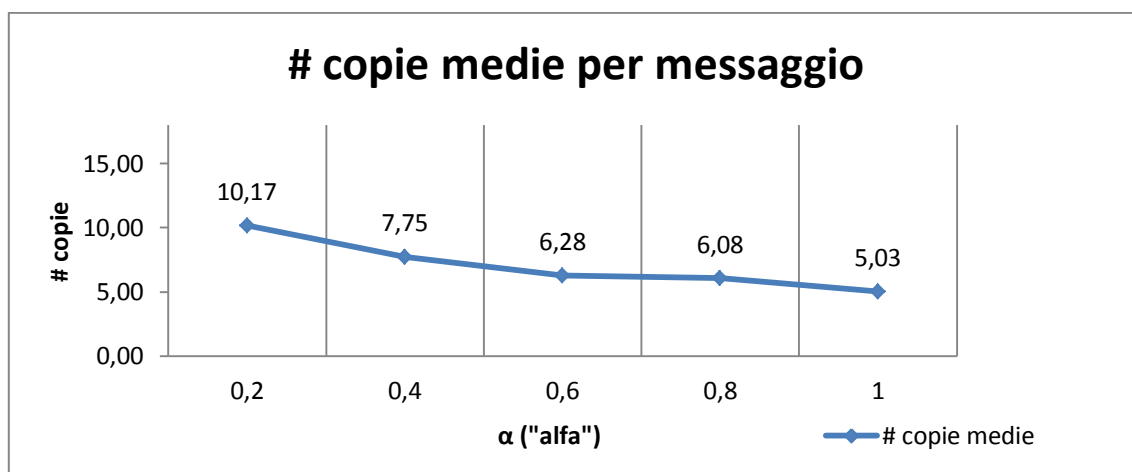
Vediamo ora i risultati per la terza serie di simulazioni:

L'overhead di rete (vedi Grafico 4.41) tende ad aumentare al crescere del parametro  $\alpha$  in quanto più messaggi verranno immessi nella rete ( $30 \alpha$ ):



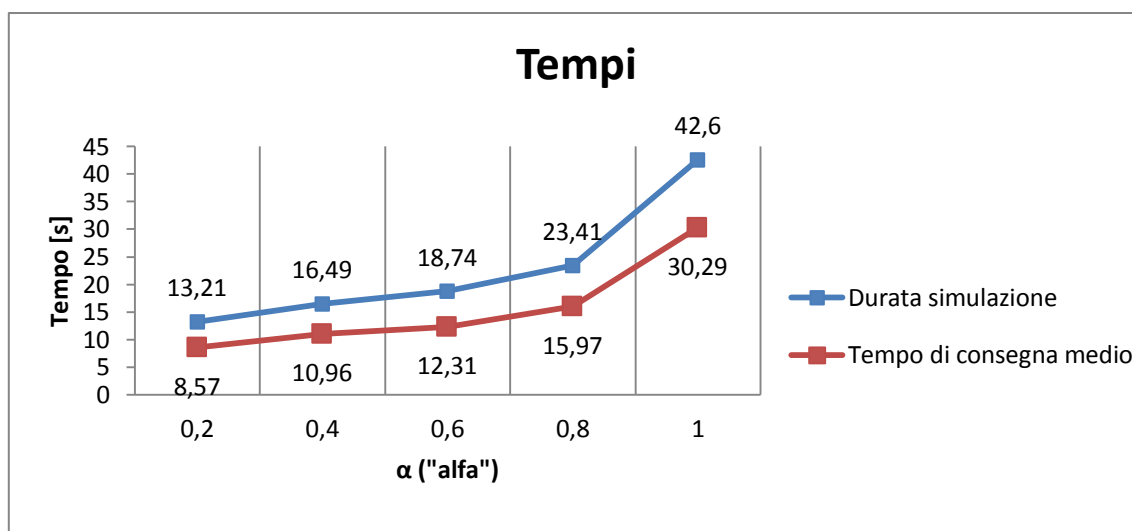
**Grafico 4.41:** mostra l'overhead della rete al variare del parametro  $\alpha$  con raggio di copertura fisso a 100 [metri]

Il numero di copie medio per messaggio (vedi Grafico 4.42) tende a diminuire al crescere di  $\alpha$  in quanto il numero di nodi "amici" diminuisce ( $30 - 30\alpha$ ):



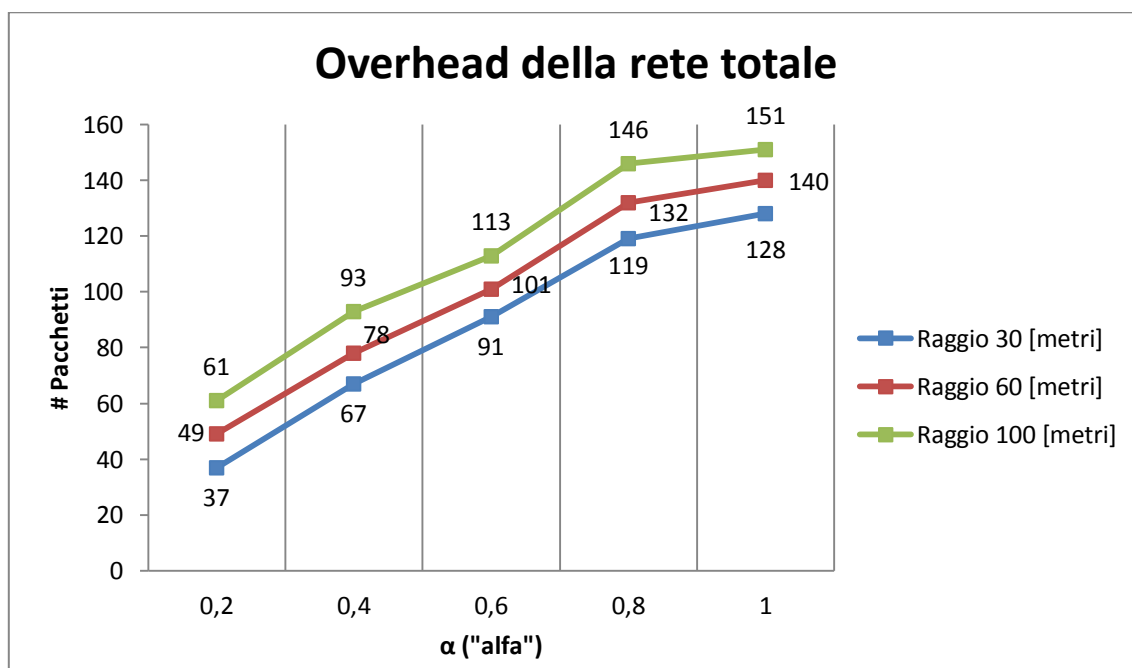
**Grafico 4.42:** mostra il numero di copie medie per messaggio al variare del parametro  $\alpha$  con raggio di copertura fisso a 100 [metri]

Il tempo di simulazione e il tempo di consegna medio (vedi Grafico 4.43), al crescere del parametro  $\alpha$ , aumentano in quanto ci saranno meno nodi "amici" ( $30 - 30\alpha$ ) che porteranno il messaggio a destinazione causando dunque un allungamento dei tempi medi:



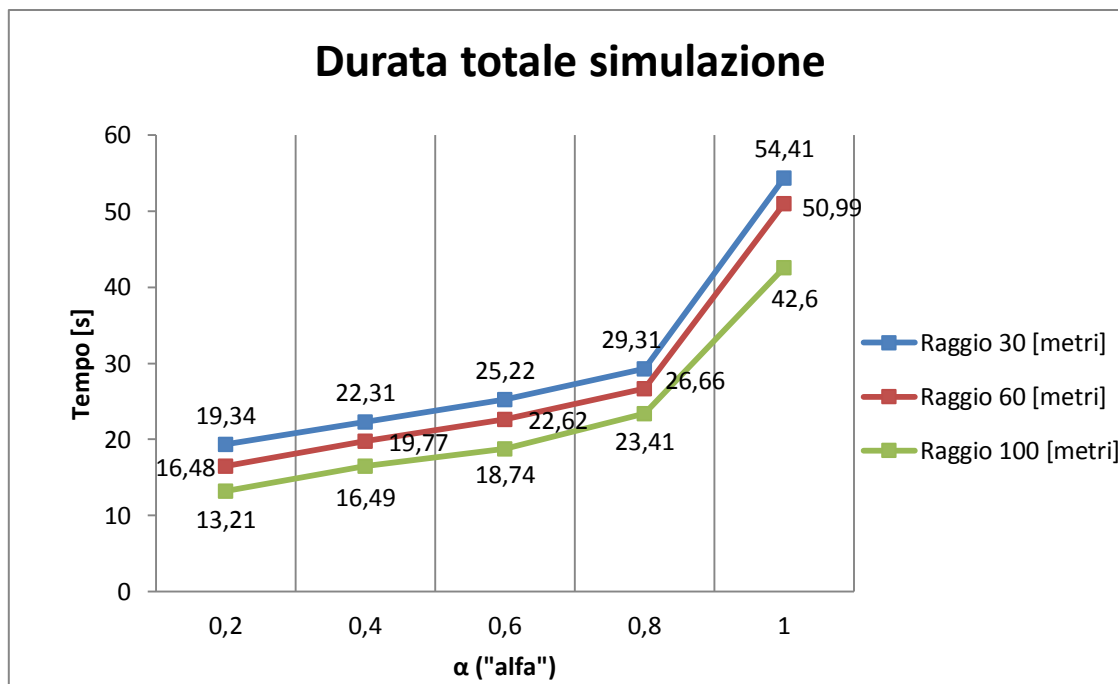
**Grafico 4.43:** mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare del parametro  $\alpha$  con raggio di copertura fisso a 100 [metri]

Riassumendo, come si può notare nel Grafico 4.44, l'overhead di rete è maggiore tanto più il raggio è grande in quanto un maggior numero di nodi viene raggiunto:



**Grafico 4.44:** mostra l'overhead della rete al variare del parametro  $\alpha$  con tutti i raggi di copertura espressi in [metri]

Nel Grafico 4.45 si mostra invece il tempo totale di simulazione e si può notare che aumentando il raggio di copertura dei nodi la simulazione termina prima in quanto i nodi destinatari sono raggiungibili molto più velocemente:



**Grafico 4.45:** mostra la durata totale della simulazione al variare del parametro  $\alpha$  con raggio di copertura fisso a 100 [metri]

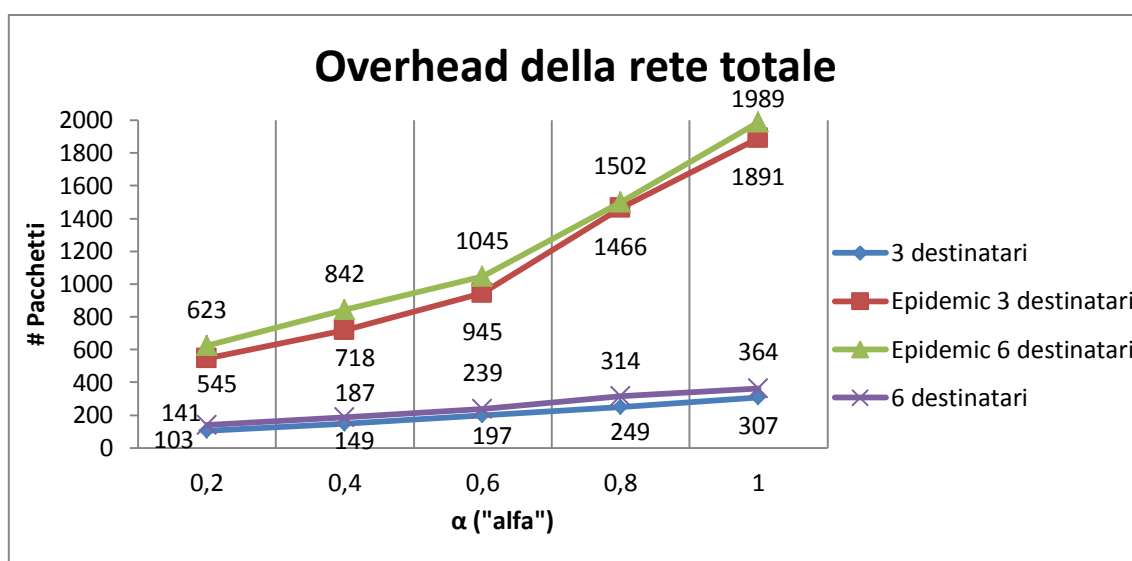
#### 4.4.2 - Protocollo Applicativo Multicast

Per simulare tale protocollo sono stati inseriti nella rete 9 nodi fissi (per l'esattezza 3 nodi fissi a 3 precisi valori sull'asse delle  $y$ ) più altri 30 nodi; di questi 30 nodi, fissato un  $0 < \alpha \leq 1$ , una parte ( $30\alpha$ ) è libera di muoversi nella rete, l'altra ( $30 - 30\alpha$ ) si muove orizzontalmente allo stesso "livello" dei nodi fissi; esattamente un terzo dei nodi che si muovono orizzontalmente attraverserà i tre nodi fissi più in alto nella rete, un altro terzo attraverserà i tre nodi fissi più in basso nella rete, mentre l'ultimo terzo attraverserà i 3 rimanenti nodi fissi di destinazione. Per semplicità tali nodi li chiamerò nodi "amici" in quanto se due nodi sono amici allora tenderanno ad incontrarsi maggiormente rispetto ad altri; infatti i nodi che si muovono orizzontalmente hanno nella loro lista amici gli ID dei nodi fissi che periodicamente attraversano; anche i nodi lineari ( $30\alpha$ ) che si muovono nella rete hanno una loro lista amici dove in modo random sono presenti 3 dei 9 nodi destinatari. Al momento dell'invio del messaggio, ogni nodo libero di muoversi linearmente ( $30\alpha$ ) selezionerà 3 (e poi 6) dei 9 ID dei nodi fissi e procederà a inoltrare il pacchetto verso di loro secondo la funzione obiettivo scritta nel Capitolo 3.

Per valutare le prestazioni di tale protocollo sono state fatte 2 serie di simulazioni:

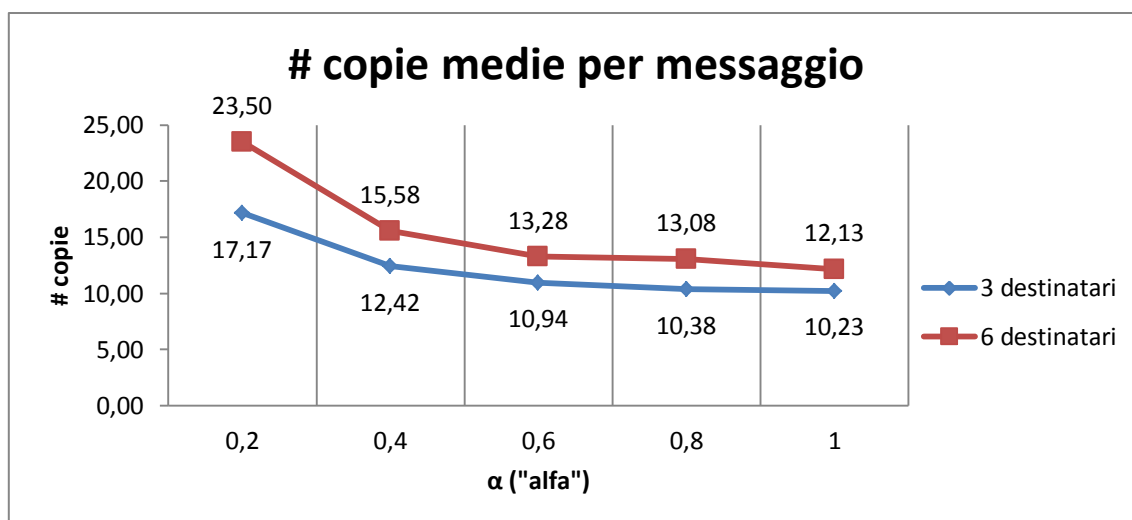
- nella prima serie di simulazioni viene fatto variare il valore di  $\alpha$  (0.2 - 0.4 - 0.6 - 0.8 - 1) tenendo fisso il raggio di copertura a 30 [metri] e selezionando 3 (e poi 6) dei 9 destinatari possibili;
- nella seconda serie di simulazioni viene fatto variare il valore di  $\alpha$  (0.2 - 0.4 - 0.6 - 0.8 - 1) tenendo fisso il raggio di copertura a 60 [metri] e selezionando 3 (e poi 6) dei 9 destinatari possibili;

Vediamo i risultati per la prima serie di simulazioni:



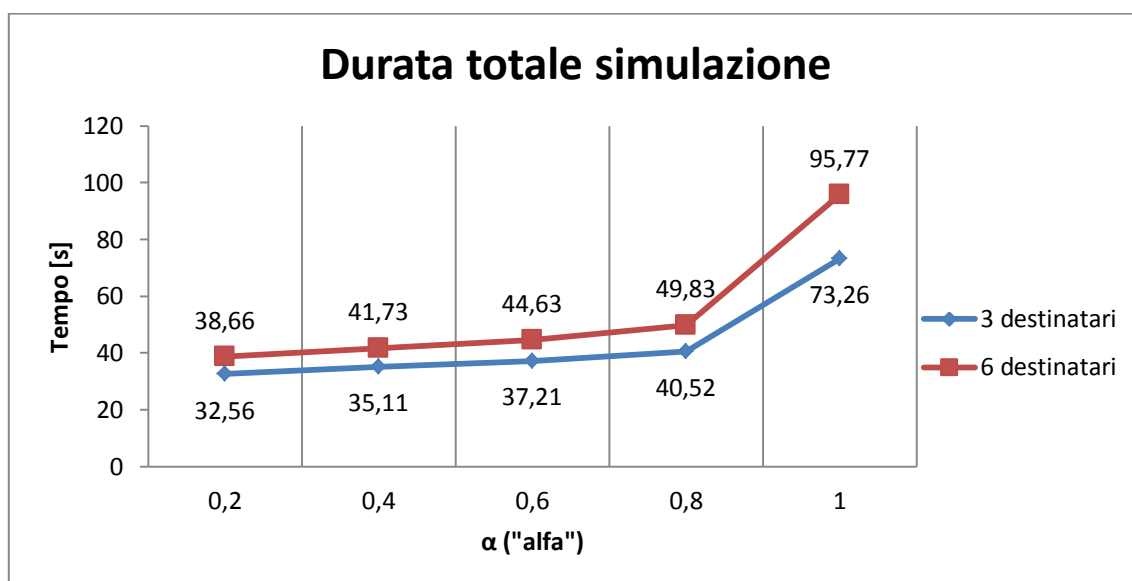
**Grafico 4.46:** mostra l'overhead della rete al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri] sia nel caso di 3 destinatari sia nel caso di 6 destinatari

Come si può notare nel Grafico 4.46, l'overhead di rete aumenta al crescere del parametro  $\alpha$  in quanto i nodi lineari che invieranno messaggi aumenteranno. Si noti anche che l'overhead, nel caso in cui i destinatari dei vari messaggi siano 3, è inferiore al caso in cui i destinatari dei messaggi siano 6; infatti, nel caso a 3 destinatari, i messaggi dovranno raggiungere un numero inferiore di nodi rispetto al caso a 6 destinatari.



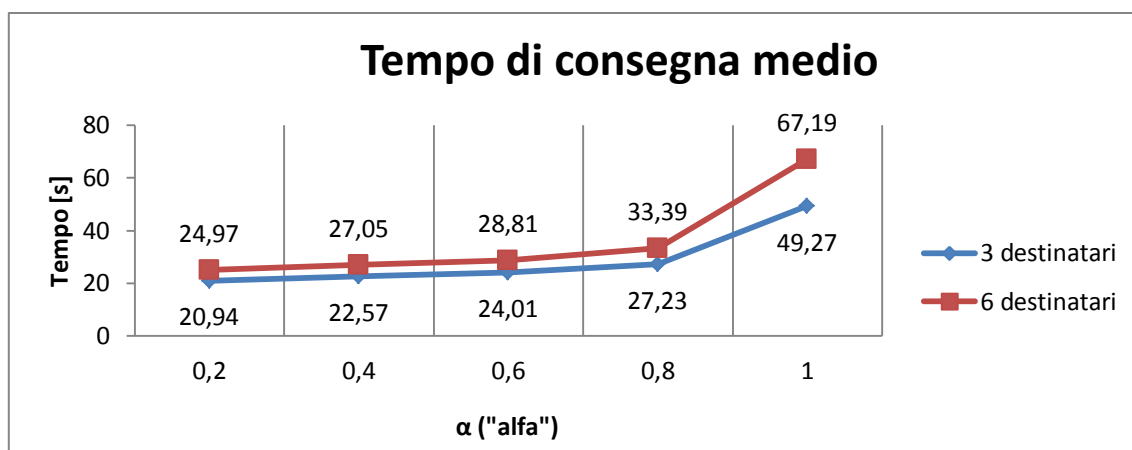
**Grafico 4.47:** mostra il numero di copie medie per messaggio al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri] sia nel caso di 3 destinatari sia nel caso di 6 destinatari

Nel Grafico 4.47 viene mostrato il numero di copie medie al variare del parametro  $\alpha$ ; il numero di copie tende a diminuire al crescere di alfa in quanto i nodi "amici" ( $30 - 30\alpha$ ) diminuiscono e dunque non vengono fatte molte copie dei vari messaggi; nel caso a 6 destinatari ovviamente il numero di copie medie sarà superiore sperchè ogni messaggio dovrà raggiungere più nodi.



**Grafico 4.48:** mostra la durata totale della simulazione al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri] sia nel caso di 3 destinatari sia nel caso di 6 destinatari

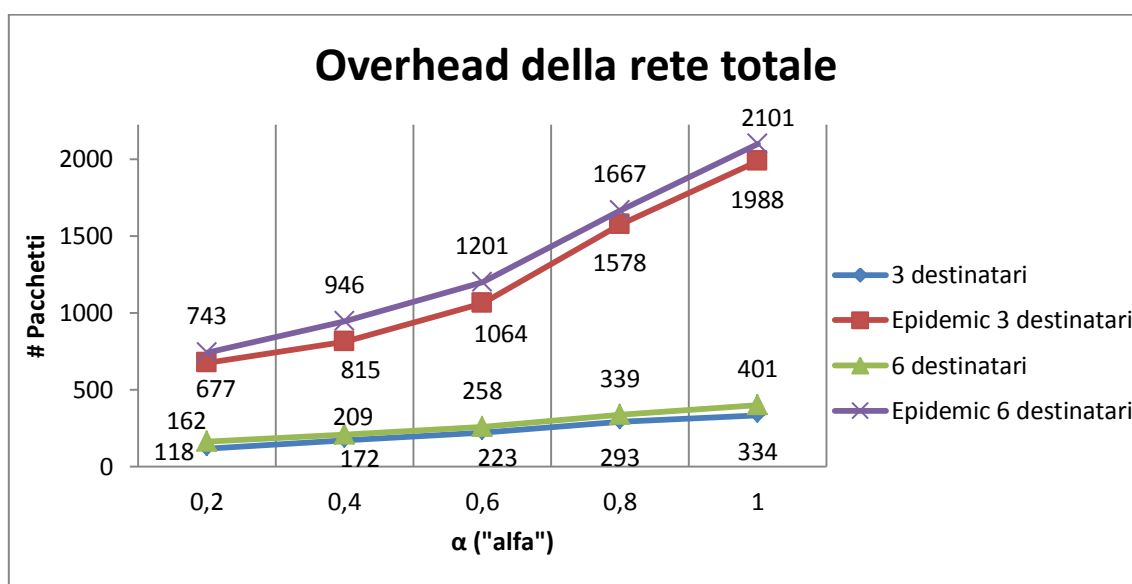
Nel Grafico 4.48 viene mostrata la durata della simulazione sia nel caso a 3 destinatari sia nel caso a 6 destinatari. Come si può osservare, la simulazione termina sempre prima nel caso a 3 destinatari rispetto al caso a 6 in quanto i messaggi devono raggiungere un numero di nodi inferiore; inoltre la durata della simulazione tende ad aumentare in quanto il numero di nodi che genera messaggi ( $30 - 30\alpha$ ) cresce al crescere di alfa.



**Grafico 4.49:** mostra il tempo di consegna medio di ogni messaggio al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri] sia nel caso di 3 destinatari sia nel caso di 6 destinatari

Stesso comportamento lo si può osservare nel Grafico 4.70 che mostra il tempo di consegna medio dei messaggi sia nel caso a 3 destinatari, sia nel caso a 6 destinatari. Il tempo di consegna medio nel caso a 3 destinatari è sempre inferiore rispetto al caso a 6 destinatari in quanto il numero di nodi da raggiungere è inferiore nel primo caso; inoltre i tempi di consegna in entrambi i casi tendono a crescere perchè il numero di nodi che generano messaggi ( $30 \alpha$ ) cresce al crescere di alfa.

