Routing with Transmission Buffer Zones in MANETs

Erlend Larsen, Lars Landmark, Vinh Pham, Øivind Kure
Q2S NTNU
{erl, larslan, vph, okure}@unik.no

Paal E. Engelstad
UniK/UiO
SimTel (Telenor/Simula)
paalee@unik.no

Abstract

Dealing with link breaks in MANETs is a challenge for the routing protocol. This paper proposes a mechanism to reduce the negative impact of link breaks on the routing. The transmission area of a node is divided into a safe zone close to the node and an unsafe zone (i.e. buffer zone) near the end of the transmission range. The probability is high that link breaks occur with neighboring nodes located in the buffer zone, while links to neighboring nodes in the safe zone are expected to be more stable. Thus, neighbors in the safe zone are preferred as relay nodes, while neighbors in the buffer zone are only used if necessary to avoid network partitioning. The main cost of this mechanism is that the mean number of hops between two nodes is higher than without the mechanism, but simulations show that the solution offers increased throughput.

1 Introduction

Mobile ad hoc networks (MANETs) are designed to function without any prior infrastructure in place, making them attractive for use in emergency and military scenarios. However, ad hoc networks are limited in performance due to their nature of distributed wireless communication and often random and rapid topology changes. One way to increase the network performance in MANETs has been by sharing and utilizing information between the network layers (cross-layering), since traditional wired networks – for which the network stack was invented – have not had to deal with the conditions that MANETs have to handle. Due to the faster timings of the lower layers, the routing protocol can for example act much faster in detecting changes in connectivity by using lower layer information, and determine the distance to neighboring nodes using the signal strength of incoming transmissions.

The routing protocol selects a route at lowest cost, and the most widely used cost metric in ad hoc networks is *shortest path*. With this metric, the routing protocol selects a route with a minimal number of hops between a source and a destination. The advantages are effective use of network resources, little delay and little overhead. However, a disadvantage of shortest path routing is that the routing protocol tends to select nodes on the edge of the transmission area as relay nodes, since this normally reduces the number of hops between a source and a destination. The problem is that with node mobility, it is the nodes close to the transmission area edge that have the highest probability to move out of the transmission area. Furthermore, links to nodes near the edge of the transmission range also have a higher probability of bit errors, as the signal strength decreases with distance.

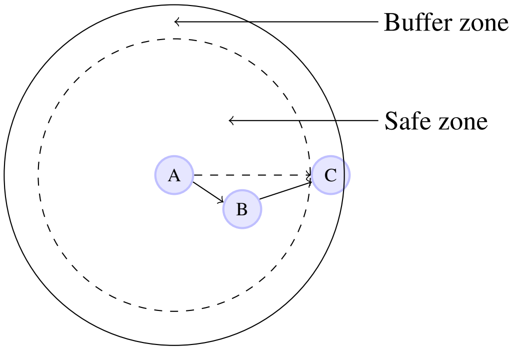

Figure 1. Transmission area zones of node A with safe node (B) and unsafe node (C).

The main contribution of this paper is the proposal of a mechanism to reduce the disadvantages of shortest path routing in MANETs. The key idea is to divide the transmission area of a node into a *safe zone* close to the node and an *unsafe zone* (i.e. *buffer zone*) near the end of the transmission range (Fig. 1). The routing protocol prefers neighbors that are located in the safe zone as relay nodes, while neighbors in the buffer zone are only used if necessary to avoid network partitioning.

The routing protocols for ad hoc networks can be divided into two groups, proactive and reactive. A proactive routing protocol aims to have an updated view of the

978-1-4244-4439-7/09/$25.00 ©2009 IEEE

network with routes to all nodes at any time, while a reactive protocol only establishes routes to the nodes where an application needs to send traffic. This paper focuses on MANETs using a proactive routing protocol.

Although the simulations and evaluations presented in this paper are based on using the Optimized Link State Routing protocol (OLSR) [9] as the proactive routing protocol, the results and analysis presented here should be applicable to the use of other proactive routing protocols as well. Without loss of generality, it is also assumed that IEEE 802.11 [12] is used as the technology at the physical link and link layer. IEEE 802.11 uses Carrier Sense Multiple Access (CSMA) with exponential backoff. Retransmissions occur until the DATA frame is successfully transmitted (i.e. the ACK frame successfully received) or until the retry counter reaches the retry limit upon which the transmitting node discards the DATA frame.

The rest of the paper is structured in the following way. First, Section 2 present some background on how routing protocols deal with link breaks. Then, in Section 3 the devastating effect link breaks have on Medium Access Control (MAC) retransmissions and on the overall throughput is documented through simulation. The simulation setup is presented in this context. Section 4 presents the proposed solution in full. The solution is then evaluated in Section 5 by simulations. Finally, related work is presented in Section 6, before the paper is concluded in Section 7.

2 Background

The normal way of detecting link breaks for a routing protocol is through lost polling packets (i.e. lost Hello packets). The Hello packets of OLSR are transmitted between one-hop neighbors at a specified time frequency (e.g. every 2 seconds, which is the recommended transmission frequency of OLSR) and provide neighborhood connectivity information and a means for link break detection. If no Hello packet from a neighbor is received within a specified time interval (e.g. within 6 seconds, the recommended interval of OLSR), the neighbor is considered unavailable and a link to this neighbor is considered as broken and invalid.

