Comparing AODV and OLSR Routing Protocols

Aleksandr Huhtonen
Helsinki University of Technology
Telecommunication Software and Multimedia Laboratory
ahuhtone@cc.hut.fi

Abstract

An ad hoc wireless network consists of mobile networks which creates an underlying architecture for communication without the help of traditional fixed-position routers. Nevertheless, the architecture must maintain communication routes although the hosts are mobile and they have limited transmission range. There are different protocols for handling the routing in the mobile environment. This paper will focus on two well know algorithms: Ad hoc On-Demand Distance Vector and Optimized Link State Routing Protocol.

KEYWORDS: ad hoc networks, wireless networks, ad hoc on-demand distance vector, optimized link state routing protocol, ad hoc network routing protocols.

1 Introduction

Wireless communication technology is increasing daily, with such growth sooner or later it would not be practical or simply physically possible to have a fixed architecture for this kind of network. Ad hoc wireless network must be capable to self-organise and self-configure due to the fact that the mobile structure is changing all the time. Mobile hosts have a limited range and sending the message to another host, which is not in the sender's host transmission range, must be forwarded through the network using other hosts which will be operated as routers for delivering the message throughout the network. The mobile host must use broadcast for sending messages and should be in promiscuous mode for accepting any messages that it receives. In the ad hoc network there can be unidirectional hosts, that can transmit only to the one direction, so that the communication is not bi-directional as in the usual communication systems. [4, 5, 8]

The routing protocols for ad hoc wireless network should be capable to handle a very large number of hosts with limited resources, such as bandwidth and energy. The main challenge for the routing protocols is that they must also deal with host mobility, meaning that hosts can appear and disappear in various locations. Thus, all hosts of the ad hoc network act as routers and must participate in the route discovery and maintenance of the routes to the other hosts. For ad hoc routing protocols it is essential to reduce routing messages overhead despite the increasing number of hosts and their mobility. Keeping the routing table small is another important issue, because the increase of the routing table will affect the control packets sent in the network and this in turn will affect large link overheads. [4, 5, 8]

Routing protocols are divided into two categories based on how and when routes are discovered, but both find the shortest path to the destination. Proactive routing protocols are table-driven protocols, they always maintain current up-to-date routing information by sending control messages periodically between the hosts which update their routing tables. When there are changes in the structure then the updates are propagated throughout the network. The proactive routing protocols use link-state routing algorithms which frequently flood the link information about its neighbours. Other routing protocols are on-demand routing protocols, in other words reactive, ones which create routes when they are needed by the source host and these routes are maintained while they are needed. Such protocols use distance-vector routing algorithms, they have vectors containing information about the cost and the path to the destination. When nodes exchange vectors of information, each host modify own routing information when needed. The ad hoc routing protocols are usually classified as a pure proactive or a pure reactive protocol, but there are also hybrid protocols. This only concern flat routing protocols, but there are also hierarchical and graphic position assisted routing protocols. [4]

The routing protocol needs to have following qualities in order to be effective: distributed operation, loop-freedom, demand-based operation, proactive operation, security, "sleep" period operation, unidirectional link support. [7]

Distributed operation means that any host can enter or leave the network whenever it wants. Loop-freedom is needed to prevent that host will be sending information uselessly creating overhead. Demand-based operation will let the protocol adapt to the traffic pattern to decrease traffic and use bandwidth resources more efficiently, but this will increase route discovery delay. Proactive operation is the opposite of demand-based operation. It can be used when demand-based operation is unsuitable or when there is enough bandwidth and energy resources for the proactive operation. Security must be taken in consideration in modern communication and mobile devices are vulnerable to snooping because of the broadcasting. The basic idea in the "sleep" period operation is to reduce the energy used

by hosts and protocol should be able to adjust such sleep periods without any consequences. Because links can be unidirectional in the mobile network, it is essential to have a unidirectional link support in routing protocol. [7]

This paper will compare the two ad hoc routing protocols: reactive Ad hoc On Demand Distance Vector (AODV) and proactive Optimized Link State Routing (OLSR) protocols. The reminding part of this paper is organized as follows. Sec. 2 will give information about AODV protocol and Sec. 3 about OLSR protocol. In the end of the sections 2 and 3 the possible information about each protocol's advantages will be given. Actual comparison will be done in Sec. 4. Sec. 5 will conclude this paper.

2 Ad hoc On Demand Distance Vector (AODV)

2.1 Introduction to AODV

The information in this section concerning the Ad Hoc On Demand Distance Vector Protocol (AODV) protocol is taken from the RFC [1]. AODV is a reactive protocol, i.e., so the routes are created and maintained only when they are needed. The routing table stores the information about the next hop to the destination and a sequence number which is received from the destination and indicating the freshness of the received information. Also the information about the active neighbours is received throughout the discovery of the destination host. When the corresponding route breaks, then the neighbours can be notified.

