
DHT Symphony Simulation

Andrea Aquino, Michele Consiglio Università di Bologna

11 gennaio 2013

Il protocollo DHT Symphony discusso da Manku, Bawa e Raghavan si propone come nuova frontiera per il mantenimento e l'accesso di dizionari distribuiti sulla rete globale. Sfruttando una topologia di rete virtuale ad anello sulla quale vengono instaurate connessioni tra nodi distanti, è possibile ridurre drasticamente i tempi di recupero di dati evitandone l'eccessiva replicazione. Tuttavia il numero di messaggi necessari al funzionamento del protocollo stesso non è oggetto di studio del documento prodotto dagli autori di cui sopra. Si intende quindi analizzare tale aspetto mediante la simulazione del protocollo a mezzo del simulatore Omnet++. Verrà inoltre introdotta una modifica protocollare con lo scopo di diminuire il numero di messaggi transitanti nella rete senza inficiare la correttezza e l'efficacia del protocollo.

1 Simulazione

1.1 Gli strumenti

Lo studio di simulazione è stato realizzato a mezzo del simulatore Omnet++. Nello specifico il protocollo oggetto dello studio è stato emulato mediante l'implementazione dei moduli stanti alla lista seguente:

- File NED descrivente la topologia di rete virtuale;
- Classe rappresentante i singoli nodi all'interno della rete;
- Pacchetti di rete strutturati per la trasmissione delle informazioni protocolli rilevanti;
- File di configurazione per gestire le informazioni parametriche della simulazione.

Sono stati inoltre utilizzati gli strumenti di analisi offline messi a disposizione da tale simulatore per l'analisi prestazionale del protocollo nella versione originale e soggetto a modifica. I risultati di tale studio verranno discussi nella sezione "Risultati" del presente elaborato.

1.2 Il livello di dettaglio

L'implementazione in esame consiste a tutti gli effetti in una emulazione puntuale del protocollo DHT Symphony allo scopo di analizzare con assoluta precisione le prestazioni dello stesso in termini di numero di messaggi scambiati tra i nodi della rete al passare del tempo. La simulazione di rete astrae dalla topologia di rete reale realizzando soltanto le connessioni virtuali previste dal protocollo medesimo. In particolare i nodi sono disposti su di un ipotetico cerchio di perimetro unitario che costituisce una rete ad anello. Nella fase di inizializzazione del protocollo ogni nodo è quindi connesso ai suoi due immediati vicini ed è dotato di un numero K (configurabile) di connessioni "lunghe" con i nodi i cui id differiscono dal proprio tra le 8 e le $8+K$ unità.

Per semplicità le connessioni con i nodi vicini (da qui in poi connessioni brevi) sono state realizzate mediante gate bidirezionali facenti parte di un vettore di gate contenente la totalità delle connessioni per il nodo corrente. In fase di sperimentazione il numero di nodi della rete è stato fissato a 16, 32 e 64, di cui la metà connessi nella struttura ad anello discussa in precedenza e la rimanente metà in principio non connessa.

Infine sono stati individuati e simulati fedelmente i seguenti aspetti del protocollo:

1. stima della lunghezza del segmento gestito da un nodo;
2. stima del numero di nodi della rete;

3. protocollo di routing per individuare il manager di un punto qualsiasi sull'anello di perimetro unitario;
4. protocollo di relink;
5. protocollo di join;