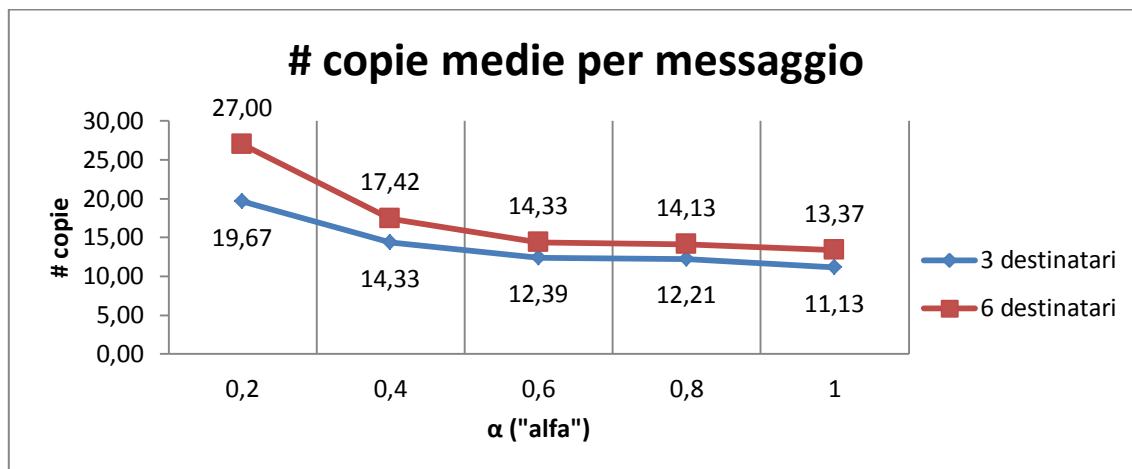
Vediamo ora i risultati della seconda serie di simulazioni:



**Grafico 4.50:** mostra l'overhead della rete al variare del parametro  $\alpha$  con raggio di copertura fisso a 60 [metri] sia nel caso di 3 destinatari sia nel caso di 6 destinatari

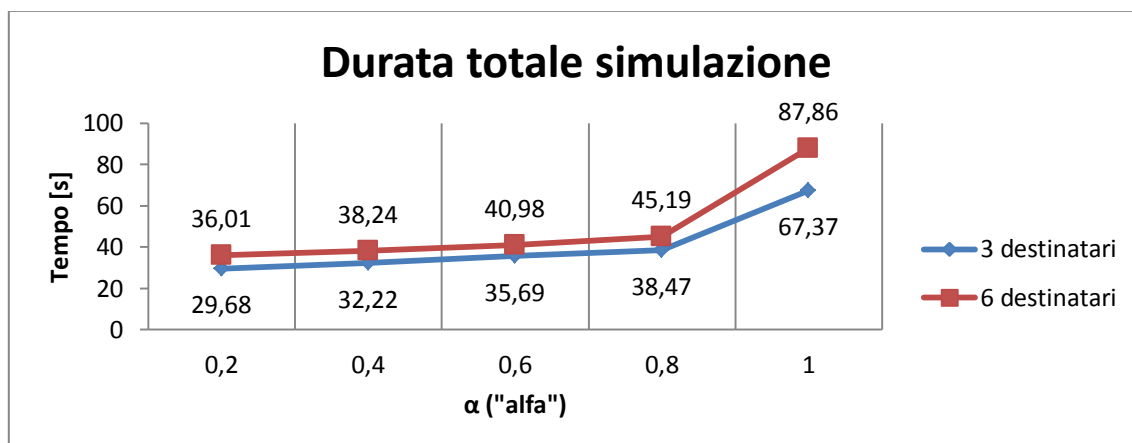
Nel Grafico 4.50 viene mostrato l'overhead di rete sia nel caso a 3 destinatari sia nel caso a 6; nel primo caso, ovviamente, l'overhead sarà inferiore in quanto il numero di nodi destinatari da raggiungere sarà minore rispetto al secondo caso; inoltre, al crescere

di  $\alpha$ , l'overhead aumenterà in quanto saranno presenti più nodi lineari ( $30 - 30\alpha$ ) che genereranno messaggi.



**Grafico 4.51:** mostra il numero di copie medie per messaggio al variare del parametro  $\alpha$  con raggio di copertura fisso a 60 [metri] sia nel caso di 3 destinatari sia nel caso di 6 destinatari

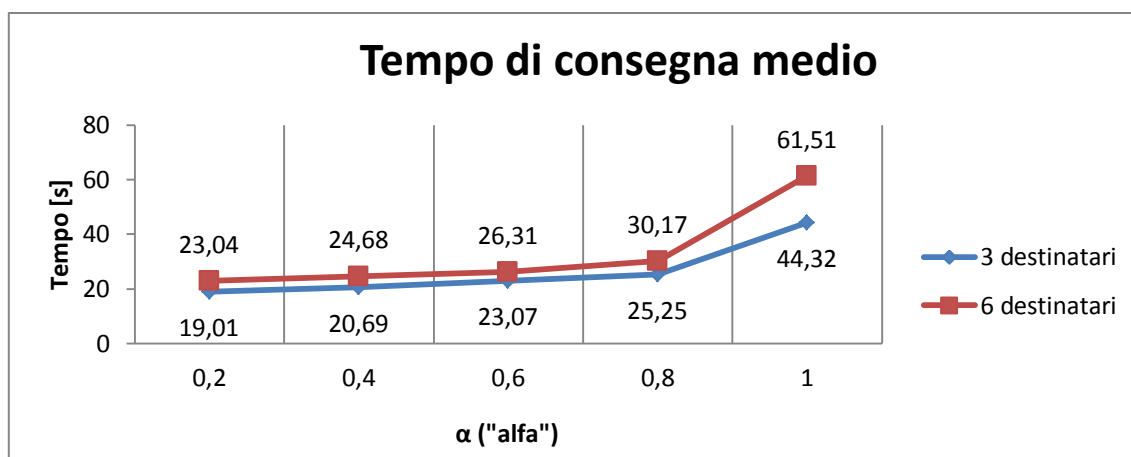
Nel Grafico 4.51 vengono mostrate il numero di copie medie per messaggio al crescere di  $\alpha$ ; tale valore diminuisce in quanto il numero di nodi "amici" ( $30 - 30\alpha$ ) è inferiore al crescere di  $\alpha$ ; il numero di copie medie per messaggio nel caso a 3 destinatari è sempre inferiore rispetto a quello a 6 in quanto i messaggi dovranno raggiungere un numero minore di nodi destinatari.



**Grafico 4.52:** mostra la durata totale della simulazione al variare del parametro  $\alpha$  con raggio di copertura fisso a 60 [metri] sia nel caso di 3 destinatari sia nel caso di 6 destinatari

Anche la durata di simulazione (vedi Grafico 4.52) aumenta in quanto il numero di messaggi distribuito nella rete cresce al crescere di  $\alpha$ ; inoltre la simulazione termina prima nel caso a 3 destinatari rispetto quello a 6 in quanto il numero di messaggi da far arrivare a destinazione è inferiore nel primo caso rispetto al secondo.

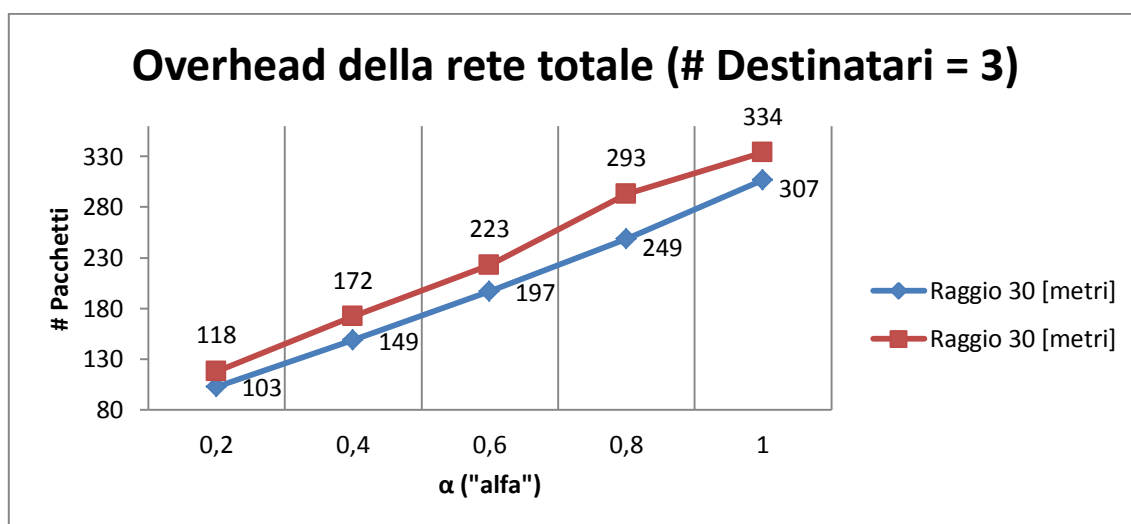




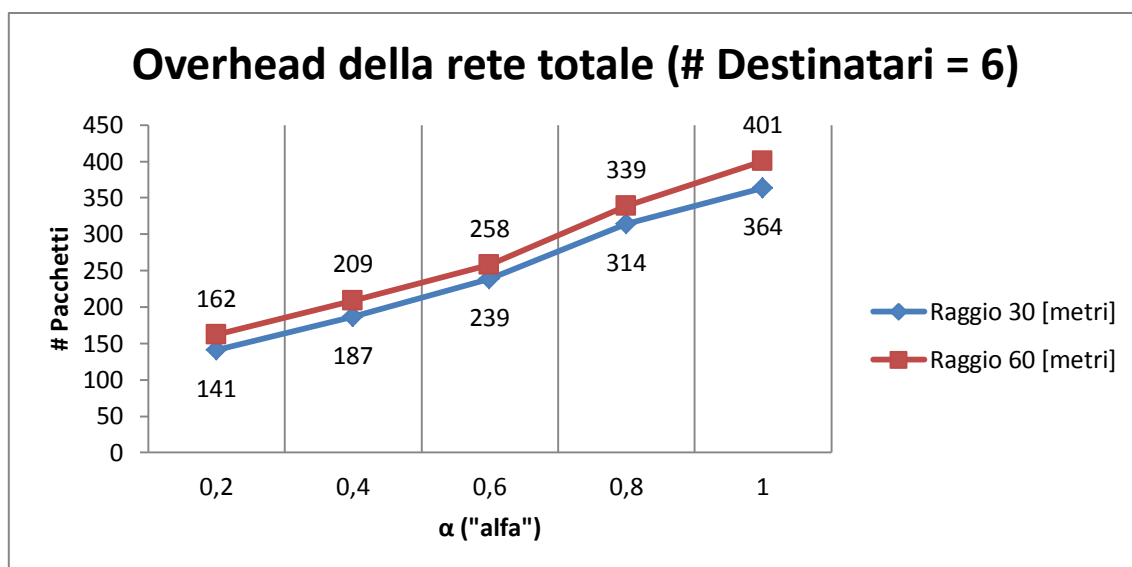
**Grafico 4.53:** mostra il tempo di consegna medio di ogni messaggio al variare del parametro  $\alpha$  con raggio di copertura fisso a 60 [metri] sia nel caso di 3 destinatari sia nel caso di 6 destinatari

Anche il tempo di consegna medio (vedi Grafico 4.53) tende ad aumentare in quanto il numero di nodi "amici" ( $30 - 30\alpha$ ) diminuisce al crescere di  $\alpha$  e quindi vengono fatte meno copie dei messaggi immessi nella rete. Per il caso a 3 destinatari si ha un tempo di consegna medio inferiore rispetto al caso a 6 in quanto il numero di messaggi da far arrivare a destinazione è inferiore nel primo caso rispetto al secondo.

Riassumendo, osservando il Grafico 4.54a (3 destinatari) e 4.54b (6 destinatari), l'overhead di rete è maggiore tanto più il raggio è grande in quanto un maggior numero di nodi viene raggiunto:

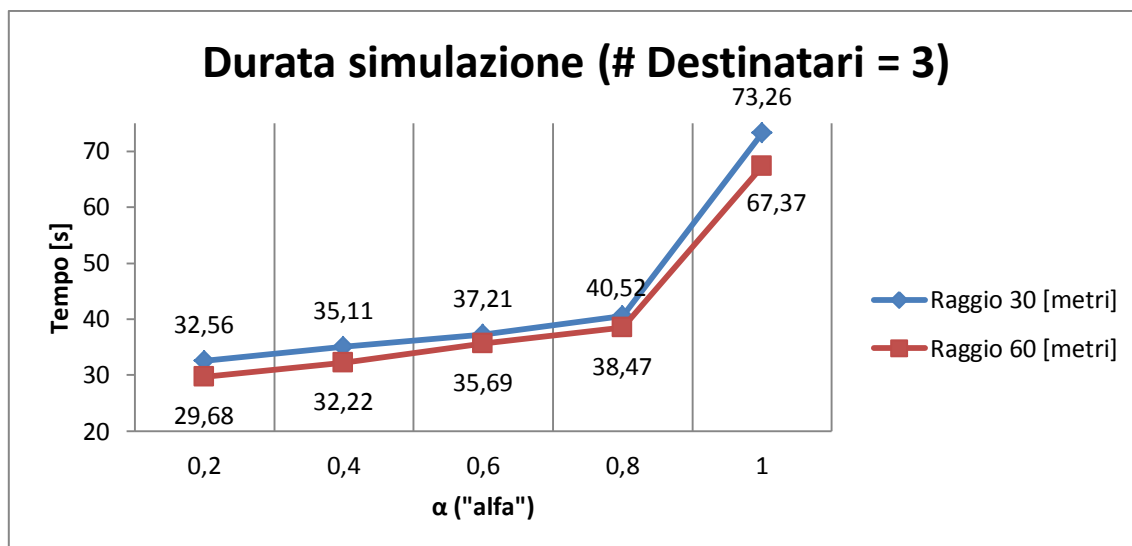


**Grafico 4.54a:** mostra l'overhead della rete al variare del parametro  $\alpha$  con tutti i raggi di copertura espressi in [metri] nel caso a 3 destinatari

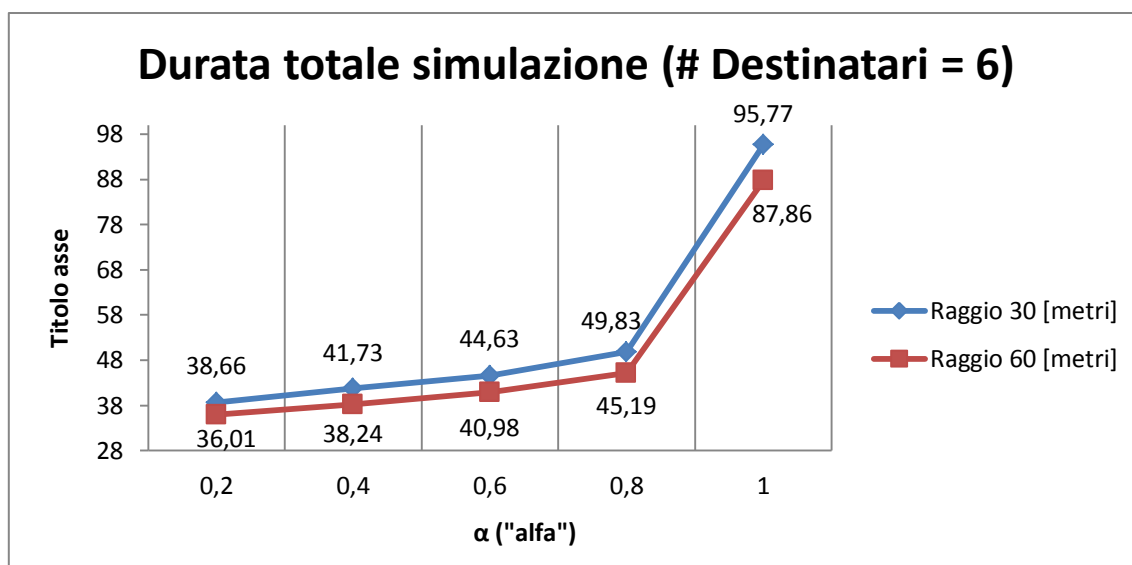


**Grafico 4.54b:** mostra l'overhead della rete al variare del parametro  $\alpha$  con tutti i raggi di copertura espressi in [metri] nel caso a 6 destinatari

D'altra parte, maggiore è il raggio di copertura, minore è il tempo di simulazione in quanto i nodi destinatari vengono raggiunti prima; a dimostrazione di ciò si osservino i Grafici 4.55a (3 destinatari) e 4.55b (6 destinatari):



**Grafico 4.55a:** mostra la durata totale della simulazione al variare del parametro  $\alpha$  con tutti i raggi di copertura espressi in [metri] nel caso a 3 destinatari



**Grafico 4.55b:** mostra la durata totale della simulazione al variare del parametro  $\alpha$  con tutti i raggi di copertura espressi in [metri] nel caso a 6 destinatari

#### 4.4.3 - Protocollo Applicativo Geocast

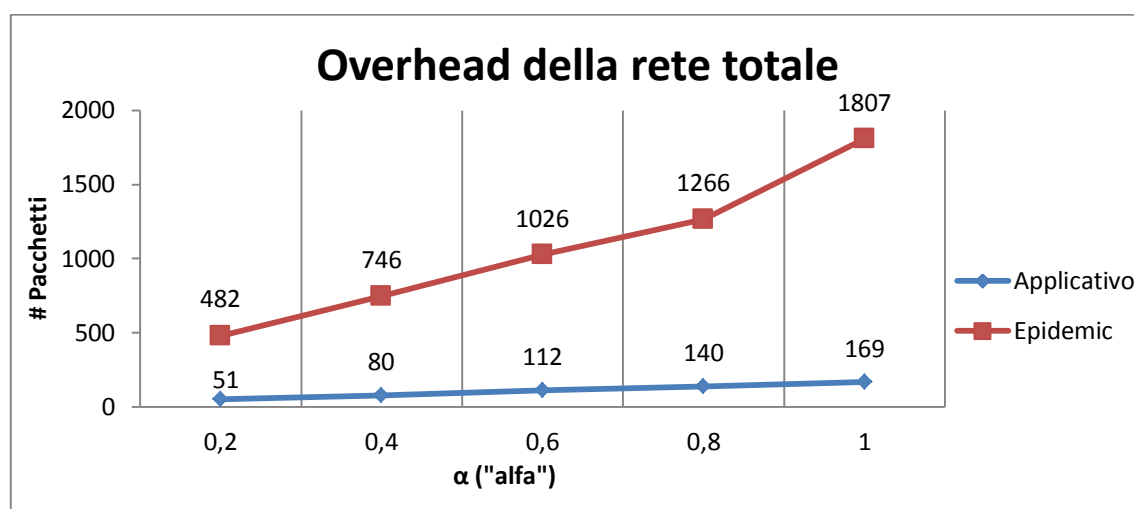
Per simulare tale protocollo sono stati inseriti nella rete 6 nodi fissi (un nodo per ognuna delle 6 aree in cui è divisa la rete) più altri 30 nodi; il nodo fisso per ogni area è stato messo al fine di avere una delivery del 100% in modo tale che ogni volta un messaggio arrivi nell'area di destinazione abbia sempre un destinatario presente; degli altri 30 nodi, fissato un  $0 < \alpha \leq 1$ , una parte ( $30 \alpha$ ) è libera di muoversi nella rete, l'altra ( $30 - 30\alpha$ ) si muove orizzontalmente allo stesso "livello" dei nodi fissi; inoltre i nodi che si spostano orizzontalmente sono stati divisi in due: una metà attraversa le aree 1 - 2 - 3, l'altra metà attraversa le aree 4 - 5 - 6, in modo da poter coprire tutte le aree della rete; per semplicità tali nodi verranno chiamati "major" in quanto possiedono un numero di Check-In maggiore in quelle aree rispetto a qualsiasi altro nodo. Prima di inviare i messaggi, si è atteso un tempo pari a 12 minuti in modo che tutti i nodi incrementassero i propri Check-In per le aree attraversate. Al momento dell'invio del messaggio, ogni nodo libero di muoversi linearmente ( $30 \alpha$ ) selezionerà una delle 6 aree della rete e procederà a inoltrare il pacchetto verso tale area secondo la funzione obiettivo scritta nel Capitolo 3. Inoltre tutti e 30 i nodi effettuano il Check-In nelle aree ad una frequenza fissa di  $1 \left[ \frac{Check-In}{5 sec} \right]$  e, dunque, in tale struttura, tutti i nodi lineari ( $30 \alpha$ ) possono essere buoni portatori in quanto ogni nodo può avere un numero di Check-In maggiore rispetto ad un altro per una rispettiva area. Da notare però che i nodi "major" (che si muovono con angolo pari a  $0^\circ$ ) avranno sempre un numero di Check-In (nelle tre aree attraversate) maggiore rispetto agli altri nodi lineari in quanto continueranno sempre e

solo ad attraversare 3 aree mentre quelli lineari saranno liberi di attraversare tutta la rete.

Per valutare le prestazioni di tale protocollo sono state fatte 2 serie di simulazioni:

- nella prima serie di simulazioni viene fatto variare il valore di  $\alpha$  (0.2 - 0.4 - 0.6 - 0.8 - 1) tenendo fisso il raggio di copertura a 30 [metri];
- nella seconda serie di simulazioni viene fatto variare il valore di  $\alpha$  (0.2 - 0.4 - 0.6 - 0.8 - 1) tenendo fisso il raggio di copertura a 60 [metri];

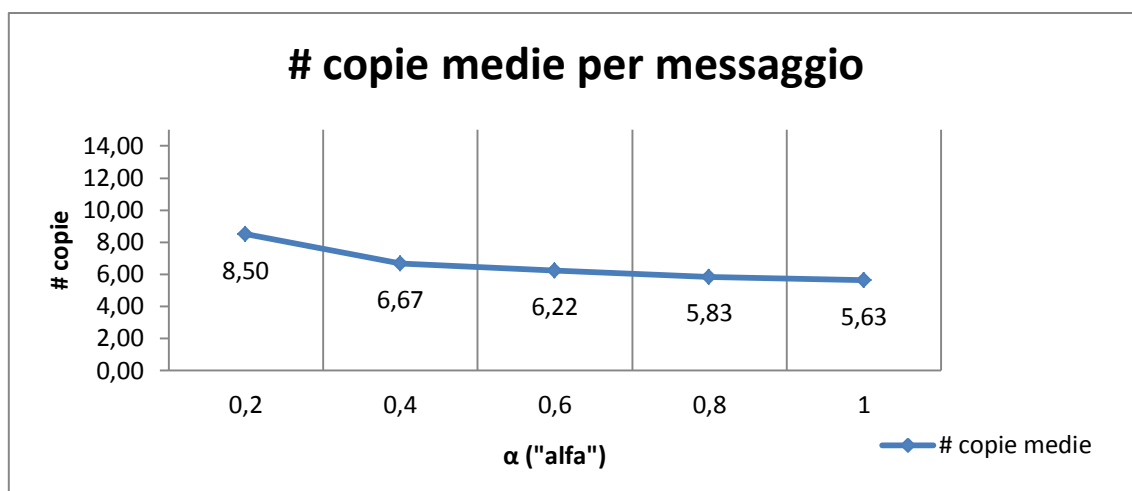
Vediamo i risultati per la prima serie di simulazioni:



**Grafico 4.56:** mostra l'overhead della rete al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri]

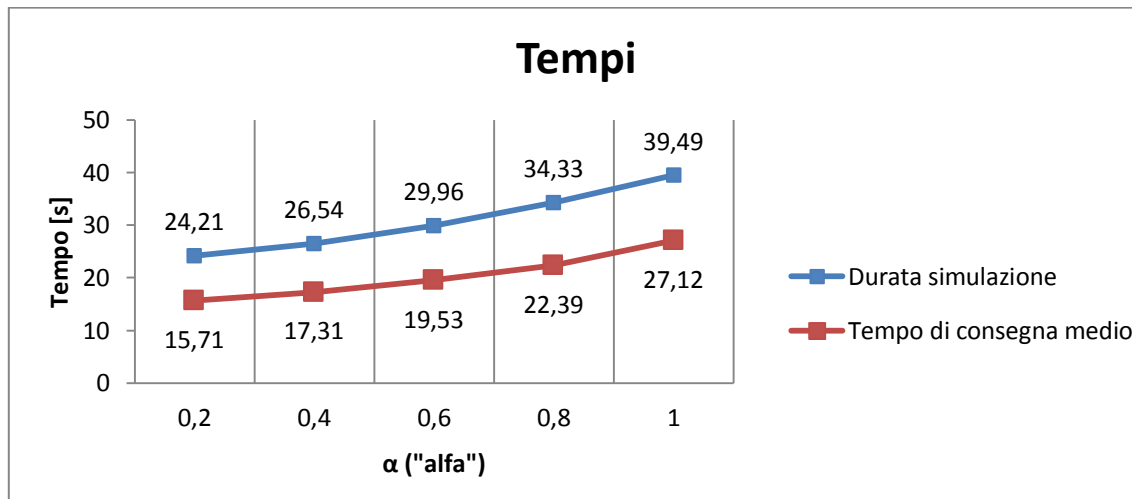
Come si può osservare nel Grafico 4.56, al crescere del valore di  $\alpha$  anche l'overhead di rete aumenta in quanto il numero di nodi che creano messaggi ( $30 \alpha$ ) aumenta.

D'altra parte, il numero di copie (vedi Grafico 4.57) diminuisce in quanto al crescere di  $\alpha$  il numero dei nodi "major" ( $30 - 30\alpha$ ) decresce.