Another way for the routing protocol to detect link breaks is to leave it up to a mechanism implemented at the underlying link layer. The routing protocol must then be notified explicitly about a link break by the link layer. The disadvantage of this *Link Layer Notification* (LLN) approach might be the cost of additional implementation complexity. However, the advantage is that the link layer is normally able to detect link breaks sooner. As a link layer, IEEE 802.11 is normally capable of detecting a link break considerably faster than a second. In contrast, without LLN and with the recommended values of OLSR, a link break will not be detected before 4 seconds at best and 6 seconds at worst. This paper focuses first on link break detection through lost Hello packets, while the use of LLN will be discussed and evaluated by the end of the paper.

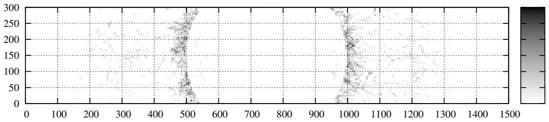

Figure 2. Accumulated transmissions diagram.

It is important for the overall performance to detect the link break in a timely fashion, since two negative effects occur in the period between the physical link break and the detection by the routing protocol. First, the packets queued in the interface queue are marked with an unreachable next hop address. This means that these packets will never reach their destination, and are at this point lost. Second, these packets will be attempted transmitted several times by the MAC layer before they are discarded. This will steal valuable medium time from packets transmitted from other nodes with a valid next hop address.

The retransmission effect is illustrated through a simulation where a node was placed in the center of the simulation area and set up to receive data from 40 nodes moving randomly inside the simulation area at 10 m/s (Fig. 2). In this simulation, a node inside the transmission area of the receiving node successfully sends traffic to the receiving centered node until it moves out of the receiving node's transmission area. At that point a link break occurs, but it is not detected by the transmitting node's routing protocol for another 4-6 seconds. At the time when the link break is detected by the routing protocol, the node may have travelled 40 to 60 m past the edge of the transmission area of the receiving node. During this time the MAC layer will transmit each packet with the receiving node as MAC destination several times. Fig. 2 shows the simulation area with the positions for all occurring transmissions plotted in. A ring of an increased number of transmissions is observed outside the transmission area of the receiving node, a direct effect of link breaks and subsequent retransmissions.

3 Analysis of the effects of link breaks

In this section the normal behavior of OLSR as a proactive routing protocol is investigated. First the simulation setup is presented, and then the results documenting the link detection problem are shown. To simplify the analysis, it is assumed that all link changes occur as a consequence of node mobility, while the radio conditions

Table 1. Simulation parameter settings

Radio-propagation model	TwoRayGround
Interface queue type	FIFO with DropTail and PriQueue
Interface queue size	300 packets
Maximum MAC retries	7
Antenna Model	OmniAntenna
Nominal transmission rate	2 Mbps
Basic rate	1 Mbps
Simulation time	500 s
Random seed	Heuristic
Traffic TTL	32
OLSR Hello interval	2 s
OLSR Hello timeout	6 s
OLSR TC interval	5 s
OLSR TC timeout	15 s

are considered as stable, assuming full radio connectivity within the radio transmission area and no connectivity outside this range.

The simulations were performed using the ns-2 network simulator [2] version 2.31. The Optimized Link State Routing Protocol (OLSR) [9], [1] was used for multi-hop routing and the IEEE 802.11 protocol [12] was used as MAC layer. The nodes were divided into two equally large groups, and each node in each group transmitted packets to all other nodes in that group. The traffic type was UDP at constant bit rate, and the packet size was 64 bytes.

The simulation area was 1500 x 300 m^2 with 40 nodes in all simulations. The dimensions were selected to get a topology with many hops and little partitioning without the need of a very large number of nodes. The mobility model is Random Direction with Reflection with a 10 s travel time before direction change and a 5 s travel time delta. All nodes had the same velocity, and there was no pause time. All simulations with the same velocity were run on the same set of 10 different topologies, to make the comparison between the different algorithms as fair as possible. The simulation results were sampled over 10 simulation runs, and all results are presented with a confidence interval of 95%. Other simulation parameter settings are presented in Table 1.

Running simulations with the standard OLSR implementation (i.e. without a buffer zone) reveal that the goodput (Fig. 3) is significantly lower with higher mobility, and the loss is at over 40% even at a total data load of 50 kbps for 10 m/s node velocity. For 10 m/s node velocity the loss (Fig. 4) up to 150 kbps load is mainly caused by discards due to maximum MAC retries (RET). (When the number of retransmissions of frames reaches the retry limit, the transmitting node discards the DATA frame, and a RET loss occurs.) This confirms the suspicion that mobility and the delayed link break reaction of the routing protocol causes considerable loss.

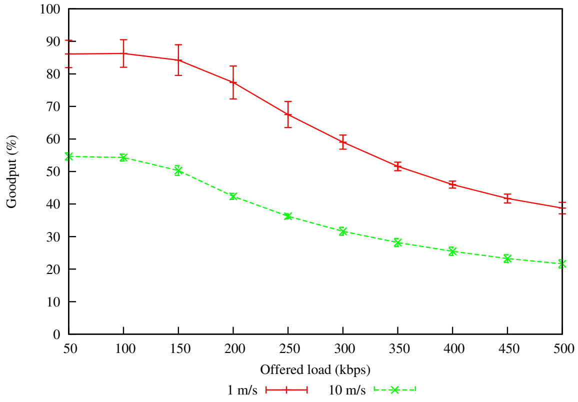

Figure 3. Goodput results, standard OLSR.

