Proposal and Evaluation of a Caching Scheme for Ad Hoc Networks

F.J. González-Cañete, E. Casilari, A. Triviño-Cabrera

Dpto. Tecnología Electrónica
University of Málaga
Málaga, Spain
{fgc, ecasilari, atc}@uma.es

Abstract. In this paper, we describe and evaluate the performance of a caching scheme for ad hoc networks. In this proposal the wireless nodes store the documents they request in a local cache and they can also work as a server for the other wireless nodes if they intercept the forwarding requests and serve the documents requested directly using their local cache. On the other hand, the wireless nodes inspect the requests and responses that they forward in order to learn where and how far the documents are located and they use this information to redirect the requests to other nodes that are closer than the original destination of the request. By means of simulations we evaluate the performance of the proposal and demonstrate that they reduce the latency perceived by the nodes.

Keywords: Ad Hoc networks, cache, replacement policy.

1 Introduction

Mobile Ad Hoc Network (MANET) technology brings the opportunity to extend the coverage area of wireless devices so non-connected nodes can communicate through the collaboration of intermediate devices. Initially, this capability made MANETs especially attractive for disaster or battlefield operations where these networks could work without any infrastructure. However, the success of wireless communications has extended the use of MANETs in commercial applications. In these new scenarios, the users require access to external networks, such as the Internet. For this connection, a gateway that provides access to the Internet and to external servers (E.g. HTTP Server) should be available. Nevertheless, the mobility of the MANET may provoke the Gateway to be temporarily unreachable. Web technologies should adapt to this circumstance to operate properly. In this paper, we study how HTTP traffic can be improved in MANETs when web caching is used.

When using web caching, devices store some documents which were previously requested to an HTTP server in their internal cache. Mobile devices in a MANET can benefit from the storage space of other nodes, so that the documents can be served without accessing the HTTP server. With this operation, HTTP requests are satisfied even when the Internet Gateway is not reachable. Furthermore, the traffic generated to get the document is reduced as an intermediate node in the route to the server serves it. As a first approach of our study, this work deals with multihop wireless networks composed of static devices.

The rest of this paper is structured as follows. In Section II, the caching scheme is described. Section III details the simulation model and the results of the simulations. Finally, Section IV outlines the main conclusion and suggests possible future work.

2 Caching Scheme Proposed

In this section we present an application level caching scheme for ad hoc networks. In this scheme the network nodes request documents that are located in data servers. Due to the limited capabilities of wireless devices, we assume that the data servers are not part of the MANET but they are accessed through the Internet Gateways. The Internet Gateways are fixed nodes in the MANET as specified in [1].

As in the case of HTTP traffic, the simplest way in which a caching scheme can be implemented is allocating a local cache for node. This cache will store the documents requested by the node once they are received from the data server. The next time the node requires the same document it will be served directly from its local cache. This situation is called a cache hit and it drastically reduces the traffic over the ad hoc network, as well as the energy consumption and the latency to receive the documents.

As the proxies in the HTTP traffic the proxy functionalities can be transferred to the wireless nodes in a similar way to the routing procedures. With this additional task, every node in the path of a request from a node to the data server can respond to this request if it has a valid copy of the requested document in its own local cache.

Figure 1 shows an example of an ad hoc network where *DS* is a data server node, that is, the node that physically stores all the documents. This node is accessed through a Gateway (*GW*). Nodes *1*, *2*, *3* and *4* are user nodes that request documents to *DS*. The connections between the nodes indicate the existing wireless links.

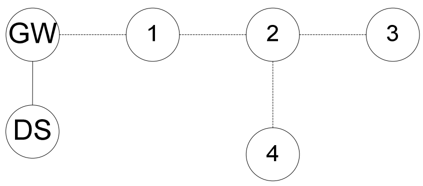


Fig. 1. Example of ad hoc network.

