Comparative Study of CBR and TCP Performance of MANET Routing Protocols

Thomas Clausen, Philippe Jacquet, Laurent Viennot

Comparative study of CBR and TCP performance of MANET routing protocols

Thomas Clausen*, Philippe Jacquet and Laurent Viennot
T.Clausen@computer.org,{philippe.jacquet,laurent.viennot}@inria.fr

Project HiPERCOM - INRIA Rocquencourt
B.P. 105, 78153 Le Chesnay Cedex, FRANCE
Phone: +33 1 3963 5263, Fax: +33 1 3963 5566

Technical subject area: Wireless Ad-hoc Networks, performance evaluation, simulation, TCP, CBR.

Abstract

In this paper, we evaluate the performance of two MANET routing protocols under varying traffic, density and mobility conditions. We observe, that a rather large fraction of the traffic being carried on the Internet today carries TCP. Thus, Internet traffic has inheritly different characteristics than that of CBR traffic, which is the commonly used traffic type for evaluating MANET routing protocol performance. Hence, in this paper, we extend our evaluations of the two protocols to include the performance of both TCP and CBR traffic. We find, that testing a protocol using CBR traffic is not a good indicator for the same protocols performance when subject to TCP traffic.

1 Introduction

In this paper, we describe the Optimized Link State Routing Protocol (OLSR) [14],[19],[15], a proactive routing protocol for Mobile Ad-hoc NETworks (MANETs). We evaluate its performance through exhaustive simulations using the Network Simulator 2 (ns2) [10], and compare with the Ad-hoc On-Demand Distance Vector (AODV) [13] routing protocol.

The scenarios, in which we compare the two protocols, span a wide range of mobility, density and traffic patterns. In particular, we include studies of the performance of the two protocols when subject to both constant bit rate (CBR) traffic and TCP-traffic. While CBR traffic, due to its non-conforming nature, is useful for stress-testing a network, approximately 95% of the traffic on the Internet today carries TCP [5], [8]. Hence, it is appropriate to study how well the different routing protocols support TCP.

1.1 Mobile Ad Hoc Networks

A mobile ad-hoc network (MANET) is a collection of nodes, connected by wireless links which are able to connect on a wireless medium forming an arbitrary and dynamic graph. That is, over time links between nodes may change, nodes may disappear and new nodes appear in the network. The physical size of a MANET is expected to be larger than the radio range of the wireless interfaces. Thus for any two nodes in the network to be able to communicate, multi-hop routing is necessary. A challenge for a routing protocol for MANETs is thus the ability to respond quickly to a high degree of topological changes in the network and still maintain routes, while at the same time to not swamp the network with excessive control traffic.

Other than acting as stand-alone networks, MANETs may find use as "edge network": connecting a cloud of mobile nodes to e.g. the global Internet or another wired and engineered infrastructure. Another use includes connecting separated wired infrastructures through an ad-hoc network infrastructure. Thus a MANET should be able to support the

*May also be contacted at: Mindpass Center for Distributed Systems, Department of Computer Science, Aalborg University, Fredrik Bajers Vej 7E, 9220 Aalborg Ø, Denmark

same types of traffic as are present in wired networks.

A large fraction of published performance studies of MANET routing protocols, including [7, 4, 2, 3, 9, 19] protocols have emphasized or focused solely on comparing or evaluating protocols based on CBR traffic. However observing that approximately 95% of the traffic on the Internet today carries TCP [5], [8], we find that it is appropriate to study how well the different routing protocols support TCP.

1.2 Paper Outline

The remainder of this paper will be organized as follows: in section 2, we describe the OLSR and AODV routing protocols in some detail. Following, in section 3, we characterize TCP and CBR traffic types, used for the simulations. In section 4, we describe the metrics we use for evaluating the protocols and section 5 presents our simulation scenarios and results. Finally, the paper is concluded in section 6.

2 Routing protocols for MANETS

Several protocols exist, addressing the problems of routing in mobile ad-hoc networks. Such protocols are, traditionally, divided into two classes, depending on when a node acquires a route to a destination. *Reactive*, demand-driven protocols are characterized by nodes acquiring and maintaining routes on-demand. In general, when a route to an unknown destination is required by a node, a query is flooded onto the network and replies, containing possible routes to the destination, are returned. Examples of reactive protocols include the "Ad Hoc On Demand Distance Vector Routing Protocol" (AODV) [13] and "Dynamic Source Routing" (DSR) [6].