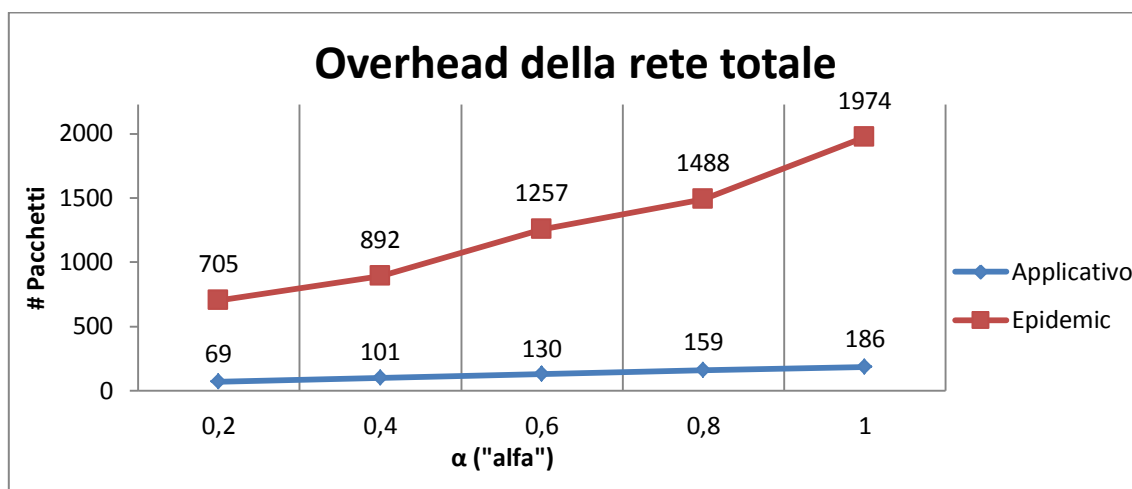
**Grafico 4.57:** mostra il numero di copie medie per messaggio al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri]

Nel Grafico 4.58 viene mostrata la durata totale della simulazione e il tempo di consegna medio; entrambi aumentano al crescere di alfa in quanto il numero dei nodi "major" ( $30 - 30\alpha$ ) decresce e perciò i messaggi impiegano più tempo ad arrivare a destinazione:



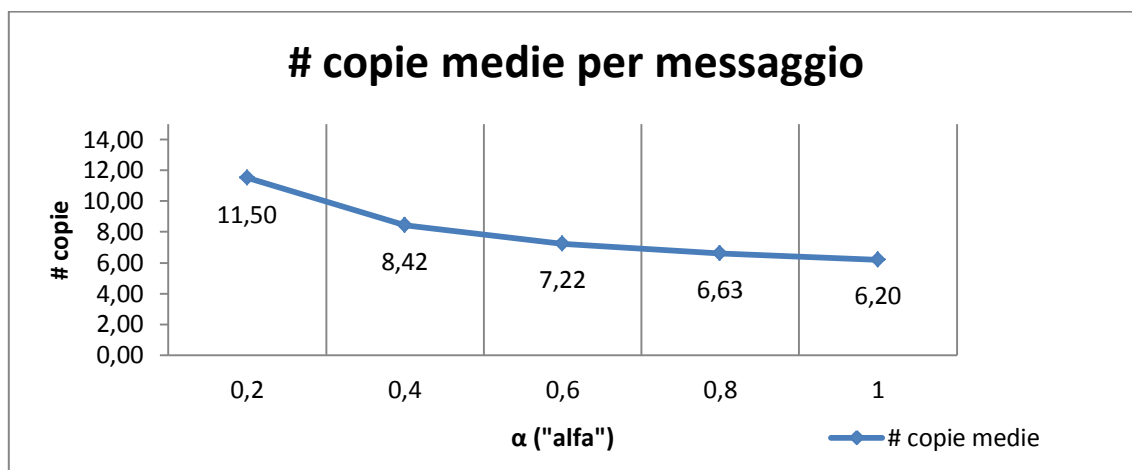
**Grafico 4.58:** mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri]

Vediamo ora i risultati relativi alla seconda serie di simulazioni:



**Grafico 4.59:** mostra l'overhead della rete al variare del parametro  $\alpha$  con raggio di copertura fisso a 60 [metri]

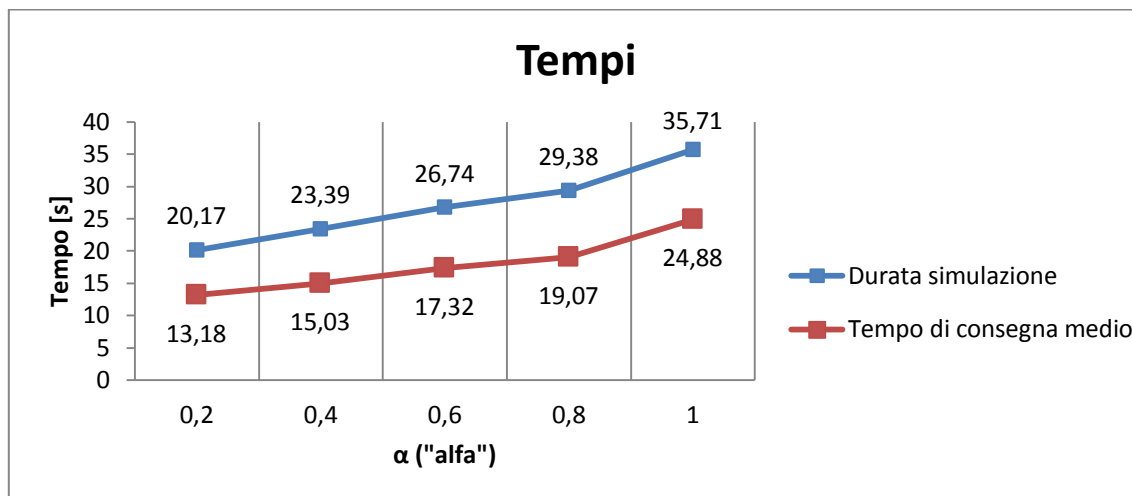
Nel Grafico 4.59 viene mostrato l'overhead della rete; al crescere di  $\alpha$  anche il numero di messaggi complessivi nella rete aumenta in quanto il numero di nodi lineari ( $30\alpha$ ) cresce.



**Grafico 4.60:** mostra il numero di copie medie per messaggio al variare del parametro  $\alpha$  con raggio di copertura fisso a 60 [metri]

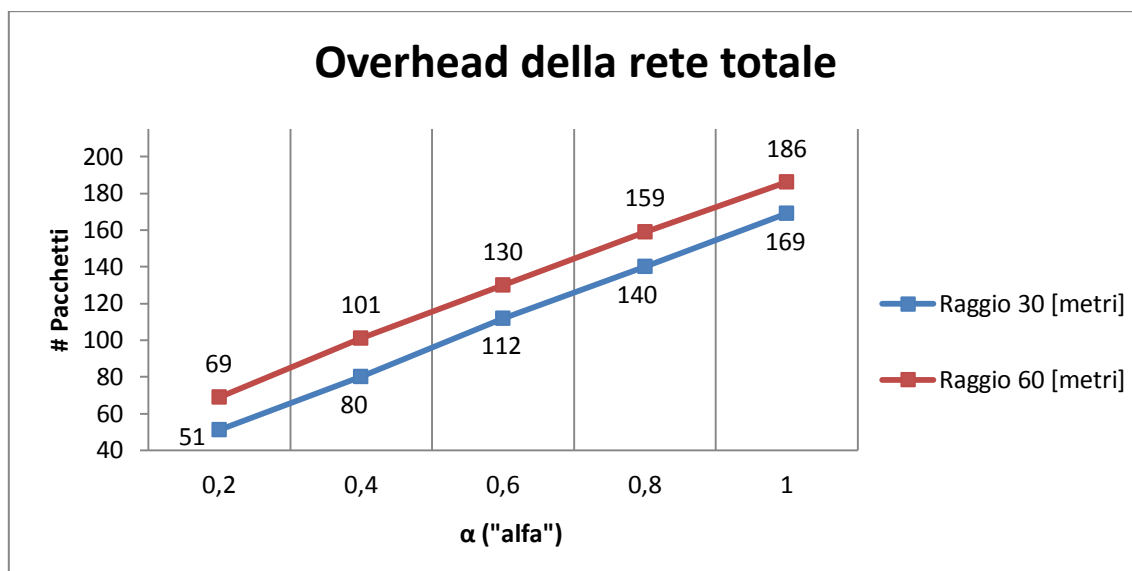
Nel Grafico 4.60 viene mostrato il numero di copie medie per messaggio; anche qui il numero di copie diminuisce in quanto al crescere di  $\alpha$  il numero dei nodi "major" ( $30 - 30\alpha$ ) decresce e dunque verranno fatte meno copie dei vari messaggi; da questo deriva dunque un tempo di simulazione e un tempo di consegna medio (vedi Grafico 4.61) maggiori al crescere di  $\alpha$  in quanto il numero di messaggi generati nella rete sarà

maggiore ( $30\alpha$ ) e il numero dei nodi "major" ( $30 - 30\alpha$ ) tenderà a diminuire allungando i tempi medi.



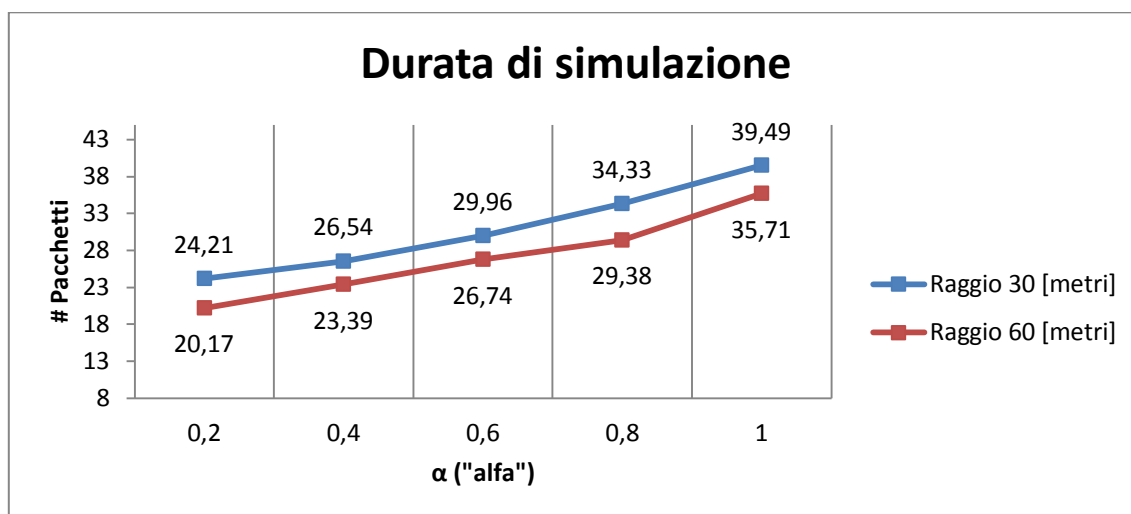
**Grafico 4.61:** mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare del parametro  $\alpha$  con raggio di copertura fisso a 60 [metri]

Riassumendo, nel Grafico 4.62 viene mostrato l'overhead di rete in funzione del raggio al variare di  $\alpha$ ; l'overhead è maggiore tanto più il raggio è grande in quanto un maggior numero di nodi viene raggiunto:



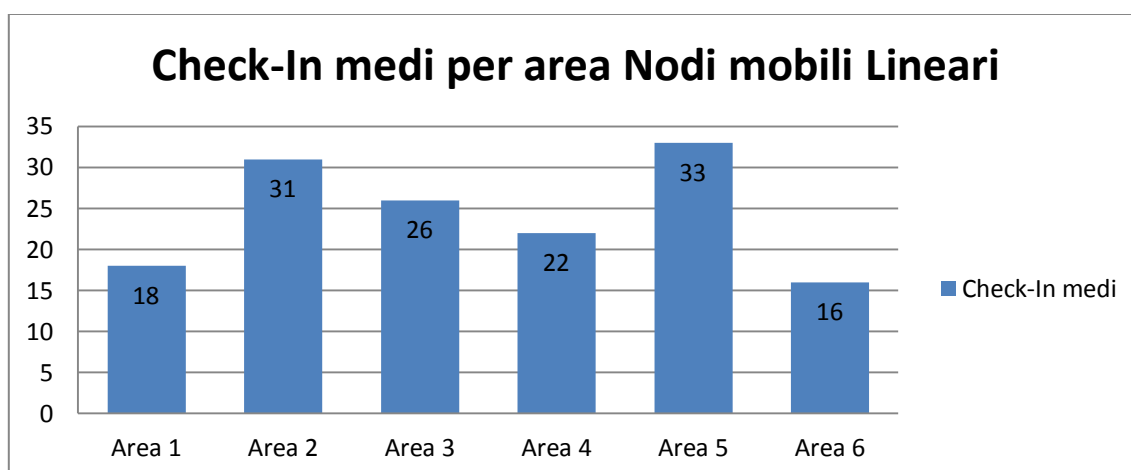
**Grafico 4.62:** mostra l'overhead della rete al variare del parametro  $\alpha$  con tutti i raggi di copertura espressi in [metri]

Nel Grafico 4.63 viene invece mostrata la durata totale di simulazione in funzione del raggio al variare di  $\alpha$ ; tale durata è minore tanto più il raggio è grande in quanto un numero più grande di nodi viene raggiunto prima e la simulazione termina prima.



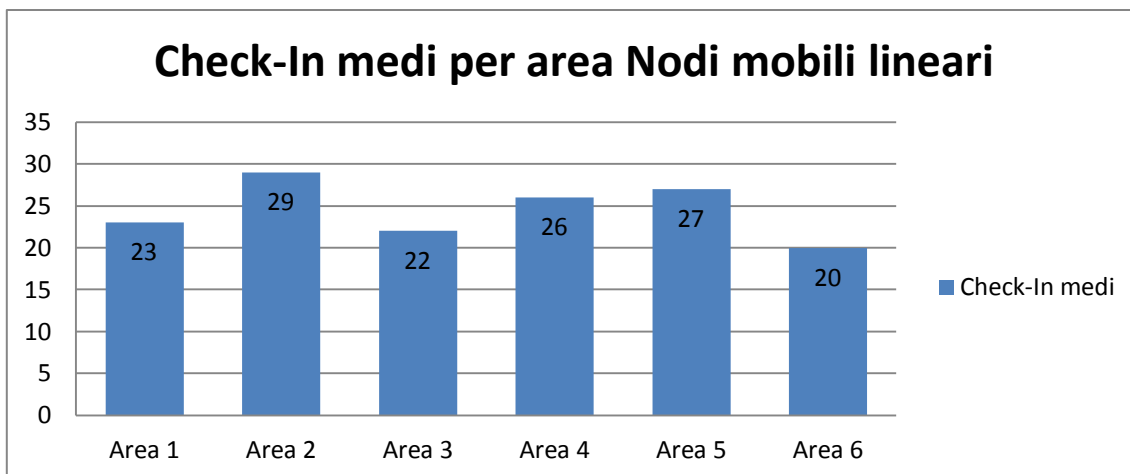
**Grafico 4.63:** mostra la durata totale della simulazione al variare del parametro  $\alpha$  con tutti i raggi di copertura espressi in [metri]

Inoltre sono stati contati i Check-In medi effettuati dai vari nodi mobili lineari (grafici 4.64a - 4.64b - 4.64c - 4.64d - 4.64e) per area al variare di  $\alpha$ :

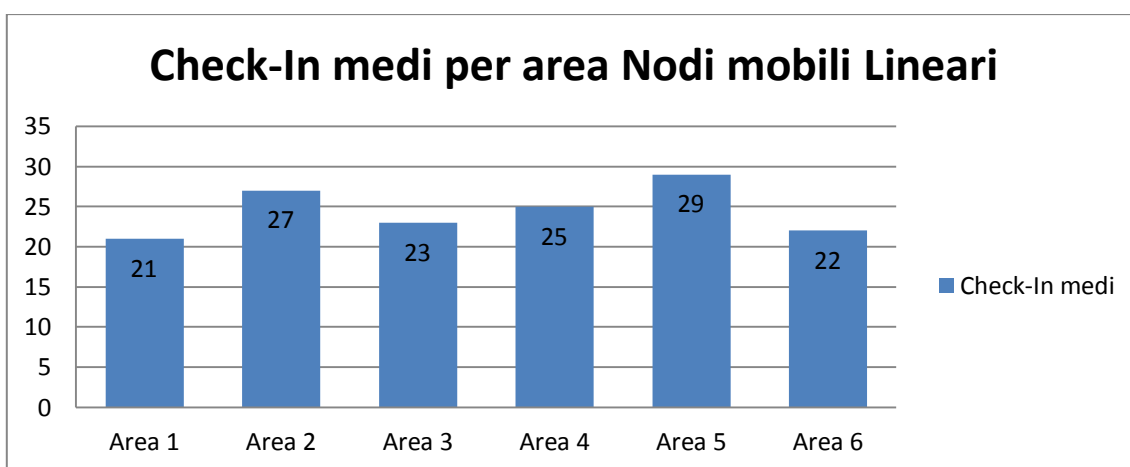


**Grafico 4.64a:** mostra il numero di Check-In medi di un nodo lineare per area per  $\alpha=0.2$

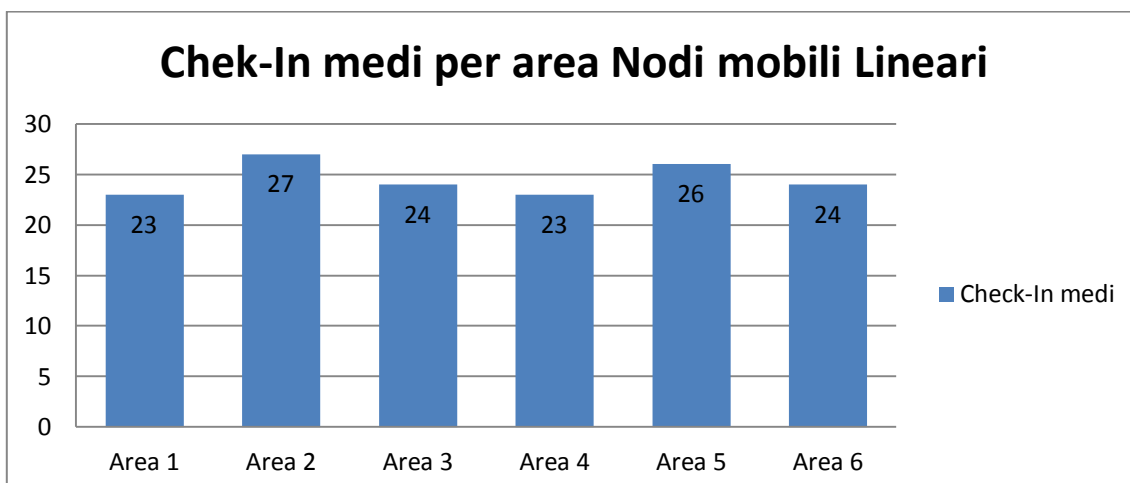




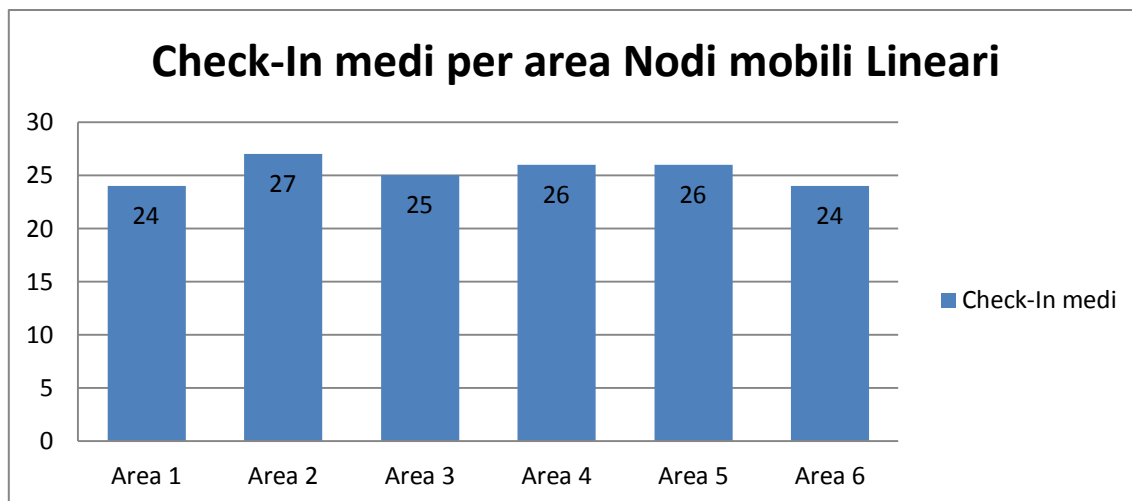
**Grafico 4.64b:** mostra il numero di Check-In medi di un nodo lineare per area per  $\alpha=0.4$



**Grafico 4.64c:** mostra il numero di Check-In medi di un nodo lineare per area per  $\alpha=0.6$



**Grafico 4.64d:** mostra il numero di Check-In medi di un nodo lineare per area per  $\alpha=0.8$



**Grafico 4.64e:** mostra il numero di Check-In medi di un nodo lineare per area per  $\alpha=1$

Come si può notare in questi grafici, all'aumentare di  $\alpha$  (e dunque del numero di nodi lineari nella rete) il numero di Check-In medio per area tende a uniformarsi allo stesso livello in quanto i nodi sono sparsi più uniformemente nella rete; ad esempio con  $\alpha = 0,2$  i nodi lineari nella rete risultano essere  $30 \cdot 0,2 = 6$ : questi 6 nodi sono inseriti nella rete in modo random, non uno per area e così risulta che alcune aree abbiano un numero di Check-In medio abbastanza diverso da altre; viceversa, con  $\alpha = 1$ , tutti i nodi si muovono linearmente e tutte le 6 aree avranno un numero di Check-In medio quasi uguale.

## 4.5 - Analisi prestazioni per Protocollo Ibrido

### 4.5.1 - Protocollo Ibrido Unicast

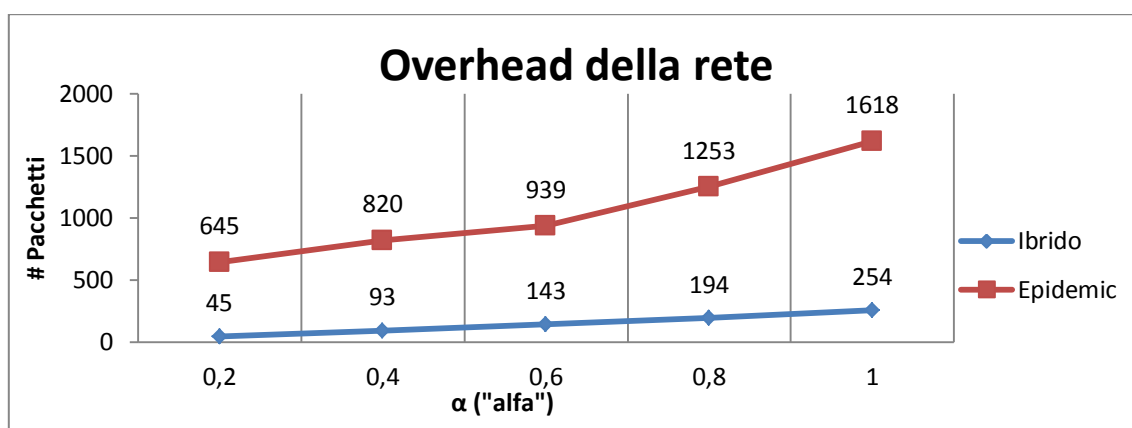
Per simulare tale protocollo è stata utilizzata la stessa struttura di rete spiegata nel Capitolo 4.4.1 con la sola differenza che si è atteso un tempo di 8 [s] prima di inviare i messaggi per fare in modo che i nodi si scambiassero i ping; il messaggio per arrivare a destinazione potrà essere passato ad altri nodi secondo la funzione obiettivo scritta nel Capitolo 3; le simulazioni fatte prendono fissi alcuni parametri come:

- Raggio di copertura: 30 [mt];
- Frequenza ping:  $1 \left[ \frac{\text{ping}}{\text{sec}} \right]$ .

Al fine di valutare le prestazioni di tale protocollo, sono state fatte 3 serie di simulazioni:

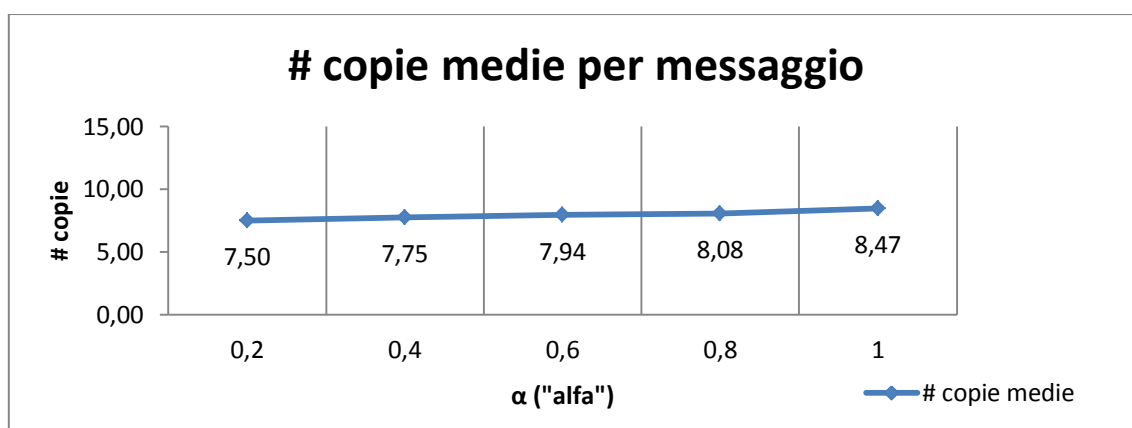
- nella prima serie viene fatto variare il valore di  $\alpha$  (0.2 - 0.4 - 0.6 - 0.8 - 1) con  $P = 0.75$  e  $Q = 0.25$ ;
- nella seconda serie viene fatto variare il valore di  $\alpha$  (0.2 - 0.4 - 0.6 - 0.8 - 1) con  $P = 0.25$  e  $Q = 0.75$ ;
- nella terza serie viene fatto variare il valore di  $\alpha$  (0.2 - 0.4 - 0.6 - 0.8 - 1) con  $P = 0.50$  e  $Q = 0.50$ ;

Vediamo ora i risultati della prima serie di simulazioni:



**Grafico 4.65:** mostra l'overhead della rete al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.75$ ,  $Q = 0.25$

Il Grafico 4.65 mostra l'overhead di rete; come si può notare, al crescere di  $\alpha$  anche l'overhead di rete sale in quanto più messaggi vengono generati dai nodi ( $30 \alpha$ ).