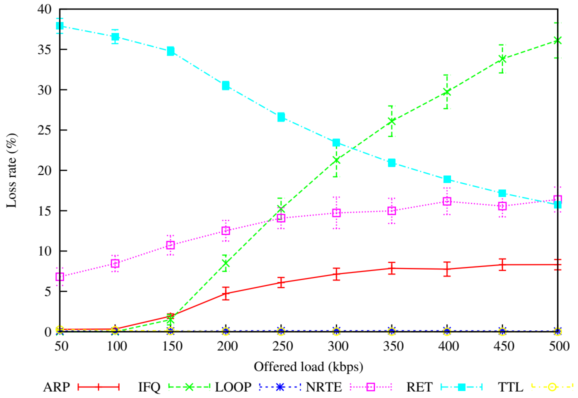

Figure 4. Loss reasons, 10 m/s, standard OLSR.

Above 150 kbps the loss caused by tail drops from the interface queue (IFQ loss) increases, together with the loss from the Address Resolution Protocol buffer (ARP loss), while the RET loss ratio declines. Thus, above 150 kbps, the network gets more and more congested. In addition, some loss is caused by the lack of route to the destination (NRTE loss), increasing slowly as more routing control packets are lost due to collisions. This means that topology information is lost, leading to more packets being dropped because of a lack of route to the destination (NRTE).

4 Proposed solution

This section presents the buffer zone solution, which is an algorithm for improved handling of mobility and link breaks in the ad hoc network.

4.1 Analysis

The closer a neighbor node comes to the edge of the transmission area, the more likely is a link break to occur. The detection of this link break will delay until no Hello packet has been received within a given time interval. Thus, a node in the area where a link break is likely to occur could be considered an unsafe node that should not be relied on to forward traffic, if an alternative is possible.

The part of the transmission area where it is no longer guaranteed that a link to a neighbor will remain stable is referred to as the unsafe buffer zone, and the other part is thus referred to as the safe zone, as can be seen in Fig. 1. Likewise, a node in the safe zone is referred to as a safe neighbor, while a node in the unsafe buffer zone is referred to as an unsafe neighbor.

The minimum time for a node at the edge of the safe zone to disappear can be expressed as the distance to the transmission edge divided by the maximum relative velocity between the two neighboring nodes of the link. With the recommended settings of OLSR, the routing protocol will detect a link break between 4 and 6 seconds after the link break has occurred. In the worst case, a node with direction directly opposite of the transmitting node with a velocity of 10 m/s would be able to travel up to $2s \cdot 20m/s = 40m$ before the first Hello packet is lost, and $6s{\cdot}20m/s = 120m$ from the link break has happened until it is detected and acted upon. This means that to be sure that a neighbor does not move out of the transmission area before it is marked as unsafe, any node closer to the edge than 120 m should be marked as unsafe.

However, 120 m would be the maximum distance from the edge where a link break could occur due to the movement of a node. Defining such a large part of the transmission area as a buffer zone has two drawbacks. First, the mean number of hops between pairs of nodes in the MANET would almost be doubled, leading to an increased number of transmissions in the network and a lower end-to-end traffic capacity. Second, nodes close to the transmission area edge are treated the same way as nodes 120 m away from the edge, despite that the latter nodes have a very low probability of a link break compared to the nodes located close to the transmission area edge. The optimal buffer zone is to be found as a tradeoff between these effects, and thus somewhere in the range between 0 and 120 m in this particular example.

4.2 Zone routing algorithm

The buffer zone solution is based on defining nodes as safe or unsafe, and either using them as relay nodes, in case they are safe, or avoiding them as relay nodes in case they are unsafe. Also, traffic to unsafe nodes inside the sending node's transmission area should be attempted relayed through safe nodes, if possible.

The signal strength of the Hello packets can be used as parameter to be able to determine which nodes are in what is considered the safe zone and the unsafe zone with varying mobility speeds. However, other means of determining this are also possible, including the use of GPS.

The zone status of each neighbor must be added to each link entry in the Hello packets and announced to the other neighbors, in order to support neighboring nodes in routing traffic to its unsafe neighbors. It is necessary to avoid routing a packet to a relay node which has the destination as an unsafe neighbor, if the source node also has the destination node as an unsafe neighbor.

Fig. 1 compares standard OLSR routing to OSLR routing with the proposed buffer zone algorithm. The dashed arrow shows the normal routing, where all packets from A to C are transmitted directly to node C. This makes the transmission vulnerable to mobility in case node C moves away from node A. The continuous arrows show the packet path using the zone algorithm, where traffic from node A to node C is routed via node B, because node C is in the unsafe buffer zone of node A. This means that the traffic path is not vulnerable to node C moving out of the transmission area of node A.

The routing table of each node is first calculated based only on nodes in the safe zone, and if this leads to partitioning, routes via nodes in the unsafe buffer zone are included in the routing table. The principle of buffer zone routing is to only use nodes in the safe zone to forward traffic. The nodes in the unsafe buffer zone should only be used for forwarding if it is impossible to obtain full connectivity without them.

As the neighbor set, two-hop neighbor set and topology set are traversed, no route updates to the already defined routes are allowed. This means that if a node already is represented in the routing table as a destination, the newly found route to the same destination is discarded, even if it is of fewer hops than the first route. The steps of the buffer zone routing algorithm are shown in Table 2.

5 Evaluation of the buffer zone solution

5.1 Outline of the evaluation

This section presents the key behavior of the buffer zone routing algorithm, and compares it to the behavior of standard OLSR. For all graphs where the x-axis represents the threshold between the safe and unsafe zone, the results at 250 m threshold correspond to the complete transmission area being the safe zone. Thus, the buffer zone results at a threshold of 250 m are equal to the performance of standard OLSR without the buffer zone solution. (In fact, the latter is not entirely true, as the buffer

Table 2. The zone routing algorithm

1. Clear routing table.
2. Add route to all neighbors (1st time: only neighbors in the safe zone).
3. *Add 2hop neighbors*
3.1. Add route to all 2hop neighbors that both are in the safe zone of the relaying neighbor and where the neighbor is in the routing table.
3.2. Add route to all 2hop neighbors that are this node's neighbors with direct route.
3.3. Add route to all 2hop neighbors in the unsafe zone of their neighbor while the neighbor is in this node's safe zone.
3.4. On 2nd iteration: Add 2hop neighbors in the unsafe zone of their neighbor while the neighbor is in this node's unsafe zone
4. Add route to all topology tuplets with increasing hop count.
5. If first time, return to step 2, else exit.

