Performance of Internet Access Solutions in Mobile Ad Hoc Networks

Hamidian, Ali; Körner, Ulf; Nilsson Plymoth, Anders

Published in:
[Host publication title missing]

DOI:
10.1007/b107131

2004

Performance of Internet Access Solutions in Mobile Ad Hoc Networks

Ali Hamidian, Ulf Körner and Anders Nilsson

Department of Communication Systems
Lund University, Sweden
Box 118, 221 00 Lund
{alexh, ulfk, andersn}@telecom.lth.se

Abstract. Although an autonomous mobile ad hoc network (MANET) is useful in many scenarios, a MANET connected to the Internet is more desirable. This interconnection is achieved by using gateways, which act as bridges between a MANET and the Internet. Before a mobile node can communicate with an Internet host it needs to find a route to a gateway. Thus, a gateway discovery mechanism is required. In this paper the MANET routing protocol Ad hoc On-Demand Distance Vector (AODV) is extended to achieve the interconnection between a MANET and the Internet. Moreover, the paper investigates and compares three approaches for gateway discovery. The question of whether the configuration phase with the gateway should be initiated by the gateway, by the mobile node or by mixing these two approaches is being discussed. We have implemented and simulated these three methods and we discuss the advantages and disadvantages of the three alternatives.

1 Introduction

A mobile ad hoc network (MANET) is an autonomous network that can be formed without need of any established infrastructure or centralized administration. It normally consists of mobile nodes, equipped with a wireless interface, that communicate with each other. Because these kinds of networks are very spontaneous and self-organizing, they are expected to be very useful. It is also highly likely that a user of the network will have the need to connect to the Internet.

The Internet Engineering Task Force (IETF) has proposed several routing protocols for MANETs, such as Ad hoc On-Demand Distance Vector (AODV) [1], Dynamic Source Routing (DSR) [2], Optimized Link State Routing Protocol (OLSR) [3] and Topology Dissemination Based on Reverse-Path Forwarding (TBRPF) [4]. However, these protocols were designed for communication within an autonomous MANET, so a routing protocol needs to be modified in order to achieve routing between a mobile device in a MANET and a host device in a wired network (e.g. the Internet). To achieve this network interconnection, gateways that understand not only the IP suite, but also the MANET protocol stack, are needed. Thus, a gateway acts as a bridge between a MANET and the Internet and all communication between the two networks must pass through any of the gateways.

The AODV routing protocol is one of the most developed and implemented routing protocols investigated by the IETF MANET working group. In this work AODV has been modified to achieve routing of packets towards a wired network [5]. Although AODV was used in this study, our approach can be applied to any reactive MANET routing protocol and with some modifications to proactive MANET routing protocols as well.

This paper evaluates three approaches for gateway discovery. An interesting question is whether the configuration phase with the gateway should be initiated by the gateway (proactive method), by the mobile node (reactive method) or by mixing these two approaches. We have implemented these three methods in Network Simulator 2 (ns-2) [6] and compare them by means of simulation. We also discuss the advantages and disadvantages of the three alternatives.

The remainder of this paper is organized as follows: Section 2 gives an overview of AODV and presents an Internet access solution for MANETs. Section 3 investigates three gateway discovery strategies. The simulation results are presented and discussed in Sect. 4. Finally, Sect. 5 concludes this paper and gives some directions for future work.

2 Protocol Description

As mentioned above, AODV was originally designed for routing packets within a MANET and not between a MANET and a wired network. In order to achieve routing across the network interconnection, the routing protocol needs to be modified. After giving an overview of AODV, we present a solution, which is referred to as AODV+ [7], where AODV is extended to provide Internet access for mobiles node in a MANET.

2.1 Ad hoc On-Demand Distance Vector (AODV)

Ad hoc On-Demand Distance Vector (AODV) is a reactive MANET routing protocol [1], where the reactive property implies that a mobile node requests a route only when it needs one. Consequently, the node maintains a routing table containing route entries only to destinations it is currently communicating with. Each route entry contains a number of fields such as *Destination IP Address*, *Next Hop* (a neighbor node chosen to forward packets to the destination), *Hop Count* (the number of hops needed to reach the destination) and *Lifetime* (the expiration or deletion time of the route). AODV guarantees loop-free routes by using sequence numbers that indicate how fresh a route is.