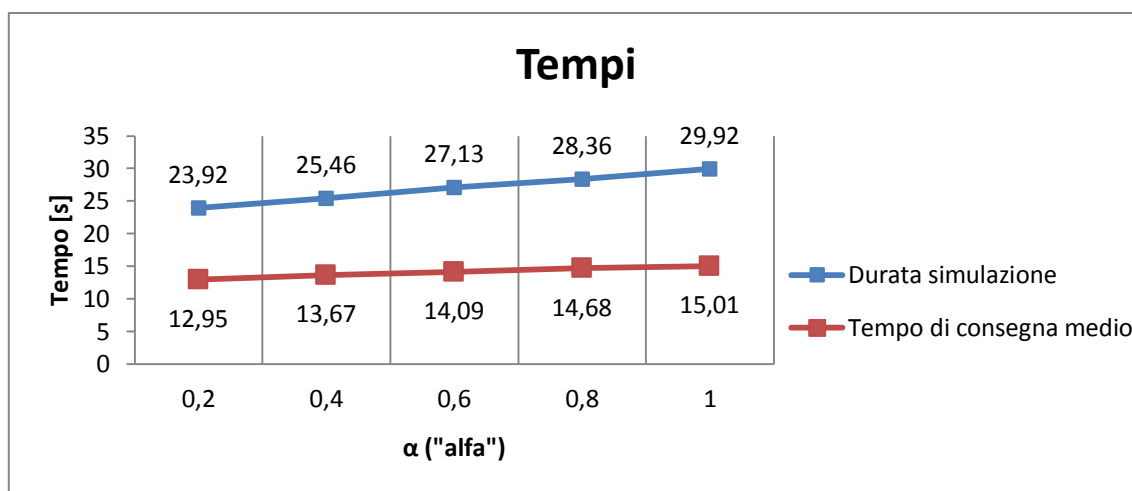


**Grafico 4.66:** mostra il numero di copie medie per messaggio al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.75$ ,  $Q = 0.25$

Nel Grafico 4.66 invece viene mostrato il numero di copie medio della rete; tale valore tende a crescere in quanto anche all'aumentare di  $\alpha$  il numero di portatori rimane il medesimo perchè i messaggi vengono passati ad altri nodi con probabilità  $P = 0.75$  se

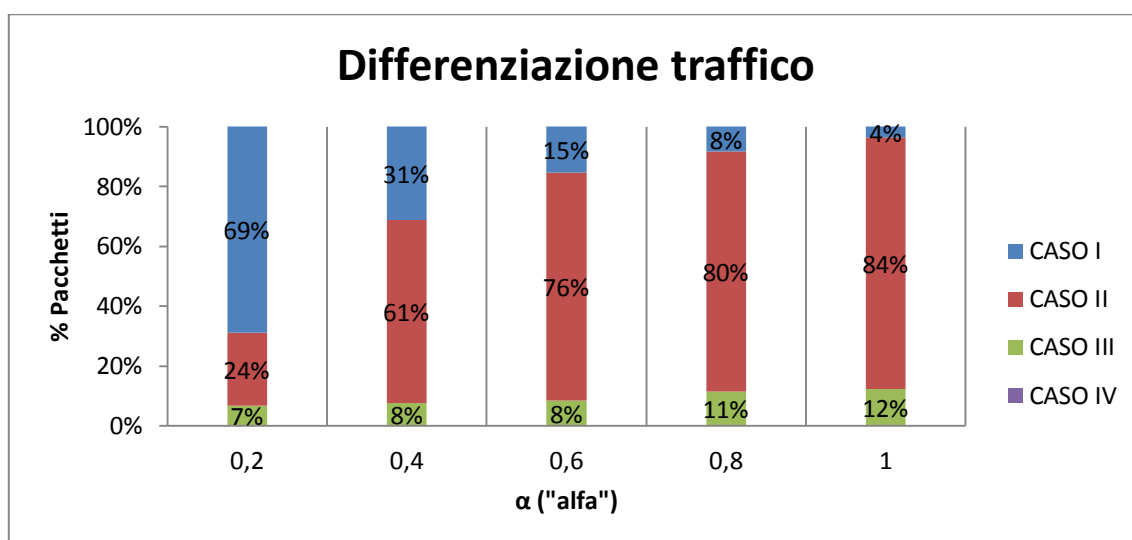
hanno un numero di ping superiore; infatti il numero di ping di un nodo è indipendente dal valore di  $\alpha$ .

Anche la durata di simulazione e il tempo di consegna medio (vedi Grafico 4.67) aumentano in quanto al crescere di  $\alpha$ , il numero di messaggi da consegnare è maggiore:



**Grafico 4.67:** mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.75$ ,  $Q = 0.25$

Infine, nel Grafico 4.68 si può notare la suddivisione dei pacchetti inoltrati nella rete:

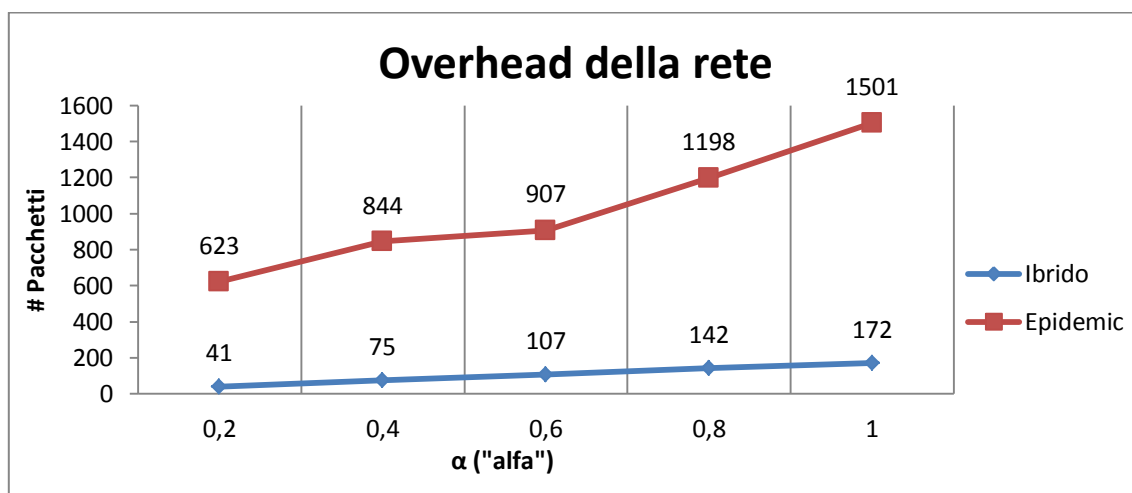


**Grafico 4.68:** mostra la divisione del traffico di tutti i messaggi della rete al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.75$ ,  $Q = 0.25$

Al crescere di  $\alpha$ , si verificano le suddette cose:

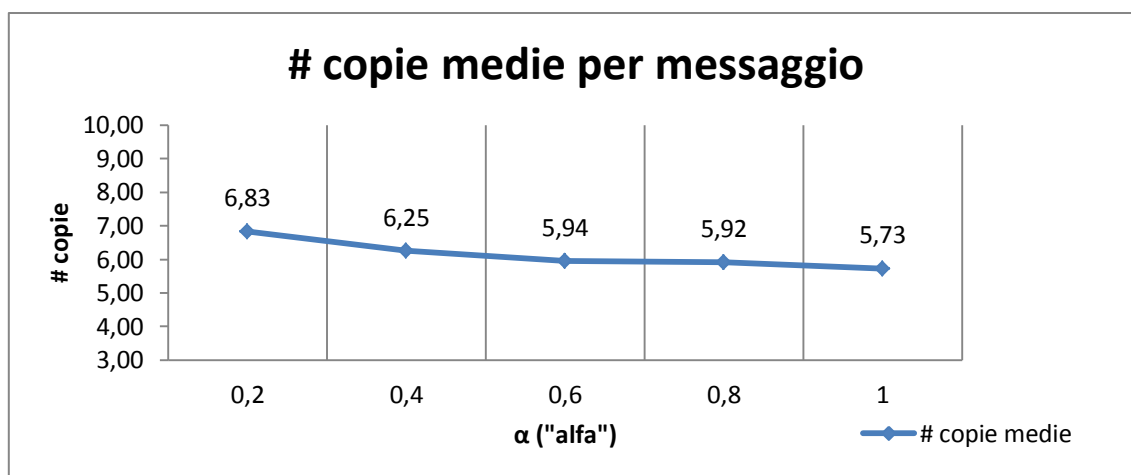
- i pacchetti inoltrati rientranti nel Caso I diminuiscono in quanto i nodi “amici” ( $30 - 30\alpha$ ) diminuiscono e, di conseguenza, anche il numero di nodi che risultano avere contemporaneamente sia un numero di ping maggiore sia il destinatario tra la propria lista amici;
- i pacchetti inoltrati rientranti nel Caso II aumentano in quanto al crescere di  $\alpha$  i nodi vengono sparsi per la rete e il numero di messaggi generati ( $30\alpha$ ) aumenta; inoltre i nodi “amici” ( $30 - 30\alpha$ ) tendono a diminuire favorendo l’inoltro con la condizione basata solo sul numero dei ping (appunto il Caso II);
- i pacchetti inoltrati rientranti nel Caso III rimangono pressoché costanti in quanto seppur il numero di nodi “amici” ( $30 - 30\alpha$ ) diminuisca al crescere di  $\alpha$ , il numero di messaggi generati aumenta e perciò vengono instradati ugualmente una piccola % di pacchetti con questa condizione;
- i pacchetti inoltrati rientranti nel Caso IV rimangono sempre 0 in quanto la loro probabilità di instradamento è fissata a 0.

Vediamo i risultati relativi alla seconda serie di simulazioni:



**Grafico 4.69:** mostra l’overhead della rete al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.25$ ,  $Q = 0.75$

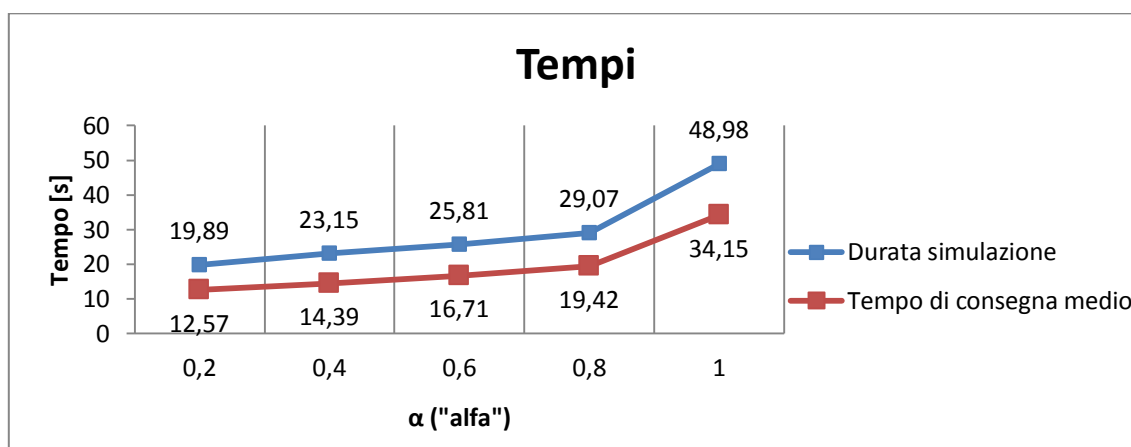
Il Grafico 4.69 mostra l’overhead di rete; al crescere di  $\alpha$  anche l’overhead aumenta in quanto il numero di messaggi generati dai nodi ( $30 \alpha$ ) cresce.



**Grafico 4.70:** mostra il numero di copie medie per messaggio al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.25$ ,  $Q = 0.75$

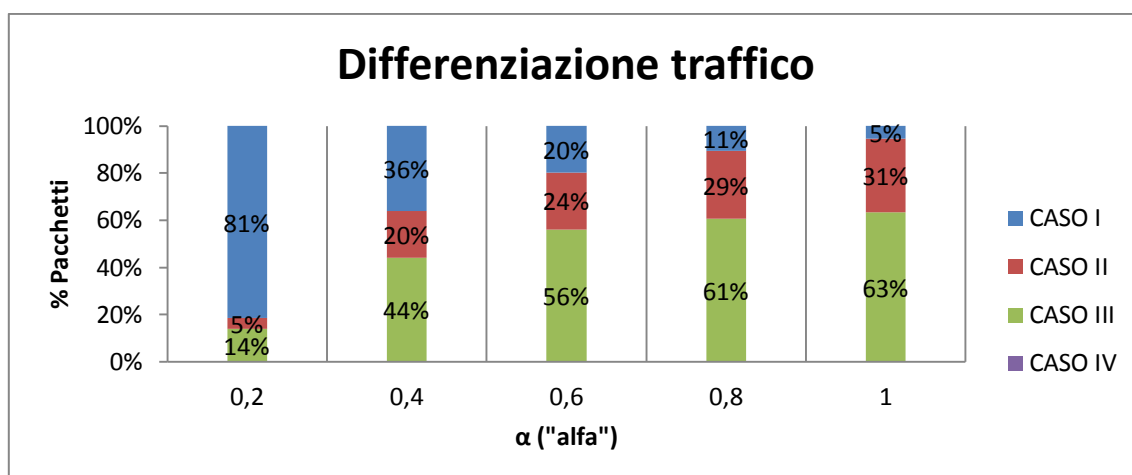
Nel Grafico 4.70 viene mostrato l'andamento del numero di copie medio dei messaggi; rispetto alla prima serie di simulazioni, al crescere di  $\alpha$ , tale valore diminuisce in quanto il numero di nodi "amici" ( $30 - 30\alpha$ ) decresce e dunque verranno fatte meno copie dei vari messaggi; la probabilità  $P$  di passare un messaggio ad un altro nodo è troppo bassa per provare a invertire il trend al ribasso; infatti, in questo caso, l'instradamento dei vari messaggi avviene con più probabilità per il Caso 3 ( $Q = 0.75$ ) ed è lecito aspettarsi questo comportamento.

La durata di simulazione e il tempo di consegna medio (vedi Grafico 4.71) all'aumentare di  $\alpha$  crescono in quanto il numero di messaggi da portare a destinazione è maggiore e il numero di "amici" diminuisce; nel caso pessimo ( $\alpha = 1$ ) per il protocollo ibrido con  $Q = 0.75$ , la probabilità  $P$  è troppo piccola (0.25) per avere una durata di simulazione e un tempo di consegna più brevi:



**Grafico 4.71:** mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.25$ ,  $Q = 0.75$

Nel Grafico 4.72 si può notare la suddivisione dei pacchetti inoltrati nella rete:

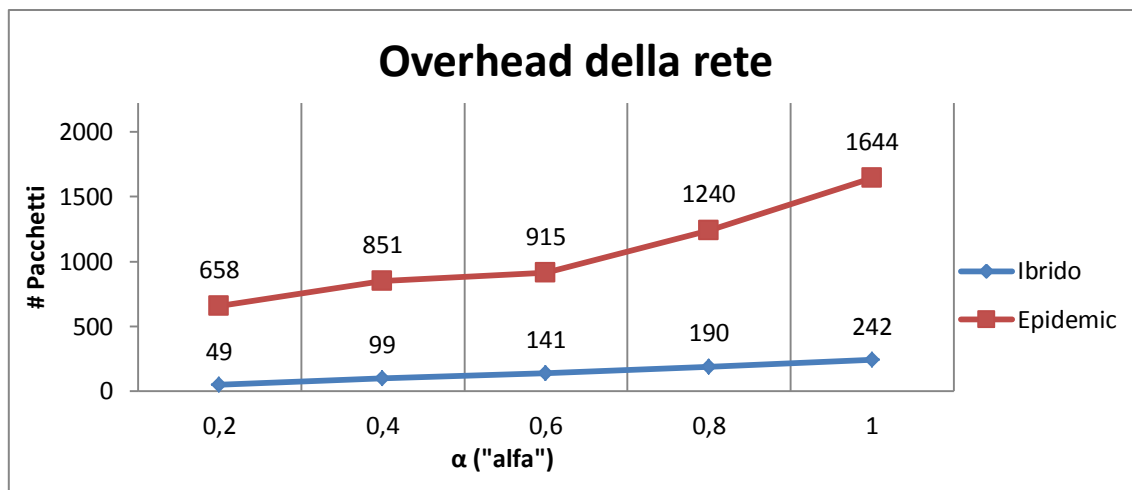


**Grafico 4.72:** mostra la divisione del traffico di tutti i messaggi della rete al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.25$ ,  $Q = 0.75$

Al crescere di  $\alpha$ , si verificano le suddette cose:

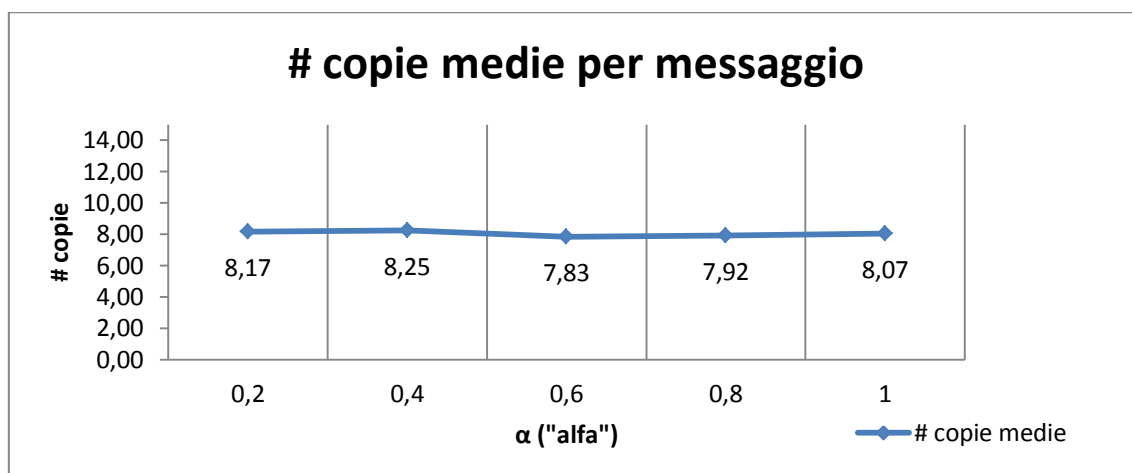
- i pacchetti inoltrati rientranti nel Caso I diminuiscono in quanto i nodi “amici” ( $30 - 30\alpha$ ) diminuiscono e di conseguenza anche il numero di nodi che risultano avere contemporaneamente sia un numero di ping maggiore sia il destinatario tra la propria lista amici;
- i pacchetti inoltrati rientranti nel Caso II aumentano in quanto al crescere di  $\alpha$  i nodi vengono sparsi per la rete e il numero di messaggi generati ( $30\alpha$ ) aumenta; nonostante il valore di  $P$  sia basso (0.25), l’inoltro con la condizione basata solo sul numero dei ping (appunto il Caso II) avviene lo stesso in buona misura;
- i pacchetti inoltrati rientranti nel Caso III aumentano per due motivi: il primo è che il numero dei messaggi da far arrivare a destinazione è maggiore ( $30\alpha$ ); il secondo è che seppur il numero di nodi “amici” ( $30 - 30\alpha$ ) diminuisca al crescere di  $\alpha$ , il valore di  $Q$  (0.75) è superiore di quello di  $P$  (0.25) e perciò la maggior parte dei messaggi viene trasmessa con questa condizione;
- i pacchetti inoltrati rientranti nel Caso IV rimangono sempre 0 in quanto la loro probabilità di instradamento è fissata a 0.

Vediamo i risultati relativi alla terza serie di simulazioni:



**Grafico 4.73:** mostra l'overhead della rete al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.50$ ,  $Q = 0.50$

Nel Grafico 4.73 viene mostrato l'overhead di rete; all'aumentare di  $\alpha$  anche l'overhead cresce in quanto vi è un numero maggiore di messaggi da consegnare ( $30\alpha$ ).

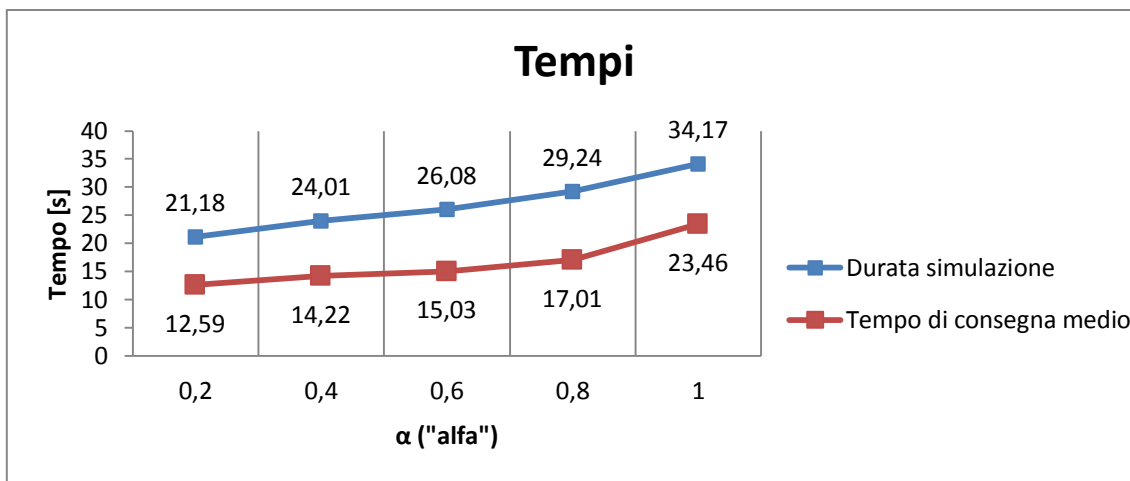


**Grafico 4.74:** mostra il numero di copie medie per messaggio al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.50$ ,  $Q = 0.50$

Nel Grafico 4.74 viene mostrato il numero di copie medie per messaggio; in questa serie di simulazioni il numero di copie si assesta sullo stesso valore; il motivo di ciò è che anche se il numero dei nodi "amici" ( $30 - 30\alpha$ ) diminuisce, il numero di copie rimane identico in quanto anche la condizione dei ping ( $P = 0.50$ ) favorisce la copia dei messaggi.

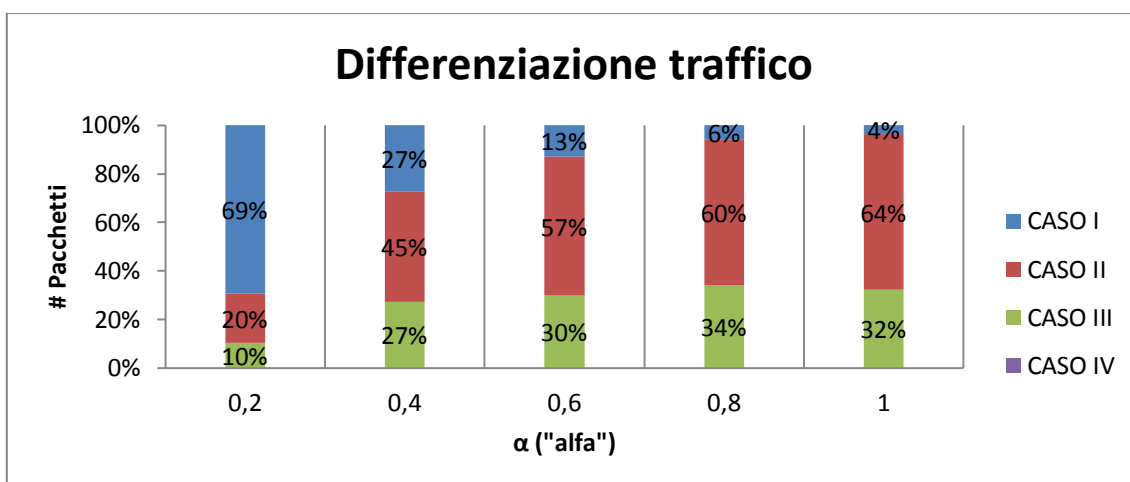
Anche in questo caso sia la durata di simulazione sia il tempo di consegna medio (vedi Grafico 4.75) aumentano in quanto il numero di messaggi da consegnare ( $30\alpha$ ) aumenta al crescere di  $\alpha$ :





**Grafico 4.75:** mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.50$ ,  $Q = 0.50$

Nel Grafico 4.76 si può notare la suddivisione dei pacchetti inoltrati nella rete:



**Grafico 4.76:** mostra la divisione del traffico di tutti i messaggi della rete al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.50$ ,  $Q = 0.50$

Al crescere di  $\alpha$ , si verificano le suddette cose:

- i pacchetti inoltrati rientranti nel Caso I diminuiscono in quanto i nodi "amici" ( $30 - 30\alpha$ ) diminuiscono e di conseguenza anche il numero di nodi che risultano avere contemporaneamente sia un numero di ping maggiore sia il destinatario tra la propria lista amici;
- i pacchetti inoltrati rientranti nel Caso II aumentano in quanto al crescere di  $\alpha$  i nodi vengono sparsi per la rete e il numero di messaggi generati ( $30\alpha$ ) aumenta;
- i pacchetti inoltrati rientranti nel Caso III aumentano in quanto il numero dei messaggi da far arrivare a destinazione è maggiore ( $30\alpha$ );

- i pacchetti inoltrati rientranti nel Caso IV rimangono sempre 0 in quanto la loro probabilità di instradamento è fissata a 0.