Proactive, table-driven protocols are characterized by all nodes maintaining routes to all destinations in the network at all times. Thus using a proactive protocol, a node is immediately able to route (or drop) a packet based on information already present in the nodes routing-table. Examples of proactive protocols include the "Topology Broadcast based on Reverse-Path Forwarding" routing protocol (TBRPF) [12] and the "Optimized Link State Routing Protocol" (OLSR) [14].

OLSR and AODV thus present two radically different approaches to routing in MANETs. OLSR is a proactive, link-state routing protocol, employing periodic message exchange to update topological information in each node in the network. AODV is a reactive on-demand routing protocols: route information is maintained only as needed through a request-reply cycle. This implies different overhead and performance profiles: [16] shows, that in terms of control traffic overhead, proactive protocols have an advantage in high-traffic networks, whereas in networks with little traffic and a high degree of mobility, reactive protocols are of preference. [17] confirms the findings of [16] and further shows that the on-demand discovery of routes in reactive protocols yield longer packet delivery delays than what is experienced with proactive protocols.

2.1 The Optimized Link-State Routing Protocol

The Optimized Link-State Routing Protocol (OLSR) is a proactive link-state routing protocol, employing periodic message exchange to update topological information in each node in the network. While having some commonalities with OSPF [11], OLSR is specifically designed to operate in the context of MANETs, i.e. in bandwidth-constrained, dynamic networks.

Conceptually, OLSR contains three generic elements: a mechanism for neighbor sensing, a mechanism for efficient diffusion of control traffic, and a mechanism for selecting and diffusing sufficient topological information in the network in order to provide optimal routes.

Neighbor sensing works based on periodic exchange of HELLO messages, through which a node may acquire topological information up to two hops away. This is utilized by each node to, individually, construct a Multipoint Relay set (MPR set) from among its neighbors. A node must select MPRs in a way such that a message emitted, and retransmitted by all MPRs, is received by all two-hop neighbors of that node. As illustrated in figure 1b,"careful" selection of MPRs (the filled nodes) may greatly reduce duplicate retransmissions.

The aim of the MPR selection is to device an efficient way of broadcasting information from one node to all other nodes in the MANET. We denote this mechanism *MPR Flooding*. [1] presents an analysis of MPR selection algorithms, and [20] presents simulation comparisons between MPR flooding and other flooding mechanisms.

Each node, selected by any other node as MPR, emits periodically a TC message containing a list of its MPR selectors[1]. This message is distributed to all nodes in the network through MPR flooding. Thus, all nodes receive partial topological information, describing all nodes in the network and a subset

[1] The MPR selectors of node a is the nodes, which have selected node a as MPR.

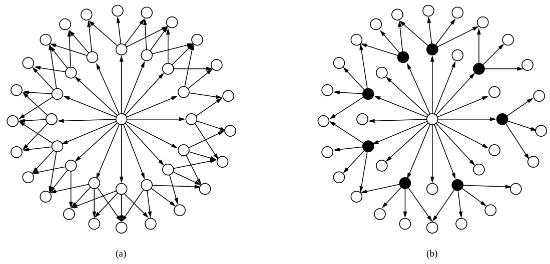

Figure 1: Example of pure flooding (a) and diffusion using Multipoint Relays (b). The source of the message is the node in the center. Each arrow pointing to a node, indicates that a node receives a copy of the message. The filled nodes are selected by the center node as Multipoint Relay.

of the links in the network[2].

It is important to emphasize that OLSR provides optimal routes (in terms of number of hops) as well as bi-directionality (i.e. if there exists a route from node a to node b, then there exists a route from node b to node a as well - although this route may not transverse the same links).

2.2 The Ad-hoc On-demand Distance Vector Routing Protocol

The common element in reactive protocols is the mechanism used for discovering routes. The source node emits a request message, requesting a route to the destination node. This message is flooded, i.e. relayed by all nodes in the network, until it reaches the destination. The path followed by the request message is recorded in the message, and returned to the sender by the destination, or by intermediate nodes with sufficient topological information, in a reply message. Thus multiple reply messages may result, yielding multiple paths - of which the shortest is to be used.

In the Ad Hoc On-Demand Distance Vector protocol (AODV), when a source requires a path to the destination, a *route request* message is flooded in the network. Upon receiving such a message, a node examines its local route-cache to check if a fresh route to the required destination is available. If so, the node unicasts a *route reply* message to the source with information about the route. Otherwise, the *route request* is retransmitted using a pure flooding mechanism with local duplicate elimination. As an optimization, AODV employs an "expanding ring" flooding, where a *route request* is issued with a limited TTL. If no *route reply* message is received within a certain time, the message is issued again with a larger TTL. If still no reply, the TTL is increased in steps, until a certain maximum value.