In the case that node 2 requests a document *A*, the request will pass through node *1* to *DS* using an ad hoc routing algorithm. The data server will respond with the document using the path from node *1* to *2*. Finally the node *2* will store the document *A* in its local cache. If node *3* requests the same document *A* to the *DS* the request will reach node *2* that checks if there is a valid copy of the document *A* in its local cache and, if so, it will respond to node *3* with the document. This interception of the request reduces the number of hops from 6 (*3-2-1-DS-1-2-3*) if there is not an interception, to 2 (*3-2-3*) and consequently the latency perceived by node *3*. This mechanism also saves energy in the nodes and reduces the traffic because terminals do not have to forward the requests and responses. In addition, the interception reduces the possible bottlenecks in the data servers because they do not process all the generated requests. The situation when a node in the path of a request to the DS intercepts the request is called an interception hit.

Aiming at reducing the route length to the serving node we make nodes keep information about the distance (measured as the number of hops) where the served documents can be found. This information is dynamically extracted from the forwarded messages (requests and responses).

To illustrate this procedure, let us suppose that node *4* in Figure 1 requests the document *B* to *DS*. The request will pass through nodes *2* and *1* to *DS* so node *2* knows that node *4* will have the document *B* and that node *4* is one hop away. When the *DS* responds with the document *B* through nodes *1* and *2* to node *4*, node *2* will register that *DS* has the document *A* and it is two hops away. If node *3* requests the same document *B* to *DS*, the request will reach node *2*. In that situation node *2* knows that node *4* and *DS* have the document *B* and they are one and two hops away respectively so node *2* will redirect the request to node *4* because the path to the document is shorter. This redirection of the request reduces the number of hops from 6 (*3-2-1-DS-1-2-3*) if there is not redirection, to 4 (*3-2-4-2-3*). Hence the latency perceived by node *3* is diminished.

Unfortunately, the redirection caching has some drawbacks that have to be considered: the mobility and disconnection of nodes and the replacement of the documents.

In order to decrease the number of redirection misses caused by the replacement of the documents stored in the local caches each node calculates the mean time that the documents are stored in its local cache. In that way, when the redirection information of a document is stored the expiration time of this information will be the minimum between the Time To Live (TTL) of the document and the mean time the documents are stored in the local cache.

3 Simulation Parameters and Performance Evaluation

In this section the simulation model and the performance evaluation of a static multihop ad hoc network are presented. In this work we study the performance of the caching scheme presented in the previous section.

The simulations are based on the network simulator NS-2.33 [3] that is one of the most popular simulators for the researches on ad hoc networks [4]. Table 1 summarizes the main simulation parameters.

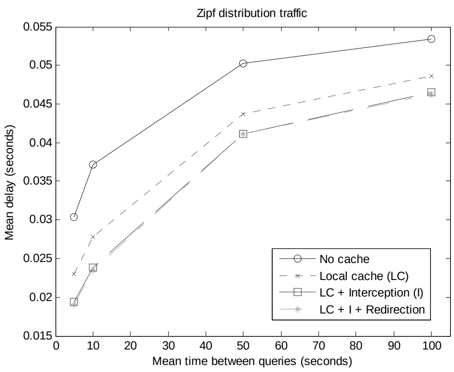
In order to quantify the performance of the network we used the next metrics: the delay (defined as the time elapsed between the request of a document and the reception of the response) and the Hit Ratio (defined as the proportion of documents served by the cache). At each node, the local, interception and redirection hit ratio are defined as the proportion of documents served by the local cache, by an intermediate node and by a node after a redirection respectively.

Table 1. Simulation parameters

Parameter	Default	Values
Simulation area (metres)	1000x1000	
Routing algorithm	AODV [2]	
Number of nodes		5x5 – 7x7 – 9x9 grids
Number of servers	2	
Number of Documents	1000	
Number of requests per node	10000	
TTL (s)	2000	500-1000-2000-4000-5000-Infinite
Mean time between requests (s)	10	5-10-50-100
Zipf slope	0.8	0.4-0.6-0.8-1.0
Replacement policy	LRU	
Cache size (documents)	100	25-50-100-200
Warm-up (requests)	2000	
Simulations per point	5	

The figures only show the results of a 5x5 grid network because the 7x7 and 9x9 networks obtain similar results.

