iOLSR: OLSR for WSNs Using Dynamically Adaptive Intervals

Erlend Larsen, Joakim Flathagen, Vinh Pham
FFI, Norway
Email: {erl, jfi, vph}@ffi.no

Lars Landmark
Q2S, NTNU, Norway
Email: larslan@q2s.ntnu.no

***Abstract*—Proactive link state routing protocols, as used within the Mobile Ad hoc NETwork framework, have not been as successful in wireless sensor networks. This is mainly due to the extensive energy usage by control traffic transmissions and state requirements. However, such protocols are in many situations a more suitable candidate than their counterparts. The benefits are their topology overview, and more importantly the already available spanning trees for information distribution. The high signaling overhead associated with proactive protocols can be reduced by taking advantage of the static nature of wireless sensor networks. In this paper, we investigate how the Optimized Link-State Routing (OLSR) protocol, as a proactive routing protocol candidate, can be adapted to work better in a wireless sensor network environment. The basis for the solution is that control messages are sent with a low frequency when the network is stable, and more often if topology changes occur. The proposed solution is investigated using simulations from no loss to lossy link environments showing promising results.**

***Keywords*-Ad hoc networks; Routing; Wireless sensor networks.**

I. Introduction

Mobile Ad Hoc Networks (MANETs) and Wireless Sensor Networks (WSNs) are regarded as two distinctly different types of ad hoc networks, requiring routing protocols with specialized attributes. While MANET routing protocols are challenged with mobility, the main limitations for routing in WSNs are energy and memory. The IETF 6LoWPAN (IPv6 over Low power Wireless Personal Area Networks) working group [1] have made great efforts to bring IP to WSNs and other low power wireless networks. Therefore, modifying a MANET protocol that natively supports IP to better fit the challenges of WSNs may help the introduction of IP in WSNs.

Radio communication is a major energy consumer in WSNs. Shwe et al. [2] proposed among other measures to minimize the number of control packet transmissions to reduce energy usage. However, limiting the control traffic could make the routing protocol less able to maintain routes and do route repair. Thus, our goal is to reduce the number of control packet transmissions while avoiding these negative consequences.

The traffic pattern types in WSNs are mainly the following, arranged in probability of occurrence from high to low:

1) Sensors to sink
2) Sink or a specific controller to all or some sensors
3) Sensor to sensor

In the first and third case, the main challenges that have been addressed are on optimizing energy preservation and memory usage. Less attention has been given to traffic flows from a controller to the sensors, for instance for software updates. The process of software updating would require a more optimized distribution tree. Hence, a proactive protocol enabling optimized message distribution and routing would in many cases be more advantages than a protocol optimized for sensor to sink.

The Optimized Link State Routing (OLSR) protocol [3] is a proactive link state MANET routing protocol. It sets up and maintains routes regardless of application layer communication demands. The route maintenance is based on the regular transmission of control traffic. A high number of control packet transmissions will make the protocol less suitable for WSNs. At the same time it offers several advantages that are not as easily available with reactive protocols. For example, it can provide: quick rerouting in case of topology changes, spanning trees for information distribution, node cooperation, and node localization. Sleep functionality may be more challenging with a proactive protocol, but on the other hand, a proactive protocol may offer a distribution tree that may allow more efficient synchronization of sleep state. WSNs often operates over lossy links. In networks where links experiencing radio silence, a protocol enabling fast recovery is a requirement. Reactive protocols tend to increase their path distance as their have no mechanism to roll back after expiring radio silence as proactive protocols.

The OLSR protocol is not specifically designed for fixed topologies. Furthermore, due to its link state properties it also has a larger state requirement, than other protocols tailored for WSNs. There are no specified mechanisms to adapt the emission interval of control messages, depending on the grade of topology change. This means that the rate of control messages must be decided before the network deployment, based on the expected dynamics and the wanted reaction time to such dynamics. WSNs can be perceived as static and fixed without any dynamics. Nonetheless dynamics will occur, due to fluctuating links, new deployed nodes or nodes disappearing due to energy depletion or other malfunction. In a more dynamic network, where links or nodes break frequently, the routing protocol needs to perform control traffic dissemination more often.