The route discovery is used by broadcasting the RREQ message to the neighbours with the requested destination sequence number, which prevents the old information to be replied to the request and also prevents looping problem, which is essential to the traditional distance vector protocols [4]. The route request does not add any new information about the passed hosts only it increases its hop metric. Each passed host makes update in their own routing table about the requested host. This information helps the destination reply to be easily routed back to the requested host. The route reply use RREP message that can be only generated by the destination host or the hosts who have the information that the destination host is alive and the connection is fresh.

New version of the AODV routing protocol [1] has also a feature that only the destination host can reply to the sent request. When the reply is sent back to the requested host the actual hop metric is counted. The intermediate hosts records information about the replied host upon receiving the reply message. The hosts must record and forward new information only when the sequence number is greater or if the sequence number is the same and hop metric is smaller. The additional RREP-ACK message must be sent in response to the RREP message when the message has an active acknowledgment option. The acknowledgment option is set up when there is possibility that the route may be unidirectional. This feature enables that the unidirectional links can be detected. When the breakage of the route is noticed the host sends RERR message to the neighbours. The Hello message is periodically sent for maintaining the route information.

Usually messages are transmitted by using IP limited broadcast address, but the messages are checked for the content so that they will not be broadcasted throughout the entire network. Some of the messages are supposed to be spread widely in the network, for example route request message (RREQ). So their distribution is restricted by the TTL field in the IP header. Usually the fragmentation of the IP packet is not required [1].

2.2 Routing

2.2.1 Sequence numbers

The sequence numbers are the key idea for removing the old and invaluable information from the network. The sequence number act as timestamps and prevent this distance vector protocol from the loop problem [1, 4, 5]. The destination sequence number for each possible destination host are stored in the routing table. The destination sequence numbers are updated in the routing table when the host receives the message with the greater sequence number. The host can change the destination sequence number in the routing table if it is offering a new route to itself or if some route expires or simply breaks. [1]

The host also keeps its own sequence number, which must be incremented only in two different cases: before it sends RREQ message and when the host sends a RREP message responding to the RREQ message. In the second case the sequence number must be incremented to the maximum of the current sequence number and the sequence number in the received RREQ message. The sequence numbers must be treated as unsigned integers so that the possible rollovers can occur, AODV protocol supports the sequence number to be rolled over without any problems. [1]

2.2.2 RREQ, RREP and RREP-ACK messages

The route request message (RREQ) is sent when the host does not know the route to the needed destination host or the existed route is expired. The RREQ message includes the destination sequence number which is the last known sequence number of the destination host entry found in the routing table. If there contains no entry for the destination host, then the unknown sequence number flag must be set. The RREQ message also contains the requesting hosts sequence number, which must be incremented beforehand. The RREQ ID field is incremented by one which is found from the last used RREQ message, which was sent by this host. Also the hop count metric must be set to zero and before sending the RREQ message the RREQ ID and its own address must be saved to the buffer for the specified amount of time, so that it recognize the replies. [1]

There is possibility that some hosts can be unidirectional then the G field can be set in RREQ message, so that every intermediate host will generate the RREP message and unicast it to the requesting host. Also the intermediate host must generate the gratuitous RREP to the destination host. There is a limit of RREQ messages that the host can send per minute, waiting before retransmitting the RREQ message, and number of RREQ message retries it can send overall. All the repeat attempts must be sent using binary exponential backoff. The expanding ring search technique is used for preventing the RREQ messages from unnecessary spreading out through the network for more information about the technique is found from [1].

First when the host receives RREQ message, it checks the time period between the last RREQ messages from the same host and discards the message if it is under the specified limit. Next host increases the hop count by one in the RREQ message and makes update in own routing table basing on the sequence number and the requested host's address. Also the hop count is copied from the RREQ message. The host marks that the route is valid to requested host and adds information about the next hop specifying to which host the message should be forwarded to. Host needs to count the lifetime of the route to the requested host. The host must set the destination sequence number in the RREQ message if the sequence number is greater in the routing table than in the received message, but the host should not modify the sequence number in the routing table. Lastly the host should broadcast the request and decrease its TTL field in the IP header. [1]