Route Discovery. Whenever a node (source) determines that it needs a route to another node (destination) it broadcasts a *route request* (RREQ) message and sets a timer to wait for the reception of a *route reply* (RREP). A node that receives a RREQ creates a *reverse route entry* for the source in its routing table. Then it checks to determine whether it has received a RREQ with the same Originator IP Address and RREQ ID within the last PATH_DISCOVERY_TIME. If such a RREQ has been received, the node discards the newly received RREQ in order to prevent duplicated RREQs from being forwarded. If the RREQ is not discarded the node continues to process it as follows: If the node is

either the destination or if it has an unexpired route to the destination it unicasts a RREP back to the source; otherwise it rebroadcasts the RREQ. If a RREP is generated, any intermediate node along the path back to the source creates a *forward route entry* for the destination in its routing table and forwards the RREP towards the source.

If the source does not receive any RREP before the RREQ timer expires, it broadcasts a new RREQ with an increased time to live (TTL) value. This technique is called *expanding ring search* and continues until either a RREP is received or a RREQ with the maximum TTL value is broadcasted. Broadcasting a RREQ with the maximum TTL value is referred to as a *network-wide search* since the RREQ is disseminated throughout the MANET. If a source performs a network-wide search without receiving any corresponding RREP, it may try again to find a route to the destination, up to a maximum of RREQ_RETRIES times after which the session is aborted.

Route Maintenance. When an active link breaks, the node upstream of the break invalidates all its routes that use the broken link. Then, the node broadcasts a *route error* (RERR) message that contains the IP address of each destination that has become unreachable due to the link break. Upon reception of such a RERR message, a node searches its routing table to see if it has any route(s) to the unreachable destination(s) (listed in the RERR message) that uses the originator of the RERR as the next hop. If such routes exist, they are invalidated and the node broadcasts a new RERR message. This process continues until the source receives a RERR message. The source then invalidates the listed routes as previously described and initiates a route discovery process if needed.

2.2 Internet Access for Mobile Ad Hoc Networks

Whenever a mobile node is about to communicate with a fixed wired node, it searches its routing table for a route towards the destination. If a route is found, the communication can be established. Otherwise, the mobile node starts a route discovery process by broadcasting a RREQ message as described above.

When an intermediate mobile node receives a RREQ message, it searches its routing table for a route towards the wired destination. If a route is found, the intermediate node would normally send a RREP back to the originator of the RREQ. But in that case, the source would think that the destination is a mobile node that can be reached via the intermediate node. It is important that the source knows that the destination is a fixed node and not a mobile node, because these are sometimes processed differently. In our solution, this problem has been solved by preventing the intermediate node to send a RREP back to the originator of the RREQ if the destination is a wired node. Instead, the intermediate node updates its routing table and rebroadcasts the received RREQ message. To determine whether the destination is a wired node or not, an intermediate node consults its routing table. If the next hop address of the destination is a default route (see Table 1), the destination is a wired node. Otherwise, the destination is a mobile node or a gateway.

Since neither the fixed node nor the mobile nodes in the MANET can reply to the RREQ, it is rebroadcasted until its TTL value reaches zero. When the timer of the RREQ

expires, a new RREQ message is broadcasted with a larger TTL value. However, since the fixed node cannot receive the RREQ message (no matter how large the TTL value is) the source will never receive the RREP message it is waiting for. This problem has been solved by letting the source assume the destination is a fixed node if a network-wide search has been done without receiving any corresponding RREP. In that case, the source must find a route to a gateway (if it does not have one already, see Sect. 3) and send its data packets towards the gateway, which will forward them towards the fixed node.

It should be mentioned that when using the expanding ring search, a considerable route discovery delay will occur if the destination is a fixed node. Modifying the parameters involved in the expanding ring search technique (such as TTL_START and TTL_THRESHOLD) can decrease the route discovery delay if the destination is a fixed node. However, the modification can also result in increased routing overhead if the destination is a mobile node. The modification could for example be to increase TTL_START. Assuming the destination is a fixed node, increasing TTL_START would result in less number of broadcasted RREQs (and consequently less delay) before the source assumes that the destination is a fixed node. Thus, different approaches are preferable depending on whether a mobile node is to communicate mostly with the MANET or the Internet.

