Optimization of OLSR Routing Protocol with Effective Resource Management

Lokesh Kumar Laddhani
Institute of Computer Science
Vikram University, Ujjain (M.P.), India

Umesh Kumar Singh
Institute of Computer Science
Vikram University, Ujjain (M.P.), India

ABSTRACT
Mobile Ad hoc networks (MANET) allow a set of wireless hosts to exchange information without any special infrastructure. Limited battery power is one of the most important issues in mobile ad hoc network as the mobile nodes operate in limited battery power. Also there occurs a problem of broken links due to the lack of energy which cause disorder in network system. Such problem occurs due to the unawareness of energy of mobile neighbor nodes. Here, we choose Proactive Routing Protocol i.e OLSR because it is not Loop free and its communication overhead is high. Due to this here paper proposes Optimization of OLSR in aspect of Hello Message Interval and Topology Control (TC) time to meet Quality of Service (QoS) requirements between source and destination node pairs. The Hello Message Interval and Topology Control is treated like an optimization problem and techniques of Genetic Algorithms (GA) are used to solve it. The solution obtained after solving the optimization problem is in the form of effects on QoS while changing the values of parameter of OLSR using OPNET Modeller 14.5 simulator.

General Terms
MANET

Keywords
Mobile ad hoc networks; Routing; Hello message; Topology Control

1. INTRODUCTION
MANET is one of the most emerging fields in research and development of wireless network. As the popularity of mobile device and wireless networks increased significantly over the past years, it has now become one of the most vibrant and active field of communication in wireless technology.

MANET is a self configuring and infrastructure- less network. Each device or node is free to move independently, and will therefore change its links with other devices frequently in any direction. The primary challenge in creating a MANET environment is to continuously maintain the information required to route the traffic properly. Such networks can operate by themselves or by connecting itself to the larger Internet. They may contain one or more transceivers. This results in a highly dynamic and autonomous topology [1].

MANET has routable networking environment to process the exchange of information or packet from one node to other node. Here in this paper optimization of OLSR protocol through simulation for measuring the packet drop rate, the overhead introduced by the routing protocol, end-to-end delay of packet, network throughput, etc. This paper proposes an Optimization of OLSR through Hello messaging and TC Topology Control time interval establish a link and efficiently utilize the energy to enhance the life of Mobile Adhoc Network.

2. WORKING OF MANET ROUTING PROTOCOL

2.1 Route Discovery
When a node desires to send packets to a destination node, it first establishes a path to it for communication. The node begins the route discovery by broadcasting a route request (RREQ) message containing the IP address of the destination. When an intermediate node receives the RREQ, it records the reverse route toward the source and checks whether it has a route to the destination. If a route to the destination is not known, the intermediate node rebroadcasts the RREQ or if it has recent information about a route to the destination, route reply (RREP) message is generated. This RREP is unicast back to the source using the reverse route that is been recorded. When a RREP reaches the source, it begins to send data packets to the destination along the discovered path. If more than one RREP is received by the source, the route with the lowest hop count to reach the destination is selected [2].

2.2 ROUTE MAINTANENCE
This is the phase where the maintenance of link is preserved when broadcasting the packets. When a link breaks along an active path, the node upstream of the break detects the break and creates a route error (RERR) message. This message lists all destinations that are now unreachable, due to the link breakage and this information is sent to the source. Each intermediate hop deletes any broken routes and forwards the RERR packet towards the source. When the source receives this, it determines whether the packet still needs to be forwarded. If so, it begins the route discovery process for forwarding [2].

3. ENERGY CONSERVATION IN MANETS
Battery energy is said to be a rare resource, and it often affects the communication activities between nodes in network. Communication takes place through direct links or through multi hop links. Due to the limited battery energy of mobile nodes, the lifetime of node becomes the key challenge. Controlling the transmission power significantly reduces the energy consumption for sending data packets and also increase lifetime of network. Nodes adjust the transmission power so as to achieve the minimum energy consumption according to the local information. The idea of distributed power control can be used to improve energy efficiency of routing algorithm in MANET. There are some control messages such as RREP in On-Demand Routing Protocol which provide a strong indication that messages should trigger a node to switch to active node from sleep. Since the communication with a neighbor is only possible if the neighbor is in active mode, it is necessary for nodes to track energy modes of neighbors i.e., active, sleep or idle. The neighbor's power mode can be discovered in two ways: the first way is through explicit local HELLO message exchanges

with piggybacked information about the energy management mode of a node, and the second way is via passive inference [2].