The main contribution of this paper is the adaption of OLSR to exploit the static nature of WSNs through dynamically

 ISBN: 978-1-61208-144-1

increasing the intervals of the control messages. This proposed solution is named iOLSR. Scaling OLSR used in WSNs has been criticized for the extensive use of state, which makes scaling a challenge. However, for medium WSNs, the required memory for holding OLSR state will likely be below current memory limitations of many commercial nodes for WSNs. The proposed solution shows its ability to reduce its control traffic, and to deal well with in environments with error prone links.

The rest of the paper is structured as follows. Related work is presented in Section II. The changes proposed to OLSR are presented in Section III. The solution is investigated and compared to alternatives in Section IV. Finally, the paper is concluded along with an outline of further work in Section V.

II. Related Work

Earlier work that address adaptation of OLSR for WSNs include [4], where Benslimane et al. propose a way to perform energy-aware routing using OLSR. Minet and Mahfoudh presents an energy-aware version of the OLSR routing protocol in [5]. These two papers focus on routing traffic over paths that minimize the energy consumed in the end-to-end transmission of a packet flow and avoiding nodes with low residual energy, increasing the network lifetime. iOLSR, on the other hand, does not consider energy levels, but instead focuses on the reduction of control traffic.

The OLSR standard allows for different nodes having different interval settings, but there are no described options or methods to vary the intervals while operating. Fast-OLSR [6] is a proposal to enable the broadcast of Fast-Hello messages with a shorter interval in case high mobility is detected. It is thus a proposal to change the control message intervals with basis in information about the relative mobility of the node, depending on if there is a high number of changes in the node's neighborhood. In addition, Fast-OLSR proposes a Fast-Hello message with a reduced set of neighbors announced, to reduce the increased routing traffic overhead. Our proposed solution does not impose a new network message type and it is tested in a link burst environment. As with Fast-OLSR, iOLSR is compatible with the standard OLSR protocol.

An IPv6 routing protocol for Low Power and Lossy Networks (RPL) is currently being developed in the IETF. Clausen and Herberg investigate RPL-Enabled Optimized Broadcast in [7]. The authors argue that Multi-Point Relay (MPR)-based efficient broadcast is a well performing mechanism for WSNs, and the MPR mechanism is essential to the OLSR routing protocol, upon which we base our proposed solution.

III. Proposed solution

The OLSR protocol uses two different control messages for its most basic routing functionality, *Hello* and *Topology Control (TC)*. The Hello messages are generated by all nodes and are periodically broadcasted to all 1-hop neighbors. Based on the information exchanged in Hello messages, a subset of the nodes in the network are selected as MPR nodes. These nodes generate Topology Control (TC) messages, which are flooded throughout the network using the other MPR nodes.

The Hello and TC emission intervals affect the reaction latency to topology changes, and the intervals can be set balancing between the energy usage and the topology change discovery latency. The default lengths of the Hello and TC intervals are 2 and 5 s, respectively, and the main motivation for these low values is the ability to cope with high mobility induced topology change.

There are two clear side effects of increasing the control packet intervals. The first is the increased latency in detecting link breaks. To reduce the link break detection time, disappearing nodes causing link failures can be detected using Link Layer Notification (LLN). The second side effect is the latency in detecting new nodes. If new nodes are introduced in the network when the network has been operating for a while, these nodes will only be employed for routing when the network has discovered them. However, until the new node has received TC messages from all elected MPRs in the network, there is a risk that the node will discard packets to destinations it is unaware of. And worse, generate loops if it has a different view of the shortest path than the upstream node. Large message intervals will delay the discovery and the use of more optimal paths.

In static WSNs, there are no topology changes caused by mobility. The topology is stable and static for the most of the time. After performing the initial discovery of the topology, the routing protocol could stop disseminating control messages. However, at any time, a node may disappear or make its appearance in the network, and links may fluctuate. Links may even be broken due to external causes, such as targets entering the the network detection zone. Therefore, even proactive protocols for WSNs must perform control with the network links to detect and recover from topology changes. To reduce the overhead of routing messages, which drains the nodes of energy, the message intervals can be increased or turned off. However, this is at the cost of slower detection of topology changes detected by necessary control packets.