Handover. Due to the multihop nature of a MANET, there might be several reachable gateways for a mobile node at some point of time. If a mobile node receives gateway advertisements from more than one gateway, it has to decide which gateway to use for its connection to the Internet. In this solution a mobile node initiates a handover when it receives an advertisement from a gateway that is closer (in terms of number of hops) than the one it is currently registered with. Apart from the hop count, there are other potential criteria that could be used to determine whether a handover is needed or not; e.g. geographical distance, radio signal level, signal delay and direction of node movement [8]. However, the question of a suitable metric for route selection is a general routing issue in MANETs.

Gateway Operation. When a gateway receives a RREQ, it consults its routing table for the destination IP address specified in the RREQ message. If the address is not found, the gateway sends a RREP with an 'I' flag (RREP_I) back to the originator of the RREQ. On the other hand, if the gateway finds the destination in its routing table, it unicasts a RREP as normal, but may also optionally send a RREP_I back to the originator of the RREQ. This will provide the mobile node a default route although it has not requested it. If the mobile node is to communicate with the Internet later, the default route is already established, and another time consuming gateway discovery process can be avoided.

Routing Table Management. Another issue that must be taken into consideration is how the routing table should be updated after a network-wide search without receiving any corresponding RREP. Once the source has determined that the destination is a fixed

node located on the Internet, it has to create a route entry for the fixed node in its routing table. If the route entry for the fixed destination would not be created in the routing table, the source would not find the address to the fixed node in its routing table when the next data packet would be generated and hence, the source would have to do another time consuming network-wide search.

Table 1 shows how the routing table of a mobile node should look like after creation of a route entry for a fixed node. The first entry tells the node that the destination is a fixed node since the next hop is specified by the default route. The second entry specifies which gateway the node has chosen for its Internet connection. The last entry gives information about the next hop towards the gateway.

Table 1. The routing table of a mobile node after creating a route entry for a fixed node

Destination Address	Next Hop Address
Fixed node	Default
Default	Gateway
Gateway	IMN

Another challenge is how to setup the routing table of an intermediate mobile node (IMN) chosen to forward data packets towards the gateway. Since the forward route entries are created for the gateway (the source of the RREP_I) and not for the fixed node, which is the final destination of the data packets, IMN will not find any valid route for the fixed node when it receives data packets from the source. Therefore, it would normally drop the data packets because it does not know how to forward them. In our solution, if IMN does not find a valid route to the destination and if the destination is a fixed node, it creates a (or updates the) route entry for the fixed node in its routing table and forwards the data packets towards the gateway.

3 Gateway Discovery

An interesting question to investigate is whether the configuration phase with the gateway should be initiated by the gateway (proactive method), by the mobile node (reactive method) or by mixing these two approaches (hybrid proactive/reactive method) has been discussed lately. In the following, the mechanisms of these three approaches are discussed.

3.1 Proactive Gateway Discovery

The proactive gateway discovery is initiated by the gateway itself. The gateway periodically broadcasts a *gateway advertisement* (GWADV) message with the period determined by ADVERTISEMENT_INTERVAL [7, 9]. The advertisement period must be chosen with care so that the network is not flooded unnecessarily.

The mobile nodes that receive the advertisement, create a (or update the) route entry for the gateway and then rebroadcast the message. To assure that all mobile nodes within

the MANET receive the advertisement, the number of retransmissions is determined by NET_DIAMETER defined by AODV. However, this will lead to enormously many unnecessary duplicated advertisements. A conceivable solution that prevents duplicated advertisements, is to introduce a “GWADV ID” field in the advertisement message format similar to the “RREQ ID” field in the RREQ message format (see Sect. 2.1).

It is worth mentioning that the mobile nodes randomize their rebroadcasting of the GWADV message in order to avoid synchronization and subsequent collisions with other nodes’ rebroadcasts.

The advantage of this approach is that there is a chance for the mobile node to initiate a handover before it looses its Internet connection. The disadvantage is that since a control message is flooded through the whole MANET periodically, limited resources in a MANET, such as power and bandwidth, will be used a lot.