Energy efficiency is measured by the duration of the time over which a network can maintain a certain performance level, which is usually called as the network lifetime. Using the power consumption is not only a single criterion for conserving energy efficiency. Hence, routing to maximize the lifetime of the network is different from minimum energy routing. Minimum energy routing sometimes attract more flows since the nodes in these route exhaust their energy very soon. Hence, the whole network cannot perform many task due to the failure of these nodes. Routing with maximum lifetime balances all the routes and nodes globally so that it can maintain certain performance level for a longer time.

Hence saving energy at the time of broadcasting in order to recover from the node failure and during re-routing around failed node is essential.

4. OLSR PROTOCOL

Optimized Link State Protocol (OLSR) falls under the class of proactive routing protocol and hence the routes are always available immediately when needed. OLSR is based on the link state protocol and is an optimized version for wireless networks taking into consideration the various issues in wireless data transmission. Mobile nodes topological changes cause flooding of topological information to available network hosts. To reduce network overhead, OLSR uses Multi Point Relays (MPR) as they lower broadcast flooding by reducing same broadcast in some network regions. Time interval reduction for control messages transmission brings reactivity to topological changes which are desired as it lowers control message bandwidth utilization. OLSR uses two differing control messages such as Hello and Topology control. Hello messages locate link status information and immediate neighbours to the host. Using the Hello message the MPR Selector set describes which neighbours choose the host to act as MPR and from this, the host calculate its own MPR set. The difference between Hello message and TC messages is that the former is sent only one hop away, whereas TC messages are broadcasted throughout the network. TC messages broadcast information about its own advertised neighbours including at least the MPR Selector list.

Only the MPR hosts can forward the TC messages which are broadcasted periodically. In an ad-hoc network, the link can be either be bidirectional or unidirectional which the host is required by to know about its neighbours. The Hello messages are broadcasted periodically to check the presence of the neighbour. Hello messages are broadcasted for only one hop nodes and are not forwarded further. When a node receives the Hello message from another node, it sets the host status to asymmetric in the routing table. When the first node sends a Hello message and includes that, it has the link to the second node as asymmetric, the second node set first node status to symmetric in its own routing table. Finally, when second node sends Hello message again, where the status of the link for the first node is indicated as symmetric, then first node changes the status from asymmetric to symmetric. In the end, both nodes knows that their neighbour is available, and the corresponding link is bi-directional. The Hello messages periodic broadcasting is used for link sensing, neighbour's detection and MPR selection process. Hello message also contains information about how often the node sends Hello messages, the degree of willingness of the node to act as a MPR, and information about its neighbour. Information about the neighbours contains interface address, the type of link which could be symmetric, asymmetric or lost and the type of neighbour. The neighbour type indicates just symmetric, MPR or not a neighbour. The MPR type indicates that link to the neighbour is symmetric and that this specific node has chosen it as MPR.