Ciononostante, i messaggi inoltrati mediante il Caso II saranno sempre maggiori di quelli inoltrati mediante il Caso III in quanto il numero di “amici” ( $30 - 30\alpha$ ) diminuisce al crescere di  $\alpha$  e l’aumento di pacchetti da portare a destinazione ( $30\alpha$ ) non colma questo gap.

#### 4.5.2 - Protocollo Ibrido Multicast

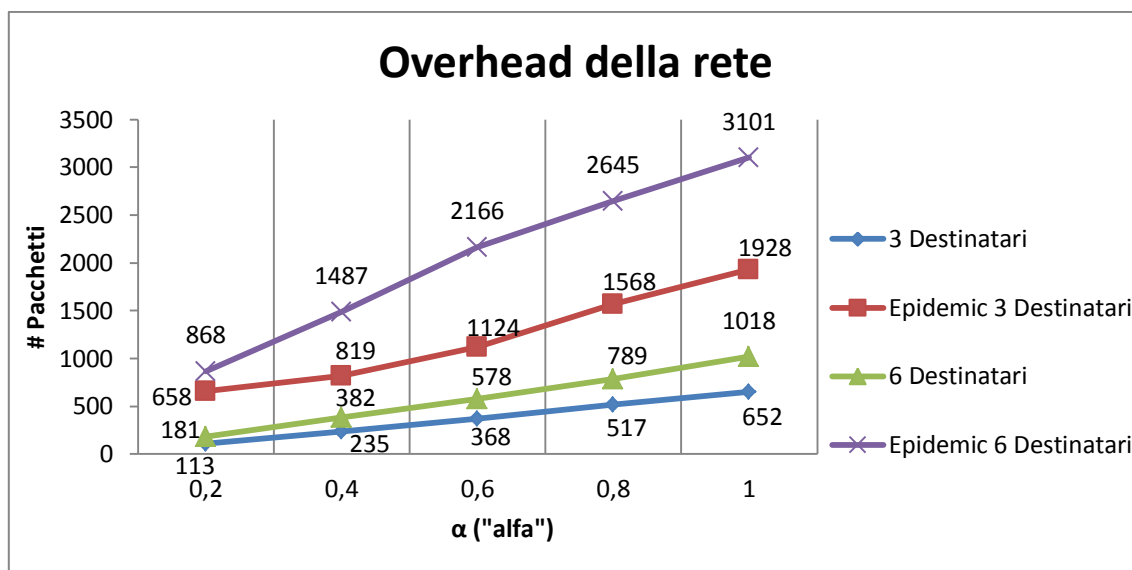
Per simulare tale protocollo è stata utilizzata la stessa struttura di rete spiegata nel Capitolo 4.4.2 con la sola differenza che si è atteso un tempo di 8 [s] prima di inviare i messaggi per fare in modo che i nodi si scambiassero i ping; il messaggio per arrivare a destinazione potrà essere passato ad altri nodi secondo la funzione obiettivo scritta nel Capitolo 3; le simulazioni fatte prendono fissi alcuni parametri come:

- Raggio di copertura: 30 [mt];
- Frequenza ping:  $1 \left[ \frac{ping}{sec} \right]$ .

Al fine di valutare le prestazioni di tale protocollo, sono state fatte 3 serie di simulazioni:

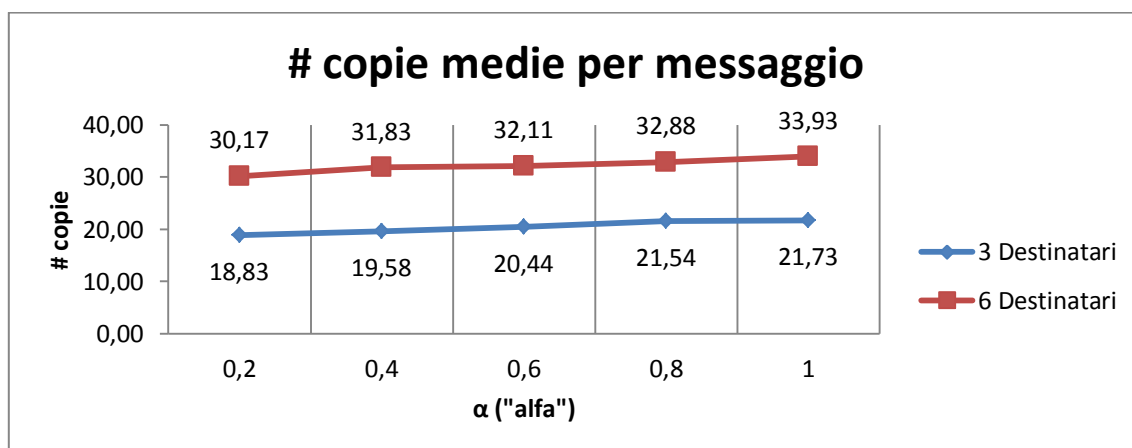
- nella prima serie viene fatto variare il valore di  $\alpha$  (0.2 - 0.4 - 0.6 - 0.8 - 1) con  $P = 0.75$  e  $Q = 0.25$  sia nel caso a 3 destinatari, sia nel caso a 6 destinatari;
- nella seconda serie viene fatto variare il valore di  $\alpha$  (0.2 - 0.4 - 0.6 - 0.8 - 1) con  $P = 0.25$  e  $Q = 0.75$  sia nel caso a 3 destinatari, sia nel caso a 6 destinatari;
- nella terza serie viene fatto variare il valore di  $\alpha$  (0.2 - 0.4 - 0.6 - 0.8 - 1) con  $P = 0.50$  e  $Q = 0.50$  sia nel caso a 3 destinatari, sia nel caso a 6 destinatari;

Vediamo ora i risultati della prima serie di simulazioni:



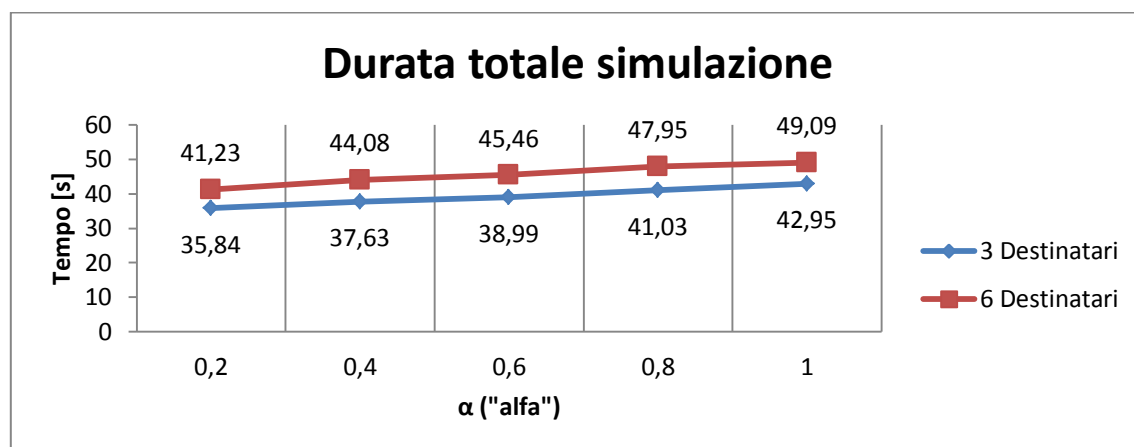
**Grafico 4.77:** mostra l'overhead della rete al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.75$ ,  $Q = 0.25$  sia a 3 destinatari sia a 6 destinatari

Nel Grafico 4.77 viene mostrato l'overhead di rete; al crescere di  $\alpha$  anche l'overhead sale in quanto il numero di messaggi da portare a destinazione aumenta ( $30\alpha$ ); comportamento simile lo si può osservare nel Grafico 4.78 dove viene mostrato l'andamento del numero di copie medie: tale valore aumenta in quanto il numero di messaggi aumenta ( $30\alpha$ ) e anche se il numero di "amici" ( $30 - 30\alpha$ ) decresce, i messaggi vengono inoltrati maggiormente secondo la condizione basata sul numero dei ping (Caso II con  $P = 0.75$ ) e dunque è corretto aspettarsi un aumento del numero di copie medie. In entrambi i grafici comunque, sia l'overhead di rete, sia il numero di copie medie è maggiore nel caso a 6 destinatari rispetto al caso a 3 destinatari in quanto il messaggio deve arrivare un numero maggiore di nodi.



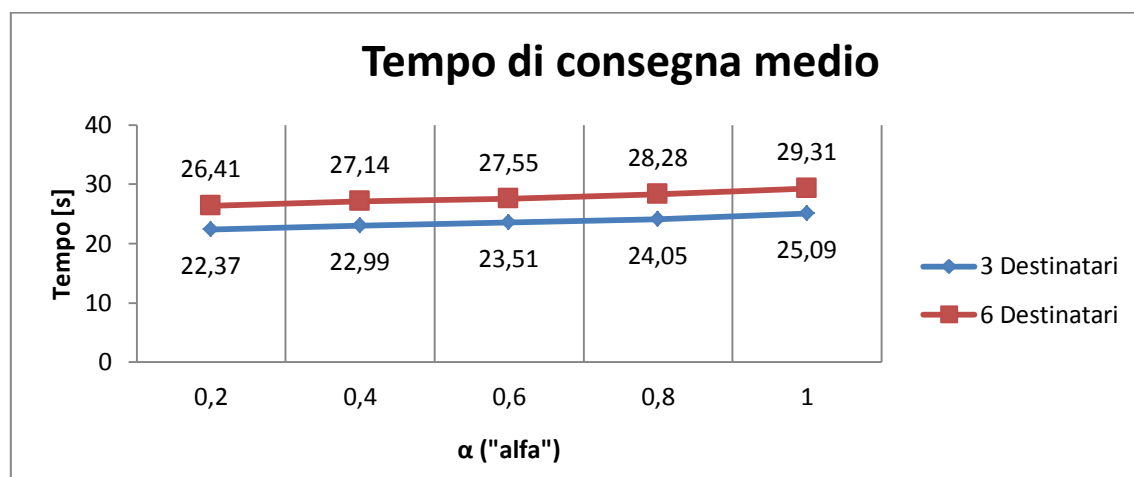
**Grafico 4.78:** mostra il numero di copie medie per messaggio al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.75$ ,  $Q = 0.25$  sia a 3 destinatari sia a 6 destinatari

Nel Grafico 4.79 viene mostrata la durata di simulazione; tale durata aumenta al crescere di  $\alpha$  in quanto il numero di messaggi da consegnare ( $30\alpha$ ) è maggiore:



**Grafico 4.79:** mostra la durata totale della simulazione al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.75$ ,  $Q = 0.25$  sia a 3 destinatari sia a 6 destinatari

Anche il tempo di consegna medio (vedi Grafico 4.80) aumenta in quanto al crescere di  $\alpha$  in quanto il numero di messaggi da consegnare ( $30\alpha$ ) è maggiore:



**Grafico 4.80:** mostra il tempo di consegna medio di ogni messaggio al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.75$ ,  $Q = 0.25$  sia a 3 destinatari sia a 6 destinatari

Nel Grafico 4.81a e 4.81b si può notare la suddivisione dei pacchetti inoltrati nella rete sia nel caso a 3 destinatari sia nel caso a 6 destinatari:

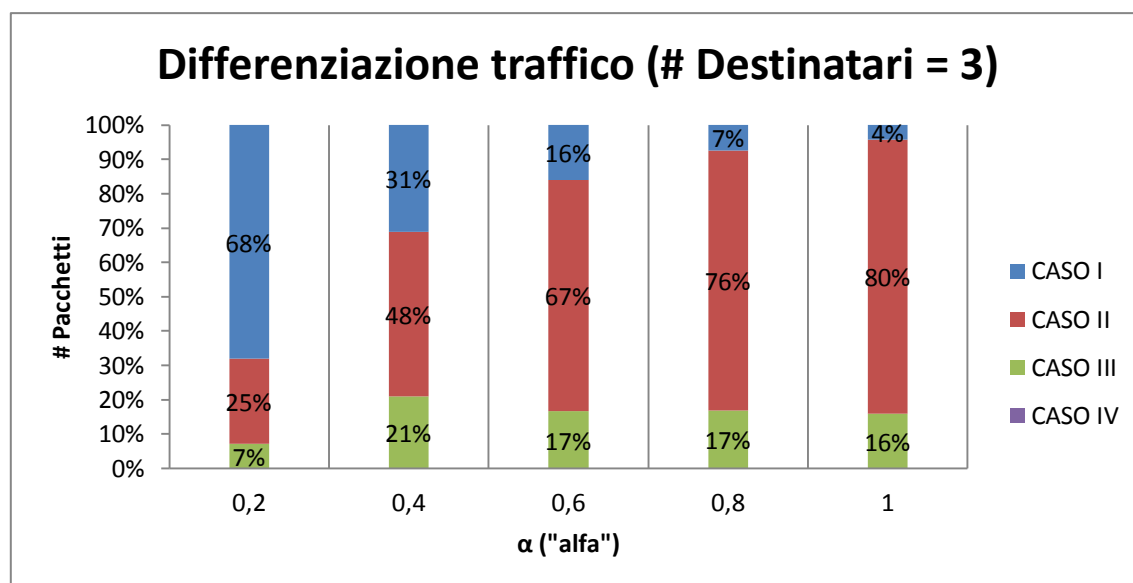


Grafico 4.81a: mostra la divisione del traffico di tutti i messaggi della rete al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.75$ ,  $Q = 0.25$  nel caso a 3 destinatari

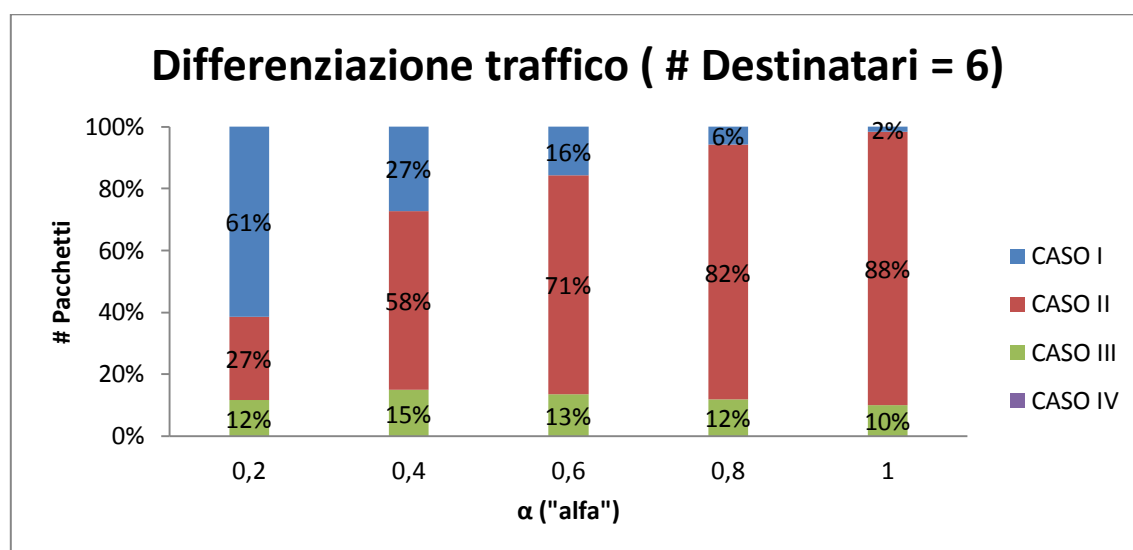


Grafico 4.81b: mostra la divisione del traffico di tutti i messaggi della rete al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.75$ ,  $Q = 0.25$  nel caso a 6 destinatari

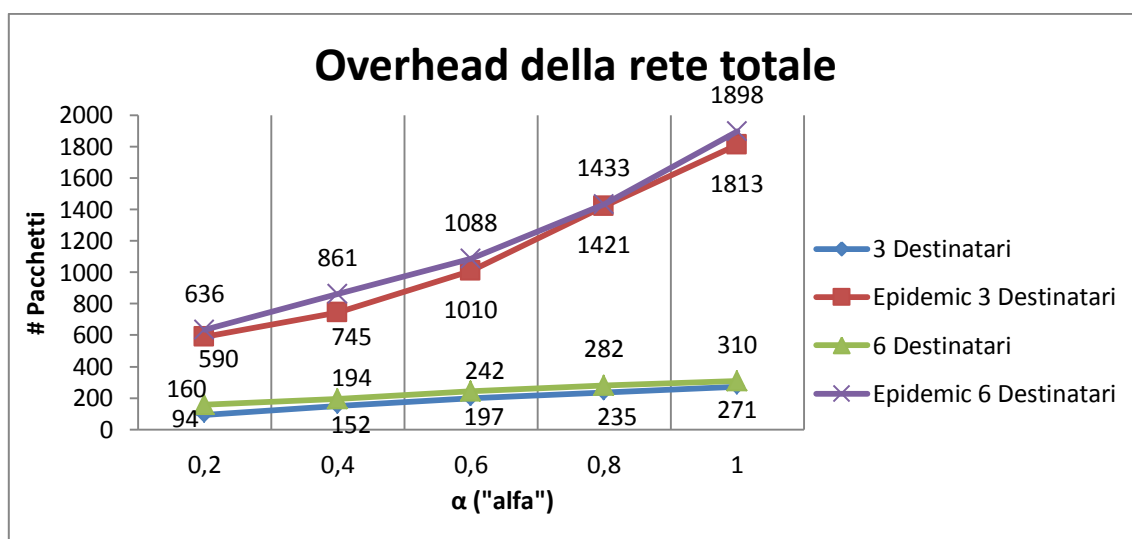
Al crescere di  $\alpha$ , sia nel caso a 3 destinatari sia nel caso a 6 destinatari, si verificano le suddette cose:

- i pacchetti inoltrati rientranti nel Caso I diminuiscono in quanto i nodi "amici" ( $30 - 30\alpha$ ) diminuiscono e di conseguenza anche i nodi che risultano avere

contemporaneamente sia un numero di ping maggiore sia il destinatario tra la propria lista amici;

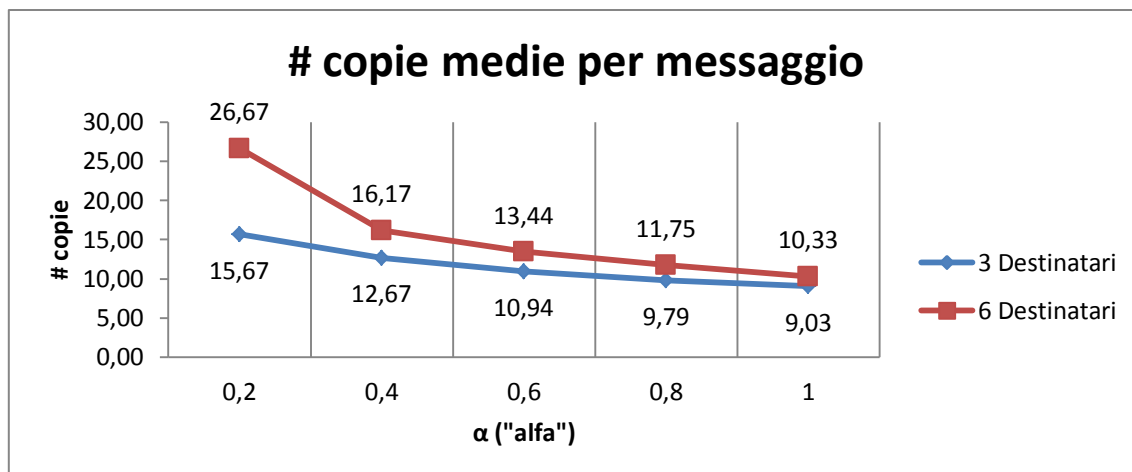
- i pacchetti inoltrati rientranti nel Caso II aumentano in quanto al crescere di  $\alpha$  i nodi vengono sparsi per la rete e il numero di messaggi generati ( $30\alpha$ ) aumenta; inoltre i nodi “amici” ( $30 - 30\alpha$ ) tendono a diminuire favorendo l’inoltro con la condizione basata solo sul numero dei ping (appunto il Caso II);
- i pacchetti inoltrati rientranti nel Caso III rimangono pressoché costanti in quanto seppur il numero di nodi “amici” ( $30 - 30\alpha$ ) diminuisca al crescere di  $\alpha$ , il numero di messaggi generati aumenta e perciò vengono instradati ugualmente una piccola % di pacchetti con questa condizione;
- i pacchetti inoltrati rientranti nel Caso IV rimangono sempre 0 in quanto la loro probabilità di instradamento è fissata a 0.

Vediamo i risultati ricavati dalla seconda serie di simulazioni:



**Grafico 4.82:** mostra l’overhead della rete al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.25$ ,  $Q = 0.75$  sia a 3 destinatari sia a 6 destinatari

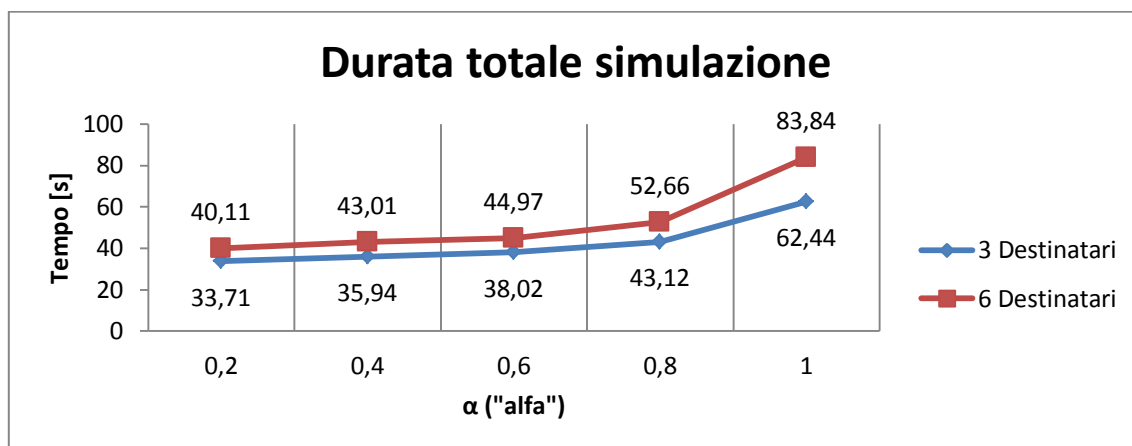
Nel Grafico 4.82 viene mostrato l’overhead di rete; al crescere di  $\alpha$  anche l’overhead sale in quanto il numero di messaggi nella rete cresce ( $30 \alpha$ ).



**Grafico 4.83:** mostra il numero di copie medie per messaggio al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.25$ ,  $Q = 0.75$  sia a 3 destinatari sia a 6 destinatari

Nel Grafico 4.83 viene mostrato il numero di copie medie per messaggio; al crescere di  $\alpha$  tale valore scende in quanto il numero di nodi “amici” ( $30 - 30\alpha$ ) decresce e dunque vengono fatte meno copie di un messaggio; infatti  $P (= 0.25)$  è troppo piccola per invertire il trend al ribasso e dunque tale valore diminuisce.

Il tempo di simulazione (vedi Grafico 4.84) diminuisce al crescere di  $\alpha$  in quanto vi sono più messaggi ( $30\alpha$ ) da far arrivare a destinazione.



**Grafico 4.84:** mostra la durata totale della simulazione al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.25$ ,  $Q = 0.75$  sia a 3 destinatari sia a 6 destinatari

Anche il tempo di consegna medio (vedi Grafico 4.85) aumenta al crescere di  $\alpha$  in quanto il numero di messaggi da portare a destinazione ( $30\alpha$ ) aumenta e il numero di nodi “amici” ( $30 - 30\alpha$ ) diminuisce.