3.2 Reactive Gateway Discovery

The reactive gateway discovery is initiated by a mobile node that is to create or update a route to a gateway. The mobile node broadcasts a RREQ with an ’I’ flag (RREQ_I) to the ALL_MANET_GW_MULTICAST [5] address, i.e. the IP address for the group of all gateways in a MANET. Thus, only the gateways are addressed by this message and only they process it. Intermediate mobile nodes that receive a RREQ_I are not allowed to answer it, so they just rebroadcast it. When a gateway receives a RREQ_I, it unicasts back a RREP_I which, among other things, contains the IP address of the gateway.

The advantage of this approach is that control messages are generated only when a mobile node needs information about reachable gateways. Hence, periodic flooding of the whole MANET, which has obvious disadvantages as discussed in 3.1, is prevented. The disadvantage of reactive gateway discovery is that a handover cannot be initiated before a mobile node looses its Internet connection. As a consequence, a situation can occur where a mobile node uses a gateway for its Internet connection although there are other gateways that are closer.

3.3 Hybrid Gateway Discovery

To minimize the disadvantages of the proactive and reactive strategies, they can be combined into a hybrid proactive/reactive method for gateway discovery. For mobile nodes in a certain range around a gateway, proactive gateway discovery is used while mobile nodes residing outside this range use reactive gateway discovery to obtain information about the gateway.

The gateway periodically broadcasts a GWADV message. Upon receipt of the message, the mobile nodes update their routing table and then rebroadcast the message. The maximum number of hops a GWADV can move through the MANET is determined by ADVERTISEMENT_ZONE. This value defines the range within which proactive gateway discovery is used. When a mobile node residing outside this range needs gateway information, it broadcasts a RREQ_I to the ALL_MANET_GW_MULTICAST address. Mobile nodes receiving the RREQ_I just rebroadcast it. When a gateway receives a RREQ_I, it sends a RREP_I towards the source.

Thus, the proactive gateway discovery method is used to handle the mobile nodes less or equal than ADVERTISEMENT_ZONE hops away from the gateway and the reactive gateway discovery method is used to handle the mobile nodes more than ADVERTISEMENT_ZONE hops away from the gateway.

4 Performance Evaluation

In order to evaluate the performance of the three gateway discovery methods, the network simulator ns-2 has been used. First, the source code of AODV in ns-2 was extended to provide Internet access to mobile nodes. Then the three gateway discovery methods were implemented. This code, which is referred to as AODV+, has been contributed [7] to ns-2 and is free to be downloaded and used by everyone. The latest version of ns-2 (ns-2.27) has been used in this study.

4.1 Simulation Scenario

The studied scenario consists of 60 mobile nodes, two gateways, two routers and two hosts. The topology is a rectangular area with 1300 m length and 800 m width. A rectangular area was chosen in order to force the use of longer routes between nodes, compared to a square area with the same node density. The two gateways were placed on each side of the area; their x- and y-coordinates in metres are (200,500) and (1100,500). All simulations were run for 1000 seconds of simulation time. Since we were interested in studying the behaviour of the network in steady state, the first 100 seconds of the simulation were ignored.

Ten of the 60 mobile nodes are constant bit rate (CBR) traffic sources sending data packets with a size of 512 bytes, to one of the two hosts, chosen randomly. The sources are distributed randomly within the MANET. The transmission range of the mobile nodes is 250 metres.

A screenshot of the simulation scenario is shown in Fig. 1. The 60 small circles represent the mobile nodes. The two hexagonal nodes at each side of the figure are the gateways and the four square nodes are the two hosts and the two routers.

4.2 The Mobility Model

The mobile nodes move according to an improved version of the commonly used random waypoint model. It has been shown that the original random waypoint model can generate misleading results [10]. With the improved random waypoint model the mobile node speed reaches steady state after a quick warm-up period.

Each mobile node begins the simulation by selecting a random destination in the defined area and moves to that destination at a random speed. The random speed is distributed uniformly in the interval [1,19] m/s. Upon reaching the destination, the mobile node pauses for 10 seconds, selects another destination, and proceeds as described. This movement pattern is repeated for the duration of the simulation.