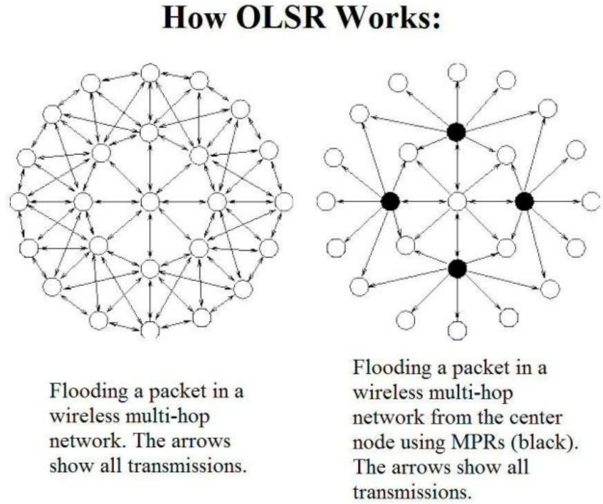

Fig.-1

The MPR is a unique concept in OLSR protocol to reduce the information exchange overhead which leads to lower bandwidth. Alternate to pure flooding, the OLSR uses MPR to reduce the number of the node which broadcasts the information throughout the network. The MPR is a node's one hop neighbour which can forward its messages. The number of nodes with MPR capabilities is kept minimal for the protocol to be efficient. To compute the efficient MPR set, each node must have information about the symmetric one hop and two hop neighbours. Hello messages provide information about the neighbours including two hop neighbours as each Hello message contain all the node neighbours. The goal of MPR selection algorithm is to select the minimum number of the one hop neighbours which covers all the two hop neighbours. On receiving a broadcast message which is to be sent to all the nodes in the network and the message's sender interface address is in the MPR Selector set, and then the node must forward the message. Due to the possible change in the ad-hoc network, the MPR Selectors sets are updated continuously using Hello messages.

Nodes selected as MPR send the TC message both to exchange topological information and build a topology information base. TC messages are broadcasted throughout the network through the MPR's, and they are generated and broadcasted periodically. A TC message is sent by a node to advertise its own network links. Minimum criteria for a node are to forward its MPR selector set links. The TC message includes own set of advertised links and each message's sequence number, which indicates message freshness and also avoids loops. Thus, if a node gets a message with smaller sequence number the message is discarded without updates. Nodes increase sequence number when links are removed from TC message and also when links are added to the message. When the node advertised links set is empty, it must still forward empty TC messages for a specific duration to invalidate earlier TC messages. Nodes maintain routing table information with destination address, next address, hops number to the destination and local interface address. Next address indicates next hop node with this information being got from a topological set and local link information base. If sets have any changes the routing table is recalculated. As OLSR is a proactive protocol the routing table has routes for all network hosts. Information on broken links/partially known links is not stored in the routing table. The routing table is updated if new neighbour link appear or a link disappears, two hops neighbour is either created or removed, topological link is available, lost or when the multiple interface association information changes. Though, the update of this information does not require sending of the messages into the network. Shortest path algorithm is used for finding the routes

4.1 State Maintenance in OLSR

An OLSR daemon must record and keep updated information in its internal tables to maintain network topology information in the presence of mobility or failure:

Tuples in a link set track link status between a node and neighbours and this status is of two types: SYM link (bi-directional) and ASYM link (uni-directional). A tuple contains local andneighbour nodes interface addresses (link end points), and validity time when a link is valid and useable.

A neighbour set includes neighbour tuples to track a node's neighbour status, including willingness and validity time, while 2-hop neighbour set records a set of 2-hop tuples describing symmetric links between neighbours and symmetric 2-hop neighbourhood. MPR set maintains a neighbour set chosen as MPRs, while MPR selector set records a set of MPR-selector tuples and describes neighbours chosen by this node as MPRs. Topology Information Base (TIB) maintains network topology information acquired by Topology Control (TC) messages and utilised for routing table calculations. Two types of control messages are used for topology information in OLSR protocol. They are HELLO and TC messages. A node forwards a HELLO message for self-identification and to report a list of neighbouring mobile nodes. A mobile node receives information from a Hello message about its immediate neighbours and 2-hop neighbours, and chooses MPRs accordingly.

MPR nodes originate TC messages revealing that what selected it as a MPR. These messages are relayed by other MPRs throughout the network, enabling remote nodes to discover links between MPR and selectors. Based on such information, a routing table is calculated using a shortest-path algorithm based on this information.

It should be understood that internal state information maintenance held at nodes is directly related to HELLO and TC messages exchange and thus anything like refresh timer intervals which are affected by such message generation in all likelihood will impact protocol performance.

