2009

OLSR-R^3: Optimised link state routing with reactive route recovery

Jerry Chun-Ping Wang
University of Wollongong, jerryw@uow.edu.au

Farzad Safaei
University of Wollongong, farzad@uow.edu.au

Mehran Abolhasan
University of Wollongong, mehran.abolhasan@uts.edu.au

Daniel R. Franklin
University of Wollongong, danielf@uow.edu.au

Recommended Citation

Wang, Jerry Chun-Ping; Safaei, Farzad; Abolhasan, Mehran; and Franklin, Daniel R.: OLSR-R^3: Optimised link state routing with reactive route recovery 2009, 359-362.
https://ro.uow.edu.au/infopapers/777

OLSR-R^3: Optimised link state routing with reactive route recovery

Abstract

The Optimised Link State Routing (OLSR) is a proactive routing protocol which relies on periodical broadcast of routing packets. However, due to the one-to-many relationship of broadcast traffic, the delivery of these packets can not be guaranteed by underlying MAC protocol, particularly in a congested condition. In this paper, the possible routing pathologies and failures of OLSR in a congested network are explored. In addition, a hybrid routing protocol which integrates OLSR with Reactive Route Recovery (OLSR-R3) is proposed to rectify the erratic routing behaviour described in this paper. Simulation studies are presented which show that the proposed solution is effective in addressing the underlining problems.

Keywords

optimised, 3, r, state, recovery, reactive, route, olsr, routing, link

Disciplines

Physical Sciences and Mathematics

Publication Details

Wang, J., Abolhasan, M., Franklin, D. R. & Safaei, F. (2009). OLSR-R^3: Optimised link state routing with reactive route recovery. Proceedings of the 15th Asia-Pacific Conference on Communication (APCC 2009) (pp. 359-362). USA: IEEE.

OLSR-R^3: Optimised Link State Routing with Reactive Route Recovery

Jerry Chun-Ping Wang, Mehran Abolhasan, Daniel R. Franklin, and Farzad Safaei
Information & Communication Technology Research Institute,
University of Wollongong, Wollongong, NSW 2522, Australia
{jcpw942,mehrana,danielf,farzad}@uow.edu.au

***Abstract*—The Optimised Link State Routing (OLSR) is a proactive routing protocol which relies on periodical broadcast of routing packets. However, due to the one-to-many relationship of broadcast traffic, the delivery of these packets can not be guaranteed by underlying MAC protocol, particularly in a congested condition. In this paper, the possible routing pathologies and failures of OLSR in a congested network are explored. In addition, a hybrid routing protocol which integrates OLSR with Reactive Route Recovery (OLSR-R^3) is proposed to rectify the erratic routing behaviour described in this paper. Simulation studies are presented which show that the proposed solution is effective in addressing the underlining problems.**

I. INTRODUCTION

The Optimised Link State Routing (OLSR) [1] is a proactive link-state routing protocol specifically tailored for multi-hop ad hoc network. The protocol, as name suggest, optimises the pure link-state routing protocol by propagating topology information via selected nodes known as multipoint relays (MPRs). This proposed mechanism promises to provide more efficient flooding of control messages, thereby alleviating the well-known “broadcast storm” [2] problems in the ad hoc networks.

Given the proactive nature of the protocol, each node requires periodical broadcast of two control messages, namely *HELLO* and *Topology Control (TC)* packets. The HELLO packet is used to discover the local 2-hop neighbours and perform MPRs selection at each node. The selection of MPRs must make sure that there exists a route to every 2-hop neighbour via selected MPRs. The TC packets which carries link state information then propagate to all nodes in the network via the relays of MPRs. After receiving sufficient link state information, the routes can be computed locally using shortest path algorithm (i.e. Dijktra Algorithm).

While the development of OLSR has been mature and is set to become part of IEEE 802.11s standard, the question remains on whether the lower protocol stack (particularly MAC protocols) can support the operation of OLSR. Given both HELLO and TC messages rely on broadcast transmissions, the IEEE 802.11 Distributed Coordinated Function (DCF) can only offer a minimal service quality for broadcast transmissions. The stations do not acknowledge received broadcast frames, nor do they have the ability to re-transmit in the event of packet loss. Therefore, when competing against data traffic (which typically is dominated by unicast data), the routing packets are prone to loss [3].