We propose to allow each node to adjust its Hello and TC intervals depending on the local state of the network. In the initial startup phase, where each new received Hello message contains new information, the node keeps the default low interval between each new originated message. As the initialization phase draws to an end, and no more changes are experienced in the neighborhood, the control message intervals are increased. In this way, the energy usage is reduced while the topology is stable and unchanged. If a change is detected in the local neighborhood, the message intervals are reset, and then incremented anew when no changes are detected (i.e., the network is perceived as stable again). The topology changes are detected by using Link Layer Notifications (LLNs) and through Hello messages containing new information. As a consequence, the protocol is able to adjust itself to operate over both stable links and more lossy environments while optimizing the overhead of routing messages through the increasing intervals.

In our proposal, the intervals of the control messages must vary between a lower and an upper limit. If the lower interval is too short, the number of control messages transmissions will drain the network nodes of energy without improving the routing protocol's reaction time. If set very low, collisions of control messages may even impair the network, causing loss of data traffic due to loops or lack of routes. In the upper end, the time fields of the OLSR message header limits the maximum interval. The time is encoded in the header in a mantissa and exponent format, each of 4 bits, into one byte. A time value i is encoded as $i = C \cdot (1 + \frac{a}{16}) \cdot 2^b$ where a is the highest 4 bits of the field, and b is the lowest 4. The scaling factor C is proposed as $\frac{1}{16}$ second, giving a time field range of 0.0625 s – 3968 s. The scaling factor could be increased to achieve a higher maximum time range, which could be advantageous for our proposed solution, but this has not been looked into in this work. In such a case, one would loose the resolution at lower numbers.

The intervals and the corresponding message timeouts (valid times) are increased each time the control message is transmitted. Upon experiencing a change in the neighborhood the intervals are reset to the default values, and the incrementation process begins over again. The rate that the intervals are increased by, can be discussed, and the simulation results show three different takes on this increase.

A generic representation of the calculation of control message intervals and their timeouts has been sought for. The following formula represents the calculation of increasing the intervals continually, either linearly or exponentially. The generic interval v can be calculated as follows:

$$v = v_d \cdot (\alpha^i + \beta \cdot i) \tag{1}$$

In (1), v is the resulting interval, v_d is the starting (default) interval, α is the base exponential value, β is the linear increment and i is a counter of successfully transmitted control messages. Upon a change in the local topology information, this counter is reset to 0.

The basis for message information timeout has been to follow the proposal of the OLSR RFC [3], using 3 times the interval rate as the timeout value. However, with an expected increasing interval, the timeout vt must be calculated as stated in (2).

$$vt = \sum_{k=0}^{2} v_d \cdot (\alpha^{(i+k)} + \beta \cdot (i + k)) \tag{2}$$

In the equations, α and β are constants, set for the entire duration of the simulation use or network deployment.

IV. SIMULATIONS

A. Setup

The proposed solution was investigated using simulations on the ns-2 network simulator [8] version 2.34. The OLSR protocol as described in [3] and implemented for ns-2 in [9] was used for unicast routing, and the IEEE 802.11 protocol [10] was used as MAC layer. LLN was enabled in all simulations.

TABLE I
SIMULATION PARAMETER SETTINGS

Radio-propagation model	TwoRayGround
Interface queue type	FIFO with DropTail, PriQueue for OLSR packets
Interface queue size	30 packets
Antenna Model	OmniAntenna
Data/basic rate	2 Mbps / 1 Mbps
Data transmission/sensing radius	250 m / 550 m
Simulation/measurement time	6000 s / 600–5900 s
Random seed	Heuristic

The solution was tested on a scenario with fluctuating links where the link loss is a consequence of link failures using an implementation of the Gilbert-Elliot link burst error model [11], [12]. The link has a certain probability of going into a 0–3 s burst error period for each received packet, and while in the burst error period, the link experiences a 100% packet loss. When not in a burst error period, the two-ray-ground radio propagation model is employed with a 250 m transmission radius.