4.2 Soft state in OLSR

OLSR uses a soft state approach to signaling. Routing state times out and if is not regularly refreshed by routing updates receipt are removed. Soft-state signaling needs no specific state/orphaned state removal when state installer crashes as non-refreshed state will finally be timed-out. Also, periodic refresh messages ensure robustness of the system against node failure to refresh messages loss/corruption. Refresh messages delivery guarantee is not required. State approaches were implemented in numerous protocols like RSVP, IGMP, SIP and also OLSR. The latter relies heavily on soft state approach to maintain topology information and routing tables consistency among network nodes. Hence, apart from normal periodic messages extra traffic is not generated by the protocol, in response to link failure and node join/leave events. OLSR soft state timers have two usage types: message generation and state maintenance. Message generation timers (HELLO and TC interval timers) send periodic HELLO and TC messages, while state-maintenance timers update state information in OLSR internal tables, removing obsolete states through time-outs (state holding timer for the neighbor set, link set and TIB. OLSR neighbor state holding time is set to 3 times the value of default OLSR HELLO interval; OLSR TIB holding time is 3 times default value of TC interval. TIB and link tuple timers' expiry interval equal TIB holding time interval. When new nodes join the network, a node forwards a HELLO message to new neighbours through a link-sensing process. Similarly when either a node leaves the network or links between nodes go down, thee the corresponding link set's link state and neighbor state in the neighbour set are discarded at expiry of state holding timers. Also, regular Topology Control (TC) messages help recovery from topology information loss due to state corruption/ nodes restarting. Thus, each node's internal state maintenance is related to refresh intervals. Changing them impacts the whole protocol.

5. RELATED WORKS

In order to assure all features of proactive routing protocol OLSR, there is an overhead imposed on any network owing to the emission of control messages. These control messages are created by the OLSR protocol. These control messages are sent by OSLR to enable it to maintain all its features during proactive routing and its control of message emission to the network. There is a general effort to reduce these overheads as it leeches useful bandwidth of the channels. In Ad-hoc networks consisting of notebooks, the amount of control data might not seem relevant compared to the networks of constrained environment. In the latter, networks where the sensors drain their batteries mostly by signal emission, the amount of control data emitted by the OLSR protocol is certainly an issue. Several techniques have already been adopted to adjust OLSR to give better results with sensor networks and also with other network deployments. The following strategies depict the most relevant proposals and techniques presented concerning OLSR optimization.

An interesting concept described in paper [3] presents an extension of the standardized OLSR routing protocol in order to make it more energy efficient. By means of energy information that is inserted to every HELLO and TC messages by every node, it presents a new routing policy and a new MPR selection policy. Instead of the shortest-path routing policy used in standardized OLSR it uses a one hop-by-hop energy efficient routing policy, where each node forwards the received packets towards the next hop on the minimum cost path. Also, it proposes an energy efficient selection of the MPRs where MPRs are selected according to their residual energy and any that are denoted as EMPRs. This approach can prolong the network lifetime by 50% compared to OLSR with a network of 200 nodes.

In [4] another appealing approach comprises an optimization scheme by reducing the size of the HELLO messages and the number and average size of the TC messages. It extends the standard neighbor tupple by adding a field named N modified. The value in this field indicates whether the link-state information was modified between two successive periods, or not. Using this field, the nodes do not describe the entire neighbourhood by their HELLO messages: Only the links which have been modified during the last HELLO interval are described. Similarly, in TC messages only the nodes and the links to those nodes whose neighbourhoods have changed are the links themselves announced. In most cases, it can decrease the routing overhead by about 17%. Here in this paper, implementation of proposed work is done.