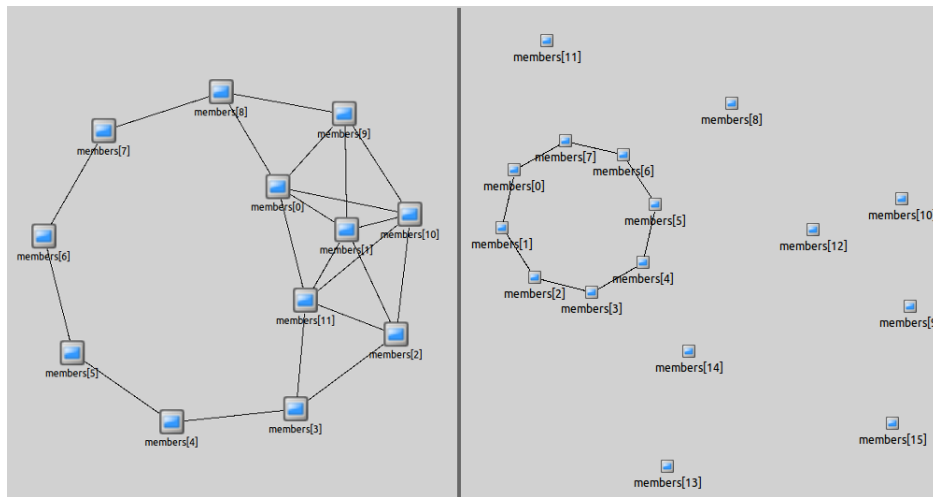


Figura 1: *Topologie di reti virtuali*

1.3 Assunzioni

Il documento “Symphony: Distributed Hashing In A Small World” non specifica una variegata serie di dettagli. Pertanto è stato necessario effettuare tutta una serie di assunzioni al fine di poter implementare il protocollo stesso.

1. La rete è supposta essere, in principio, popolata;
2. Il numero di nodi della DHT è inferiormente limitato da un valore positivo configurabile, supposto maggiore di 8;
3. Ogni nodo non ha conoscenza della posizione all'interno della circonferenza di perimetro unitario dei propri vicini;
4. Il lookup dei file della DHT non è ritenuto di interesse e pertanto non è simulato;
5. Nessuna coppia di nodi facente parte della DHT ha la stessa posizione sulla circonferenza di perimetro unitario.

2 Modifica protocollare

L'implementazione originale del protocollo di routing prevede, in virtù del punto 3. della sezione “Assunzioni”, la necessità di effettuare un broadcast ai nodi vicini per poter definire quali di questi dista meno dal nodo destinazione e quindi instradare le successive richieste attraverso di esso. In fase di simulazione suddetto dettaglio sembra inficiare duramente la qualità del protocollo in termini prestazionali (quantità di messaggi trasmessi). Quindi, sotto l'assunzione che ogni nodo mantenga traccia della posizione dei suoi vicini sulla circonferenza di perimetro unitario, è stata effettuata una modifica protocollare che snellisce sostanzialmente il protocollo di routing (massivamente sfruttato anche da parte dei protocolli di join e di relink) sotto tale punto di vista.

2.1 Implementazione

In teoria sarebbe stato necessario tenere traccia per ogni nodo della posizione di ogni altro nodo ad esso connesso mediante un vettore locale. Si noti che la posizione dei vicini di un dato nodo può variare solamente a seguito dell'accesso o dell'uscita di un nodo dalla rete; si ritiene che tale evento occorra assai più raramente rispetto al protocollo di routing (che è cuore del meccanismo di inserimento e ricerca di chiavi nel dizionario distribuito). Per semplicità, tale vettore è stato assunto esistere e la posizione dei nodi vicini al nodo corrente è stata recuperata mediante le direttive del simulatore Omnet++ stesso.

2.2 Correttezza

La modifica protocollare precedentemente discussa non modifica in alcun modo la semantica del protocollo. Segue una dimostrazione informale di tale evidenza:

1. la modifica in esame riguarda esclusivamente il protocollo di routing; se la semantica di tale protocollo rimane la medesima, di conseguenza la semantica del protocollo complessivo non varia.
2. per dimostrare che la semantica del protocollo di routing non cambia è sufficiente dimostrare che in ogni caso in cui la versione originale del protocollo effettua una scelta tra i nodi vicini per l'instradamento di una richiesta, il protocollo modificato effettua la medesima scelta.
3. dimostriamo il punto 2. per induzione sul numero di connessioni lunghe del nodo corrente.

- **caso base:** il nodo corrente ha solo le connessioni brevi con i nodi che lo seguono immediatamente in senso orario e antiorario. Poiché sono supposti esistere almeno 3 nodi all'interno della rete simulata, deve esistere una relazione d'ordine totale tra le posizioni di questi sulla circonferenza di perimetro unitario. Siano

$$0 \leq a \neq b \neq c < 1$$

tali posizioni e sia

$$0 \leq x < 1$$

il valore di cui si intende individuare il nodo manager per mezzo del protocollo di routing. Ne consegue che

$$|a - x| \neq |b - x| \neq |c - x|$$

e quindi una sola distanza è quella minima. Non essendoci alcuna forma di non determinismo ed essendo la stima di a, b e c uguale per entrambe le versioni del protocollo la scelta di instradamento non subisce variazioni.