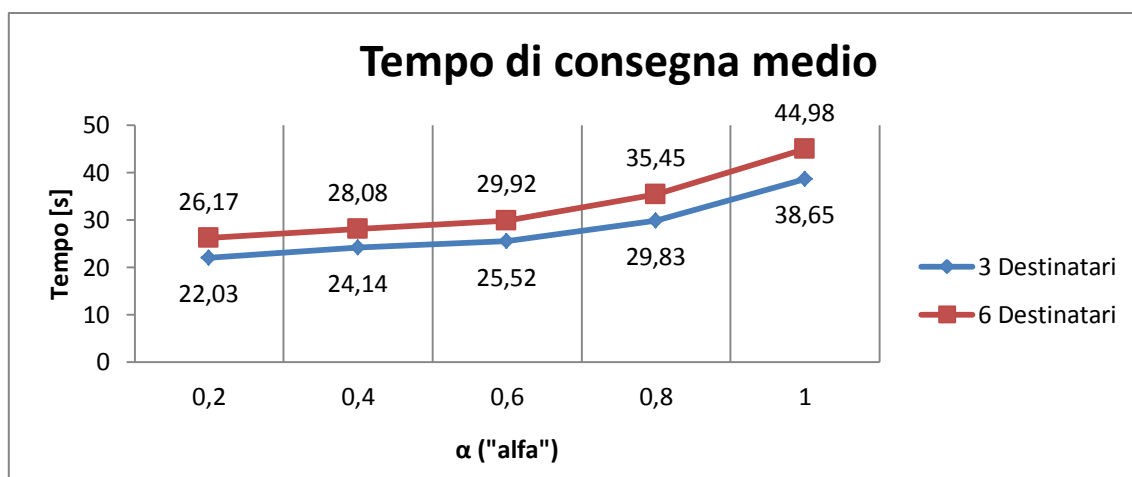


Grafico 4.85: mostra il tempo di consegna medio di ogni messaggio al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.25$ ,  $Q = 0.75$  sia a 3 destinatari sia a 6 destinatari

Nel Grafico 4.86a e 4.86b si può notare la suddivisione dei pacchetti inoltrati nella rete sia nel caso a 3 destinatari sia nel caso a 6 destinatari:

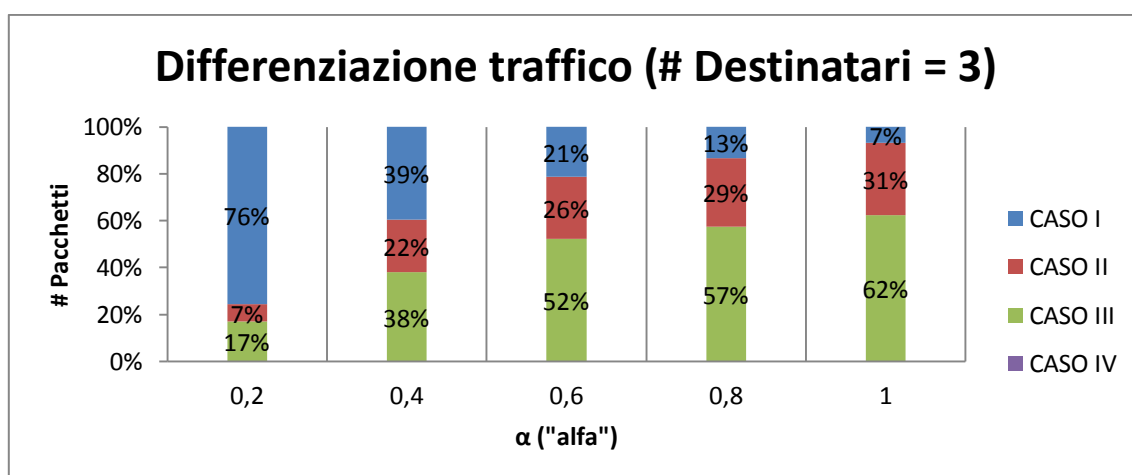


Grafico 4.86a: mostra la divisione del traffico di tutti i messaggi della rete al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.25$ ,  $Q = 0.75$  nel caso a 3 destinatari

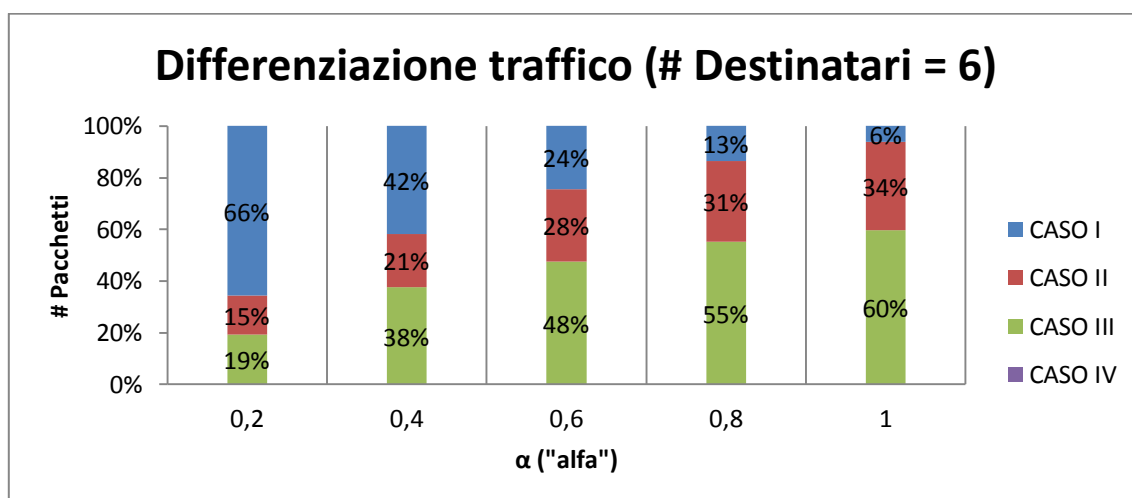


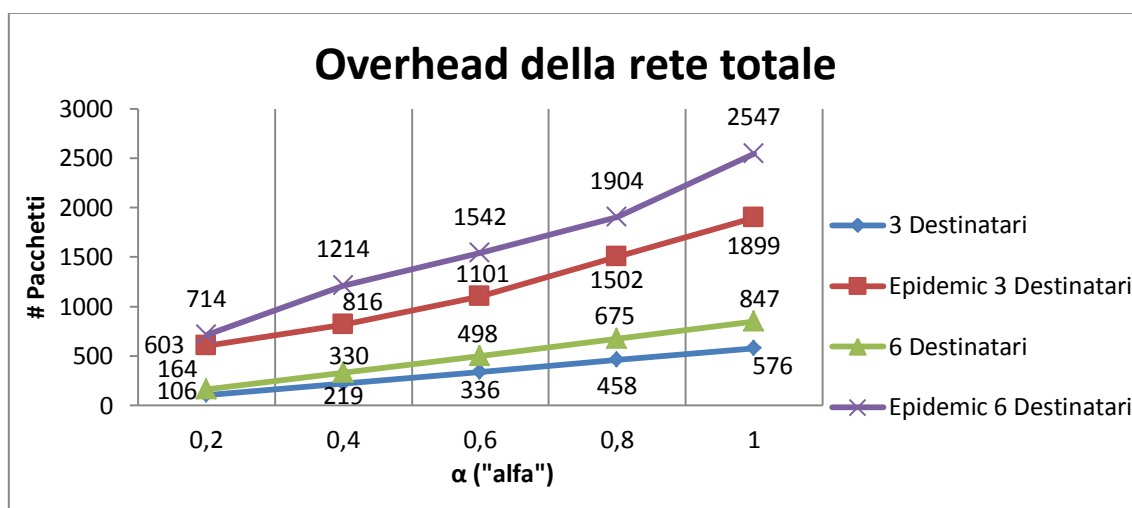
Grafico 4.86b: mostra la divisione del traffico di tutti i messaggi della rete al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.25$ ,  $Q = 0.75$  nel caso a 6 destinatari



Al crescere di  $\alpha$ , sia nel caso a 3 destinatari sia nel caso a 6 destinatari, si verificano le suddette cose:

- i pacchetti inoltrati rientranti nel Caso I diminuiscono in quanto i nodi “amici” ( $30 - 30\alpha$ ) diminuiscono e di conseguenza anche il numero di nodi che risultano avere contemporaneamente sia un numero di ping maggiore sia il destinatario tra la propria lista amici;
- i pacchetti inoltrati rientranti nel Caso II aumentano in quanto al crescere di  $\alpha$  i nodi vengono sparsi per la rete e il numero di messaggi generati ( $30\alpha$ ) aumenta; nonostante il valore di  $P$  sia basso (0.25), l’inoltro con la condizione basata solo sul numero dei ping (appunto il Caso II) avviene ugualmente in buona misura;
- i pacchetti inoltrati rientranti nel Caso III aumentano in quanto il numero dei messaggi da far arrivare a destinazione è maggiore ( $30\alpha$ ); inoltre, seppur il numero di nodi “amici” ( $30 - 30\alpha$ ) diminuisce al crescere di  $\alpha$ , il valore di  $Q$  (0.75) è superiore di quello di  $P$  (0.25) e perciò la maggior parte dei messaggi viene trasmessa con questa condizione;
- i pacchetti inoltrati rientranti nel Caso IV rimangono sempre 0 in quanto la loro probabilità di instradamento è fissata a 0.

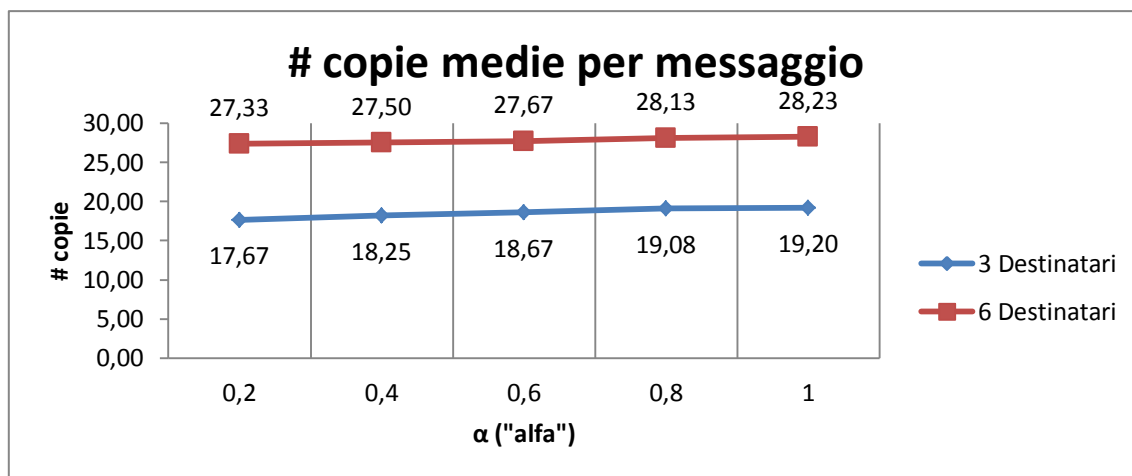
Vediamo i risultati relativi alla terza serie di simulazioni:



**Grafico 4.87:** mostra l’overhead della rete al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.50$ ,  $Q = 0.50$  sia a 3 destinatari sia a 6 destinatari

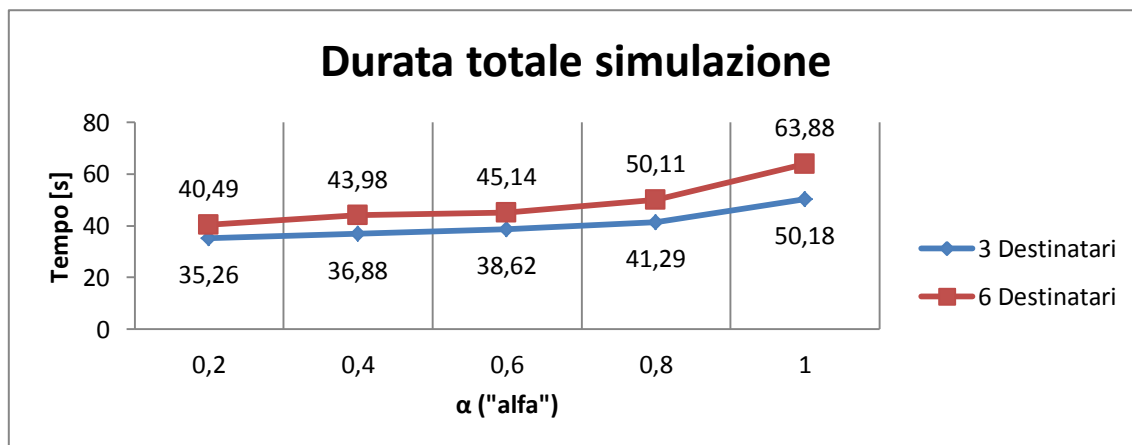
Come mostrato nel Grafico 4.87 l’overhead di rete aumenta al crescere di  $\alpha$  perchè il numero di messaggi immessi nella rete ( $30\alpha$ ) cresce; l’overhead nel caso a 6 destinatari

sarà sempre maggiore rispetto al caso a 3 destinatari in quanto il numero di nodi destinatari è maggiore.



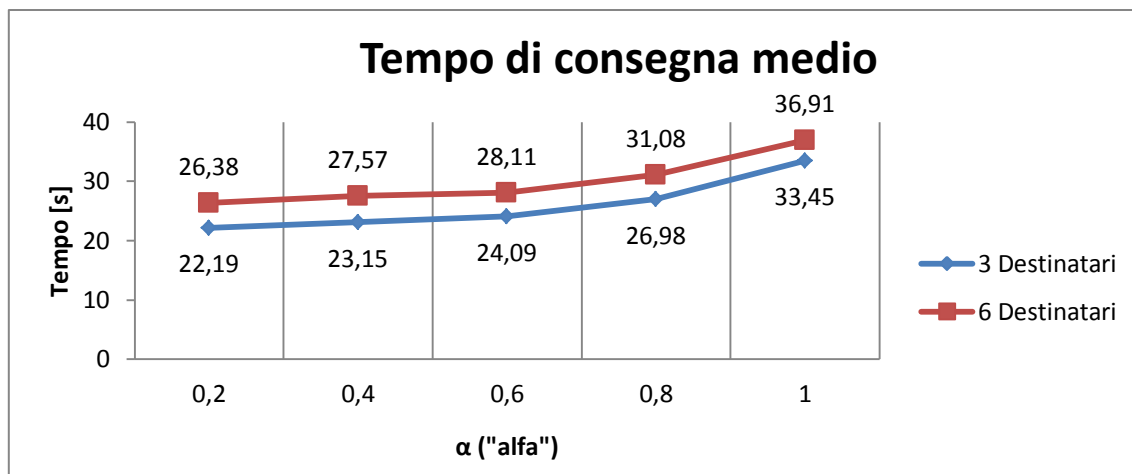
**Grafico 4.88:** mostra il numero di copie medie per messaggio al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.50$ ,  $Q = 0.50$  sia a 3 destinatari sia a 6 destinatari

Nel Grafico 4.88 viene mostrato il numero di copie medie; tale valore tende leggermente ad aumentare in quanto seppur il numero di "amici" ( $30 - 30\alpha$ ) diminuisce al crescere di  $\alpha$ , il numero di portatori non decresce perchè anche con una probabilità  $P$  ( $= 0.50$ ) vengono instradati i messaggi (e dunque fatte copie) ai nodi con più ping.



**Grafico 4.89:** mostra la durata totale della simulazione al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.50$ ,  $Q = 0.50$  sia a 3 destinatari sia a 6 destinatari

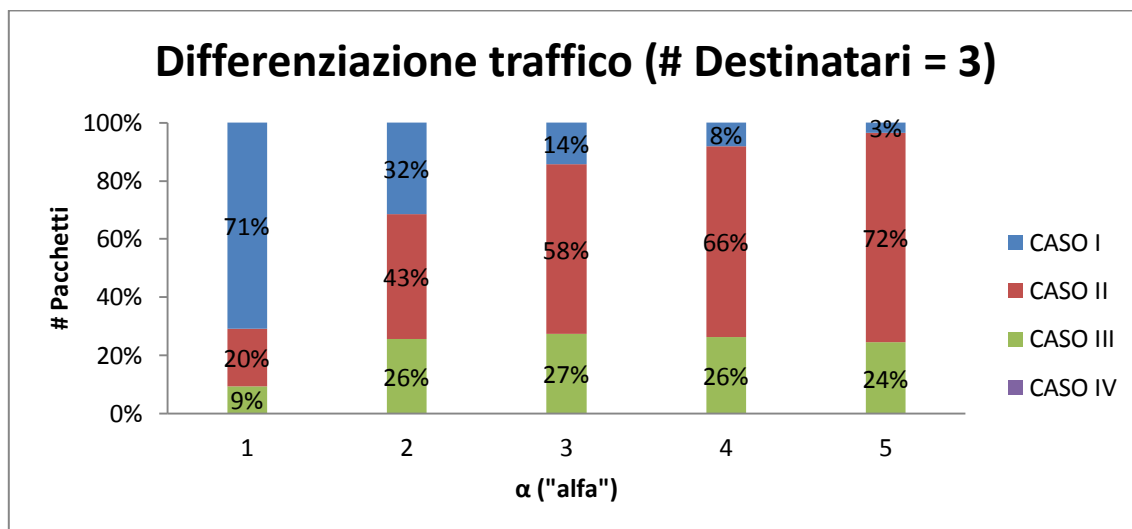
Nel Grafico 4.89 viene mostrata la durata totale della simulazione; si può notare che al crescere di  $\alpha$  tale valore aumenta in quanto il numero di messaggi ( $30 \alpha$ ) da portare a destinazione cresce; tale durata è sempre maggiore nel caso a 6 destinatari rispetto al caso a 3, in quanto il numero di destinatari è maggiore.



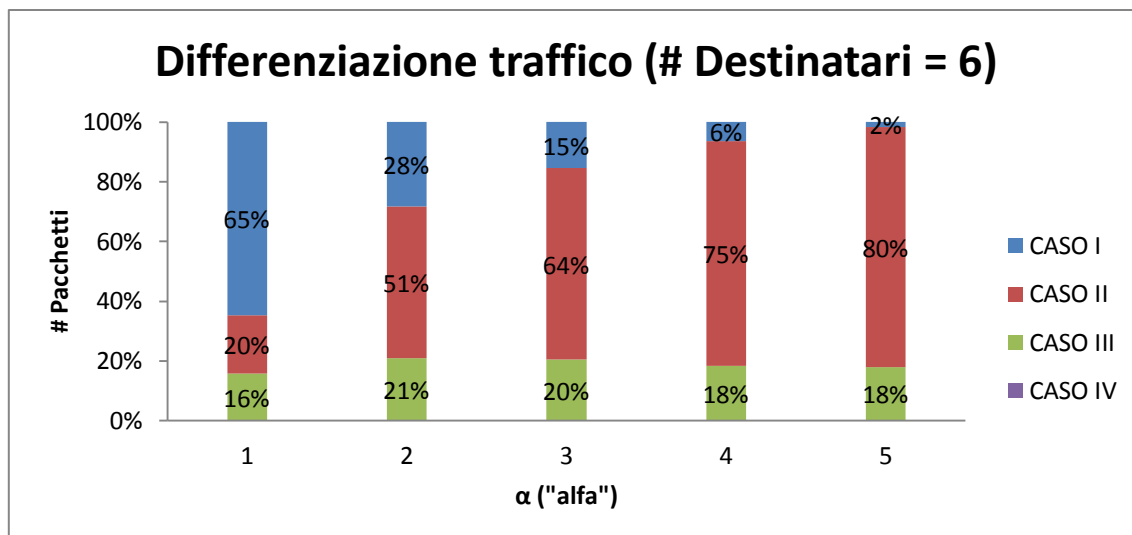
**Grafico 4.90:** mostra il tempo di consegna medio di ogni messaggio al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.50$ ,  $Q = 0.50$  sia a 3 destinatari sia a 6 destinatari

Nel Grafico 4.90 viene riportato il tempo di consegna medio per il messaggio; tale valore cresce all'aumentare di  $\alpha$  in quanto il numero di "amici" ( $30 - 30\alpha$ ) decresce e l'instradamento con probabilità  $P (= 0.50)$  se un nodo ha un numero maggiore di ping non è sufficiente a tenere costante il tempo di consegna.

Nel Grafico 4.91a e 4.91b si può notare la suddivisione dei pacchetti inoltrati nella rete sia nel caso a 3 destinatari sia nel caso a 6 destinatari:



**Grafico 4.91a:** mostra la divisione del traffico di tutti i messaggi della rete al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.50$ ,  $Q = 0.50$  nel caso a 3 destinatari



**Grafico 4.91b:** mostra la divisione del traffico di tutti i messaggi della rete al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.50$ ,  $Q = 0.50$  nel caso a 6 destinatari

Al crescere di  $\alpha$ , sia nel caso a 3 destinatari sia nel caso a 6 destinatari, si verificano le suddette cose:

- i pacchetti inoltrati rientranti nel Caso I diminuiscono in quanto i nodi “amici” ( $30 - 30\alpha$ ) diminuiscono e di conseguenza anche il numero di nodi che risultano avere contemporaneamente sia un numero di ping maggiore sia il destinatario tra la propria lista amici;
- i pacchetti inoltrati rientranti nel Caso II aumentano in quanto al crescere di  $\alpha$  i nodi vengono sparsi per la rete e il numero di messaggi generati ( $30\alpha$ ) aumenta;
- i pacchetti inoltrati rientranti nel Caso III rimangono pressoché costanti in quanto il numero dei messaggi da far arrivare a destinazione è maggiore ( $30\alpha$ ); inoltre, seppur il numero di nodi “amici” ( $30 - 30\alpha$ ) diminuisce al crescere di  $\alpha$ , il valore di  $Q$  ( $= 0.50$ ) permette ugualmente un buon numero di messaggi instradati con questa condizione;
- i pacchetti inoltrati rientranti nel Caso IV rimangono sempre 0 in quanto la loro probabilità di instradamento è fissata a 0.

#### 4.5.3 - Protocollo Ibrido Geocast

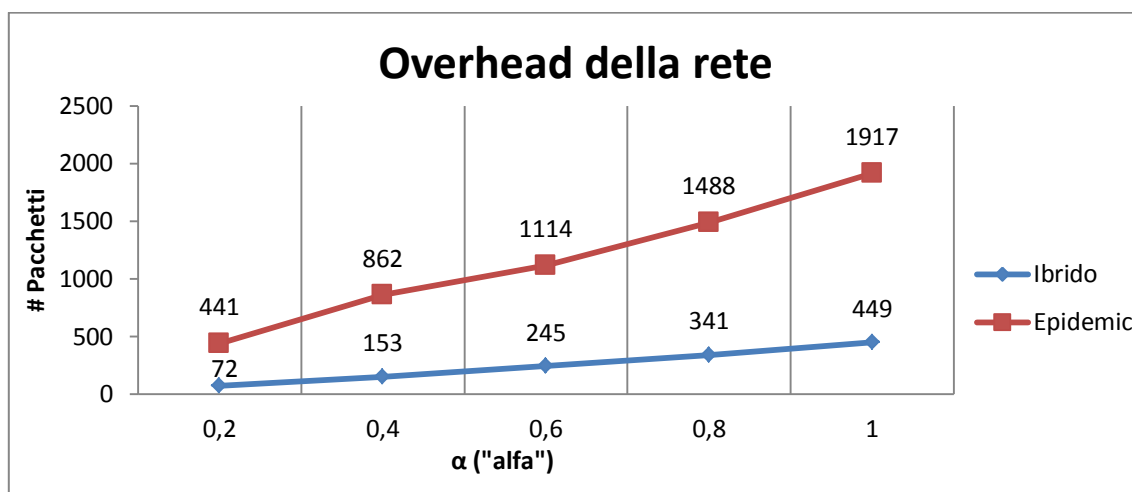
Per simulare tale protocollo è stata utilizzata la stessa struttura di rete spiegata nel Capitolo 4.4.3 con la sola differenza che si è atteso un tempo di 8 [s] prima di inviare i messaggi per fare in modo che i nodi si scambiassero i ping; il messaggio per arrivare a destinazione potrà essere passato ad altri nodi secondo la funzione obiettivo scritta nel Capitolo 3; le simulazioni fatte prendono fissi alcuni parametri come:

- Raggio di copertura: 30 [mt];
- Frequenza ping: 1 [ $\frac{ping}{sec}$ ].

Al fine di valutare le prestazioni di tale protocollo, sono state fatte 3 serie di simulazioni:

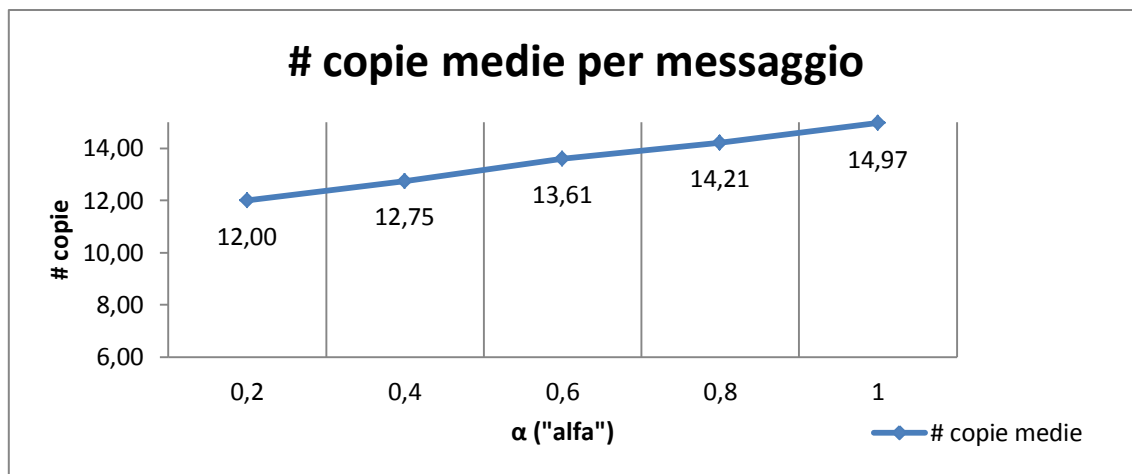
- nella prima serie viene fatto variare il valore di  $\alpha$  (0.2 - 0.4 - 0.6 - 0.8 - 1) con  $P = 0.75$  e  $Q = 0.25$ ;
- nella seconda serie viene fatto variare il valore di  $\alpha$  (0.2 - 0.4 - 0.6 - 0.8 - 1) con  $P = 0.25$  e  $Q = 0.75$ ;
- nella terza serie viene fatto variare il valore di  $\alpha$  (0.2 - 0.4 - 0.6 - 0.8 - 1) con  $P = 0.50$  e  $Q = 0.50$ ;

Vediamo ora i risultati della prima serie di simulazioni:



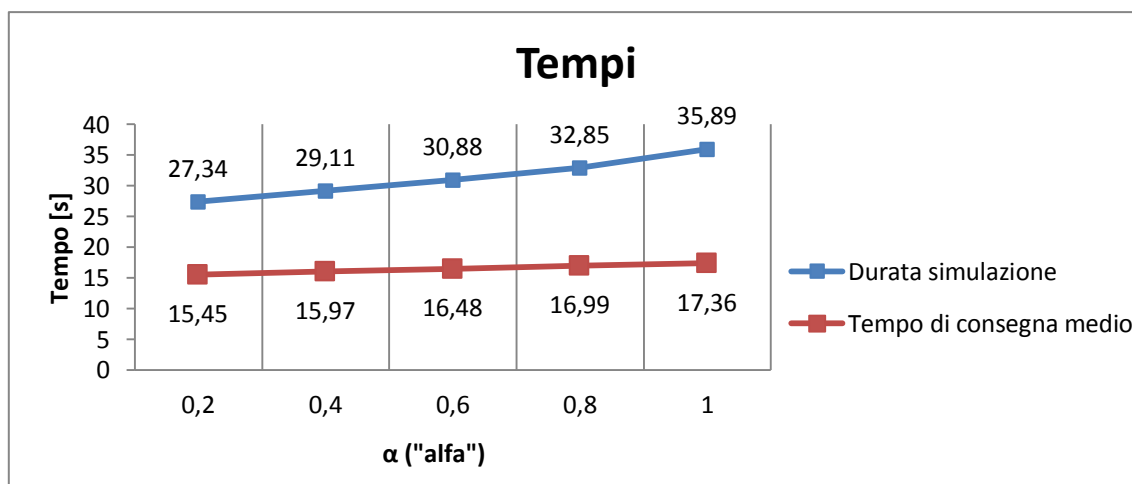
**Grafico 4.92:** mostra l'overhead della rete al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.75$ ,  $Q = 0.25$

Nel Grafico 4.92 viene mostrato l'overhead di rete; tale valore tende ad aumentare al crescere di  $\alpha$  in quanto il numero di messaggi generati ( $30\alpha$ ) sarà maggiore; anche per il numero di copie medie (vedi Grafico 4.93) si può osservare lo stesso andamento in quanto il numero di messaggi aumenta al crescere di  $\alpha$  e il numero di portatori rimane pressoché uguale in quanto la probabilità  $P$  ( $= 0.75$ ) di inviare i messaggi in base al numero di ping è maggiore della probabilità  $Q$  ( $= 0.25$ ) di inviare i messaggi in base al numero di Check-In.