While this route discovery is performed, any IP-packets to the destination are buffered in the source node. When a route is established, the packets are transmitted. If no route can be established, the packets are dropped.

When a link is detected to be broken (either through a neighbor discovery protocol, as in OLSR, or through a link-layer notification), the detecting node issues a *route error* message to those neighbors who have been using a route over the now broken link. These nodes will then have to issue new *route requests* to repair the broken routes.

3 Data Traffic Types

As indicated in section 1, the traffic carried over a MANET may have different characteristics. In this section, we describe two types of traffic: constant bit-rate traffic, traditionally used for stress-testing networks, and TCP-traffic. We keep the descriptions brief, and aim at exposing those characteristics that are of importance when comparing MANET performances.

3.1 Constant Bit Rate (CBR) Traffic

"Constant Bit Rate" traffic is a terminology borrowed from the ATM world. It implies that data are sent at a fixed bit rate – a CBR stream is thus characterized by data being sent in packets of a fixed size with a fixed interval between each packets. The source of a CBR stream makes no attempt to detect if the destination receives the transmitted data – or even discovering if the destination exists. I.e. no connection establishment phase is required and traffic is flowing only from the source to the destination with no feedback from the destination or from intermediate nodes.

3.2 TCP Traffic

Contrary to CBR traffic, TCP is a connection oriented, reliable and conforming transport protocol. I.e., prior to transmitting data, a connection establishment phase must be completed, denoted a *three-way handshake*. During transfer, TCP employs both *flow control* and *congestion control*. The purpose of flow control is to avoid overloading the recipient, while congestion control is employed to shape the traffic such that it conforms to the available network capacity. Positive acknowledgments, timeouts

[2] Since all nodes in a multi-hop network are required to select a non-empty MPR set, all nodes will be advertised.

and retransmissions are employed to ensure reliable data delivery.

A source for TCP data will maintain two "windows": a "receive window" for each destination, representing the available buffer capacity of the destination, and a "congestion window", representing the available capacity of the network. As the source transmits data, the size of both windows are reduced by an amount equal to the size of the data sent. When either window reaches zero, the source is not allowed to transmit.

The receive window is, initially, set to a value negotiated with the destination during the connection establishment phase. For each byte of data sent, the window size is reduced by one byte. When acknowledgments are received from the destination, the window size is increased accordingly: for each byte of data acknowledged, the window size is increased by one byte. I.e. the transfer rate by the sender is controlled by the capacity of the destination.

The congestion window is maintained in two phases, denoted *slow start* and *congestion avoidance* respectively. During the slow start phase, TCP starts with a very low data transfer rate. Indeed, the congestion window is initially set to the maximum size of one TCP segment, allowing exactly one TCP segment to be transmitted. In the slow-start phase, if an acknowledgment is received before its timeout expired, the congestion window is doubled. I.e. the congestion window grows exponentially.

When the congestion window has reached a specific threshold, the slow-start phase ends and is replaced by the congestion avoidance phase. During congestion avoidance, each timely acknowledged TCP segment causes the congestion window to grow by one. I.e. during congestion avoidance, the growth of the congestion window is linear.

If an acknowledgment is not received before it is timed out, TCP retransmits the data (to ensure reliable delivery). Further, the abcense of an acknowledgment is taken as an indication of the network being congested. To accommodate for this congestion, the congestion window is reset to the maximum size of one TCP segment, and the threshold for going between slow-start and congestion avoidance is set to the current size of the congestion window, divided by two.

3.3 CBR, TCP and MANETs

CBR and TCP traffic impose different conditions on MANETs. In this subsection, we will try to outline the impacts of some of these conditions.

First, we observe that for TCP, both during connection establishment and data transfer, bidirectional traffic between the source and the destination is required in order for data to be successfully delivered. With CBR traffic, traffic from the source to the destination is sufficient. This implies that, for TCP traffic, it is required that the routing protocol maintains effectively two routes for each connection – whereas for CBR, only one route is required per stream of data.

Second, we observe that a long delay of an acknowledgment is interpreted similarly to an abcense of acknowledgment – as an indication of network congestion. If the topology changes in a MANET, a reactive protocol may have to execute a renewed route discovery in order to acquire a new route. The delay involved in this can be long enough to cause an acknowledgment to be delayed and, hence, time out in the source. I.e. a topological change in an otherwise not congested network will be treated as if the network was congested, and the sender will have its transmission rate drastically decreased.