The topology size was set to 40 nodes in a 1500 x 300 m^2 area, and the nodes were placed randomly using software from [13], to allow the examination of a wide network without the very long simulation processing time that follows using ns-2 with a high number of simulated nodes. The sink was randomly positioned. The simulation time was 6000 seconds unless otherwise stated. All nodes generated packets, except those nodes that appeared or disappeared, and the packets were set with the sink as destination, to test the paths toward the sink. The traffic load was 1 packet per second from each traffic-generating node. The traffic type was UDP unicast with a packet size of 50 bytes, and the traffic flows were started at 500 s. All data points are an average of 10 simulation runs, and are presented with a 95% confidence interval. The topologies were the same 10 topologies for each of the simulations. Other simulation parameter settings are presented in Table I.

For our proposed iOLSR solution, the interval incrementation counters were reset at the following events:

- Hello messaging causing link change or timeout.
- A new MPR selector or a timeout of an existing one.
- Link break causing a LLN.

When the static Hello intervals were increased, the TC intervals were increased correspondingly, so that for example a Hello interval of 10 had a TC interval of 25.

B. Results

1) Increasing intervals: First we investigated how the iOLSR solution compared to the regular static interval behavior of OLSR in varying link stability conditions. Three different static Hello intervals were simulated: 2, 20 and 100 s. The corresponding TC intervals are: 5, 50 and 250 s. The iOLSR default intervals were 2 s for Hello and 5 s for TC, and the interval increment was 2-base exponential. Examining the goodput results (Fig. 1) we see that all the variations manage to perform well when the topology is stable without many link errors. When the link error probability

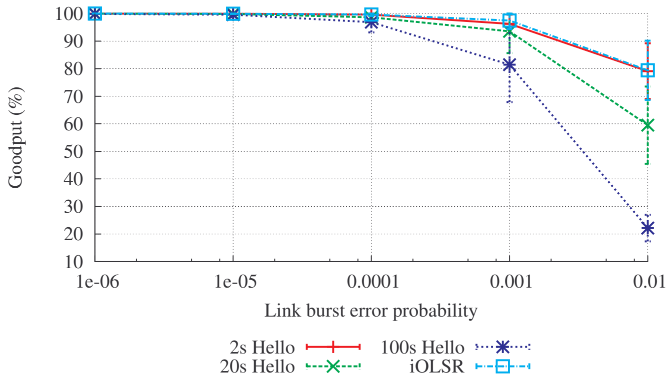

Fig. 1. Average goodput for iOLSR compared to OLSR with static intervals.

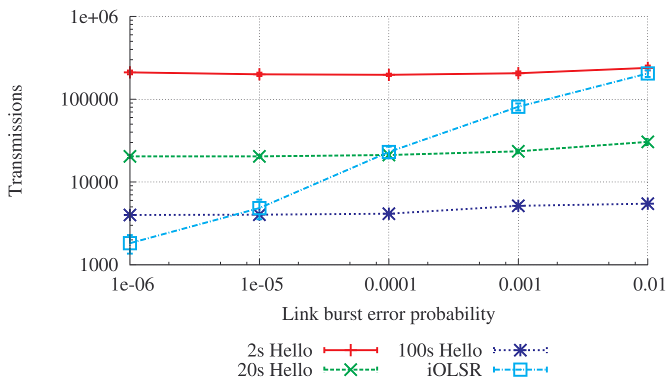

Fig. 2. Number of control packet transmissions for iOLSR compared to OLSR with static intervals.

increases, a higher interval between the Hello messages makes the routing protocol less capable of taking advantage of the links rebounding from a burst error, and this leads to a logical partitioning that reduces the goodput. Interestingly, we mark that iOLSR is able to offer the same performance as that of standard OLSR with 2 s Hello intervals, even at the highest link burst error probability.