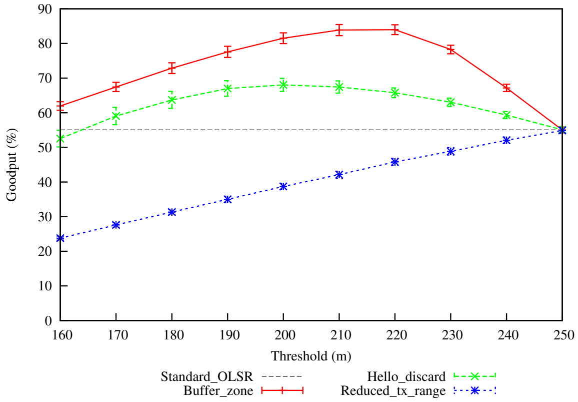

Figure 5. Goodput with node speed 10 m/s and 50 kbps system load.

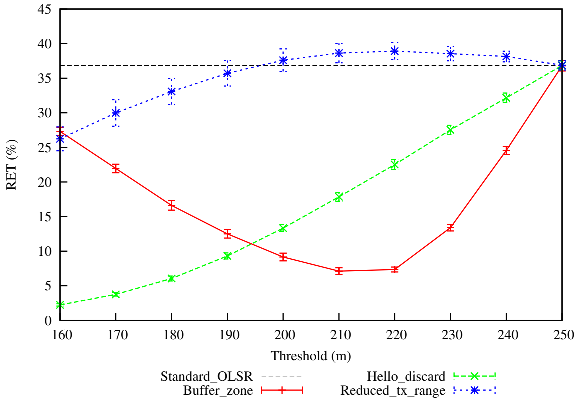

Figure 6. RET loss. (Loss caused by MAC maximum retransmissions discards.)

zone algorithm has a marginally higher routing overhead, as will be shown in Section 5.3.)

To give a fair comparison between the buffer zone routing and standard OLSR, not only is the buffer zone routing compared to standard OLSR at a transmission range of 250 m. It is also compared to standard OLSR with a reduced range of next hop neighbors (see the curves marked '*Hello_discard*'). This way it can be ensured that the advantages of the buffer zone routing do not stem from some effects of reducing the range of next hop neighbors, but rather from dividing the transmission area into a safe zone and an unsafe zone. Thus, the graphs marked as '*Hello_discard*' imply a simple discard mechanism for the Hello packets at thresholds below 250 m, preventing the Hello packets received from nodes in the unsafe zone from being processed, but allowing reception and acknowledgement of data packets. In addition, the buffer zone algorithm is compared to an implementation of standard OLSR configured with an overall reduced transmission range (see the curves marked '*Reduced_tx_range*'), where the effect of reducing the reception radius for both data traffic and control traffic is shown.

In the following, first results with low traffic load (i.e. 50 kbps of traffic in total) and high node mobility (i.e. 10 m/s velocity for each node) are presented. Both the goodput, loss, average path length and routing load are investigated. After having gained insight about the performance at high mobility, the goodput results are compared with results for low mobility. Then, these low traffic load results are compared with similar results generated at a high traffic load (i.e. of 500 kbps of traffic in total), for both low and high mobility. Finally, the goodput when LLN is implemented is explored.

5.2 Exploring the benefits of the buffer zone solution

Fig. 5 shows the goodput results with a relatively high node velocity of 10 m/s and a light total traffic load of 50 kbps. Comparing the results of the buffer zone algorithm at lower thresholds than 250 m to the result of standard OLSR (which is equal to the result of the buffer zone algorithm with a threshold of 250 m), the gain of using the buffer zone algorithm over standard OLSR is $84\% - 55\% = 29\%$ at 220 m. Fig. 6 indicates that the main advantage of the buffer zone algorithm stems from the fact that the retransmission (RET) loss is considerably reduced compared to the RET loss of standard OLSR.

One could on the other hand suspect that the advan-

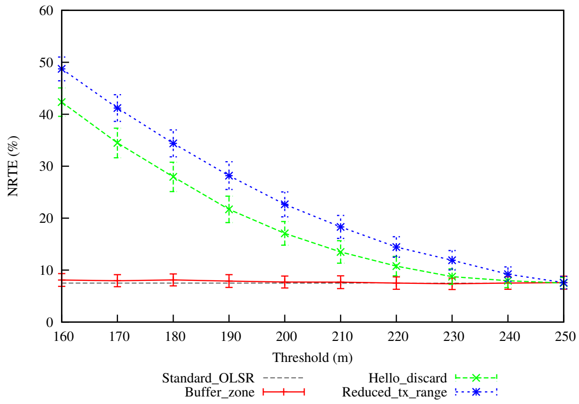

Figure 7. NRTE loss. (Loss caused by lack of route.)

tages of the buffer zone routing stems from some effects of reducing the range of next hop neighbors. However, Fig. 5 shows that by letting standard OLSR discard Hello packets at a threshold lower than 250 m (*Hello_discard*), it still performs worse than the buffer zone solution. This is mainly due to the increased probability of partitioning caused by the discard of Hello packets. This again results in more packets being lost due to lack of route (NRTE loss), as observed in Fig. 7. The figure shows that the probability of partitioning for the *Hello_discard* method increases with a decreasing threshold distance.