4 Metrics

In order to compare and evaluate the protocols, this section presents the metrics we use for representing the performance of the protocols.

control traffic overhead
The control traffic overhead, measured in bytes per second, represents the amount of routing protocol specific traffic in the network. The control traffic overhead is measured in number of bytes transmitted, including UDP and IP headers. This metric represents one component of the "cost" of employing a routing protocol.

delivery ratio (for CBR)
The delivery ratio represents the fraction of data traffic that is successfully received at the intended destination, relative to the total amount of transmitted data traffic. For CBR traffic, this metric is useful as a "success measure": a high delivery ratio, close to 1, implies that a high ratio of transmitted data are actually delivered to the intended recipients.

path length
The path length, measured in number of hops, represents the average length of a path a packet takes from the source to the destination. Suboptimal routes represent another component of the "cost" of employing a routing protocol[18].

delay
The delay, measured in milliseconds, is the number of milliseconds elapsing from a packet is

sent from the source and until it is received by the recipient. This is linked to the metric of path length (longer paths, in general, introduce longer delays), however is also affected by the size of the queues in intermediate nodes (i.e. also dependent on both the control traffic overhead and on other traffic in the network). The delay includes all possible delays caused by buffering, during route discovery latency, processing to determine the path using QoS values during data transmission, queuing at the interface queue, retransmission delays at the MAC, and propagation and transfer times.

total transfer time (for TCP)
The total transfer time for TCP represents the time it takes from the first packet in the TCP connection is transmitted by the source and until the last packet is received by the destination.

normalized routing load (for TCP)
The normalized routing load expresses the ratio of bytes transmitted in order to ensure that one data byte is successfully delivered at the destination. TCP employs retransmissions and acknowledgments in order to ensure the reliability of data. Hence, for TCP traffic this metric is useful as "success measure": a low normalized routing, preferably close to 1, implies efficiency in delivering a data packet to the intended recipient.

5 Simulations

In this section, we present our simulations of OLSR and AODV under different scenarios and with different traffic characteristics.

For each sample point, we specify an abstract description of the scenario. This description is fed through an automatic scenario generator to produce 30 different scenarios, conforming to the same abstract description. We present the mean taken over the 30 different scenarios, and emphasize, that the set of 30 scenarios per sample point are the same for both of the tested protocols. I.e. for a given sample point, OLSR and AODV are tested with the same 30 scenarios. By using randomized scenario generation, and by running a large number of scenarios for each sample point, we eliminate any possible bias that might come from a specific instance of a scenario favoring either of the protocols.

5.1 Simulation parameters

Our simulations are conducted with a basic set of parameters, shown in table 1, describing the nodes and their mobility in the network.

Parameter	Value
Number of nodes:	50
Field size:	1000 x 1000 m^2
Simulation time:	250 s
Node speed	1-5 $\frac{m}{s}$
Node rest time	0-5 s
Node distance	1000 m

Table 1: Basic simulation parameters describing the network topology

The mobility model employed is the "random waypoint" model [2].

Further, the basic parameters describing the traffic pattern for CBR and TCP traffic are shown in table 2 and table 3, respectively.

Parameter	Value
Streams	25
Packet size	64 bytes
Packet interval:	0.1 s
Stream duration:	10 s

Table 2: Basic simulation parameters describing the CBR traffic characteristics

Parameter	Value
No. transfers	25
Transfer Amount	16384 bytes

Table 3: Basic simulation parameters describing the TCP traffic characteristics

We note that these are basic parameters. I.e., in our simulations we choose to study the impact of tuning one of these parameters while keeping the remaining constant.

Thus, we will in the following subsections test the protocols behavior under the following scenarios:

varying traffic
the parameter *streams* (for CBR) and *no transfers* (for TCP) is varied from 10 to 140

varying mobility
the parameter *node speed* is varied from 0 to 20 $\frac{m}{s}$

5.2 CBR traffic simulations

In figure 2 and figure 3, we present the traffic delivery ratio using AODV and OLSR under various mobility and traffic scenarios.