While the goodput performance for all alternatives was very good at the lower link break probabilities, it is the number of transmitted control packets that is most interesting in WSNs, since the number of transmissions directly affect the energy use of the nodes. The number of control packet transmissions (Fig. 2) show iOLSR as being highly adaptive to the environment it operates in. When there is low probability of burst errors, the number of routing packets is kept at a much lower level than the comparable 2 s Hello interval results, and even compared to the 100 s Hello results. As the burst error probability increases, the routing protocol dynamically increases the number of Hello and TC messages generated locally in the area around each failing link.

2) Interval increment rate: The control message intervals' rate of increase is important when evaluating iOLSR. We have investigated three increment options where α and β refer to (1) and (2):

1) Linear (lin) ($\alpha = 1$ and $\beta = 1$)

TABLE II
CONTROL MESSAGE INTERVAL PROGRESSION

Hello			TC			
Lin	Exp2	Exp3	Lin	Exp2	Exp3	Exp3 vt
2	2	2	5	5	5	65
4	4	6	10	10	15	195
6	8	18	15	20	45	585
8	16	54	30	40	135	1755
10	32	162	35	80	405	5265
...	...	...	...	...	...	...

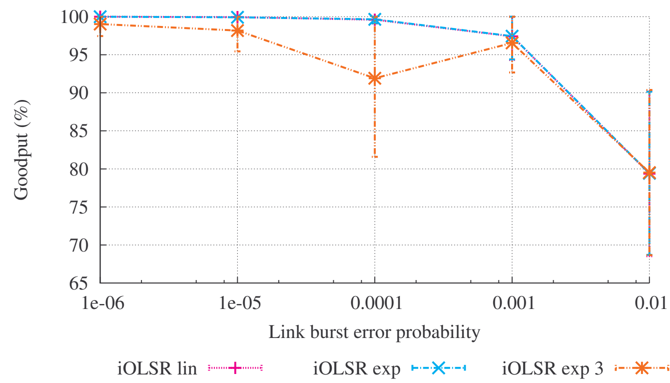

Fig. 3. Average goodput for linear contra exponential increase of iOLSR intervals.

2) 2-base exponential (exp2) ($\alpha = 2$ and $\beta = 0$)
3) 3-base exponential (exp3) ($\alpha = 3$ and $\beta = 0$)

As we see in Table II, the linear option will increase the interval by the default value for each successful transmission, while the 2-base and 3-base exponential options increment the message intervals exponentially according to (1).

The simulation results with varying control message interval incrementation rate shows that the goodput (Fig. 3) is not affected adversely by choosing a 2-base exponential increase of the intervals, even in an environment with a high probability of link burst errors. It follows the linear increment results very closely. The 3-base exponential increase is more prone to errors.

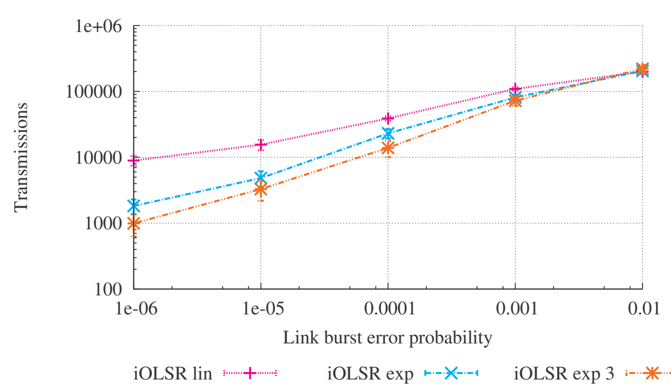

Fig. 4. Number of control packet transmissions for linear contra exponential increase of iOLSR intervals.

 ISBN: 978-1-61208-144-1

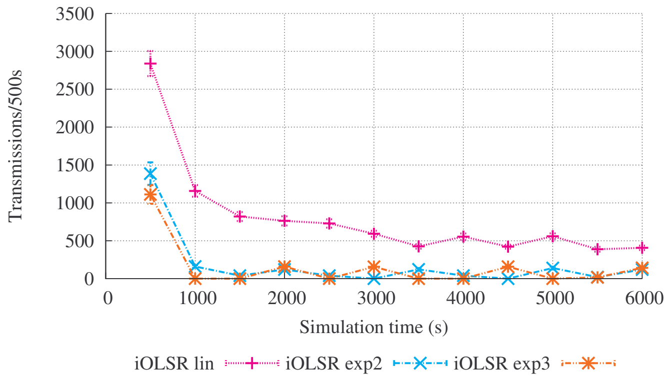

Fig. 5. Number of control packet transmissions for linear contra exponential increase of iOLSR intervals at increasing simulation time, 0–6000 s.