- **caso induttivo:** supponiamo che i protocolli si comportino allo stesso modo per un certo numero n di link lunghi. Dimostriamo che continuano a comportarsi allo stesso modo anche per n+1 link lunghi. In virtù del punto 5. della sezione “Assunzioni” il nodo connesso al nodo corrente mediante l'n+1-esimo link lungo ha una posizione diversa da tutti gli altri sulla circonferenza di perimetro unitario. I casi sono dunque due:
 - * il nuovo nodo è il più vicino al nodo destinazione; ma in tal caso lo è per entrambe le versioni del protocollo e quindi i protocolli continuano ad instradare i messaggi mediante lo stesso nodo.
 - * il nuovo nodo è più distante rispetto ad un altro vicino del nodo corrente dal nodo destinazione; ma in tal caso il protocollo si comporta come nel caso in cui l'n+1-esimo link lungo non esista affatto, ma in questa situazione i due protocolli hanno lo stesso comportamento per ipotesi induttiva.

Fino ad ora abbiamo supposto che la posizione dei nodi vicini al nodo che sta correntemente eseguendo il protocollo di routing sia costantemente aggiornata per la versione modificata del protocollo. Ciò è tuttavia certamente vero sotto l'ipotesi che nel momento in cui un nodo accede alla rete o immediatamente prima di abbandonarla comunichi ai propri vicini la necessità di aggiornare le posizioni dei nodi adiacenti.

3 Risultati

É stata effettuata la simulazione a mezzo del simulatore Omnet++ del protocollo Symphony sia nella versione originale che nella versione modificata. Quindi è stato analizzato il parametro prestazionale di interesse (numero di messaggi inviati dai singoli nodi) mediante gli strumenti di analisi offline messi a disposizione da tale simulatore. Di seguito verranno esposti i risultati dell'analisi prestazionale effettuata sul protocollo, impostando il numero di nodi all'interno della rete a 2048 di cui 1024 inizialmente facenti parte della rete virtuale. Nelle figure che seguono sono rappresentati i grafici dei messaggi inviati dai nodi in fase di simulazione. In particolare nell'asse delle ascisse si trova il tempo simulato, mentre in quello delle ordinate il numero di messaggi inviati dal nodo in questione.

In figura 2 e 3 vengono rappresentati rispettivamente il numero di messaggi inviati dal nodo 0 nel protocollo originale e nella sua versione modificata. Come si può facilmente verificare, in questo caso la modifica protocollare apporta una riduzione dei messaggi inviati da parte del nodo 0 del 67% circa.