The host can generate the route reply message (RREP) if the destination is the host itself or if the route to the destination is valid and has the same or greater destination sequence number, but only if the D field is not set. D field in the RREQ message indicates that only the destination host can reply to the RREQ message. When generating the RREP message host copies the destination address and the requested host's sequence number to the corresponding RREP message's fields. If the receiver is the destination host then its own sequence number is incremented and copied to the destination sequence number field. In addition, the hop count is set to zero and the lifetime field of the RREP message is set to the initial timeout value of the host. If the receiver is the intermediate host, then it just copies destination sequence number from the routing table and adds the host address from where it has received RREQ message to the destination address field. Also the host must add the hop count with the lifetime from the routing table to the RREP. The lifetime is calculated by subtracting the current time and the expiration time from the routing table. When the RREP message is created it is sent using unicast to the next hop in order to be delivered to the requested host. The hop count metric is incremented along the path, so at the end, it corresponds to the actual distance between the hosts. [1]

The gratuitous RREP is like the original RREP only it is sent to the destination host and all of the fields are generated in the same manner only gratuitous RREP destination address is set to the requested host's address. If the gratuitous node is sent to the destination node and the destination node has already sent its own RREQ message, then the contents of the RREQ message and RREP message which was sent in response to the earlier requested host are actually the same [1].

When the host receives the RREP message it searches for the previous hop and increases hop metric by one. If there is no routing entry for the previous hop, then the route is created but without a valid sequence number. Also it is necessary that the route to the destination host is created in the routing table. First the host must compare the destination sequence numbers. The routing table entry is modified only in the following situations: the sequence number is marked as invalid in routing table, the destination sequence number is greater than the routing table entry and the route is marked as valid, the sequence number is the same but the route is marked as inactive, the sequence number is same and the hop count metric is smaller than the information in the routing table. If the routing table is updated or created then the route must be marked as active and the destination sequence number field as valid. Also in the routing table the next hop is assigned to the host address from which the RREP message was received. The hop count is increased and the expiry time is set to current time plus the lifetime from the RREP message. The destination sequence number is copied from the message. Finally RREP message is forwarded to the requester using the next hop address from the routing table. If the address to which the RREP message is forwarded can have errors or maybe unidirectional then the A flag is set, which correspond to the receiver of the message to generate the RREP-ACK message back to the sender. [1]

2.2.3 RERR messages, route expiry and route deletion

When the link breakage happens the host must invalidate the existing route in the routing table entry. The host must list the affected destinations and determine which neighbours can be affected with this breakage. Finally the host must send the route error (RERR) message to the corresponding neighbours. The RERR message can be broadcasted if there are many neighbours which need that information or unicasted if there is only one neighbour. The host can also iteratively unicast the message to needed neighbours if the broadcast is not possible. However, iterative unicasting must be considered as a single broadcast RRER message, so that RERR messages per second limit is essential [1].

If the host detects the link breakage of the active route, then the host makes a list of unreachable destinations based on the routing table entries where the unreachable neighbour acts as a next hop address. If host gets RERR messages, then the unreachable destinations is consisted from the routing table which has same addresses as in RERR message and routing table next hop address entries. The destination sequence numbers for the entries in the routing table for

the unreachable destinations must be incremented or if the host received RERR message, then simply copied from it. After this the entry for the unreachable hosts must be set to invalid lifetime. Lifetime is set to the current time plus specific deletion time, so that the entry is not deleted from the routing table before the lifetime expires. Then the RERR message with the unreachable destinations should be unicasted for one neighbour or broadcasted to the many neighbours with TTL value set to 1. The DestCount field in the RERR message describes the number of the unreachable host addresses. [1]

2.2.4 Repairing

When the link breakage occurs then the host can try to locally repair the link if the destination is no further than specified amount of hops. In order to repair the link the host increase the destination sequence number and broadcasts the RREQ message to the host. The TTL for the IP header must be calculated, so that locally repair process would not spread throughout the network. The host waits for the RREP messages to its RREQ message for specified amount of time. If the RREP message is not received, then it changes the routing table status for the entry to invalid. If host receives the RREP message then the hop count metric is compared. If the hop metric from the message is greater than the previous one then the RERR with the N field set up is broadcasted. The N field in the RERR message indicates that the host has locally repaired the link and the entry in the table should not be deleted. The received RREP message is handled as original RREP message. The repairing of the link before the data is sent to unavailable host is a proactive repairing [1]. Proactive repairing can be inefficient because the risk of repairing the routes that are not used anymore. So the proactive repairing can be used basing on the local traffic and the workload of the network. [1]

2.2.5 Hello messages

Although AODV is a reactive protocol it uses the Hello messages periodically to inform its neighbours that the link to the host is alive. The Hello messages are broadcasted with TTL equals to 1, so that the message will not be forwarded further. When host receives the Hello message it will update the lifetime of the host information in the routing table. If the host does not get information from the host's neighbour for specified amount of time, then the routing information in the routing table is marked as lost. This action generates needed RRER message to inform other hosts of the link breakage. The routes that were created by the Hello message and were not used for any routing actions should not generate the RERR message when the link breakage occurs. [1]