Nevertheless, Fig. 5 shows that even the simple *Hello_discard* method has a higher goodput than standard OLSR. The reason is that the discard of Hello packets forces the routing protocol to use shorter links. A bulk of link breaks is then avoided, because nodes outside the discard zone still can receive the packet and reply with an acknowledgment before the neighbor moves beyond the transmission radius. The RET loss is therefore reduced also for the *Hello_discard* method (Fig. 6). This benefit outweighs the disadvantage of a higher probability of partitioning, leading to a totally higher goodput than the goodput of standard OLSR (Fig. 5).

Reducing the transmission range itself (i.e. the *Reduced_tx_range* method) offers no buffer zone outside the threshold where nodes that have moved outside the threshold can continue to communicate. Therefore, a clear advantage of a reduced RET loss compared to that of standard OLSR is not observed in Fig. 6. Instead, the *Reduced_tx_range* method performs worse than OLSR (Fig. 5). The main reason is that the reduced transmission range causes more network partitioning as the number of neighbors is reduced, leading to the high NRTE loss observed in Fig. 7. The loss is even higher than the loss of the *Hello_discard* method, because TC messages from nodes outside the threshold are also discarded.

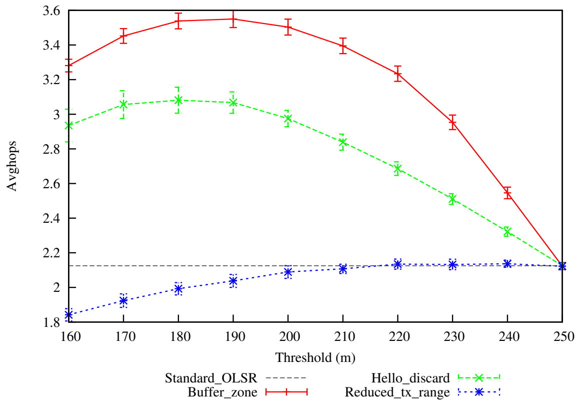

Figure 8. Average number of hops.

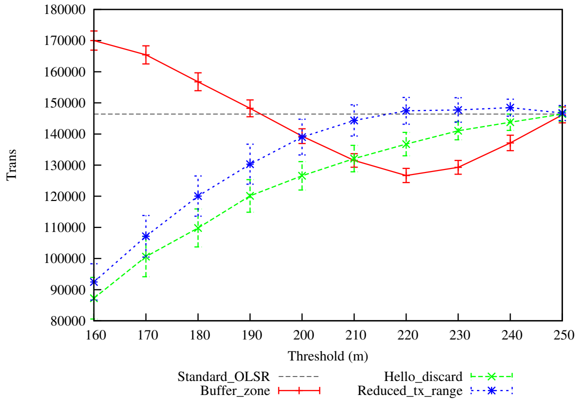

Figure 9. Total number of MAC transmissions including retransmissions.

5.3 Exploring the costs of the buffer zone solution

Having identified a reduction in the RET loss as a main benefit of the buffer zone algorithm, it is interesting to explore the cost of this solution.

There was no difference between the buffer zone solution and standard OLSR in term of packets being lost due to lack of route (Fig. 7). Thus, the buffer zone algorithm does not increase the chances of network partitioning compared to standard OLSR. This is expected, since the buffer zone algorithm is forming links to neighbors in the buffer zone whenever necessary.

However, there is a difference between the buffer zone solution and standard OLSR in terms of the mean number of hops between a source and a destination. The number of hops per path (Fig. 8) is increased with the buffer zone solution, as it favors nodes in the safe zone as relay nodes. The increased hop length is a main disadvantage of the buffer zone solution.

First, the increased hop length leads to an increased number of needed transmissions for the same end-to-end

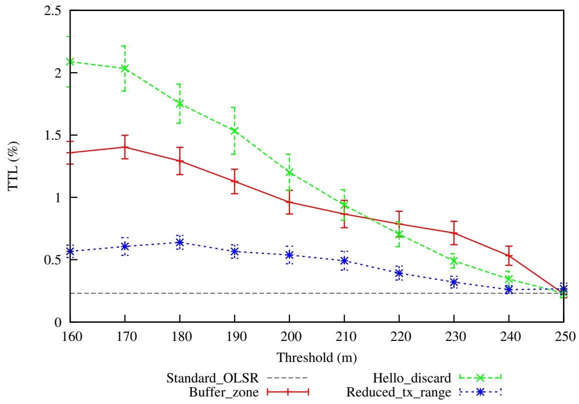

Figure 10. TTL loss. (Loss caused by exhausted time to live.)

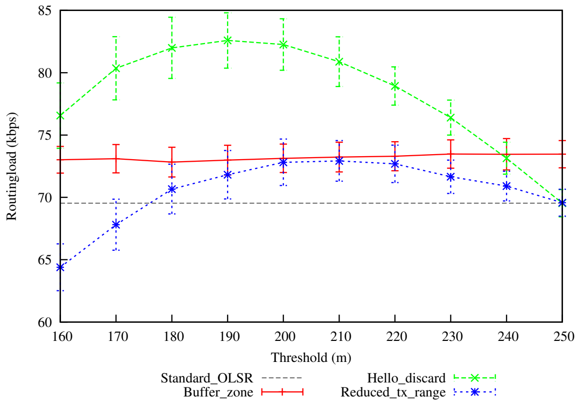

Figure 11. Routing load.

traffic streams, thus reducing the total available capacity per traffic stream. However, since the buffer zone solution reduces the number of packets lost due to RET, the total number of transmissions (Fig. 9) is actually lower for the thresholds above 190 m.