In this paper, the possible routing pathologies and failures of OLSR in a congested environment is explored. Based on these observations, a hybrid solution is proposed to enable reactive route recovery in OLSR. The specific contributions of this paper include:

- A study of routing pathologies in OLSR due to network congestion. (Section II)
- A framework of OLSR with Reactive Route Recovery (OLSR- R^3) - a hybrid routing protocol that rectifies the problems emerged as OLSR fails due to network congestion. (Section III)
- An evaluation of OLSR- R^3 which highlights the effectiveness of the proposed protocol. (Section IV)

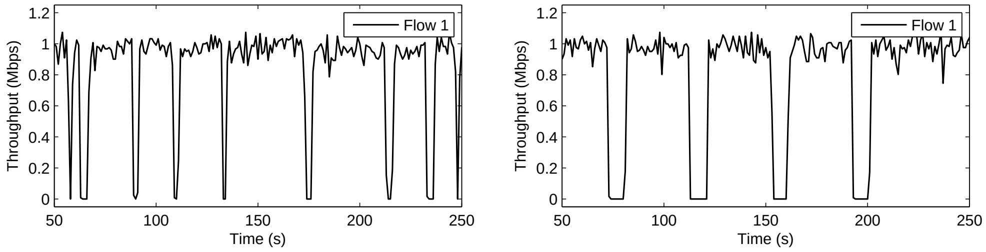
II. ROUTING PATHOLOGIES

This section demonstrates the impact of network congestion and possible routing pathologies on OLSR protocol.

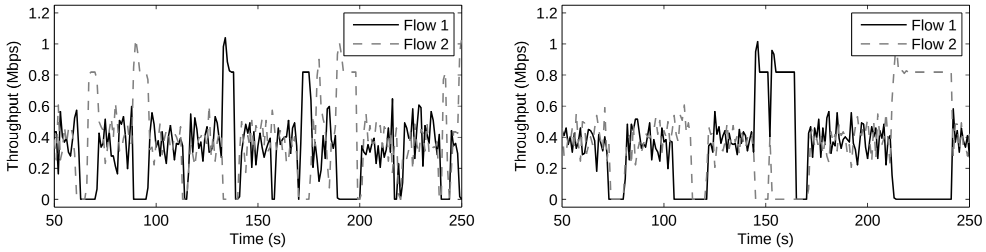
A. Simulation Setup

The evaluations were performed using Qualnet (version 4.0). Each station is equipped with an IEEE 802.11-compliant interface and an omni-directional antenna positioned 1.5 meters above the ground. The RF channel is represented by a Two-Ray Pathloss propagation model, and the data bitrate is set at a fixed rate of 11 Mb/s. Under these conditions, each station’s maximum transmission range is approximately 280 meters and its carrier sensing range is 500 meters. The *OLSRv2_Niigata* library is used as the choice of OLSR implementation. The simulation considers two different OLSR settings. The first OLSR setting adopts the default values from OLSR specification (HELLO interval = 2 second, TC interval = 5 seconds) , whereas the second OLSR setting doubles the default values (HELLO interval = 4 second, TC interval = 10 seconds).

The simulations consider several scenarios with one or more identical 6-hop unidirectional traffic flows. For each flow, the network traffic traverses 7 nodes with a hop distance of 250 meters between successive nodes. Figure 1 shows the topology for one-flow and two-flow scenarios. In the one-flow scenario, the network is simply a string topology of 7 nodes. In the two-flow scenario, two 6-hop linear flows share a common central node. All network traffic flows are constant bit rate (CBR) streams of 1024 byte UDP datagrams. The total offered load is 200 packets per second, equally distributed between



(a) OLSR (H=5,TC=2) - 1 Flow
(b) OLSR (H=10,TC=4) - 1 Flow

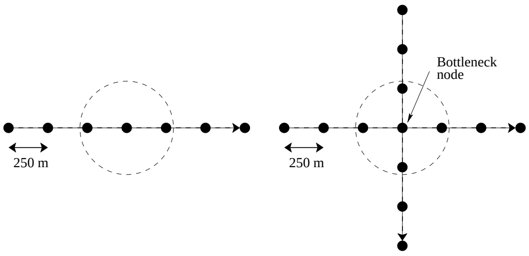


(c) OLSR (H=5,TC=2) - 2 Flows
(d) OLSR (H=10,TC=4) - 2 Flows

Fig. 2. Performance of ad hoc routing protocols in one-flow and two-flow scenarios