2.2.6 Routing table structure

This is the main data structure where all needed information about the routes is stored. The routing table must include at least the following fields: destination address, destination sequence number, hop count, next hop, lifetime, precursor list, and route state. The precursor list contains the information about which hosts can possible forward the messages to this route. Precursor list contains the information to which neighbour the errors should be forwarded when the possible break occurs. [1]

2.3 Advantages

Because the AODV protocol is a flat routing protocol it does not need any central administrative system to handle the routing process. Reactive protocols like AODV tend to reduce the control traffic messages overhead at the cost of increased latency in finding new routes [8].

In addition, AODV tries to keep the overhead of the messages small. If host has the route information in the Routing Table about active routes in the network, then the overhead of the routing process will be minimal. The AODV has great advantage in overhead over simple protocols which need to keep the entire route from the source host to the destination host in their messages. The RREQ and RREP messages, which are responsible for the route discovery, do not increase significantly the overhead from these control messages. AODV reacts relatively quickly to the topological changes in the network and updating only the hosts that may be affected by the change, using the RRER message. The Hello messages, which are responsible for the route maintenance, are also limited so that they do not create unnecessary overhead in the network. [5]

The AODV protocol is a loop free and avoids the counting to infinity problem, which were typical to the classical distance vector routing protocols, by the usage of the sequence numbers. [1, 4, 5]

3 Optimized Link State Routing Protocol (OLSR)

3.1 Introduction of OLSR

The information in this section concerning the Optimized Link State Protocol is taken from its RFC 3561 [2]. Optimized Link State Protocol (OLSR) is a proactive routing protocol, so the routes are always immediately available when needed. OLSR is an optimization version of a pure link state protocol. So the topological changes cause the flooding of the topological information to all available hosts in the network. To reduce the possible overhead in the network protocol uses Multipoint Relays (MPR). The idea of MPR is to reduce flooding of broadcasts by reducing the same broadcast in some regions in the network, more details about MPR can be found later in this chapter. Another reduce is to provide the shortest path. The reducing the time interval for the control messages transmission can bring more reactivity

to the topological changes. [3, 4, 5, 10, 11, 12, 13]

OLSR uses two kinds of the control messages: Hello and Topology Control (TC). Hello messages are used for finding the information about the link status and the host's neighbours. With the Hello message the Multipoint Relay (MPR) Selector set is constructed which describes which neighbours has chosen this host to act as MPR and from this information the host can calculate its own set of the MPRs. the Hello messages are sent only one hop away but the TC messages are broadcasted throughout the entire network. TC messages are used for broadcasting information about own advertised neighbours which includes at least the MPR Selector list. The TC messages are broadcasted periodically and only the MPR hosts can forward the TC messages. [2, 3, 10, 11, 12, 13]

There is also Multiple Interface Declaration (MID) messages which are used for informing other host that the announcing host can have multiple OLSR interface addresses. The MID message is broadcasted throughout the entire network only by MPRs. There is also a "Host and Network Association" (HNA) message which provides the external routing information by giving the possibility for routing to the external addresses. The HNA message provides information about the network- and the netmask addresses, so that OLSR host can consider that the announcing host can act as a gateway to the announcing set of addresses. The HNA is considered as a generalized version of the TC message with only difference that the TC message can inform about route cancelling while HNA message information is removed only after expiration time. The MID and HNA messages are not explained in more details in this chapter, the further information concerning these messages can be found in [2].

3.2 Routing

3.2.1 Neighbour Sensing

The link in the ad hoc network can be either unidirectional or bidirectional so the host must know this information about the neighbours. The Hello messages are broadcasted periodically for the neighbour sensing. The Hello messages are only broadcasted one hop away so that they are not forwarded further. When the first host receives the Hello message from the second host, it sets the second host status to asymmetric in the routing table. When the first host sends a Hello message and includes that, it has the link to the second host as asymmetric, the second host set first host status to symmetric in own routing table. Finally, when second host send again Hello message, where the status of the link for the first host is indicated as symmetric, then first host changes the status from asymmetric to symmetric. In the end both hosts knows that their neighbour is alive and the corresponding link is bidirectional. [2, 8, 11, 9]