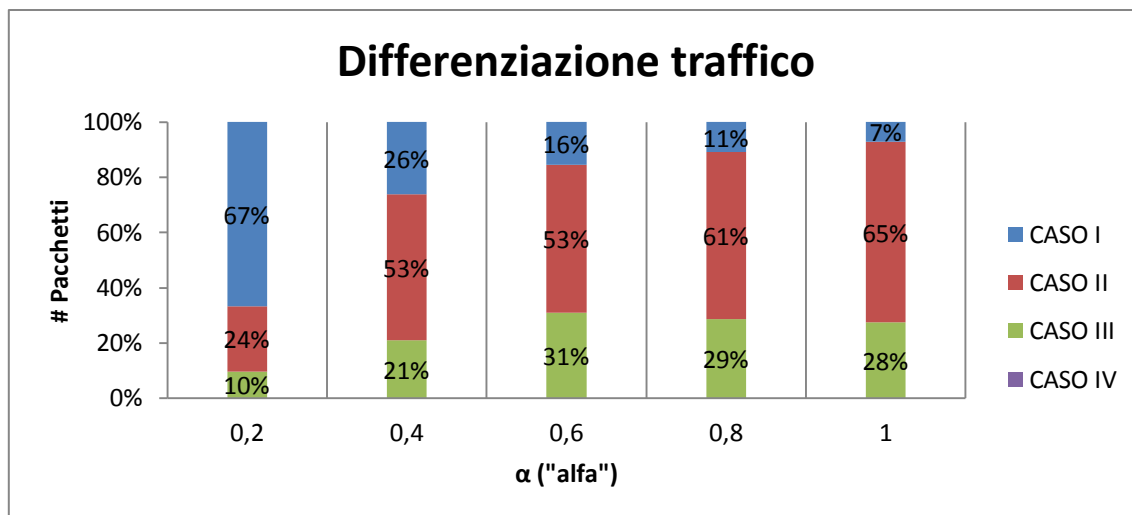
**Grafico 4.93:** mostra il numero di copie medie per messaggio al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.75$ ,  $Q = 0.25$

Nel Grafico 4.94 vengono mostrati la durata totale di simulazione e il tempo di consegna medio; entrambi aumentano al crescere di  $\alpha$  in quanto il numero di messaggi generati ( $30\alpha$ ) cresce:



**Grafico 4.94:** mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.75$ ,  $Q = 0.25$

Infine, nel Grafico 4.95 si può notare la suddivisione dei pacchetti inoltrati nella rete:

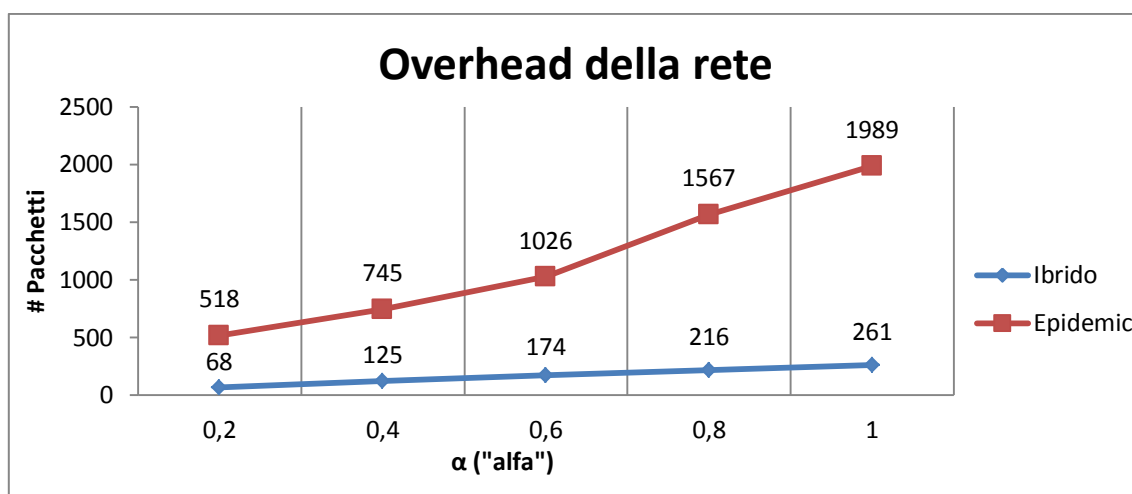


**Grafico 4.95:** mostra la divisione del traffico di tutti i messaggi della rete al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.75$ ,  $Q = 0.25$

Al crescere di  $\alpha$ , si verificano le suddette cose:

- i pacchetti inoltrati rientranti nel Caso I diminuiscono in quanto i nodi “major” ( $30 - 30\alpha$ ) diminuiscono e di conseguenza anche il numero di nodi che risultano avere contemporaneamente sia un numero di ping sia un numero di Check-In in quell’area maggiori;
- i pacchetti inoltrati rientranti nel Caso II aumentano in quanto al crescere di  $\alpha$  i nodi vengono sparsi per la rete e il numero di messaggi generati ( $30\alpha$ ) aumenta; inoltre seppure i nodi “major” ( $30 - 30\alpha$ ) tendono a diminuire, il numero di portatori per il Caso II non cambia;
- i pacchetti inoltrati rientranti nel Caso III tendono ad aumentare in quanto seppur il numero di nodi “major” ( $30 - 30\alpha$ ) diminuisca al crescere di  $\alpha$ , il numero di messaggi generati aumenta e perciò vengono instradati ugualmente una % di pacchetti con questa condizione;
- i pacchetti inoltrati rientranti nel Caso IV rimangono sempre 0 in quanto la loro probabilità di instradamento è fissata a 0.

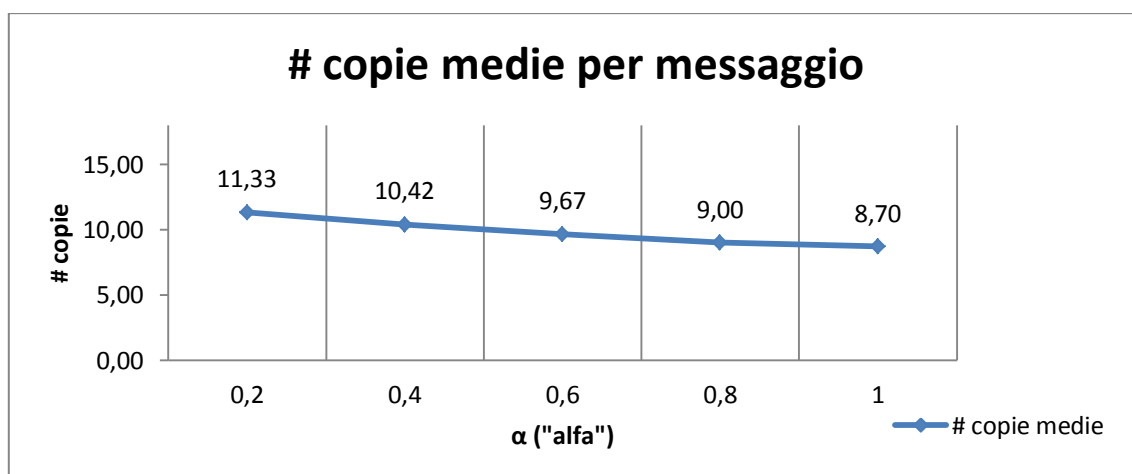
Vediamo i risultati relativi alla seconda serie di simulazioni:



**Grafico 4.96:** mostra l'overhead della rete al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.25$ ,  $Q = 0.75$

Nel Grafico 4.96 viene mostrato l'overhead di rete; al crescere di  $\alpha$  anche l'overhead aumenta in quanto il numero di messaggi generati ( $30\alpha$ ) sale.

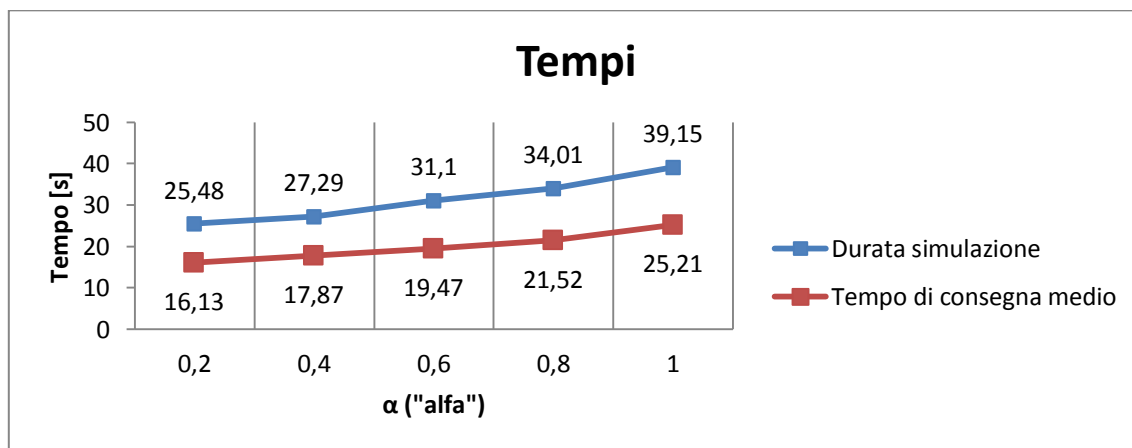
Per il numero di copie medie (vedi Grafico 4.97) si osserva invece un comportamento inverso; infatti al crescere di  $\alpha$  il numero di "major" ( $30 - 30\alpha$ ) decresce e di conseguenza anche il numero di possibili copie; nonostante nella rete anche gli altri nodi lineari ( $30\alpha$ ) possono fare i Check-in (e dunque essere migliori portatori), tale fatto non permette di invertire il trend al ribasso:



**Grafico 4.97:** mostra il numero di copie medie per messaggio al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.25$ ,  $Q = 0.75$

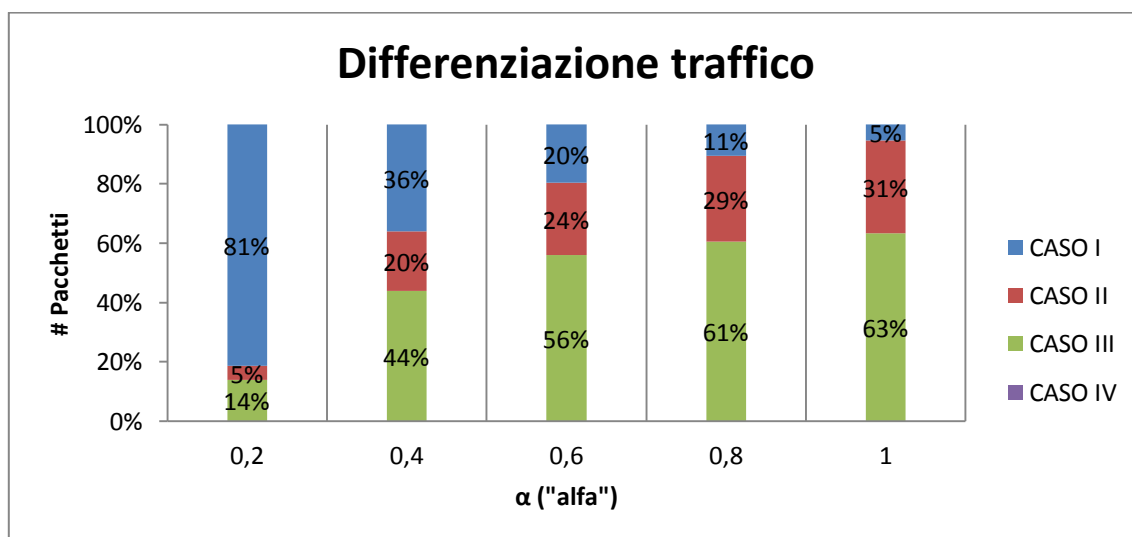


Nel Grafico 4.98 viene mostrata la durata totale di simulazione e il tempo di consegna medio; al crescere di  $\alpha$  entrambi aumentano in quanto il numero di messaggi generati ( $30\alpha$ ) è maggiore e il numero di “major” decresce e dunque i messaggi impiegano più tempo ad arrivare a destinazione:



**Grafico 4.98:** mostra la durata totale della simulazione a partire dall'invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.25$ ,  $Q = 0.75$

Nel Grafico 4.99 si può notare la suddivisione dei pacchetti inoltrati nella rete:

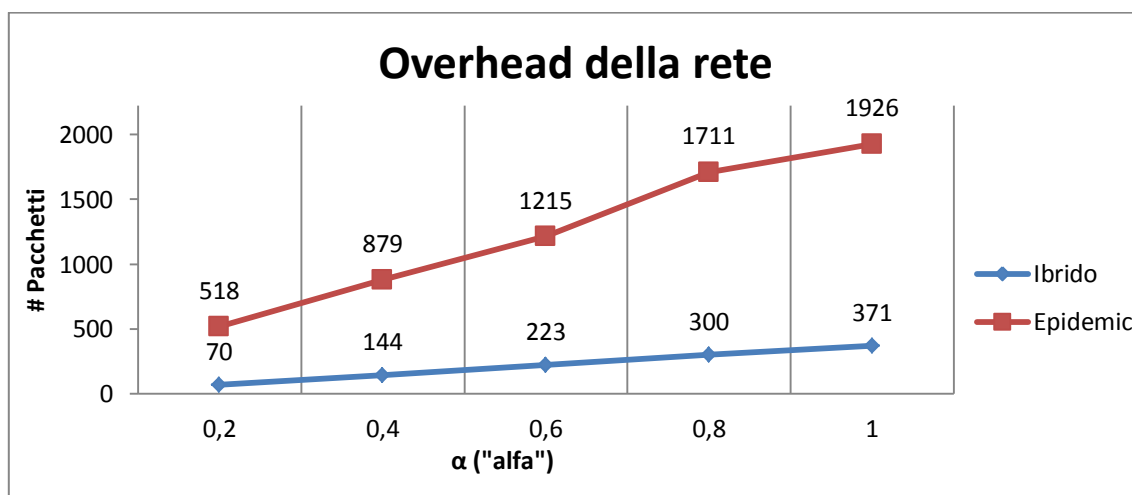


**Grafico 4.99:** mostra la divisione del traffico di tutti i messaggi della rete al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.25$ ,  $Q = 0.75$

Al crescere di  $\alpha$ , si verificano le suddette cose:

- i pacchetti inoltrati rientranti nel Caso I diminuiscono in quanto i nodi “major” ( $30 - 30\alpha$ ) diminuiscono e di conseguenza anche il numero di nodi che risultano avere contemporaneamente sia un numero di ping maggiore sia il destinatario tra la propria lista amici;
- i pacchetti inoltrati rientranti nel Caso II aumentano in quanto al crescere di  $\alpha$  i nodi vengono sparsi per la rete e il numero di messaggi generati ( $30\alpha$ ) aumenta; nonostante il valore di P sia basso (0.25), l’inoltro con la condizione basata solo sul numero dei ping (appunto il Caso II) avviene lo stesso in buona misura;
- i pacchetti inoltrati rientranti nel Caso III aumentano per due motivi: il primo è che il numero dei messaggi da far arrivare a destinazione è maggiore ( $30\alpha$ ); il secondo è che seppur il numero di nodi “major” ( $30 - 30\alpha$ ) diminuisca al crescere di  $\alpha$ , il valore di Q (0.75) è superiore di quello di P (0.25) e perciò la maggior parte dei messaggi viene trasmessa con questa condizione;
- i pacchetti inoltrati rientranti nel Caso IV rimangono sempre 0 in quanto la loro probabilità di instradamento è fissata a 0.

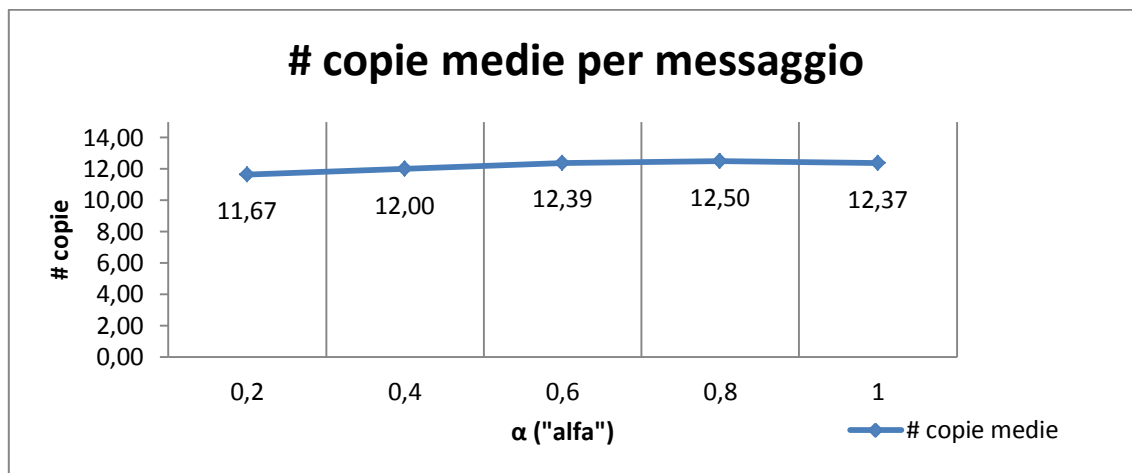
Vediamo i risultati relativi alla terza serie di simulazioni:



**Grafico 4.100:** mostra l’overhead della rete al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.50$ ,  $Q = 0.50$

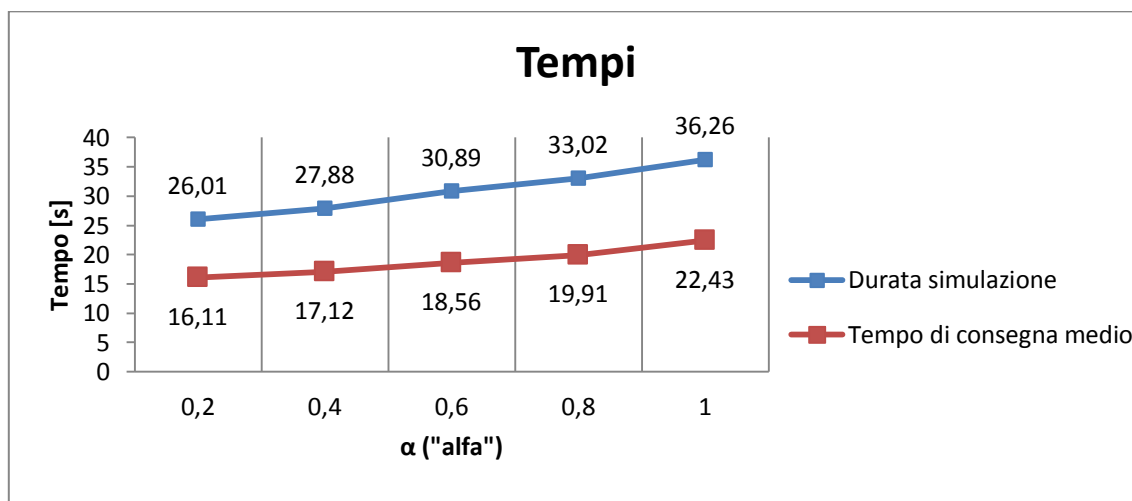
Nel Grafico 4.100 viene mostrato l’overhead di rete; al crescere di  $\alpha$  anche l’overhead aumenta in quanto il numero di messaggi generati ( $30\alpha$ ) è maggiore.

Nel Grafico 4.101 viene invece mostrato il numero di copie medie; al crescere di  $\alpha$  tale valore tende di poco ad aumentare in quanto seppure il numero di “major” ( $30 - 30\alpha$ ) decresce, i messaggi vengono instradati con probabilità  $P (= 0.50)$  se un nodo ha un numero di ping maggiore; dunque, il numero di copie medie tende ugualmente ad aumentare.



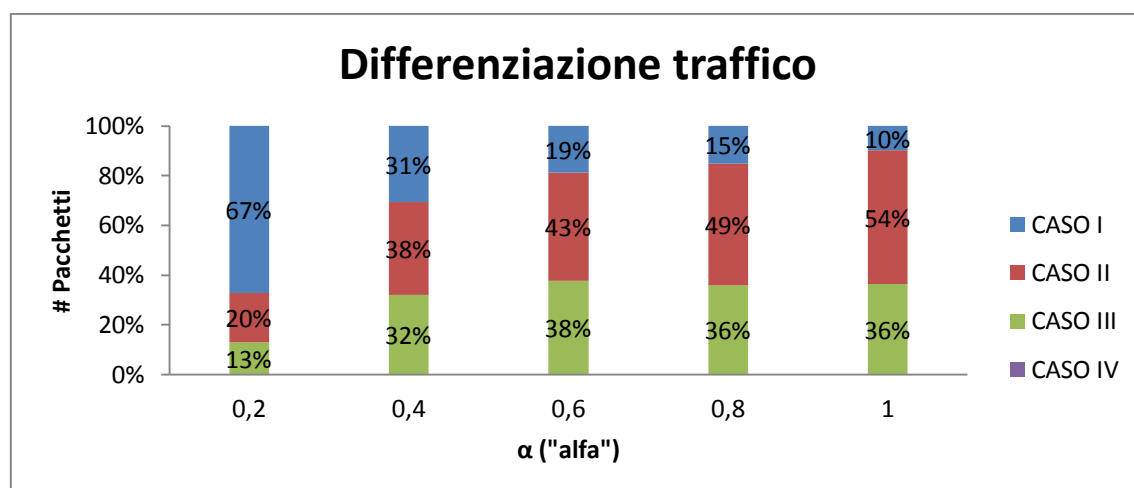
**Grafico 4.101:** mostra il numero di copie medie per messaggio al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.50$ ,  $Q = 0.50$

Nel Grafico 4.102 viene mostrata la durata totale di simulazione e il tempo di consegna medio; al crescere di  $\alpha$  tali valori aumentano in quanto il numero di messaggi da portare a destinazione ( $30\alpha$ ) è maggiore e il numero di “major” ( $30 - 30\alpha$ ) diminuisce al crescere di  $\alpha$ ; la probabilità  $P (= 0.50)$  di inviare il messaggio ad altri nodi se questi hanno un numero di ping maggiore rispetto al possessore è troppo bassa per garantire un tempo di consegna minore:



**Grafico 4.102:** mostra la durata totale della simulazione a partire dall’invio dei messaggi e il tempo di consegna medio di ogni messaggio al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.50$ ,  $Q = 0.50$

Nel Grafico 4.103 si può notare la suddivisione dei pacchetti inoltrati nella rete:



**Grafico 4.103:** mostra la divisione del traffico di tutti i messaggi della rete al variare del parametro  $\alpha$  con raggio di copertura fisso a 30 [metri],  $P = 0.50$ ,  $Q = 0.50$

Al crescere di  $\alpha$ , si verificano le suddette cose:

- i pacchetti inoltrati rientranti nel Caso I diminuiscono in quanto i nodi “major” ( $30 - 30\alpha$ ) diminuiscono e di conseguenza anche il numero di nodi che risultano avere contemporaneamente sia un numero di ping sia un numero di Check-In maggiori;
- i pacchetti inoltrati rientranti nel Caso II aumentano in quanto al crescere di  $\alpha$  i nodi vengono sparsi per la rete e il numero di messaggi generati ( $30\alpha$ ) aumenta;
- i pacchetti inoltrati rientranti nel Caso III aumentano in quanto il numero dei messaggi da far arrivare a destinazione è maggiore ( $30\alpha$ );
- i pacchetti inoltrati rientranti nel Caso IV rimangono sempre 0 in quanto la loro probabilità di instradamento è fissata a 0.

Ciononostante, i messaggi inoltrati mediante il Caso II saranno sempre maggiori di quelli inoltrati mediante il Caso III in quanto il numero di “major” ( $30 - 30\alpha$ ) diminuisce al crescere di  $\alpha$  e l’aumento di pacchetti da portare a destinazione ( $30\alpha$ ) non colma questo gap tra i due Casi.