Routing Protocols	1-Flow Scenario					2-Flow Scenario				
	Throughput	Freq.	MTBF	MTTR	Avail.	Throughput	Freq.	MTBF	MTTR	Avail.
STATIC	0.977 *Mbps*	0.00	850.00*s*	0.00*s*	100.00*%*	0.384 *Mbps*	0.00	850.00*s*	0.00*s*	100.00*%*
OLSR (H=5,TC=2)	0.837 *Mbps*	35.52	21.38*s*	2.18*s*	90.91*%*	0.364 *Mbps*	48.64	18.97*s*	6.99*s*	73.55*%*
OLSR (H=10,TC=4)	0.821 *Mbps*	17.58	40.88*s*	6.44*s*	86.72*%*	0.361 *Mbps*	24.54	35.50*s*	15.93*s*	70.00*%*

TABLE I
NUMERIC PERFORMANCE OF ONE-FLOW AND TWO-FLOW SCENARIOS OVER A SERIES OF LONG RUNS (900 SECONDS)



(a) One flow
(b) Two flows

Fig. 1. One-flow and two-flow scenarios

the number of flows. This aggregated load is sufficient to cause network saturation in a 6-hop network. The simulation allows OLSR to setup the routing table in advance and all transmissions commence at 50th seconds of simulation time.

B. One-Flow Scenario

Figure 2(a) and 2(b) illustrate the network flow performance for first 200 seconds of one-flow scenario using different OLSR settings. According to the figures, the performance of OLSR in general maintains approximately stable performance in the one-flow scenario when OLSR is fully operational. As the congestion emerges, OLSR becomes less efficient due the loss of routing packets. This will subsequently lead to sudden drop in throughput. When using default OLSR setting, Figure 2(a) shows that the network flow suffers from frequent but transient route failures, which is largely driven by frequent route update. Interestingly, given the nodes are stationary, Figure 2(b) also indicates that doubling the OLSR interval settings does not necessarily yield better performance. The larger interval can reduce the chance of involuntary route change by updating routing information less frequently, but it also has the tradeoff of higher recovery latency as routing information is not refreshed quickly enough to re-establish the routes.

The statistical results presented in Table I highlights the performance of the network flows averaged over 30 runs, each lasting 850 seconds. As expected, OLSR fails to achieve the optimal performance (i.e. static routing performance) even in a simple string topology. The throughputs for both OLSR settings are approximately 15% lower than static routing. Moreover, doubling the OLSR setting in effect reduces the route breakage frequency by half and doubles the link life time. However, the link recover time becomes three times higher than using default setting. As a result, the overall performance for OLSR using larger interval is lower than OLSR using shorter interval.

C. Two-Flow Scenario

Figure 2(c) and 2(d) demonstrate three possible types of route failures in two-flow scenario. When the routes are available for both flows, the network capacity is equally shared by two flows. However, when the routes are only available to one flow, this would immediately cause an active flow to dominate the network capacity, while the other flow remains disconnected. Further, the congestion can also cause route failures for both flow and the throughput of both flow will drop to zero until the routes are recovered.

Figure 2(d) and Table I show that the route recovery time can be adversely affected by the dominant flow in two-flow scenario. Since the common node (shown in Figure 1(a)) is dominated by an active flow, it becomes more difficult to pass routing packet through common node. OLSR requires to exchange more HELLO and TC packets before the routes can be re-established. This will further delay the route recovery process for an inactive flow as there is only a limited amount of HELLO and TC packets can be transmitted within any given intervals. Hence, as shown in Table I, the route recovery time is drastically increased, particularly for the larger interval settings.

III. OLSR WITH REACTIVE ROUTE RECOVERY

According to Section II, the network congestion can cause OLSR to fail, resulting in frequent disconnections. This section presents Reactive Route Recovery (R^3) process which fills gap at time of disconnection. The combination of OLSR and R^3 forms a new type of hybrid routing protocol. The proposed protocol primarily relies on OLSR running at background and transforms to a reactive process when OLSR fails. The synergy of both proactive and reactive processes ensures the seamless data delivery over the duration of an end-to-end session.

The OLSR-R^3 combines and compliments the use of both proactive and reactive routing protocols without compromising existing benefits. In particular, the network still retains the advantage of lower transmission latency without suffering from frequent disconnections as shown in Section II. Moreover, since the routes are proactively maintained, OLSR-R^3 also prevents the reactive process from “over-reaction” - a situation where reactive routing protocols involuntarily trigger routing process due to congestion [4].