The Hello messages are used for getting the information about local links and neighbours. The Hello messages periodic broadcasting is used for link sensing, neighbour's detection and MPR selection process. Hello message contains: information how often the host sends Hello messages, willingness of host to act as a Multipoint Relay, and information about its neighbour. Information about the neighbours contains: interface address, link type and neighbour type. The link type indicates that the link is symmetric, asymmetric or simply lost. The neighbour type is just symmetric, MPR or not a neighbour. The MPR type indicates that the link to the neighbour is symmetric and that this host has chosen it as Multipoint Relay. [2]

3.2.2 Multipoint Relays

The Multipoint Relays (MPR) is the key idea behind the OLSR protocol to reduce the information exchange overhead. Instead of pure flooding the OLSR uses MPR to reduce the number of the host which broadcasts the information throughout the network. The MPR is a host's one hop neighbour which may forward its messages. The MPR set of host is kept small in order for the protocol to be efficient. In OLSR only the MPRs can forward the data throughout the network. [2]

Each host must have the information about the symmetric one hop and two hop neighbours in order to calculate the optimal MPR set. The Fig. 1 is taken from [4] to illustrate these concepts. Information about the neighbours is taken from the Hello messages. The two hop neighbours are found from the Hello message because each Hello message contains all the hosts' neighbours. Selecting the minimum number of the one hop neighbours which covers all the two hop neighbours is the goal of the MPR selection algorithm. Also each host has the Multipoint Relay Selector set, which indicates which hosts has selected the current host to act as a MPR. [9, 10, 12, 13]

When the host gets a new broadcast message, which is need to be spread throughout the network and the message's sender interface address is in the MPR Selector set, then the host must forward the message. Due to the possible changes in the ad hoc network, the MPR Selectors sets are updated continuously using Hello messages. [2]

3.2.3 Multipoint Relays Selection

In this section the proposed algorithm for the selection of Multipoint Relay set is described. This algorithm is found from [2]. The algorithm constructs the MPR set which includes minimum number of the one hop symmetric neighbours from which it is possible to reach all the symmetrical strict two hop neighbours. The host must have the information about one and two hop symmetric neighbours in order to start the needed calculation for the MPR set. All the exchange of information are broadcasted using Hello messages. The neighbours which have status of willingness different than WILL_NEVER in the Hello message can be chosen to act as MPR. The neighbour must be symmetric in order to become an MPR.

Proposed algorithm for selecting Multipoint Relay set:

1. Take all the symmetric one hop neighbours which are willing to act as an MPR.
2. Calculate for every neighbour host a degree, which is a number of the symmetric neighbours, that are two hops away from the calculating source and does not include the source or its one hop neighbours.
3. Add the neighbour symmetric host to the MPR set. If it is the only neighbour from which is possible to get to the specific two hop neighbour, then remove the chosen host neighbours from the two hop neighbour set.
4. If there are still some hosts in the two hop neighbour set, then calculate the reachability of the each one hop neighbour, meaning the number of the two hop neighbours, that are yet uncovered by MPR set. Choose the node with highest willing value, if the values are the same then takes the node with greater number of reachability. If the reachability is the same, then take the one with greater degree counted in the second step. After choosing the neighbour for MPR set remove the reachable two hop neighbour from the two hop neighbour set.
5. Repeat previous step until the two hop neighbours set is empty.
6. For the optimization, set the hosts in the MPR set in the increasing order basing on the willingness. If one host is taken away and all the two hop neighbours, covered by at least one host and the willingness of the host is smaller than WILL_ALWAYS, then the host may be removed.

The possible improvements of this algorithm are needed, for example, when there are multiple possible interface addresses for one host [2]. The finding the optimum MPR set for the two hop neighbour coverage is considered to be an NP problem based on [9, 10, 12, 13].

3.2.4 Topology Information

In order to exchange the topological information and build the topology information base the host that were selected as MPR need to sent the topology control (TC) message. The TC messages are broadcasted throughout the network and only MPR are allowed to forward TC messages. The TC messages are generated and broadcasted periodically in the network. [2]

The TC message is sent by a host in order to advertise own links in the network. The host must send at least the links of its MPR selector set. The TC message includes the own set of advertised links and the sequence number of each message. The sequence number is used to avoid loops of the messages and for indicating the freshness of the message, so if the host gets a message with the smaller sequence number it must discard the message without any updates. The host must increment the sequence number when the links are removed from the TC message and also it should increment the sequence number when the links are added to the message. The sequence numbers are wrapped around. When the hosts advertised links set becomes empty, it should still send empty TC messages for specified amount of time, in order to invalidate previous TC messages. This should stop sending the TC messages until it has again some information to send. [2, 8, 11, 9]

The size of the TC message can be quite big, so the TC message can be sent in parts, but then the receiver must combine all parts during some specified amount of time. Host can increase its transmission rate to become more sensible to the possible link failures. When the change in the MPR Selector set is noticed, it indicates that the link failure has happened and the host must transmit the new TC message as soon as possible.[2]