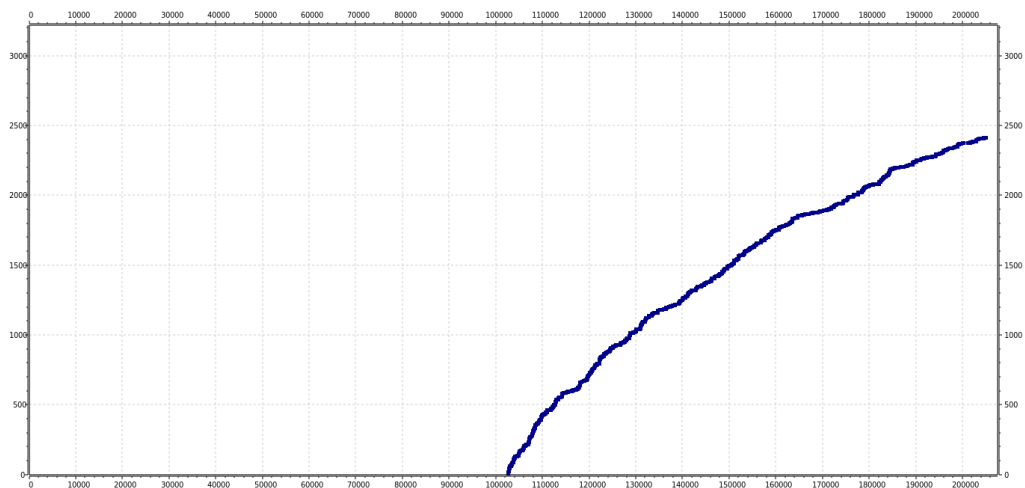


Figura 2: *Messaggi inviati dal nodo 0 col protocollo originale*

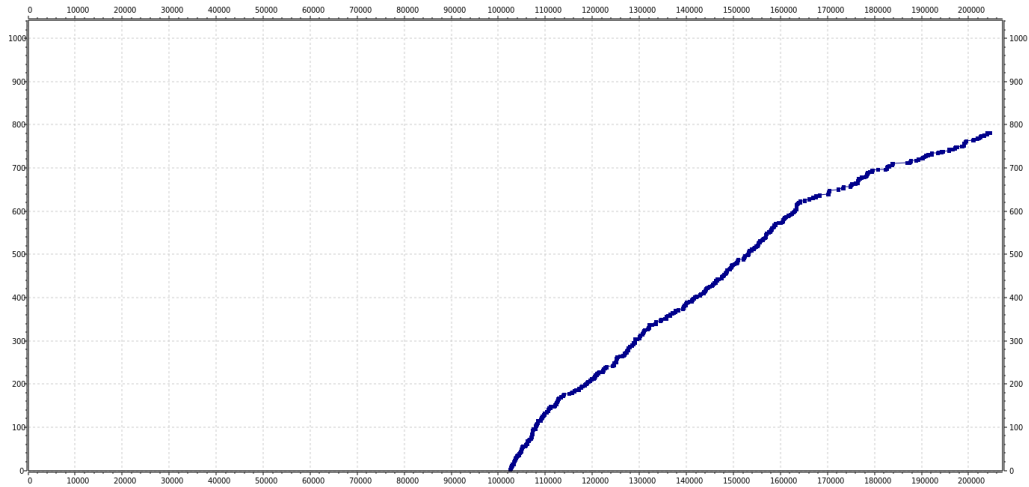


Figura 3: *Messaggi inviati dal nodo 0 col protocollo modificato*

In figura 4 e 5 vengono rappresentati rispettivamente il numero di messaggi inviati dal nodo 3 nel protocollo originale e nella sua versione modificata. In particolare tale nodo è risultato essere quello che invia il maggior numero di messaggi all'interno della simulazione. Come si può facilmente verificare, in questo caso la modifica protocollare apporta una riduzione dei messaggi inviati da parte del nodo 3 del 75% circa.