Examining the results for the number of control packet transmissions (Fig. 4), there is evidently great gain in using an exponential interval increment compared to a linear increment. However, the gain is much less in using a larger base exponent such as 3 compared to 2. The reason is twofold. First, the number of transmissions required to reach the maximum time field limit is lower with an increasing increment. After the incrementation phase, there is no difference between the increment values, since the intervals are no longer increased. For the exp3, the maximum interval for TC is reached at the fifth transmission, since the vt (control message information timeout) will be 5265 (Table II), thus exceeding the maximum time (3968). Second, the beginning of the incrementation phase is the phase where changes are most likely happen, especially at the initialization of the network. An interval increment that moves too quickly towards higher control message intervals may actually harm the initialization and convergence of the protocol.

Since there is great impact of how long the initialization phase is, when it comes to the number of transmissions, we have run some simulations without data traffic, and only routing traffic present, where the number of control packet transmissions is examined for the lin, exp2 and exp3 increment options. For a simulation lasting for 6000 s (Fig. 5), clearly the lin option is unable to reach the maximum interval. The two other options, however, reach the maximum interval very early.

In a longer simulation, 0–300000 s, the maximum time limit is even more pronounced, examining the control packet transmissions (Fig. 6). Except for the first measured step, the exponential increase options operate at the maximum for the entirety of the simulation.

The conclusion of the investigation into the increment alternatives is thus that the 2-base exponential increment represents a middleway between a too slow move away from the default intervals toward the maximum, and at the same time a more slow move away from the default intervals in the first phase of a stabilizing network.

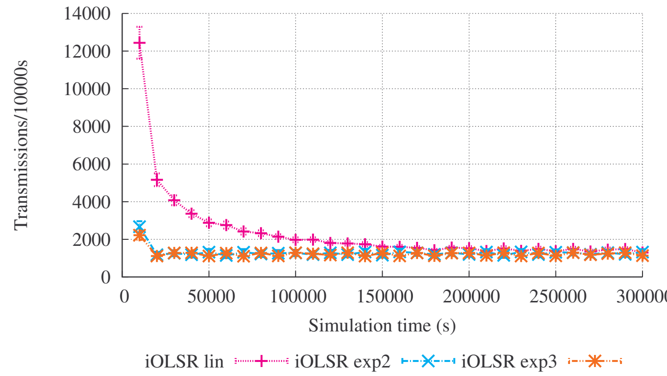

Fig. 6. Number of control packet transmissions for linear contra exponential increase of iOLSR intervals at increasing simulation time, 0–300000 s.

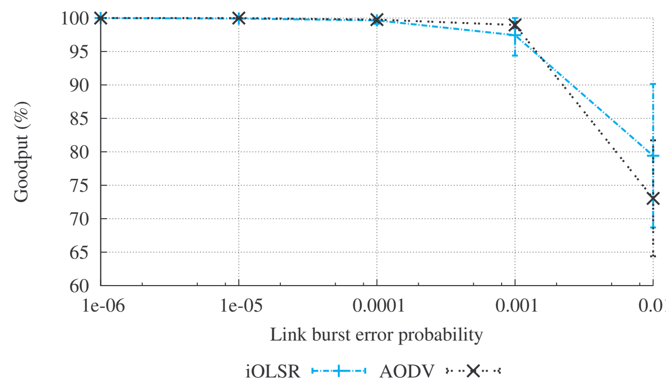

Fig. 7. Average goodput for iOLSR contra AODV.