Second, as the paths get longer, there is an increased risk that the topology information held by the forwarding nodes is wrong. There is a higher probability that the topology (both the real topology and the topologies given by the routing tables) changes while the packet is in transit between the source and destination nodes. Thus, the risk of a packet loop, or a considerable detour, is increased. Both an increased average path length, an increased risk of a packet detour and an increased risk of a packet loop add to the probability of a Time-To-Live (TTL) exhaustion. Indeed, the ratio of packets being discarded because of exhausted Time To Live (i.e. TTL loss) is higher for the buffer zone algorithm than for standard OLSR (Fig. 10).

One might argue that the increased TTL loss of the buffer zone algorithm is quite small (e.g. only 1% at a threshold of 200 m). However, the main problem with packets discarded by TTL is that they are transmitted extensively, and at least a number of times equivalent to the original TTL value set by the source node. As all source nodes in the simulations set the TTL value of the packets they are originating to 32, it means that at a threshold of 200 m, 1% of all sent packets have been transmitted at least 32 times. All these transmissions ultimately proved to be worthless. Thus, even a low percentage of TTL loss might represent a large and unnecessary consumption of the network resources.

In addition to a higher mean path length, the cost of the buffer zone solution also includes a higher routing load, in terms of a higher payload overhead of the Hello messages. The reason is that the buffer zone solution depends on publishing the zone status of the neighbor nodes in the Hello messages. At a 250 m threshold the increase in the routing load for 40 nodes at 10 m/s is around 3 kbps (Fig. 11), which is quite low compared to the overall network capacity. In fact, for simplicity the zone status field in the Hello message was implemented in the simulator as one extra byte for each address in the message, although the information – safe zone or buffer zone – could have been represented by only one bit per address. Furthermore, only symmetric links need this information, so the zone information could have been skipped altogether for the asymmetrical links. Thus, the additional routing overhead of the buffer zone algorithm of 3 kbps might be made much smaller in an optimized implementation. Nevertheless, when it was stated above that the buffer zone algorithm has equal performance and behavior as standard OLSR when the threshold is set to 250 m, this is not exactly true, due to the marginal extra routing overhead of the buffer zone algorithm.

Interestingly, the routing overhead of the *Hello_discard* and *Reduced_tx_range* methods is increasing when the threshold decreases from 250 m. The reason is that these methods require an increased number of multi-point relay (MPRs) nodes – nodes that generate and forward TC messages in the network – as the threshold decreases. This leads to higher routing overhead. However, as the threshold gets even lower, the routing load starts to decrease again. This is due to the increased probability of network partitioning observed at low thresholds (Fig. 7).

5.4 The impact of node mobility

Reducing the node velocity reduces the number of occurring link breaks. Since the buffer zone solution is aimed at reducing the number of packets lost due to link breaks, it is expected that the advantage of the solution is decreasing with decreasing mobility. However, the buffer

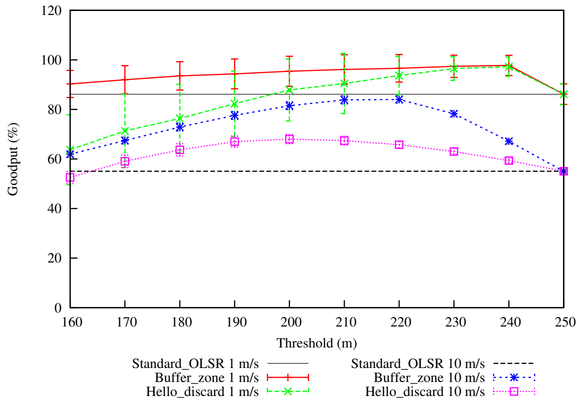

Figure 12. Goodput for 1 and 10 m/s at 50 kbps load.

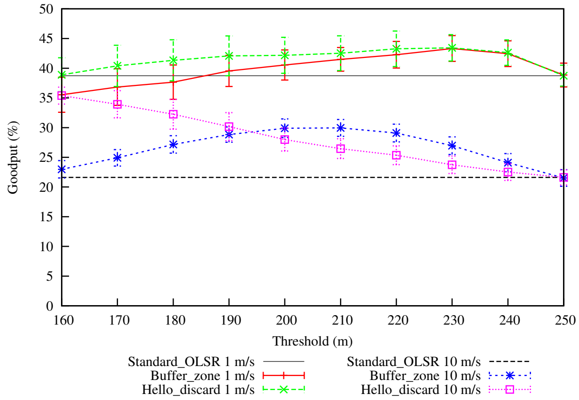

Figure 13. Goodput for 500 kbps load.

zone solution still provides a gain in throughput also at a low node velocity of 1 m/s (Fig. 12), obviously because the performance cost of the buffer zone solution is quite low. For the lowest level of mobility (i.e. 1 m/s) the optimum goodput of the buffer zone solution is over 95% at 240 m threshold, providing a gain of $95\% - 85\% = 10\%$. The *Hello_discard* mechanism also achieves this gain, but performs worse at lower thresholds due to partitioning.

With reduced velocity, the probability of a link break due to a neighbor moving out of the transmission area is lower, simply due to a lower node speed. Therefore, the threshold range can easily be set higher to achieve the same advantage, while the disadvantage of increased path lengths is reduced. It is expected that the optimal threshold range in terms of maximized goodput is increasing with decreasing node mobility, and that the optimal threshold range is 250 m when the node mobility is zero. This expectation is supported by the simulation results. The optimal threshold range is 240 m at a node velocity of 1 m/s and only 220 m for a node velocity of 10 m/s.