A. Reactive Route Recovery (R^3)

R^3 mimics the operation of Adhoc On-Demand Vector (AODV) without the need of route maintenance. When routes are not available at time of data transmission, R^3 will take the initiative to look for the routes by broadcasting a route request (RREQ) packet to its neighbours. The neighbours then relay this request message to their neighbours and so on and so forth until reaching the destination. After receiving the RREQ, the destination then sends a unicast route reply (RREP) packet back to the sender following the trace of RREQ packets. A symmetric path will be established between the requesting node and destination when RREP packets is successfully arrived at requesting node.

The routes generated by R^3 remains in the routing table until the routes can be fully recovered by OLSR. The current setting assumes life time of reactive path is *3 * HELLO INTERVAL + 2 * TC INTERVAL* where 3 * HELLO INTERVAL is the minimum time for OLSR to detect the route change and 2 * TC INTERVAL is the time to obtain new topology information. During this period, a conflict may emerge as routing table contains same destination entries from both OLSR and R^3. In this case, higher preference will be given to protocol with least amount of cost (i.e. number of hops).

B. Explicit HELLO Update

Since the routes generated by R^3 is temporary, OLSR must complete its proactive route recovery within the life time of reactive routes. Conventionally, the nodes rely on exchanging several HELLO packets to maintain symmetric links and obtain relevant link state information with their neighbours. However, the transmission of HELLO packet is not instantaneous - this can attract extended route recovery delay. To help expediting recovery process, the explicit HELLO update is proposed to utilise the exchange of R^3 messages by encapsulating proactive routing information in R^3 packets.

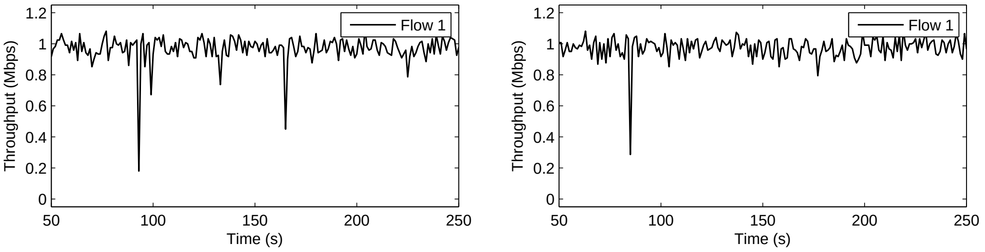
The explicit HELLO update allows OLSR to setup symmetric links and transmit on-demand HELLO packets by using RREQ and RREP packets respectively. The RREQ packets, which are flooded across the network, do not carry any additional information as it can induce excessive and redundant routing overhead. However, the reception of RREQ packets can ensure that there exists at least a uni-directional link between sender and receiver. Hence, OLSR should update the link status accordingly. Once the RREQ packets reach the destination, the HELLO packet will be encapsulated in RREP packets and returns to sender via unicast transmission. This process allows prompt delivery of HELLO messages via more reliable transmission. Moreover, the symmetric links and corresponding neighbour information will also be updated as RREP packets traverse back to the source. As a result, the routes between source and destination can be fully recovered by OLSR at next TC update.

IV. EVALUATION AND COMPARISON

A. One-Flow Scenario

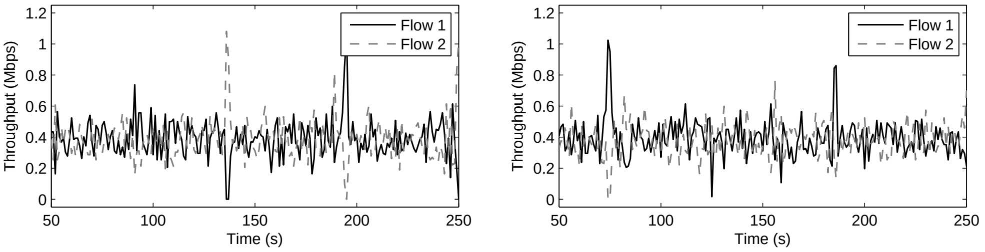
Figure 3(a) and 3(b) illustrate the performance of OLSR-R^3 in one-flow scenario. The spikes appeared in both Figure 3(a) and 3(b) represent the point where OLSR fails to maintain the route due to network congestion. While route failure is inevitable in a congested condition, OLSR-R^3 has demonstrated its ability to quickly recover the route using reactive recovery process. R^3 allows immediate recovery of route rather than waiting for the exchange of HELLO and TC packets. The network flow then returns to a steady state as soon as the route is recovered. The explicit HELLO update further extends the life time of routes and greatly reduces the frequency of route failures.