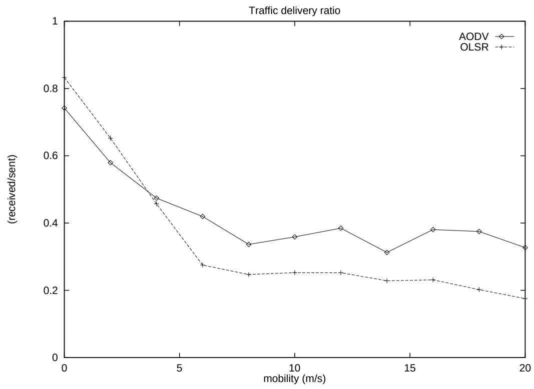

Figure 2: *Data packet delivery ratio with varying mobility.*

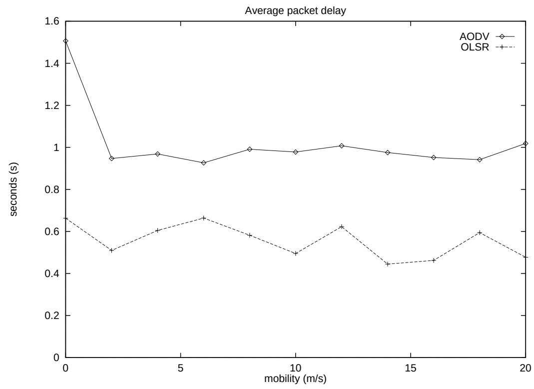

Figure 4: *Packet delays with varying mobility.*

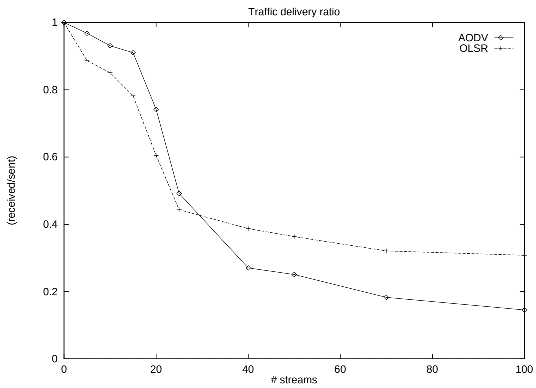

Figure 3: *Data packet delivery ratio with number of traffic streams.*

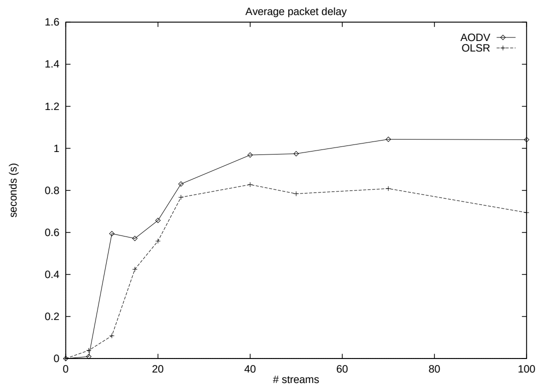

Figure 5: *Packet delays with varying number of traffic streams.*

We observe that for low mobility, OLSR performs slightly better than AODV, while for mobility rates higher than 4 $\frac{m}{s}$, AODV has an advantage. For high traffic loads, figure 3 shows, that OLSR yields a slightly lower delivery rate than AODV for a low number of traffic streams, while for high traffic rates, above 30 concurrent streams, OLSR yields a delivery rate of approximately 40% - twice that of AODV.

Figure 4 and figure 5 present the average packet delay, i.e. the delay from a packet has been transmitted until it is received. We observe that OLSR consistently presents the lowest packet delay, regardless of traffic and mobility patterns. Since the delay measurement only takes those packets that are successfully delivered into account, this result must be correlated with the delivery rate of the protocols. We observe, that even in situations where OLSR yields the highest delivery rate, it still yields the lowest delay.

Figure 6 and figure 7 present the control traffic overhead, resulting from scenarios with varying mobility and traffic, respectively. We observe, that the tested version of OLSR does not react explicitly to link-breaks, and thus that the control traffic overhead consistently is just above 3000 bytes/second. Except for very static networks or networks with very few (less than 6) communicating pairs, this is significantly lower than the overhead exposed by AODV.

In figure 8 and figure 9, we present the average path lengths for successfully delivered data packets, obtained using AODV and OLSR in scenarios with a varying mobility and traffic patterns respectively.

We observe that OLSR consistently provides shorter paths than AODV. Again, this measure only takes successfully delivered packets into account and, hence, must be correlated with the delivery rate. We observe from figure 8 that the path length of OLSR becomes < 1 at about 10 $\frac{m}{s}$ - corresponding with the fact that figure 6 shows the delivery rate of OLSR to be lower than that of OLSR. Conversely, we also observe that for the situations where OLSR yields the highest delivery rate, the paths over which the data are delivered remains shorter than the path lengths provided by OLSR.