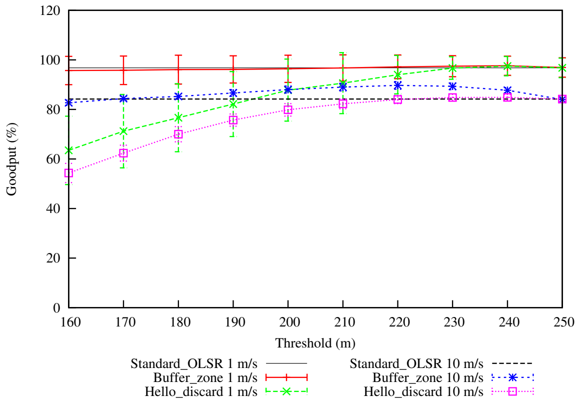

Figure 14. Goodput with LLN.

5.5 The impact of traffic load

The buffer zone algorithm increases the throughput compared to the Standard_OLSR results even for high traffic loads (Fig. 13). The relative gains of the buffer zone solution as a percentage of the goodput of the standard OLSR in the high load results are comparable with the relative gains of the low load results in Fig. 12. However, the total gain of the buffer zone solution is naturally lower for such high traffic loads. The reason is that the entire network is stressed with a large number of unnecessary transmissions and increased packet loss, leaving a lower share of the total network capacity to the successfully transmitted traffic, be it traffic routed by standard OLSR or traffic where the buffer zone algorithm is used.

For thresholds lower than 190 m, the *Hello_discard* algorithm yields better throughput than the buffer zone solution (Fig. 13), because the *Hello_discard* algorithm increases the partitioning of the network, and this reduces the load. At the same time it mends the link-break-induced retransmission problem, and the combination of partitioning and reduced retransmissions in the event of link breaks makes the throughput higher for the lower thresholds. However, due to the partitioning, the packets are highly probable to be traveling only a few hops, thus increasing the unfairness between the short path and the long path traffic.

5.6 The buffer zone solution with LLN

Even with LLN, the buffer zone algorithm gives better goodput than both the *Hello_discard* method and standard OLSR (Fig. 14). However, at very low node mobility (i.e. 1 m/s) the gain is marginal. Furthermore, the gain is considerably less than for the results without LLN (Fig. 12). The reason for the gain over standard OLSR is the fact that the buffer zone solution prevents the effects of the link break, as less traffic is routed over the link when it breaks. This also helps the other neighboring nodes to be aware that the neighbor is in danger of being lost. When LLN is employed without the buffer zone solution, any neighboring nodes must wait until the next Hello packet to get knowledge of the lost link. This could be up to 2 seconds after the link break has occurred. With the buffer zone solution, on the other hand, the neighbors are aware in advance that the node is unsafe.

5.7 Discussion

The presented results show that the buffer zone solution offers a throughput gain over both the standard OLSR routing protocol and the simpler *Hello_discard* algorithm, and the buffer zone solution is robust to variations, with both varying node velocity and traffic load. The throughput gain comes from the reduced packet loss caused by maximum MAC retransmissions (RET). The RET loss is reduced because the buffer zone solution prefers neighbors in the safe zone as relay nodes, and since these links are less apt to break, less links that break are in active use.

The optimal threshold range is affected by mobility. At 1 m/s node speed the buffer zone solution delivers high goodput in a broad threshold range, while for 10 m/s the threshold range where the throughput gain is greatest is much smaller, and focused around 200-220 m. Thus, if a static zone threshold is preferred, 210 m could be a good compromise. However, the threshold could also be set dynamically, based on one or several various parameters such as mobility, safe-to-unsafe nodes ratio, link break probability based on learning, etc.

Signal strength was used as the parameter to classify the neighbor nodes as either safe or unsafe. Because the TwoRayGround propagation model was used, the distance to each neighbor was available through the signal strength parameter, and this made the prediction and classification of which neighbors that were safe or unsafe straightforward. Signal strength is however a parameter that may only be of high value in a simulator environment. Other research, such as [8] and [7], has shown that in real experiments the signal strength is highly variable and must be filtered over many samples to provide a trustworthy value. Likewise, the transmission range threshold – where the packet loss by only moving a centimeter at the edge of the transmission area instantly goes from full to nothing – is not equal to the real world experience. Another problem with using the signal strength is illustrated through the problem of gray zones [13]. Since broadcast and unicast packets are sent with different transmission rates, the signal strength of a Hello packet, which is broadcasted, would not be directly transferable to the signal strength of a unicast packet from the same node at the same distance.

Instead of signal strength, other parameters could have been used, such as geographical position, packet loss, the number of errors corrected with forward error correction (FEC), or MAC retransmissions. These would not give the same distance precision as has been achieved through the signal strength of Hello packets. However, the buffer zone algorithm could also be seen as independent from the distance and mobility perspective that has been used in this paper. Through classifying nodes as either safe or unsafe, based on for example the number of MAC retries for the last number of transmitted packets, the classification could instead determine what links are more reliable. Using the same algorithm for constructing the routing table, the result would perhaps be enhanced throughput. This should be researched further.

6 Related work

This paper is a follow up of [15], where the retransmission problem is investigated thoroughly in relation with the interface queue size. However, there exist several other works dealing more directly with the problem of link breaks due to mobility and proposing solutions to mend this problem. Qin and Kunz [16], for example, present a solution to detect and mend the effect of mobility through evaluating the rate of link breaks.