3.2.5 Routing Table Calculations

The host maintains the routing table, the routing table entries have following information: destination address, next address, number of hops to the destination and local interface address. Next address indicates the next hop host. The information is got from the topological set (from the TC messages) and from the local link information base (from the Hello messages). So if any changes occur in these sets, then the routing table is recalculated. Because this is proactive protocol then the routing table must have routes for all available hosts in the network. The information about broken links or partially known links is not stored in the routing table. [2, 8, 3]

The routing table is changed if the changes occur in the following cases: neighbour link appear or disappear, two hops neighbour is created or removed, topological link is appeared or lost or when the multiple interface association information changes. But the update of this information does not lead to the sending of the messages into the network. For finding the routes for the routing table entry the shortest path algorithm is used. [2, 8, 3]

3.3 Advantages

OLSR is also a flat routing protocol, it does not need central administrative system to handle its routing process. The proactive characteristic of the protocol provides that the protocol has all the routing information to all participated hosts in the network. However, as a drawback OLSR protocol needs that each host periodic sends the updated topology information throughout the entire network, this increase the protocols bandwidth usage. But the flooding is minimised by the MPRs, which are only allowed to forward the topological messages.

The reactiveness to the topological changes can be adjusted by changing the time interval for broadcasting the Hello messages. It increases the protocols suitability for ad hoc network with the rapid changes of the source and destinations pairs. Also the OLSR protocol does not require

that the link is reliable for the control messages, since the messages are sent periodically and the delivery does not have to be sequential. [3, 5]

Due to the OLSR routing protocol simplicity in using interfaces, it is easy to integrate the routing protocol in the existing operating systems, without changing the format of the header of the IP messages. The protocol only interacts with the host's Routing Table. [3, 5]

OLSR protocol is well suited for the application which does not allow the long delays in the transmission of the data packets. The best working environment for OLSR protocol is a dense network, where the most communication is concentrated between a large number of nodes. [8]

OLSR has also extensions to allow for hosts to have multiple OLSR interface addresses and provide the external routing information giving the possibility for routing to the external addresses [2]. Based on this information there is possibility to have hosts in the ad hoc network which can act as gateways to another possible network.

4 Comparison of the Protocols

4.1 Performance and Scalability

As proactive protocol, OLSR reduce the control overhead forcing the MPR to propagate the updates of the link state, also the efficiency is gained compared to classical link state protocol when the selected MPR set is as small as possible. But the drawback of this is that it must maintain the routing table for all the possible routes, so there is no difference in small networks, but when the number of the mobile hosts increase, then the overhead from the control messages is also increasing. This constrains the scalability of the OLSR protocol. The OLSR protocol work most efficiently in the dense networks.

The overhead of reactive protocols like AODV is related mostly to the discovery of the new route and from the updates of the usable routes. So in the network with light traffic and low mobility the reactive protocols scales perfectly to the larger networks with low bandwidth and storage overhead. As the undesirable environment for reactive protocols is the network with heavy traffic with large number of destinations with high mobility. This situation will result that a big number of routes will break resulting repeated route discoveries and error reports in the network.

From the information above it is obvious that proactive protocols produce higher routing efficiency than reactive protocols in the network with scattered traffic. Because the updates come from periodic updates and no additional overhead occurs for finding new routes, but then the proactive protocols use more bandwidth and resources than reactive protocols. Thus, the proactive protocols cannot be used in resource critical solutions. The AODV protocol need to discover the route first in order to send the actual data, so the search latency affects of the AODV protocol, OLSR does not need to do the extra work for the discovery of the route so it provides low single packet transmission latency. The reactivity of the detecting topological changes in OLSR can be improved by shortening the time interval of periodic control messages. The OLSR drawback is that it use constantly the bandwidth but AODV is trying to keep the bandwidth usage low for the maintaining of the routes. [4, 5]

The one great advantage of the OLSR protocol is that it immediately knows the status of the link and it is possibly to extend the quality of service information to such protocol so that the hosts know in advantage the quality of the route, this feature is completely impossible in AODV, because of it reactiveness. Extending the OLSR protocol the quality of service feature will result additional latency and overhead. [3, 4]

The paper [14] presented the performance of the common mobile ad hoc network protocols. They used four hours simulation with 19 mobile nodes and a base station with different generated traffic categories to measure the qualitative and quantitative metrics. They used the default parameter settings for each protocol. The conclusion of the paper was that the AODV protocol performed the best, with slight advantage in overall throughput and lower overall delay per packet. OLSR showed good performance with the constantly changing hosts, so that the network structure is always changing.