The gateways broadcast GWADVs every ADVERTISEMENT_INTERVAL=5 seconds when the proactive or hybrid discovery method is used (see Sect. 3.1 and 3.3).

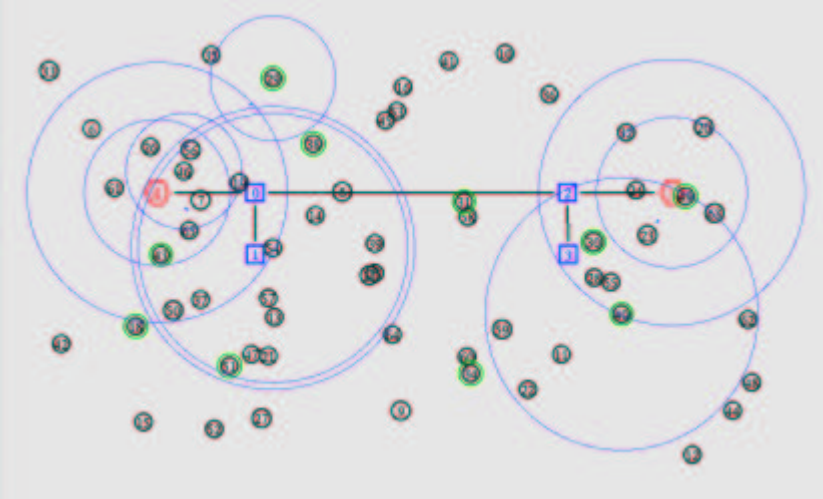

Fig. 1. Screenshot of the simulation scenario.

ADVERTISEMENT_ZONE, which is set to three, is used for the hybrid gateway discovery method and defines the range within which proactive gateway discovery is used. Outside this range the reactive gateway discovery is used.

4.3 Performance Metrics

In comparing the gateway discovery approaches, the evaluation has been done according to the following three metrics:

- The packet delivery ratio is defined as the number of received data packets divided by the number of generated data packets.
- The end-to-end delay is defined as the time a data packet is received by the destination minus the time the data packet is generated by the source.
- The overhead is defined as the amount of AODV messages in bytes divided by the sum of the AODV messages plus the data packets in bytes.

Each data point is an average value of ten runs with different randomly generated movement patterns.

4.4 Simulation Results

In all figures discussed in this section it should be noted that the term “traffic load” denotes only the data traffic that each source generates, which is ten times less than the total data traffic in the whole network. To that come also control packets sent by the data link and network layers.

Figure 2 shows the impact of the advertisement interval on the average end-to-end delay when the traffic load changes for the proactive gateway discovery method. It can

be observed that the curve representing the advertisement interval of one second differs greatly from other curves representing higher advertisement intervals. The reason is that a very short interval leads to a lot of advertisements and thus a lot of overhead, which in turn means many collisions, retransmissions and route discoveries that increase the end-to-end delay.

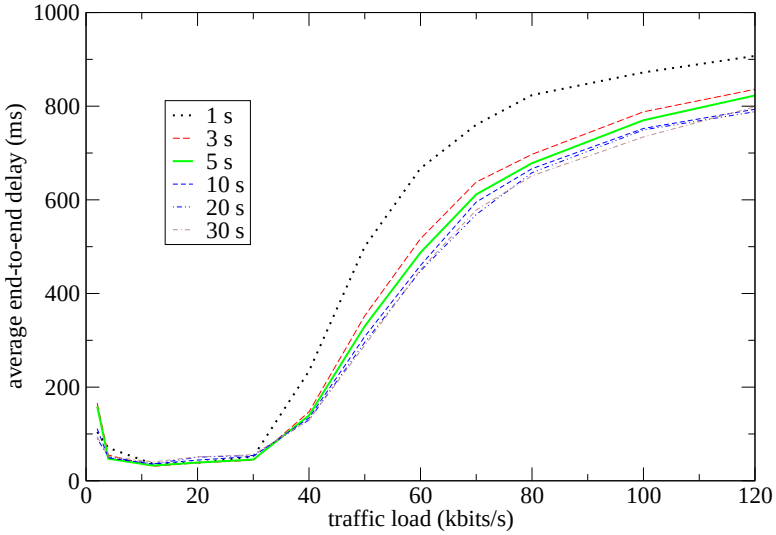

Fig. 2. The impact of advertisement interval.