Many solutions focus only on reactive routing protocols, where the rerouting overhead reduction potential is high. These include [10], which is based on GPS location information, along with [11], [14] and [5], all based on signal strength evaluation. In fact, the solution by Goff et al. [11] bears some resemblance to the *Hello_discard* algorithm presented in this paper, with a *preemptive region* comparable to the unsafe buffer zone.

On the other hand, Su, Lee and Gerla [17] present a mobility prediction solution implemented for both proactive and reactive protocols, where both GPS and signal strength are proposed used to establish the positions and relative distances of network nodes. A link break prediction table is presented in [6], to be utilized by both reactive and proactive routing protocols.

Ali et al. [3] propose to use signal strength with hysteresis in OLSR to both anticipate link breakages and avoid establishing links that are transient. Fast-OLSR [4] is another modification to OLSR where the Hello interval is varied with the degree of mobility.

7 Conclusions and further work

The introduction of a transmission buffer zone in OLSR gives improved throughput compared to standard OLSR (or compared to no buffer zone, which is approximately equal to standard OLSR). The advantage of using a buffer zone is observed both for low and high traffic loads.

A too large buffer zone, however, leads to an unnecessary higher mean number of hops between pairs of nodes in the MANET and a higher probability of network partitioning. Thus, the size of the buffer zone should be optimized.

The optimal size of the buffer zone (which is given directly by the optimal threshold range of the buffer zone algorithm) is increasing with increasing node mobility. At no node mobility, the optimal size of the buffer zone is zero, assuming that all link breaks are caused by mobility. However, in a realistic network scenario where link breaks are also caused by changing radio conditions, it is reason to believe that the buffer zone algorithm is useful also at no mobility.

Finding a means to estimate the optimal size of the buffer zone, depending on parameters such as node mobility and network load, is an important issue for future work. Furthermore, it is also a need for investigating the buffer zone algorithm with a more realistic radio model than used in this paper, i.e. both using a better radio channel model and investigating scenarios where link breaks are caused by changing radio conditions. Finally, the presented buffer zone algorithm can be improved and extended, using other criteria apart from distance to classify neighbor nodes as safe or unsafe.

8 Acknowledgment

This work was supported by the ITEA Easy Wireless and CELTIC DeHiGate projects.

References

[1] Manet simulation and implementation at the university of murcia (masimum). http://masimum.dif.um.es/.

[2] Network simulator 2 - ns2. http://nsnam.isi.edu/nsnam/.

[3] H. M. Ali, A. M. Naimi, A. Busson, and V. Vèque. An efficient link management algorithm for high mobility mesh networks. *Proceedings of the 5th ACM international workshop on Mobility management and wireless access (MobiWac)*, pages 42–49, 2007.

[4] M. Benzaid, P. Minet, and K. Al Agha. Analysis and simulation of fast-olsr. *The 57th IEEE Semiannual Vehicular Technology Conference (VTC)*, 3:1788–1792, April 2003.

[5] G. Chauhan and S. Nandi. Qos aware stable path routing (qasr) protocol for manets. *First International Conference on Emerging Trends in Engineering and Technology (ICETET)*, pages 202–207, July 2008.

[6] M. Chegin and M. Fathy. Optimized routing based on mobility prediction in wireless mobile adhoc networks for urban area. *Fifth International Conference on Information Technology: New Generations (ITNG)*, pages 390–395, April 2008.

[7] K.-W. Chin. The behavior of manet routing protocols in realistic environments. *Asia-Pacific Conference on Communications*, pages 906–910, Oct. 2005.

[8] K.-W. Chin, J. Judge, A. Williams, and R. Kermode. Implementation experience with manet routing protocols. *SIGCOMM Computer Communication Review*, 32(5):49–59, 2002.

[9] T. Clausen, P. J. (editors), C. Adjih, A. Laouiti, P. Minet, P. Muhlethaler, A. Qayyum, and L.Viennot. Optimized link state routing protocol (OLSR). RFC 3626, pages 1–75, October 2003. Network Working Group.

[10] S. Crisostomo, S. Sargento, P. Brandao, and R. Prior. Improving aodv with preemptive local route repair. *International Workshop on Wireless Ad-Hoc Networks*, pages 223–227, 2004.

[11] T. Goff, N. B. Abu-ghazaleh, D. S. Phatak, and R. Kahvecioglu. Preemptive routing in ad hoc networks. *Proceedings ACM/IEEE MobiCom*, pages 43–52, 2001.

[12] IEEE. Wireless LAN medium access control (MAC) and physical layer (PHY) specification. IEEE standard 802.11, June 1999.

[13] H. Lundgren, E. Nordströ, and C. Tschudin. Coping with communication gray zones in ieee 802.11b based ad hoc networks. In *Proceedings of the 5th ACM international workshop on Wireless mobile multimedia (WOWMOM)*, pages 49–55, New York, NY, USA, 2002. ACM.

[14] L. Meng, J. Zang, W. Fu, and Z. Xu. A novel ad hoc routing protocol research based on mobility prediction algorithm. *Proceedings of International Conference on Wireless Communications, Networking and Mobile Computing*, 2:791–794, Sept. 2005.

[15] V. Pham, E. Larsen, K. Ovsthus, P. Engelstad, and O. Kure. Rerouting time and queueing in proactive ad hoc networks. In *IEEE International Performance, Computing, and Communications Conference (IPCCC)*, pages 160–169, 2007.

[16] L. Qin and T. Kunz. Mobility metrics to enable adaptive routing in manet. *IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, pages 1–8, June 2006.

[17] W. Su, S.-J. Lee, and M. Gerla. Mobility prediction and routing in ad hoc wireless networks. *International Journal of Network Management*, 11(1):3–30, 2001.