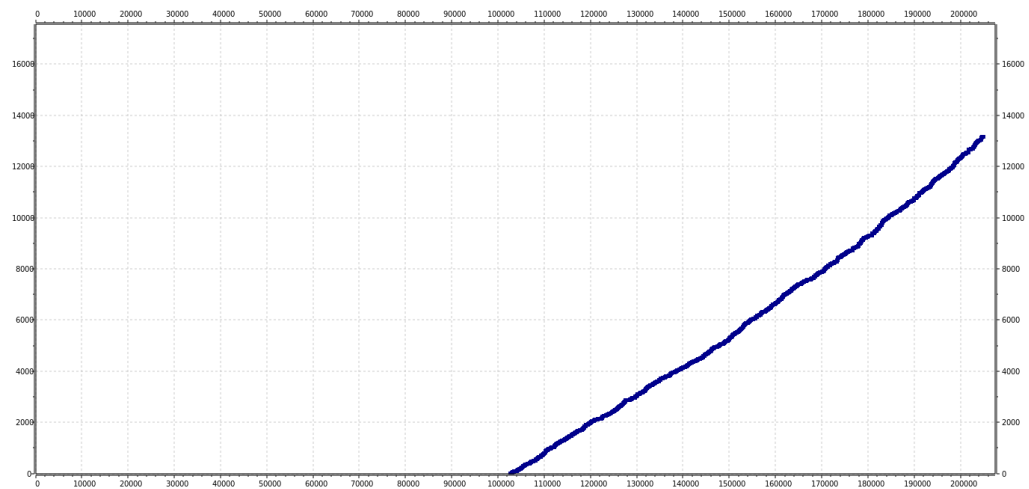


Figura 4: *Messaggi inviati dal nodo 3 col protocollo originale*

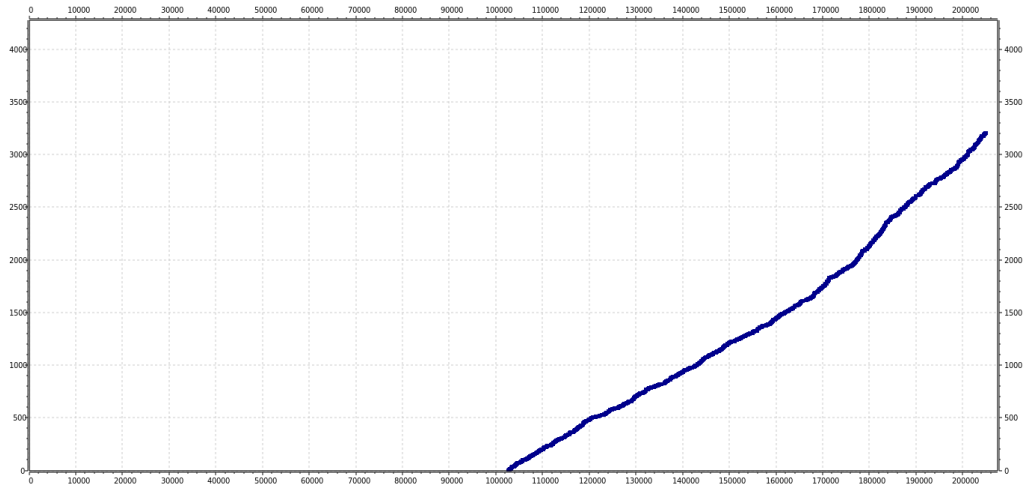


Figura 5: *Messaggi inviati dal nodo 3 col protocollo modificato*

In figura 6 e 7 vengono rappresentati rispettivamente il numero di messaggi inviati dal nodo 1023 nel protocollo originale e nella sua versione modificata. Come si può facilmente verificare, in questo caso la modifica protocollare apporta una riduzione dei messaggi inviati da parte del nodo 1023 del 46% circa.

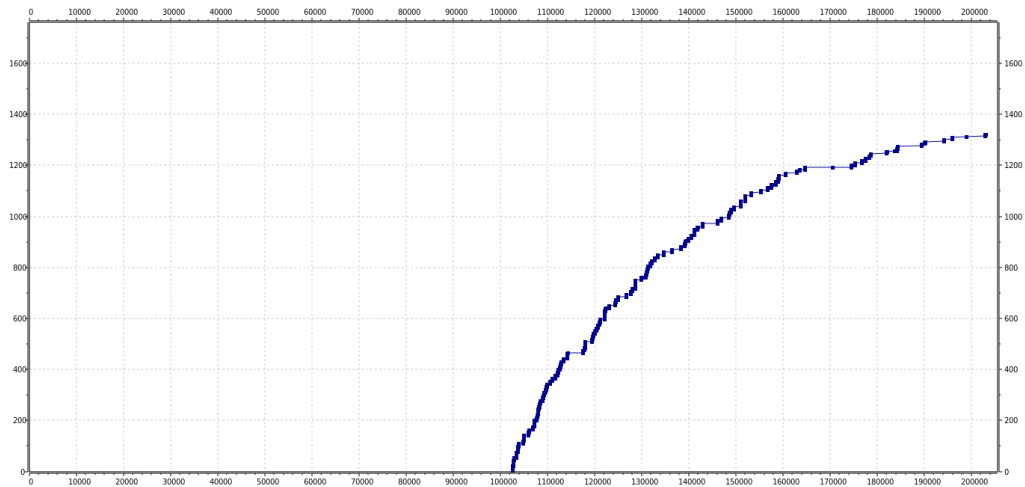


Figura 6: *Messaggi inviati dal nodo 1023 col protocollo originale*

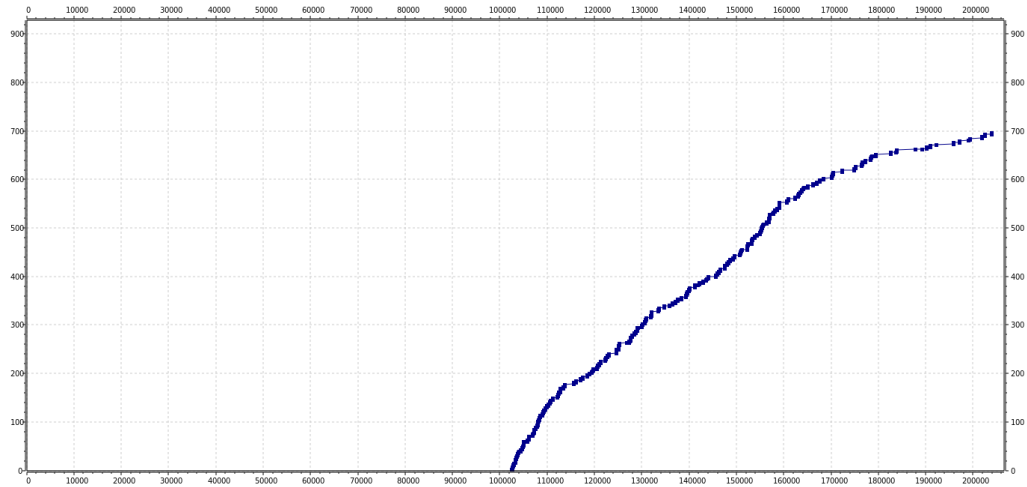


Figura 7: *Messaggi inviati dal nodo 1023 col protocollo modificato*

In figura 8 e 9 vengono rappresentati rispettivamente il numero di messaggi inviati dal nodo 1024 nel protocollo originale e nella sua versione modificata. Come si può facilmente verificare, in questo caso la modifica protocollare apporta una riduzione dei messaggi inviati da parte del nodo 1024 del 27% circa.