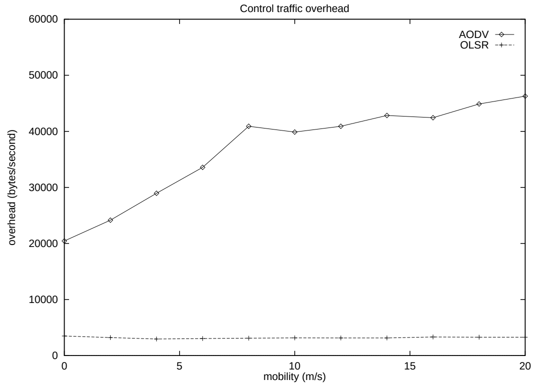

Figure 6: *Total control traffic overhead with varying mobility.*

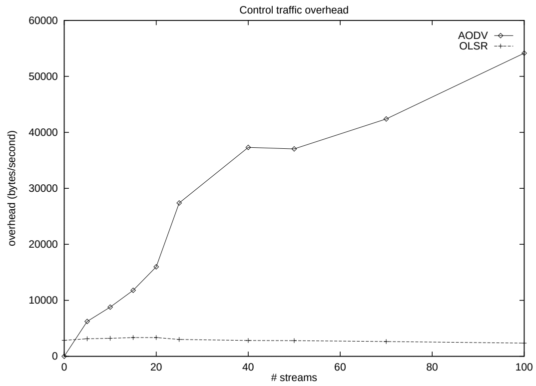

Figure 7: *Total control traffic overhead with varying number of traffic streams.*

5.3 TCP traffic simulations

Our observations from the CBR traffic simulations are, that while the two protocols definitely exhibit different performance characteristics, the general tendency for the scenarios simulated is the same: at their peak, both protocols deliver a delivery rate of 80-100 % - which falls to and stabilizes at a level.

Comparing figure 7, showing the control traffic overhead with varying number of CBR streams with figure 10, showing the control traffic overhead with varying number of TCP streams, we notice that the overhead exposed by OLSR is the same, regardless of TCP and CBR traffic. For AODV, we observe that the control traffic overhead is approximately 20% higher with TCP traffic than with CBR traffic. Since TCP requires traffic to flow in both directions, twice as many routes are effectively required as compared to CBR traffic. Thus in the worst case, AODV would for TCP generate twice as much overhead as it would for CBR. This is not the actual case in the scenarios presented since the lifetime of routes is sufficiently long to allow routes to be reused and

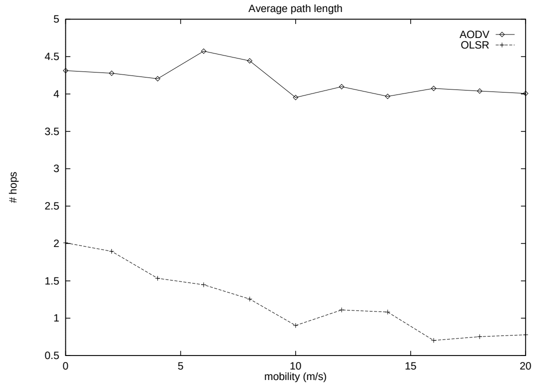

Figure 8: *Average path length with varying mobility.*

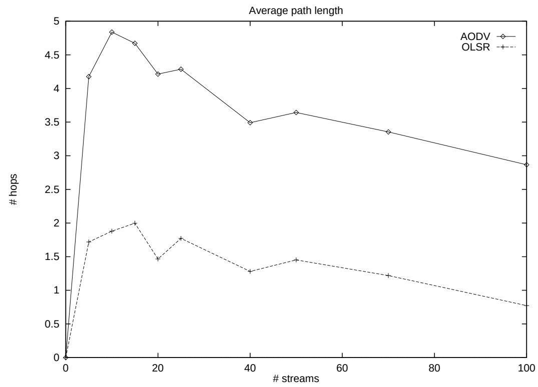

Figure 9: *Average path length with varying number of traffic streams.*

since AODV allows nodes to cache routes which they overhear.

Table 4 shows the average time required to transfer a 16Kb using TCP. The figures presented show the average over 30 runs, each of which had 25 concurrent TCP streams, yielding 750 streams total. Table 5 shows the same, although for transferring 160Kb of data.

We observe, that with AODV, 18% longer time is required to complete the 16Kb data transfer, than with OLSR. For 160Kb, 24% longer time is required with AODV.