3) Comparison with AODV: The final comparison in this paper is between the iOLSR (2-base exponential increase) and Ad hoc On-demand Distance Vector (AODV) [14]. Protocols for WSNs normally establish routes from sink to sensors to reduce the control traffic. AODV establishes routes from sensors to sink. In lossy environments it is likely that sensors must take part in path or link recovery. Hence, AODV in this sense resembles many protocols for WSNs in its behavior. In this work we want to compare the impact of burst error on iOLSR to a reactive protocol handling path/link recovery, such as AODV. With AODV, the start of the traffic flows were spaced up with 1 s intervals, to prevent effects of a synchronized route setup process. The goodput results (Fig. 7) show the interesting fact that AODV and iOLSR follow each other closely.

Examining only the number of control packets transmitted (Fig. 8) with the goodput results in mind, AODV is clearly better at low link burst error rates, yielding a much lower number of control packet transmissions than iOLSR.

However, the control packet results only tell part of the story. Investigating further, the number of hops for the data traffic (Fig. 9) is much higher for AODV than for iOLSR. This is due to the way AODV sets up routes only once, flooding the network with a route request from each source in the network. Although the total number of control packets may be low, the

 ISBN: 978-1-61208-144-1

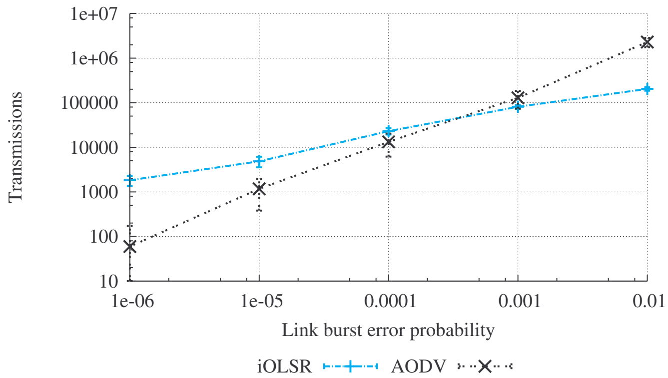

Fig. 8. Number of control packet transmissions for iOLSR contra AODV.

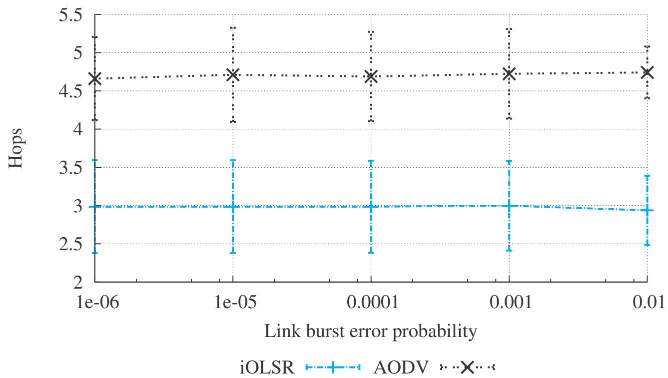

Fig. 9. Average number of hops for the data traffic for iOLSR contra AODV.

forwarding transmissions of the route requests are at risk of collisions, resulting in a failure to propagate the shortest path outwards to the destination. Furthermore, in case of low data traffic and low error rate, the result indicates the benefit of a reactive protocol. However, as traffic increases more traffic is traveling over more hops, thus draining more resources.

The consequence of the higher number of hops for the data traffic using AODV is the higher total number of transmissions (Fig. 10) where both the data traffic and control traffic transmissions are counted.

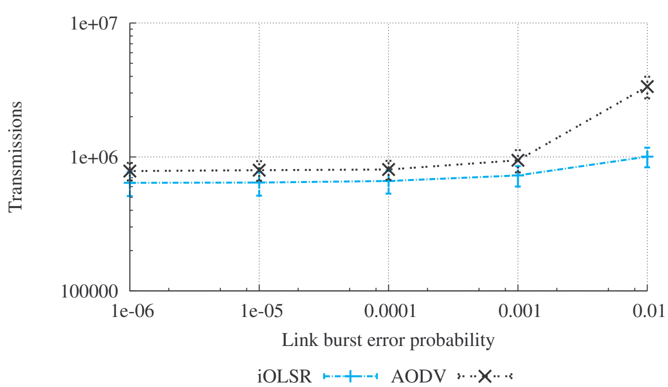

Fig. 10. Total number of transmissions (control and data traffic) for iOLSR contra AODV.