Another paper [6] presented a framework for wireless ad hoc routing protocols based on the concept of a relay node set. Using this framework the paper presents an analytical model for comparing the overhead of AODV and OLSR protocols. The analytical model of AODV protocol in the framework showed that the AODV protocol may suffer large overhead when establishing the routes in the network with high mobility and retransmission of the packets in the poor communication environment. In the case with OLSR it showed that the overhead is independent of the traffic profiles, so it has the fixed upper bound for the overhead in a network regardless to the network's traffic.

For the summary of this section the AODV protocol performs better in networks with static traffic and OLSR has advantage in networks with high density and highly sporadic traffic. But their scalability is limited when network size increases. In the case of the AODV protocol the huge flooding of packets occur for the search of the routes. In the case of the OLSR protocol the routing table size grows nonlinearly and the control messages can block the actual data packets.

4.2 Resource Usage

The storage complexity of the OLSR protocols is related on how much hosts are in the network, but the storage

complexity of AODV is related to the number of the communication pairs [4]. It is because the OLSR has to have all possible routes in Routing Table, while for AODV the active routes are necessary. In the addition, the OLSR must keep the topology information in the topology set, MPR information in MPR selector set and also update the state information about the links and neighbours [5]. So the OLSR must maintain the information about the hosts that it does not need.

The function for periodic maintainability of the routes consumes a lot of resources. In the AODV it is done by periodic Hello messages and in the OLSR by TC messages. Based on the document [5] The AODV protocol tries to minimise this traffic by making only the hosts that participates in the communication to periodically send Hello messages with the hop limitation of the one hop and OLSR tries to minimise this flooding allowing only MPR to broadcasting these messages through the network. But in addition to this the OLSR protocol also uses the Hello messages for maintaining the neighbour's status. Following sentence is taken from the [5] document from the AODV's advantage section: "Although each node sends out periodic Hello message to monitor connectivity, it is limited and the size of the control message is smaller than those used by OLSR, hence using less bandwidth for route maintenance.". Also another document [3] concludes that one of the disadvantages of OLSR is that it needs more bandwidth and energy resources.

From the information above it is quite obvious that the OLSR spend more resources than AODV in such cases where the environment is suitable for the protocols. But because the core architecture of the protocol is completely different, the resource usage mostly depends on the network suitability of the protocols.

4.3 Security Considerations

Both protocols RFC pages [1] and [2] state that the protocols do not specify any special security measurements, but there are recommendations how the security could be done.

The main points in the AODV and OLSR protocols is that the control messages must be protected, that the malicious information sent by some attacking host could not affect the routing processes in the network. Both protocols should use the IPsec authentication headers for the authentication of the hosts. The AODV needs less protection of the control messages it is enough to protect the RREP and RRER messages in order for the protocol to be secured, but in the case of OLSR all the control messages are needed to be secured. If the OLSR includes gateways hosts, then they have to be statically configured in order to advertise the routes to the valid addresses into the ad hoc network. Based on this information it is obvious that the AODV is more flexible to security solutions, because not all the AODV control messages are in need of the protection, so it can save the resource usage of the AODV protocol compared to OLSR.

The protection of the network from the other hosts can be done by encrypting all messages with some public key cryptography. However, there were not any issues about denial of service attack, because it seems impossible task to implement in such networks.

5 Conclusion

In this paper the characteristic of the ad hoc network were introduced and was explained how does it differs from the original fixed wired network. The characterization was given for the ad hoc routing protocols. Possible metrics to measure the performance and suitability of ad hoc routing protocols were given basing on the RFC paper [7].

AODV and OLSR protocols were introduced and their core architecture was described. The basic actions related to the routing process were studied in details. Also the advantages of the protocols based on their routing processes were given in the end of the chapters.

The comparison chapter were made from the possible protocols advantages and from the found literature related to these protocols. Also the chapter included some results from the papers which compared the following protocols.

The AODV protocol will perform better in the networks with static traffic with the number of source and destination pairs is relatively small for each host. It uses fewer resources than OLSR, because the control messages size is kept small requiring less bandwidth for maintaining the routes and the route table is kept small reducing the computational power. The AODV protocol can be used in resource critical environments.

The OLSR protocol is more efficient in networks with high density and highly sporadic traffic. But the best situation is when the between a large number of hosts. The quality metrics are easy to expand to the current protocol. OLSR requires that it continuously have some bandwidth in order to receive the topology updates messages.