Figures 3, 4 and 5 show the packet delivery ratio, the average end-to-end delay and the AODV overhead respectively for the three gateway discovery methods when the traffic load changes.

Packet losses occur frequently due to many reasons, e.g. when a source sends packets along a path that recently has broken but the source has not been informed about that yet; or when a source has no other nodes within its transmission range (i.e. the node is isolated) for some time and its outgoing buffer is full. Since we have omitted the TCP protocol and its retransmission function from our model high packet losses may occur.

As Fig. 3 shows, the packet delivery ratio is high when the traffic load is light but decreases when the traffic increases. This result is expected but it can also be seen that increasing the traffic affects all three approaches pretty much the same way. One can also see that the delivery ratio is somewhat lower for very light loads (5 kbits/s/source) compared to light loads (20 kbits/s/source). The reason for this is that once a connection has been established, it is not fully used when the traffic is very light. Therefore, only a few number of packets are sent before the connection breaks and a new route must be discovered.

Figure 4 shows that the average end-to-end delay increases as expected when the load increases, since increased load means more collisions, retransmissions and route

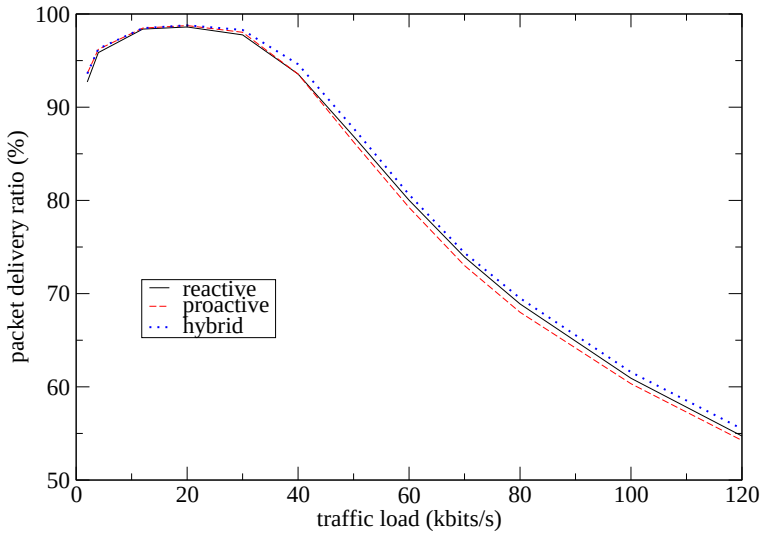

Fig. 3. Packet delivery ratio vs. traffic load.

discoveries. We can also see that the difference between the different strategies is negligible.

One might have expected that the delivery ratio and the average end-to-end delay would have been different for the reactive method compared to e.g. the proactive. From one point of view, the reactive method should perform better since it generates less overhead, which should cause less number of collisions. On the other hand, the reactive method should perform worse because it does not send periodic advertisements, which would give shorter routes (in terms of number of hops) in the long term. Since a number of other aspects need to be taken into account, it is our belief that the given scenario and the assumptions made for the simulation have a significant impact on the results.

There are some problems with the ARP [1] implementation in ns-2, which is based on the BSD [2] implementation of ARP [11], that have negative impact on our results. Each node has an ARP queue that can hold only one packet for each neighbour while requesting the MAC address of the next hop. If other packets arrive to the queue before the MAC address is resolved, all but the last one will be dropped [12]. This can lead to loss of important messages from upper layers, such as the RREP or the RREP_I messages from AODV. Consequently, if the source does not receive any RREP or RREP_I before its timer expires, it has to reattempt its gateway discovery process where the reply could be lost again. Remember that this important message can be dropped by ARP on each hop between the gateway and the source where an address resolution process is started. In the worst case, the source will give up after some attempts and the session is aborted. Increasing the buffer size of ARP can prevent situations like this to occur.