V. Conclusions and Future Work

This paper has presented an adaptation of OLSR for WSNs by introducing dynamically adaptive intervals. The advantages of employing dynamic intervals for control packets were demonstrated. We achieved less control packet overhead than by using the default control packet interval. Also, we demonstrated a faster detection and integration of new nodes than by using a large control packet interval.

The solution induces costs in terms of less route maintenance. Even so, the proposed solution represents a much better alternative for reducing the number of transmissions than that of preset large intervals, since it will depend on the real dynamics of the network whether the routing protocol transmits many or few packets. Last, but not least, using a proactive protocol provide the ability for a more optimized traffic pattern from sink to sensors.

Next, we will elaborate on the benefit of turning off the TC functionality to further reduce the control traffic. Destinations located further than two hops away would be searched for by a request using MPR for request forwarding.

References

[1] IPv6 over Low power WPAN working group. (2010, May) IETF. (Last accessed 2011-05-18). [Online]. Available: http://datatracker.ietf.org/wg/6lowpan/charter/

[2] H. Y. Shwe, X.-H. Jiang, and S. Horiguchi, "Energy saving in wireless sensor networks," *Journal of Communication and Computer*, vol. 6, no. 5, pp. 20–28, May 2009.

[3] T. Clausen, P. Jacquet (editors), C. Adjih, A. Laouiti, P. Minet, P. Muhlethaler, A. Qayyum, and L.Viennot, "Rfc 3626: Optimized link state routing protocol (OLSR)," RFC 3626, pages 1–75, pp. 1–75, October 2003, network Working Group.

[4] A. Benslimane, R. El Khoury, R. El Azouzi, and S. Pierre, "Energy power-aware routing in OLSR protocol," in *Mobile Computing and Wireless Communication International Conference, 2006. MCWC 2006. Proceedings of the First*, 17-20 2006, pp. 14 –19.

[5] S. Mahfoudh and P. Minet, "EOLSR: an energy efficient routing protocol in wireless ad hoc and sensor networks," *Journal of Interconnection Networks*, vol. 9, no. 4, pp. 389–408, 2008.

[6] M. Benzaid, P. Minet, and K. Al Agha, "Analysis and simulation of Fast-OLSR," *The 57th IEEE Semiannual Vehicular Technology Conference (VTC)*, vol. 3, pp. 1788–1792, April 2003.

[7] T. Clausen and U. Herberg, "Comparative study of rpl-enabled optimized broadcast in wireless sensor networks," INRIA, Research Report RR-7296, May 2010, (Last accessed 2011-05-18). [Online]. Available: hal.inria.fr/inria-00488030/PDF/RR-7296.pdf

[8] J. Heidemann and T. Henderson (Editors). (2009, October) Network Simulator 2. (Last accessed 2011-05-18). [Online]. Available: http://nsnam.isi.edu/nsnam/

[9] F. J. Ros and P. M. Ruiz. MANET Simulation and Implementation at the University of Murcia (MASIMUM). (Last accessed 2011-05-18). [Online]. Available: http://masimum.dif.um.es/

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

[11] E. N. Gilbert, "Capacity of a burst-noise channel," *Bell System Technical Journal*, vol. 39, pp. 1253–1265, 1960.

[12] E. O. Elliot, "Estimates of error rates for codes on burst-noise channels," *Bell System Technical Journal*, vol. 42, pp. 1977–1997, 1963.

[13] S. PalChaudhuri. Ns-2 code for random trip mobility model. (Last accessed 2011-05-18). [Online]. Available: http://monarch.cs.rice.edu/~santa/research/mobility

[14] C. Perkins and E. Belding-Royer, "Ad-hoc On-Demand Distance Vector Routing," in *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications*, February 1999, pp. 90–100, New Orleans, LA.

 ISBN: 978-1-61208-144-1