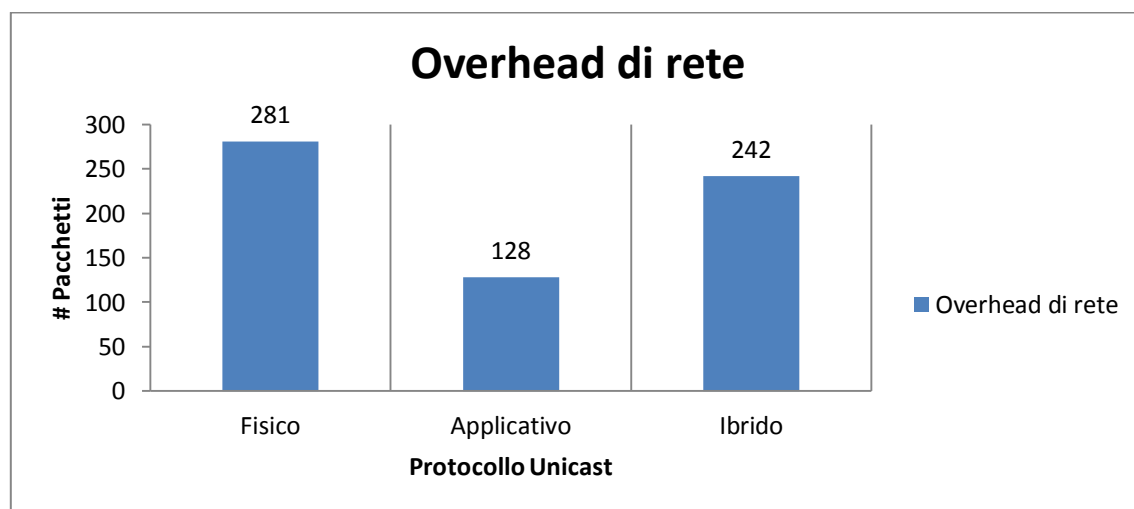
#### 4.6 - Confronto protocolli

In questi 3 sotto-Capitoli verranno messi a confronto fra di loro i protocolli Unicast, Multicast e Geocast in termini di:

- overhead di rete
- durata totale di simulazione

#### 4.6.1 - Protocolli Unicast

Confronteremo dunque i protocolli Fisico Unicast, Applicativo Unicast e Ibrido Unicast (con  $P = 0.50$ ,  $Q = 0.50$ ); come mostrato nel Grafico 4.104 è possibile vedere l'overhead di rete (in termini di soli messaggi) nel caso dei 3 protocolli; l'overhead è relativo al caso in cui ogni nodo si muove linearmente ed invia un messaggio:



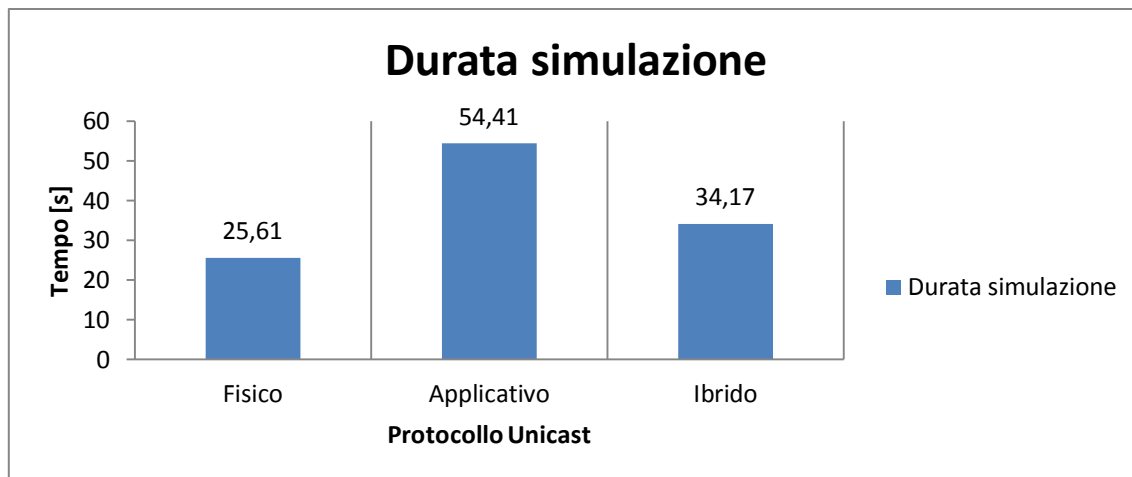
**Grafico 4.104:** mostra l'overhead di rete nel caso dei 3 protocolli Unicast dove ogni nodo invia un messaggio

Come si può notare, nel caso Applicativo Unicast l'overhead di rete è più basso in quanto si prende in considerazione il caso  $\alpha = 1$  che prevede che ogni nodo è libero di muoversi linearmente nella rete con vincoli di amicizia random. Proprio per questo motivo i nodi invieranno meno messaggi; il protocollo Ibrido ha un overhead minore rispetto al Fisico in quanto la probabilità di inoltrare un messaggio secondo la condizione del numero dei Ping ( $P$ ) è uguale a 0.50, così come quella relativa al numero degli amici ( $Q = 0.50$ ).

Si deve inoltre considerare un fattore importante: nel caso Fisico e Ibrido, oltre all'invio dei messaggi, è necessario inviare anche i ping ad una frequenza fissa di  $1 \left[ \frac{ping}{sec} \right]$ ; dunque l'overhead totale sarà molto più alto nei due protocolli che utilizzano i ping rispetto a quello Applicativo in quanto è necessario inviare altri "pacchetti" (ping) oltre i messaggi; quindi i nodi dovranno avere una capacità abbastanza alta per evitare il

satramento della rete oppure si potrebbe perdere in accuratezza (diminuendo la frequenza di invio dei ping) ma aumentando il tempo di simulazione.

Nel Grafico 4.105 viene mostrata la durata di simulazione relativa ai 3 protocolli:



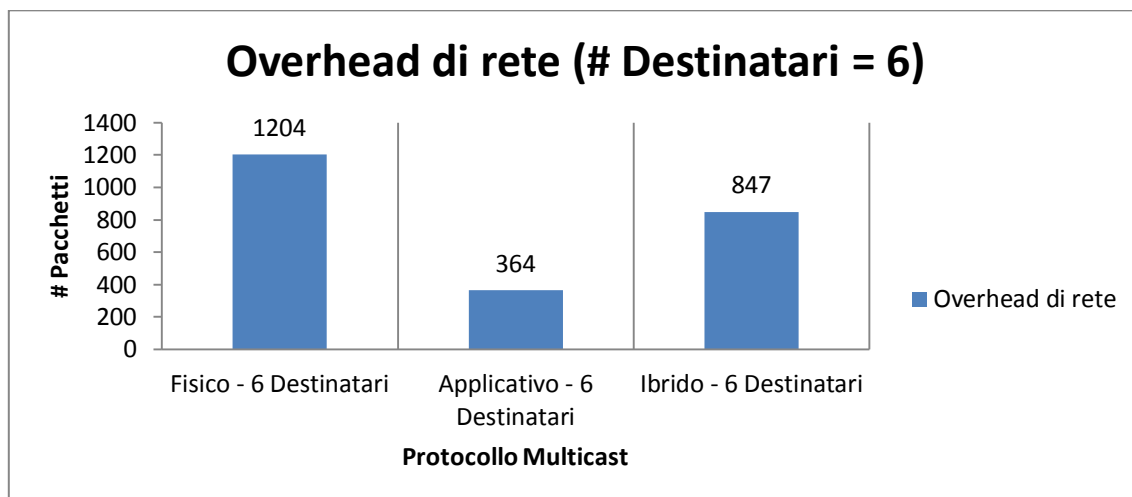
**Grafico 4.105:** mostra la durata di simulazione nel caso dei 3 protocolli Unicast dove ogni nodo invia un messaggio

Nel caso del protocollo Applicativo la durata di simulazione è molto elevata in quanto si è considerato il caso di  $\alpha = 1$  in cui ogni nodo è libero di muoversi e dove il vincolo di amicizia è random; è lecito aspettarsi dunque un tempo di simulazione maggiore rispetto agli altri due casi. Il caso migliore è il protocollo Fisico che termina in un tempo pari a 25.61 [s]; quello Ibrido invece termina quasi in un tempo intermedio tra i due in quanto i pacchetti vengono inviati con probabilità  $P = 0.50$  se un nodo ha un numero di ping maggiore rispetto a un altro e con probabilità  $Q = 0.50$  se un nodo ha il destinatario tra la propria lista amici; è dunque corretto aspettarsi un tempo di simulazione maggiore rispetto al caso Fisico in quanto si “pesano” allo stesso modo due condizioni, ping e lista amici, con la prima nettamente migliore della seconda per  $\alpha = 1$ .

Si deve però aggiungere un fattore molto importante: nel caso Fisico e nel caso Ibrido, non si è tenuto conto nel Grafico 4.105 del tempo di inizializzazione dovuto all’invio dei soli ping pari a 8 [s]; dunque, mentre il caso Applicativo non ha bisogno assolutamente di tale periodo in quanto le informazioni riguardo agli amici sono intrinseche nei nodi, negli altri due casi è strettamente necessario in quanto c’è bisogno di una prima distinzione tra i nodi della rete.

#### 4.6.1 - Protocolli Multicast

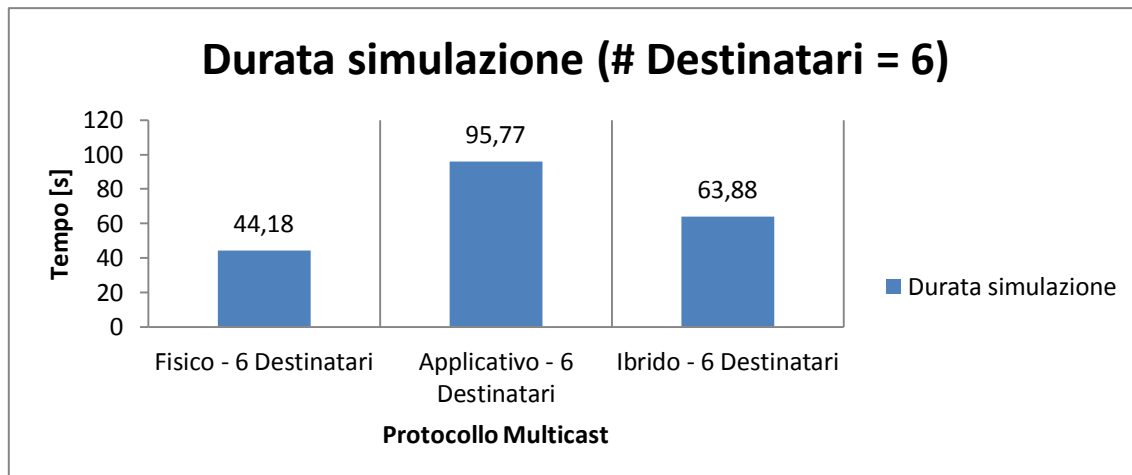
Confronteremo dunque i protocolli Fisico Multicast, Applicativo Multicast e Ibrido Multicast (con  $P = 0.50$  ,  $Q = 0.50$ ); come mostrato nel Grafico 4.106 (caso a 6 Destinatari) è possibile vedere l'overhead di rete (in termini di soli messaggi) nel caso dei 3 protocolli; l'overhead è relativo al caso in cui ogni nodo si muove linearmente ed invia un messaggio:



**Grafico 4.106:** mostra l'overhead di rete nel caso dei 3 protocolli Multicast dove ogni nodo invia un messaggio con 6 Destinatari diversi

Per tutti e tre i protocolli, all'aumentare del numero dei destinatari l'overhead sale in quanto più nodi dovranno essere raggiunti da ogni messaggio; il caso migliore in termini di overhead è il protocollo Applicativo in quanto ha un numero di messaggi abbastanza basso rispetto agli altri due; il motivo di ciò è dovuto al fatto che viene considerato il valore di  $\alpha = 1$  dove ogni nodo ha una lista di 10 nodi "amici" random sui 30 nodi totali. Il caso peggiore è quello Fisico con i messaggi inoltrati a seconda del numero di ping; quello Ibrido invece rappresenta una via di mezzo per entrambi con le probabilità di inoltro dei messaggi uguali a 0.50; anche in questo grafico non si è tenuto conto, in termini di overhead, del numero di ping inviati nella rete; infatti il protocollo Fisico e quello Ibrido si scambiano ping durante il corso della simulazione ad una frequenza di  $1 \left[ \frac{ping}{sec} \right]$  e tale valore infatti potrebbe pregiudicare le condizioni della rete in quanto quest'ultima si potrebbe saturare a causa dell'eccessivo traffico. Quindi, oltre al numero dei messaggi, si deve considerare anche il numero dei ping che aumentano di conseguenza l'overhead totale.

Nel Grafico 4.107 (caso a 6 Destinatari) viene mostrata la durata di simulazione per i 3 protocolli:



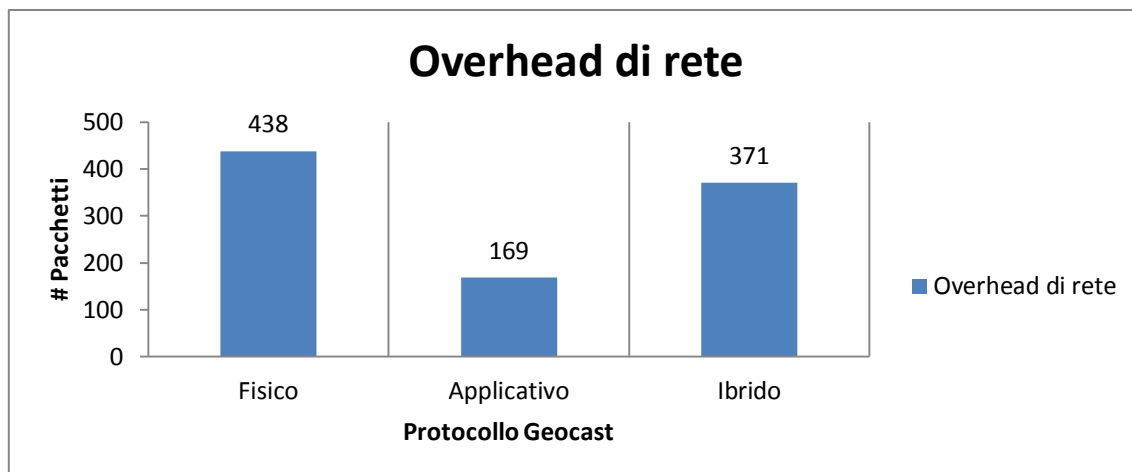
**Grafico 4.107:** mostra la durata di simulazione nel caso dei 3 protocolli Multicast dove ogni nodo invia un messaggio con 6 Destinatari diversi

Si può notare, per ogni protocollo, che se il numero di destinatari è maggiore allora che il tempo di simulazione sarà più lungo; il caso migliore lo si ha con il protocollo fisico dove l'overhead è appunto maggiore (visto nel Grafico 4.106) e i messaggi arrivano prima a destinazione; il caso peggiore è quello applicativo in quanto il valore di  $\alpha$  è stato preso pari a 1 (amici scelti in modo random); il prezzo che si paga ad avere un overhead minore è appunto nella durata di simulazione più lunga; infine il protocollo Ibrido è quasi una via di mezzo tra i due, in quanto i messaggi vengono inoltrati con probabilità  $P (= 0.50)$  se il numero di ping è maggiore e con probabilità  $Q (= 0.50)$  se il destinatario è presente nella lista amici. Tuttavia in questo grafico, non è stato considerato il tempo di inizializzazione, pari a 8 [s], necessario ai protocolli Fisico e Ibrido per inviare solamente i ping; dunque mentre per il protocollo Applicativo i messaggi possono partire immediatamente, per il protocollo Fisico e quello Ibrido si deve aspettare un tempo pari a 8 [s].

#### 4.6.1 - Protocolli Geocast

Confronteremo dunque i protocolli Fisico Geocast, Applicativo Geocast e Ibrido Geocast (con  $P = 0.50$ ,  $Q = 0.50$ ); come mostrato nel Grafico 4.108 è possibile vedere l'overhead di rete (in termini di soli messaggi) nel caso dei 3 protocolli; l'overhead è relativo al caso in cui ogni nodo si muove linearmente ed invia un messaggio:

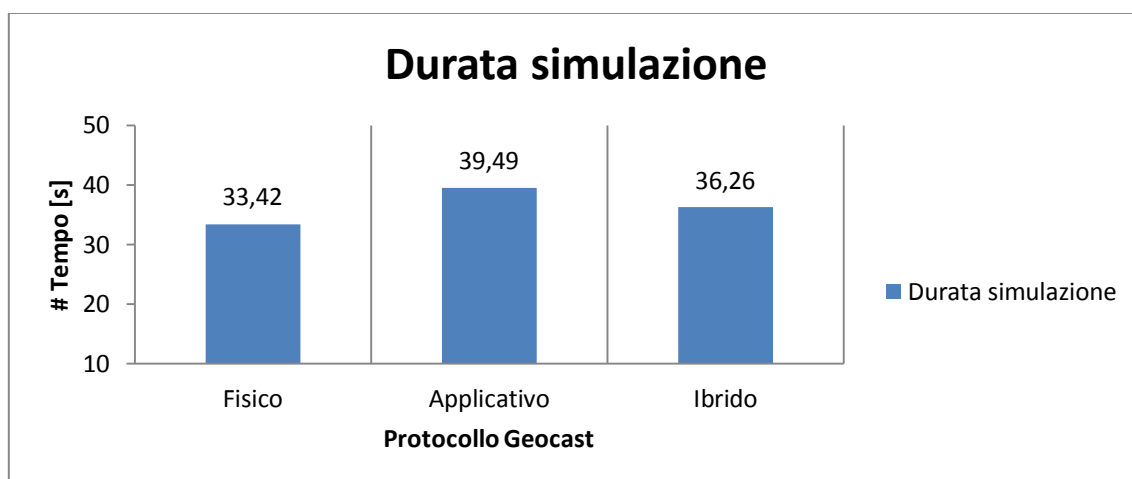




**Grafico 4.108:** mostra l'overhead di rete nel caso dei 3 protocolli Geocast dove ogni nodo invia un messaggio

Anche in questo caso, il caso migliore in termini di overhead lo si ha con il protocollo applicativo che invia i messaggi in base al numero di Check-In; il protocollo fisico invece ha l'overhead più alto, mentre con il protocollo ibrido si ha una via di mezzo tra i primi due; i protocolli fisico e ibrido, oltre all'overhead dei messaggi hanno l'overhead dovuto allo scambio dei ping ad una frequenza di  $1 \left[ \frac{ping}{sec} \right]$ ; si potrebbe diminuire la frequenza dei ping al fine di saturare meno la rete perdendo però in accuratezza in termini di tempo.

Per la durata di simulazione si osservi il Grafico 4.109:



**Grafico 4.109:** mostra la durata di simulazione nel caso dei 3 protocolli Geocast dove ogni nodo invia un messaggio

Il caso migliore lo si ha con il protocollo fisico in quanto vengono instradati un maggior numero di pacchetti che arrivano prima a destinazione; il caso peggiore è quello applicativo anche se non è molto peggiorativo in quanto, anche per  $\alpha = 1$ , l'informazione riguardante il numero di Check-In non è random come per la lista amici

(caso Unicast e Multicast) e dunque si ottengono ottime prestazioni considerando anche il basso overhead; infine, il protocollo ibrido è una sintesi dei due protocolli (fisico e applicativo) e anche in termini di tempo è una via di mezzo; nel grafico sopra non vengono considerati i tempi di inizializzazione della rete, ovvero:

- un tempo pari a 8 [s] dove vengono inviati solo i ping ogni secondo per il protocollo fisico e per quello ibrido;
- un tempo pari a 12 minuti dove tutti i nodi effettuano il Check-In ogni 5 secondi per il protocollo applicativo e per quello ibrido;

## Conclusioni

In questo elaborato sono state brevemente descritte le caratteristiche tipiche delle Delay Tolerant Networks (DTN); sono stati presentati i vari parametri che caratterizzano tali reti come il ritardo medio e l'overhead di rete; inoltre è stato analizzato il problema dell'instradamento.

A tal proposito sono state fatte diverse proposte al fine di avere delle comunicazioni efficienti all'interno di queste reti; considerando una DTN di dimensioni pari a quelle di un campus universitario, sono state proposte diverse metriche di routing riguardanti le comunicazioni Unicast, Multicast e Geocast; tali metriche sono basate o sui contatti fisici (Protocollo Fisico) o sulle relazioni sociali come gli amici di Facebook o i Check-In di Foursquare (Protocollo Applicativo) o su entrambe (Protocollo Ibrido).

I meccanismi di instradamento proposti sono stati valutati tramite diverse simulazioni usando il simulatore a eventi discreti OmNet++.

In ogni simulazione è stato fatto variare almeno un parametro come ad esempio la frequenza d'invio dei ping, il raggio di copertura, il numero dei destinatari, il numero di amici, etc.

Si è notato per il protocollo Fisico un overhead maggiore rispetto a quello Applicativo e a quello Ibrido; di conseguenza anche il numero di copie medie per messaggio sarà maggiore in tale protocollo rispetto agli altri due; d'altra parte il tempo di simulazione e quello di consegna media sono minori nel protocollo Fisico in quanto essendoci un maggior numero di pacchetti nella rete, questi arrivano prima ad ogni destinatario. Il protocollo Ibrido, prendendo alcune informazioni sia fisiche sia applicative riesce ad avere prestazioni intermedie tra i due protocolli, avendo un buon compromesso tra tempo di consegna e overhead totale della rete.

Gli sviluppi futuri per tali reti tolleranti ai ritardi potrebbero essere:

- estendere le metriche appena descritte a scenari più ampi quali ad esempio una città intera;
- considerare che alcuni nodi possano essere anche non cooperativi, ovvero non desiderino trasportare messaggi anche per conto di altri;

- considerare che alcuni messaggi possano avere una priorità di consegna maggiore rispetto ad altri e che dunque debbano essere consegnati per primi;
- considerare che i nodi, ovvero i dispositivi mobili quali smartphone, hanno una capacità di banda limitata e che dunque i tempi di consegna si allungano; inoltre è possibile che alcuni pacchetti possano essere eliminati in quanto il buffer in quel preciso istante sia pieno (capacità di memoria limitata);

## Bibliografia

- [1] John Burgess, Brian Gallagher, David Jensen, Brian Neil Levine, “ *MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networks* ”, INFOCOM 2006 25th IEEE International Conference on Computer Communications. Proceedings, Aprile 2006, Pag 1-11
- [2] Robert Nagel, Stephan Eichler, “ *Efficient and Realistic Mobility and Channel Modeling for VANET Scenarios using OMNeT++ and INET-Framework* ”, Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops, 2008, Pag 1-8
- [3] Ian Katz, “ *A Delay-Tolerant Networking Framework for Mobile Underwater Acoustic Networks* ”, Proc. of the Fifteenth International Symposium on Unmanned Untethered Submersible Technology, Agosto 2007, Pag 1-6
- [4] Ian F. Akyildiz, Dario Pompili, Tommaso Melodia, “ *Underwater acoustic sensor networks: research challenges* ”, Febbraio 2005, Pag 1-23,
- [5] Sushant Jain, Kevin Fall, Rabin Patra, “ *Routing in a Delay Tolerant Network* ”, Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications, 2004, Pag 1-13
- [6] Paolo Costa, Mirco Musolesi, Cecilia Mascolo, Gian Pietro Picco, “ *Socially-aware Routing for Publish-Subscribe in Delay-Tolerant Mobile Ad Hoc Networks* ”, Selected Areas in Communications, IEEE Journal on, Giugno 2008, Pag 1-12
- [7] Elmurod Talipov, Yohan Chon, Hojung Cha, “ *Content Sahring Over Smartphone-based Delay-tolerant Networks* ”, Mobile Computing, IEEE Transactions on, 2011, Pag 1-14
- [8] Philo Juang, Hidekazu Oki, Yong Wang, Margaret Martonosi, Li-Shiuian Peh, Daniel Rubenstein, “ *Energy Efficient Computing for Wildlife Tracking: Design Tradeoffs and Early Experiences with ZebraNet* ”, Proceedings of the 10th international conference on Architectural support for programming languages and operating systems, 2002, Pag 1-12
- [9] Anders Lindgren, Avri Doria, Olov Schelén, “ *Probabilistic Routing in Intermittently Connected Networks* ”, ACM SIGMOBILE Mobile Computing and Communications Review , 2003, Pag 1-8

## *Ringraziamenti*

*Riuscire a raggiungere questo traguardo mi rende una persona molto felice e consapevole dei propri mezzi...per arrivare a ciò però devo ringraziare tutte le persone che mi sono state accanto, sopportando tutti i miei momenti difficili e aiutandomi a vivere con serenità quest'avventura durata 5 anni. Inoltre un ringraziamento particolare lo vorrei fare a mia mamma, che più di tutti mi ha spinto ad affrontare quest'avventura con la giusta calma e tranquillità, a mio papà, che ad ogni esame aveva il suo modo di darmi forza dicendomi "auguri", alla mia ragazza, con cui ho affrontato buona parte degli esami e che mi ha aiutato nei momenti di maggiore stress, al mio Relatore, che mi ha seguito sempre e con la massima disponibilità, permettendomi di lavorare a questo elaborato con la giusta serenità e armonia, all'Ing. Filippini Ilario per avermi aiutato nei primi passi con il simulatore, ai miei compagni di corso (e aggiungerei di sventure...) e ai miei amici di sempre .*

*Infine, un ringraziamento a parte lo vorrei fare a Nicola, un ragazzo che è venuto a mancare nella mia vita quattro anni fa... una sera me l'avevi detto che ce l'avrei fatta a raggiungere questa laurea, io ero molto diffidente, ma grazie alle tue parole e ai tuoi insegnamenti sono riuscito a diventare una persona migliore e a centrare questo obiettivo...ora lo posso urlare Nicky: ce l'abbiamo fatta!!!*