[1] Address Resolution Protocol
[2] Berkeley Software Distribution

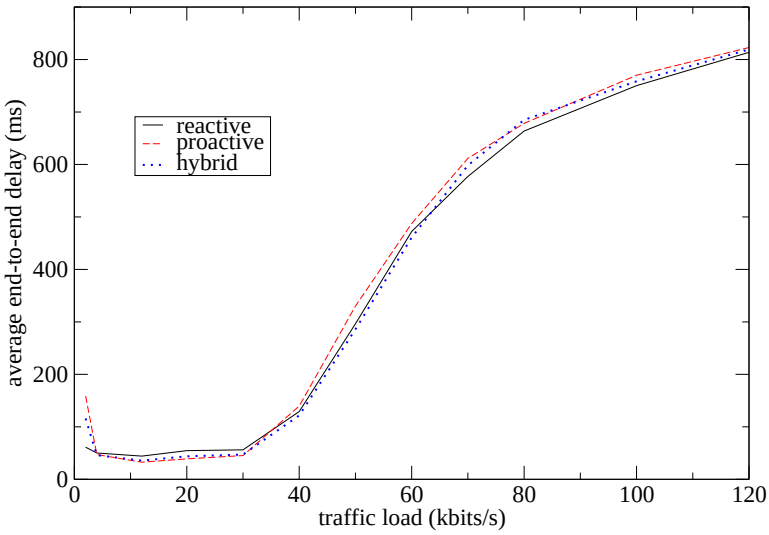

Fig. 4. Average end-to-end delay vs. traffic load.

There is another problem, where ARP is involved, which cannot be solved by increasing the buffer size. Since there is no timer involved in the address resolution process, a retransmission will not occur until it is triggered by a new incoming packet. This can have a significant impact on the end-to-end delay. Suppose that a data packet is sent to ARP from the routing protocol. Because of some reason (e.g. collision) the address resolution fails. Before a new data packet is sent to ARP to trigger an ARP request retransmission, the routing protocol changes its route towards the destination (with a new next hop) and, hence, no MAC address resolution is needed for the old next hop anymore. So far there is no problem except that the old data packet remains in the ARP queue. If the node much later needs to resolve the MAC address of the old next hop and the ARP resolution succeeds, the data packet waiting in the queue will be sent to the next hop resulting in a very long end-to-end delay. Increasing the buffer size will in fact only make the problem even worse since then there are more than a single data packet that will be delivered to the next hop with a very long end-to-end delay.

Furthermore, the lack of retransmissions means that one single loss of an ARP request or an ARP reply means that the data (e.g. RREP_I) cannot be sent to the source, which will be forced to reattempt its gateway discovery process.

The first problem caused by ARP has been investigated in [13] which shows that increasing the ARP buffer size makes the situation much better (although another solution is preferred). The second problem is discussed in [14], which suggests a cross-layer feedback mechanism from MAC to ARP.

Another thing that affects the simulation results in a negative way is when sources become isolated from the MANET such that they cannot reach any gateway. Isolated sources result in decreased packet delivery ratio and increased end-to-end delay.

In Fig. 5 the AODV overhead is dominated by the periodically broadcasted GWADV messages. As the figure shows, the AODV overhead is significantly larger for the proactive approach than for the reactive approach, especially for light traffic loads. This is an expected result since the proactive approach periodically broadcasts gateway information no matter if the mobile nodes need them or not, while the reactive approach broadcasts gateway information only when a mobile node sends a request for it. Moreover, the figure shows that the overhead of the hybrid approach, which is a mixture of both the proactive and the reactive approach, is between the two other methods.