Figure 2 represents the delay and hit ratio as a function of the mean time between requests. The use of local caching reduces the delay perceived by the nodes. As the time between requests is increased the mean delay is also increased because of the expiration of the documents in the local caches. This fact causes the reduction of the local cache hit ratio and hence the amount of documents that have to be requested again to the server is increased. The Interception scheme outperforms the local caching and also reduces the delay. Finally, the Redirection does not outperforms the Interception because the Interception hit ratio is very low for all the mean time between requests and it even reaches nearly zero for a request time of 100 seconds. For high loaded networks (a high request rate) the reduction of the delay if the combined caching scheme is adopted is about 30% compared with the scheme without caching, and this difference remains constant even for low loaded networks (with a low request rate). In very loaded networks with very active the amount of requests served by the local cache or another intermediate cache is about 55%.



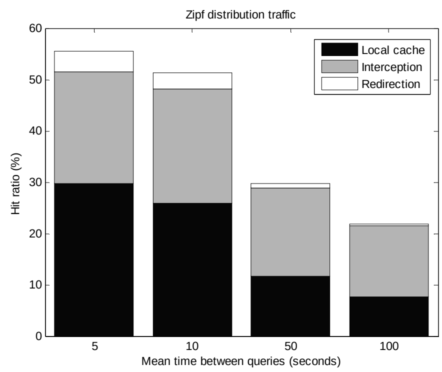


Fig. 2. Delay (left) and Hit ratio (right) as a function of the mean time between requests.

Figure 3 shows how the mean TTL of the documents influences on the delay and the hit ratio. As the TTL of the documents increases the time they can be stored in the caches increases and hence they can be useful during more time. As it can be observed from the figure the delay is reduced asymptotically as the TTL increases until the optimal value where the TTL is infinite, that is, the documents do not expire. In the case of infinite TTL the percentage of hits reaches near 60%. This fact causes the reduction of the delay in about 40% compared to the scheme without caching. In the case of short living documents (low TTL) the reduction of the delay is about 20%. The local caching and interception clearly outperform the schema without caching. The Redirection caching obtains a constant and low hit rate for all TTLs and hence practically does not reduce the delay.

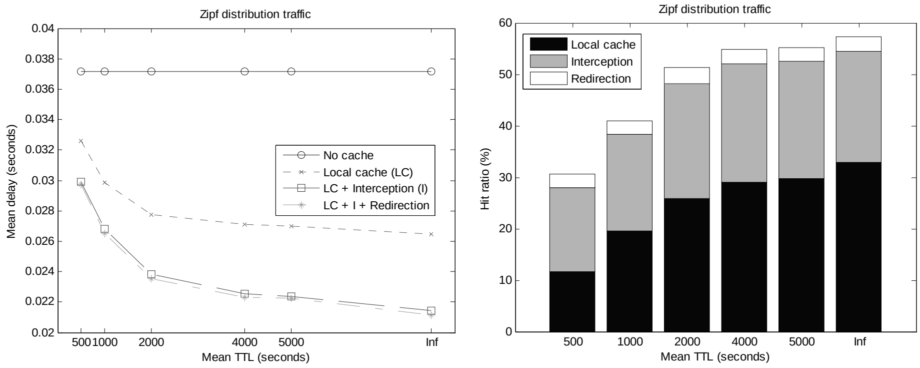


Fig. 3. Delay (left) and Hit ratio (right) as a function of the mean TTL of the documents.

Figure 4 compares the delay and hit ratio as we change the slope of the Zipf distribution of the request pattern. As the slope increases the local cache hit is also increased due to the fact that the most popular documents are frequently requested. On the other hand the interception hits are decreased as the slope increases because most traffic is served by the local caches. Consequently the delay is widely reduced as the Zipf slope increases following a similar behaviour as the previous studies. The delay is reduced about by 20% and 40% for the 0.4 and 1.0 slope respectively.