The numerical results shown in Table II indicates that the performance of OLSR-R^3 closely matches to the static routing. Given the routes are recovered in time of transmission, the route breakage frequency is almost close to zero. This directly



(a) OLSR-R^3 (H=5,TC=2) - 1 Flow

(b) OLSR-R^3 (H=10,TC=4) - 1 Flow



(c) OLSR-R^3 (H=5,TC=2) - 2 Flows

(d) OLSR-R^3 (H=10,TC=4) - 2 Flows

Fig. 3. Performance of OLSR-R^3 in one-flow and two-flow scenarios

Routing Protocols	1-Flow Scenario					2-Flow Scenario				
	Througput	Freq.	MTBF	MTTR	Avail.	Throughput	Freq.	MTBF	MTTR	Avail.
OSLR-R^3 (H=5,TC=2)	0.962 *Mbps*	0.06	824.47*s*	0.06*s*	99.99*%*	0.397 *Mbps*	15.5	74.48*s*	2.65*s*	96.76*%*
OLSR-R^3 (H=10,TC=4)	0.969 *Mbps*	0.04	832.98*s*	0.04*s*	99.99*%*	0.393 *Mbps*	11.17	100.96*s*	2.96*s*	97.41*%*

TABLE II
NUMERIC PERFORMANCE OF OLSR-R^3 IN ONE-FLOW AND TWO-FLOW SCENARIOS OVER A SERIES OF LONG RUNS (900 SECONDS)

reflects on the performance of end-to-end path, showing significantly improvement on route lifetime and approximately 100% of route availability.

B. Two-Flow Scenario

Figure 3(c) and 3(d) depict the behaviour of network flows using OSLR-R^3 in two-flow scenario. Unlike the results shown in Section II, both network flows appear to be stable and last longer. The capacity is fairly shared by the two flows. While both figures also have indicated that the network capacity can be occasionally dominated by one flow (i.e. sudden burst/drop of the traffic), such dominant condition is shown to be temporal and not as severe as previous case.

The use of R^3 can enhance the performance of OLSR, particularly for the larger HELLO and TC intervals. As shown in Section II, different OLSR intervals have the tradeoff between route failure frequency and route recovery latency. When using in conjunction with R^3, the larger interval can retain the advantage of lower route failure frequency while leveraging fast route recovery from R^3. Table II shows that OLSR-R^3 can keep the route recovery time under 3 seconds for both OLSR settings. This is close to 3 times enhancement for shorter interval and over 5 times improvement for the larger interval. Given the larger interval has lower route failure frequency, the larger interval has better overall performance than the shorter interval, which is different from the observation in Section II.

V. CONCLUSION

In this paper, the OLSR is shown to be unstable and inefficient in highly congested IEEE 802.11 ad hoc networks. It is observed that the network congestion can cause OLSR to fail due to the loss of important routing information. As a result, the end-to-end session experiences frequent but transient drop in throughput as the routes can temporarily unavailable.

A hybrid routing protocol - OLSR-R^3 is presented to rectify the erratic behaviour by integrating a reactive route recovery in OLSR. Simulations have proven the effectiveness of the proposed solution. Future work will focus on more complex scenarios as well as the introduction of mobility to further enhance the proposed hybrid routing protocol.

REFERENCES

[1] P. Jacquet, P. Muhlethaler, T. Clausen, A. Laouiti, A. Qayyum, and L. Viennot, "Optimized link state routing protocol for ad hoc networks," in *Proceedings of IEEE INMIC 2001*, 2001, pp. 62–68.

[2] Y.-C. Tseng, S.-Y. Ni, Y.-S. Chen, and J.-P. Sheu, "The broadcast storm problem in a mobile ad hoc network," *Wireless Network*, vol. 8, no. 2/3, pp. 153–167, 2002.

[3] J. C.-P. Wang, D. R. Franklin, M. Abolhasan, and F. Safaei, "Characterising the interactions between unicast and broadcast in IEEE 802.11 ad hoc networks," in *ATNAC 2008*, 2008.

[4] P. C. Ng and S. C. Liew, "Re-routing instability in IEEE 802.11 multi-hop ad-hoc networks," in *Proceedings of IEEE LCN '04*, November 2004, pp. 602–609.