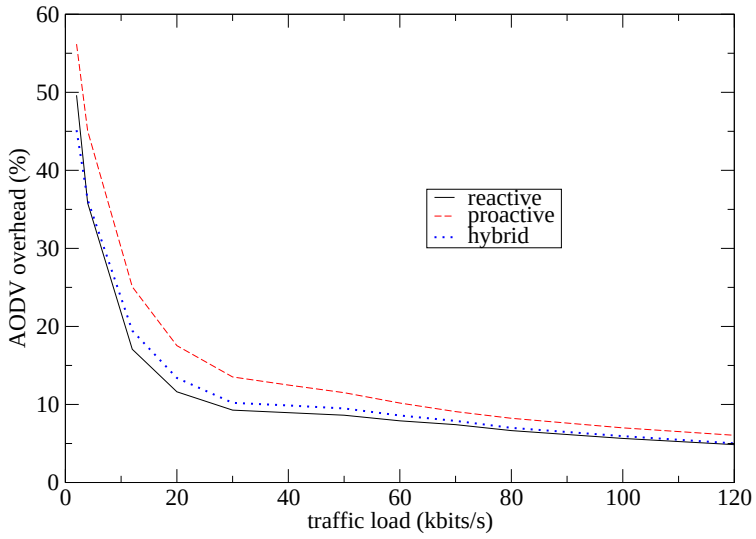

Fig. 5. AODV overhead vs. traffic load.

5 Conclusion

We have presented a solution for Internet access for mobile nodes in a MANET. The MANET routing protocol AODV has been extended to route packets, between a wireless MANET and the wired Internet. To achieve this, some nodes must be able to communicate with the MANET and with the fixed Internet. As all communication between the wireless and the wired network must pass through these nodes, they are referred to as gateways. In this paper, three methods for detection of these gateways have been presented, implemented and compared. The three methods for gateway detection are referred to as reactive, proactive and hybrid gateway discovery. When it comes to end-to-end delay and packet delivery ratio, the three methods show surprisingly similar behaviour. The fact that the proactive method shows much higher overhead in terms of control packets than the other methods is more obvious.

In order to fully understand the reasons behind the large delays and the rather low packet delivery ratio that were found, the authors plan to do a more detailed study. This would provide a better understanding of which parts of the end-to-end path that contribute most to the discovered delays and packet losses.

References

1. C. Perkins, E. M. Belding-Royer and S. Das. *"Ad hoc On-Demand Distance Vector (AODV) Routing"*. Experimental RFC 3561.
2. D. B. Johnson, D. A. Maltz, Y. Hu and J. G. Jetcheva. *"The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR)"*. IETF Internet Draft, April 2003. Work in progress.
3. T. Clausen, P. Jacquet, A. Laouiti, P. Minet, P. Muhlethaler, A. Qayyum and L. Viennot. *"Optimized Link State Routing Protocol"*. Experimental RFC 3626.
4. R. Ogier, M. Lewis and F. Templin. *"Topology Dissemination Based on Reverse-Path Forwarding (TBRPF)"*. Experimental RFC 3684.
5. R. Wakikawa, J. Malinen, C. Perkins, A. Nilsson and A. J. Tuominen. *"Global Connectivity for IPv6 Mobile Ad Hoc Networks"*, IETF Internet Draft, February 2003. Work in progress.
6. S. McCanne and S. Floyd. *"The Network Simulator - ns-2"*. www.isi.edu/nsnam/ns/. K. Fall, K. Varadhan. *"The ns Manual"*.
7. *"The Network Simulator: Contributed Code"*. www.isi.edu/nsnam/ns/ns-contributed.html.
8. M. Bernard. *"Gateway Detection and Selection for Wireless Multihop Internet Access"*. Master's thesis. Olching, Germany, May 2002.
9. A. Hamidian. *"A Study of Internet Connectivity for Mobile Ad Hoc Networks in NS 2"*. Master's thesis. Department of Communication Systems, Lund Institute of Technology, Lund University. January 2003.
10. J. Jungkeun, M. Liu and B. Noble. *"Random Waypoint Considered Harmful"*. IEEE INFOCOM 2003, San Francisco, April 2003.
11. W. R. Stevens. *"TCP/IP Illustrated, Volume 1"*. Addison Wesely, 1994.
12. J. Broch, D. Maltz, D. B. Johnson, Y. Hu and J. Jetcheva. *"A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols"*. In proceedings of the 4th ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom '98), pages 85-97, October 1998.
13. C. Carter, S. Yi and R. Kravets. *"ARP Considered Harmful: Manycast Transactions in Ad Hoc Networks"*. Proceedings of the IEEE Wireless Communications and Networking Conference, 2003.
14. S. Perur, L. Wadia and S. Iyer. *"Improving the Performance of MANET Routing Protocols using Cross-Layer Feedback"*. www.it.iitb.ac.in/ srinath/pubs/cit03.pdf.