Both protocols scalability is restricted due to their proactive or reactive characteristic. In the AODV protocol it is the flooding overhead in the high mobility networks. In the OLSR protocol is the size of the routing table and topological updates messages. Also the security of the protocols is yet undone, but because the AODV spend less resources the more cryptographically resource demanding solution can be chosen for this protocol. The scalability of these protocols is quite good and their performance depend a lot from the network environment.

References

[1] C. Perkins, E. Belding-Royer and S. Das "Ad hoc On-Demand Distance Vector (AODV) Routing." RFC 3561, IETF Network Working Group, July 2003.

[2] T. Clausen and P. Jacquet "Optimized Link State Routing Protocol (OLSR)." RFC 3626, IETF Network Working Group, October 2003.

[3] Ying Ge, Thomas Kunz and Louise Lamont "Quality of Service Routing in Ad-Hoc Networks Using OLSR." Proceeding of the 36th Hawaii International Conference on System Science(HICSS'03)

[4] Xiaoyan Hong, Kaixin Xu and Mario Gerla "Scalable Routing Protocols for Mobile Ad Hoc Networks." Computer Science Department, University of California, Los Angeles, August 2002.

[5] Koey Huishan, Chua Huimin and Koh Yeow Nam "Routing Protocols in Ad hoc Wireless Networks." National University of Singapore.

[6] Toa Lin, Scott F. Midkiff and Jahng S. Park "A Framework for Wireless Ad Hoc Routing Protocols." Bradley Department of Electrical and Computer Engineering. Virginia Polytechnic Institute and State University. Blacksburg Virginia. 2003

[7] S. Corson and J. Macker "Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations." RFC 2501, IETF Network Working Group, January 1999.

[8] P.Jacquet, P. Mühlethaler, T Clausen, A. Laouiti, A. Qayyum and L. Viennot "Optimized Link State Protocol for Ad Hoc Networks." IEEE INMIC Pakistan 2001.

[9] A. Laouti, P. Mühlethaler, A. Najid and E. Plakoo "Simulation Results of the OLSR Routing Protocol for Wireless Network." 1st Mediterranean Ad-Hoc Networks workshop (Med-Hoc-Net). Sardegna, Italy 2002.

[10] P. Jacquet, A. Laouiti, P. Minet and L. Viennot "Performance of multipoint relaying in ad hoc mobile routing protocols." Networking 2002. Pise(Italy)2002.

[11] T.H. Clausen, G. Hansen, L. Christensen and G. Behrmann "The Optimized Link State Routing Protocol, Evaluation through Experiments and Simulation." IEEE Symposium on "Wireless Personal Mobile Communications". September 2001.

[12] A. Laouiti, A. Qayyum and L. Viennot "Multipoint Relaying for Flooding Broadcast Messages in Mobile Wireless Networks." 35th Annual Hawaii International Conference on System Sciences (HICSS'2002)

[13] P.Jacquet, A. Laouiti, P. Minet and L. Viennot "Performance Analysis of OLSR Multipoint Relay Flooding in Two Ad Hoc Wireless Network Models." Research Report-4260. INRIA, September 2001. RSRCP journal special issue on Mobility and Internet.

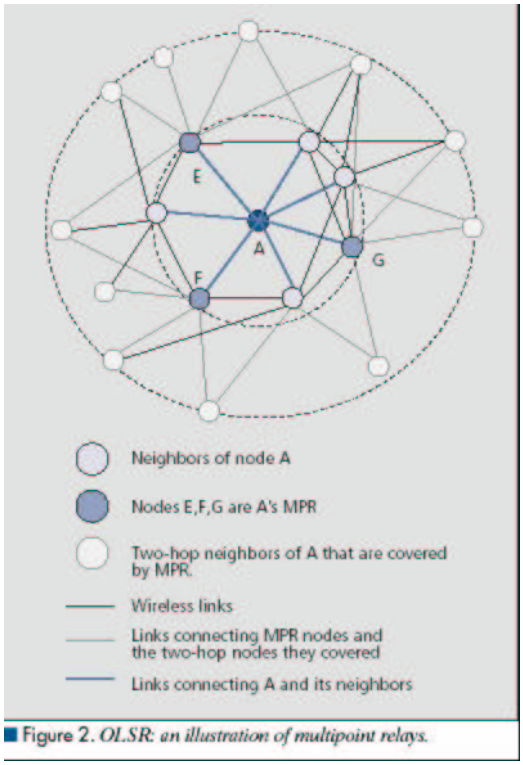

Figure 1: The basic consepts of Multipoint Relays [4]

[14] Julian Hsu, Sameer Bhatia, Mineo Takai, Rajive Bagrodia and Michael J. Acriche. "Performance of Mobile Ad Hoc Networking Routing Protocols in Realistic Scenarios."