The longer time is explained by the congestion control mechanism of TCP: if node mobility changes the topology such that an active route is broken in AODV, packets are buffered while an alternative route is discovered. Likewise, the acknowl-

	OLSR	AODV
Average TCP transfer time:	3.12 s	3.68 s
Std. deviation	0.66	1.16

Table 4: Transfer time for 16Kb of data using TCP

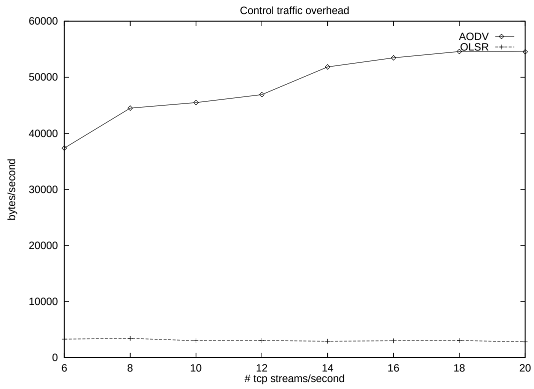

Figure 10: *Control traffic overhead with varying number of TCP streams.*

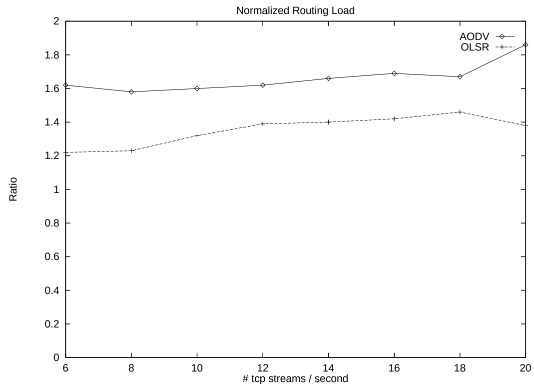

Figure 11: *Normalized routing load with varying number of TCP streams.*

	OLSR	AODV
Average TCP transfer time:	5.12 s	6.34 s
Std. deviation	0.63	1.56

Table 5: Transfer time for 160Kb of data using TCP

edgment from the data packet will need to transverse the reverse route, which must also be rediscovered/repaired. This delays the TCP acknowledgment, interpreted by the TCP congestion control mechanism as the network being congested and hence restricting the transmission rate by resetting the sender to the "slow start" phase. In OLSR, an alternative route, if such exists in the network, is already available when an existing route is broken,

Figure 11 shows the normalized routing load for OLSR and AODV with varying number of TCP streams. We observe, that the normalized routing load for OLSR is consistently lower than that of AODV. I.e. OLSR is consistently more efficient for delivering TCP traffic than AODV.

6 Conclusion

The simulations reveal that, as expected, the proactive and reactive protocol classes both excel, although in different scenarios: OLSR exhibits better performance in a network with a highly dynamic topology and many changing communicating pairs. Conversely, the overhead generated by AODV is lower than that of OLSR when the network remains mostly static, both in terms of topology and communication patterns.

An interesting observation is that, while the protocols may perform comparatively when exposed to CBR-traffic, the same scenarios, exposed to TCP-traffic, exhibit significantly different performance characteristics. With CBR traffic, for example, AODV yields a better delivery rate than OLSR for low traffic ratios. The same scenarios, but with TCP traffic, yield a quite different results: OLSR achieves a significantly and consistently better normalized routing load, indicating that a MANET with OLSR is better suited for transporting TCP traffic.

The differences between the performance results from CBR and TCP traffic are to be found in the observations than TCP is bi-directional, thus requiring that the routing protocols maintain bidirectional routes in order to operate.

Comparing the time it takes for each of the two protocols to transmit data using TCP, we notice that transfer of 16Kb with AODV takes, on average, 18% longer than with OLSR. For transferring 160Kb of data, 24% longer time is required with AODV. This is due to the congestion control mechanisms in TCP, which interprets a delayed acknowledgment (due to buffering and route discovery in AODV) as a network congestion and forces the to reduce its transmission rate.

Thus we notice that while simulations using CBR-traffic are useful for stress-testing a network, the performance achieved from such simulations is not indicative for the performance of TCP.

We also notice that in the case of TCP-traffic, OLSR as a proactive protocol has an obvious advantage from having bi-directional routes immediately available - and from continuously maintaining such routes.

References

[1] Laurent Viennot Amir Qayyum and Anis Laouiti. Multipoint relaying: An efficient technique for flooding in mobile wireless networks. Technical report, Project HiPERCOM, INRIA

Rocquencourt, 2000. INRIA research report RR-3898.

[2] J. Broch, D.A. Maltz, D.B. Johnson, Y.-C. Hu, and J. Jetcheva. A performance comparison of multi-hop wireless ad hoc network routing protocols. In *MobiCom'98*, October 1998. Dallas.

[3] S. Das, R. Castaneda, and J. Yan. Simulation based performance evaluation of mobile. *ACM/Baltzer Mobile Networks and Applications (MONET)*, pages 179–189, July 2000.