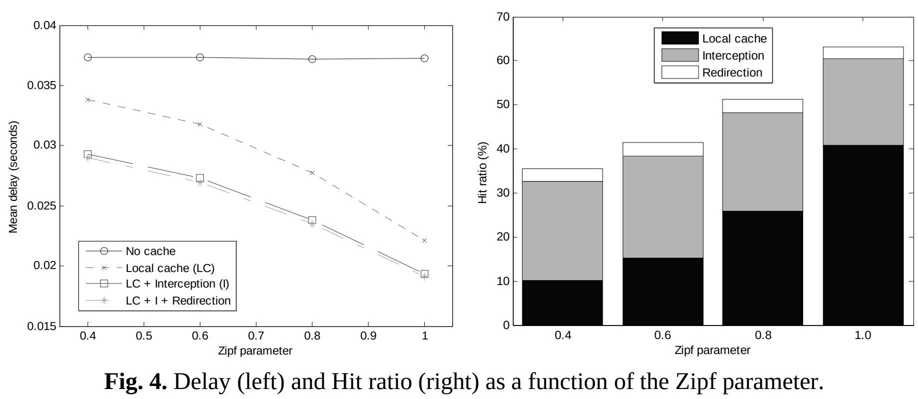


Fig. 4. Delay (left) and Hit ratio (right) as a function of the Zipf parameter.

Finally Figure 5 shows the delay and hit ratio as a function of the cache size. As in the previous scenarios, the Interception of the requests outperforms the schema without caching. The performance improvement reaches its limit when the cache size is 100 Kbytes as the results obtained for the 200 Kbytes cache size are similar. For a cache of 25 documents the improvement obtained with the Redirection caching is close to zero but the redirection hit ratio increases as the cache size increases. For the 25 documents cache the reduction of the delay is about 20% while for largest caches it is reduced by 35%.

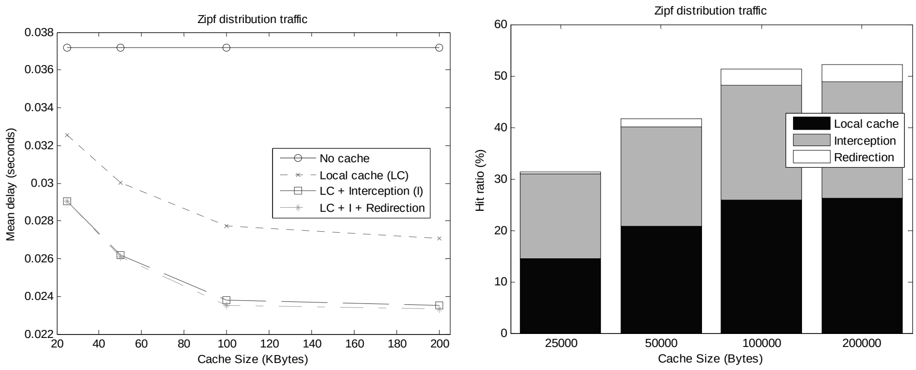


Fig. 5. Delay (left) and Hit ratio (right) as a function of the cache size.

4 Conclusions

In this work we have proposed a caching scheme for ad hoc networks. The scheme suggests implementing a local cache at each node of the ad hoc network in order to intercept or redirect the requests by the intermediate nodes from the source to the destination request. In that way the number of hops is reduced, and hence, the delay perceived by the ad hoc nodes. As the number of hops is decreased, the number of forwarding messages also decreases and the power consumption is reduced. We have studied by mean of simulations the influence of the mean time between requests, the effect of the TTL, the influence of the traffic pattern and the cache size.

We can conclude that the use of local caching combined with the use of the interception reduces drastically the delay perceived by the nodes. The reduction of the delay can vary between 20% and 45% depending on the characteristics of the traffic and cache size. The Redirection of the requests obtains poor hit ratios and hence, the reduction of the delay is not significant compared with the Interception caching.

References

1. Wakikawa, R., Malinen, J.T., Perkins, C.E., Nilsson, A., Tuominen, A.J.: Global Connectivity for IPv6 Mobile Ad Hoc Networks, draft-wakikawa-manet-globalv6-05.txt, Internet Draft, Internet Engineering Task Force (2006)
2. Perkins, C. E., Belding-Royer, E. M., and Das, S.: Ad Hoc On Demand Distance Vector (AODV) Routing. IETF RFC 3561 (2003)
3. NS-2 Home page: http://isi.edu/nsnam/ns/
4. Kurkowski, S., Camp, T., Colagrosso, M.: MANET Simulation Studies: The Incredibles. ACM's Mobile Computing and Communications Review, vol. 9, no. 4, pp. 50-61 (2005)