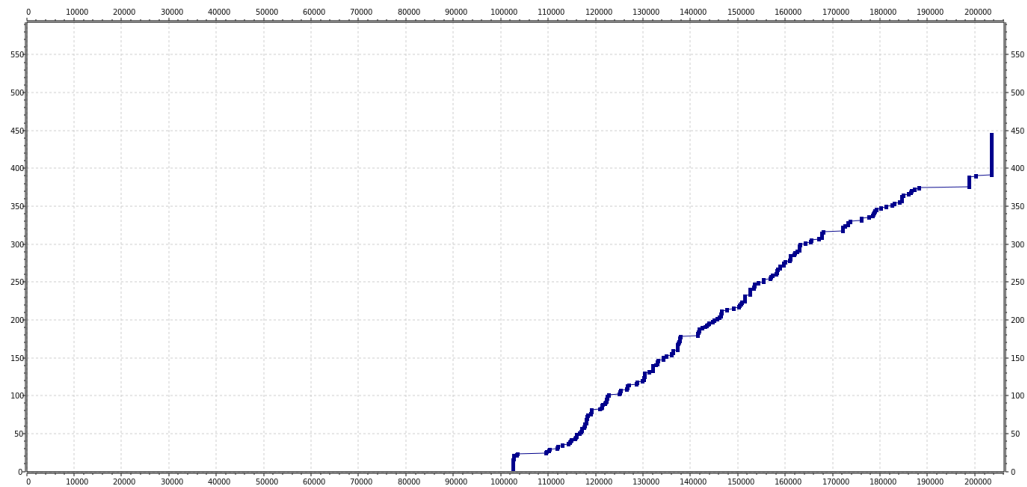


Figura 8: *Messaggi inviati dal nodo 1024 col protocollo originale*

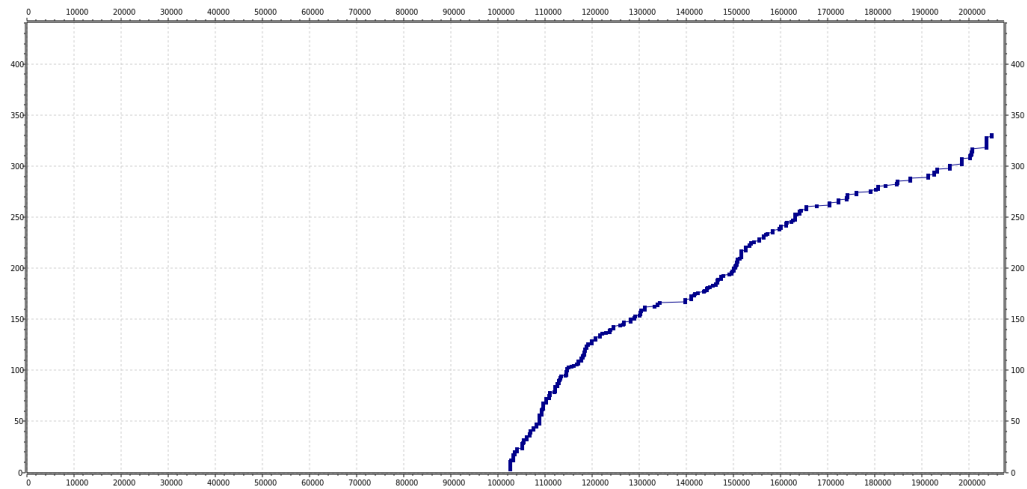


Figura 9: *Messaggi inviati dal nodo 1024 col protocollo modificato*

Come ci aspettavamo, in tutti i casi analizzati la riduzione del numero di messaggi inviati a seguito della modifica protocollare è sostanziale.

4 Bibliografia

1. Gurmeet Singh Manku, Mayank Bawa, and Prabhakar Raghavan. 2003. Symphony: distributed hashing in a small world. In Proceedings of the 4th conference on USENIX Symposium on Internet Technologies and Systems - Volume 4 (USITS'03), Vol. 4. USENIX Association, Berkeley, CA, USA, 10-10
2. Omnet++ Network Simulation Framework - <http://www.omnetpp.org/>