[4] Samir R. Das, Charles E. Perkins, and Elizabeth M. Royer. Performance comparison of two on-demand routing protocols for ad hoc networks. In *Proceedings of the IEEE Conference on Computer Communications (INFOCOM)*, March 2000. Tel Aviv, Israel.

[5] Kevin Jeffay F. Donelson Smith, Felix Hernandez Campos and David Ott. What TCP/IP Protocol Headers Can Tell Us About the Web. In *SIGMETRICS*, October 2001.

[6] J. G. Jetcheva, D. Johnson, D. Maltz, and Y.C. Hu. Dynamic source routing (DSR). Internet Draft, draft-ietf-manet-dsr-06.txt, November 2001, Work in progress.

[7] P. Johansson, T. Larsson, N. Hedman, B. Mielczarek, and M. Degermark. Scenario-based performance analysis of routing protocols for mobile ad hoc networks. In *MobiCom'99*, August 1999. Seattle.

[8] Gregory J. Miller Kevin Thompson and Rick Wilder. Wide-Area Internet Traffic Patterns and Characteristics. *IEEE Network*, November/December 1997.

[9] S.-J. Lee. *Routing and Multicasting Strategies in Wireless Mobile Ad hoc Networks*. PhD thesis, University of California, 2000.

[10] S. McCanne and S. Floyd. ns Network Simulator. Available at http://www.isi.edu/nsnam/ns/.

[11] J. Moy. Ospf version 2. Internet Standard, Request For Comments 2328, April 1998.

[12] Richard G. Ogier, Fred L. Templin, Bhargav Bellur, and Mark G. Lewis. Topology broadcast based on reverse-path forwarding (tbrpf). Internet Draft, draft-ietf-manet-tbrpf-05.txt, March 2002, Work in progress.

[13] C. E. Perkins, E. M. Royer, and S. R. Das. Ad hoc on-demand distance vector (AODV) routing. Internet Draft, draft-ietf-manet-aodv-09.txt, November 2001, Work in progress.

[14] Philippe Jacquet, Paul Muhlethaler, Amir Qayyum, Anis Laouiti, Laurent Viennot and Thomas Clausen. Optimized Link-State Routing Protocol. Technical report, Project HiPERCOM, INRIA Rocquencourt, March 2002. draft-ieft-olsr-06.txt - work in progress.

[15] T. Clausen, P. Jacquet, A. Laouiti, P. Muhlethaler, A. Qayyum and L. Viennot. Optimized link state routing protocol. In *Proceedings of IEEE INMIC, Lahore, Pakistan (INMIC 2001)*. MindPass Center for Distributed Systems, Aalborg University, 5th IEEE National Multi-Topic Conference (INMIC 2001), December 2001.

[16] T. Clausen, P. Jacquet and L. Viennot. Analyzing control traffic overhead in mobile adhoc network protocols versus mobility and data traffic activity. In *Proceeding of The First Annual Mediterranean Ad Hoc Networking Workshop*. MindPass Center for Distributed Systems, Aalborg University and Project Hipercom, INRIA Rocquencourt, The First Annual Mediterranean Ad Hoc Networking Workshop, September 2002.

[17] T. Clausen, P. Jacquet and L. Viennot. Comparative study of routing protocols for mobile ad-hoc networks. In *Proceeding of The First Annual Mediterranean Ad Hoc Networking Workshop*. MindPass Center for Distributed Systems, Aalborg University and Project Hipercom, INRIA Rocquencourt, The First Annual Mediterranean Ad Hoc Networking Workshop, September 2002.

[18] T. Clausen, P. Jacquet and L. Viennot. Optimizing route length in reactive protocols for ad hoc networks. In *Proceeding of The First Annual Mediterranean Ad Hoc Networking Workshop*. MindPass Center for Distributed Systems, Aalborg University and Project Hipercom, INRIA Rocquencourt, The First Annual Mediterranean Ad Hoc Networking Workshop, September 2002.

[19] Thomas Clausen, Gitte Hansen, Lars Christensen and Gerd Behrmann. The optimized link state routing protocol - evaluation through experiments and simulation. In *Proceeding of Wireless Personal Multimedia Communications*. MindPass Center for Distributed Systems, Aalborg University, Fourth International

Symposium on Wireless Personal Multimedia Communications, September 2001.

[20] Thomas Clausen, Tue Olesen and Nicolai Larsen. Investigating data broadcast performance in mobile ad-hoc networks. In *Proceeding of Wireless Personal Multimedia Communications*. MindPass Center for Distributed Systems, Aalborg University and project Hipercom, INRIA Rocquencourt, Fifth International Symposium on Wireless Personal Multimedia Communications, October 2002.