Yet another interesting proposal presented at [5] suggests an extension that tries to decrease the overhead of the control messages exploiting a deployment of the MANET network into a form of units. These units are the predefined groups of the Users working together to accomplish a specific task. These units are set up manually by a User before OLSR initialization. In fact, it separates a network to the several smaller MANETs (groups) that have gateways nodes. A new field to HELLO messages is introduced specifying a predefined Group ID (GID). By distinguishing different predefined groups, only neighbours having the same GID are taken into account when constructing 2-hop neighbours set. The nodes hearing two networks with different GIDs become gateways. These gateways send a list of the nodes lying inside their group to all the gateway nodes. Afterwards, every node adjusts its routing table accordingly, in such a way that to reach every other node in the network it uses a proper gateway. An advantage of this proposal is that the number of TC messages does not depend on the number of MPRs but it depends on the number of gateways nodes. In denser topologies, there should be less gateway nodes than MPRs. This is because an MPR may be located inside a group and it does not act as a gateway node. Although it saves considerable overheads caused by some TC messages, it does not save much on lightweight routing tables. Furthermore, it requires the User to know a network topology in advance.

6. SIMULATION SETUP

The OPNET simulator is used to optimize the OLSR routing protocol. Firstly create the scenario of SoET **(School of Engineering and Technology, Vikram University, Ujjain**). A set of 20 and 30 mobile nodes are created with data rate of 11 Mbps and transmit power of 0.005 watts. Each node is capable of creating a raw unformatted data and is programmed to transmit data at t=100s. The destination for data transmission is selected randomly. Topology Control (TC) interval is fixed at 5 second with IPV4 as the addressing mode. Each node moves randomly within the network range of 1000 m X 1000 m. Evaluation was carried out for three scenarios by varying the mobility speed. Experiments were carried out at 20 Kmph, 60 Kmph and 100 Kmph. The simulation is run with ‘Hello’ intervals of 2 seconds, 4 seconds and 8 seconds. Simulations were carried for a period of 600 seconds in each case. Packet Delivery Ratio, Throughput, End to End Delay and Topology Control traffic were the parameters studied.

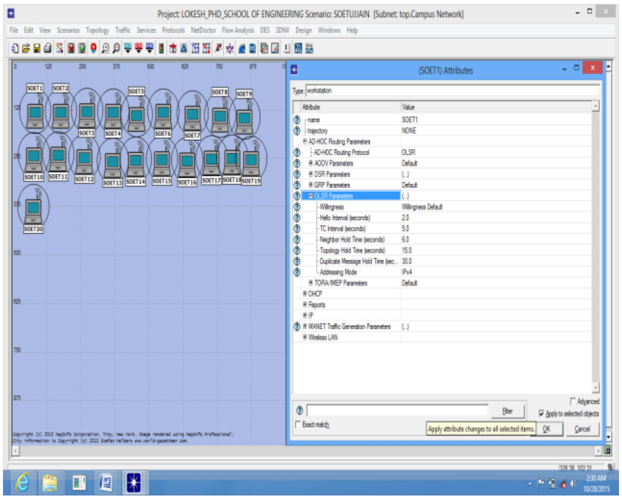

Figure 2. Scenario of School of Engineering & Technology, V.U., Ujjain of 20 Nodes with OLSR protocol parameter using OPNET Modeller 14.5

As the time between ‘Hello’ messages is increased among all the nodes, the total number of PDR decreases in the network as shown in Table no.1. It is also seen that the mobility speed in conjunction with the ‘Hello’ interval time affects the Packet Delivery Ratio. The PDR decreases from 0.98 to 0.721 when the ‘Hello’ message interval is increased from 2 to 8 sec. The degradation of the performance is drastically reduced when the time interval is greater than 6 sec. Similarly, with the increase in mobility speed the PDR decreases. When the speed is 20 Kmph, the PDR is 0.98 whereas with the increase in speed to 100 Kmph the PDR drops to 0.91, for message interval time of 2 seconds.

The packet end-to-end delay gives the average time required to navigate the packet inside the network. The delay time includes the time from creating the packet from source to the reception of the packet by destination and is expressed in seconds. This includes the overall delay of networks including buffer queues, transmission time and induced delay due to routing activities.

Table-1

Mobility Speed	**Time in Sec**			
	2 Sec	**4 Sec**	**6 Sec**	**8 Sec**
20 Kmph	0.98	0.979	0.934	0.721
60 Kmph	0.962	0.942	0.865	0.648
100 Kmph	0.91	0.862	0.764	0.566

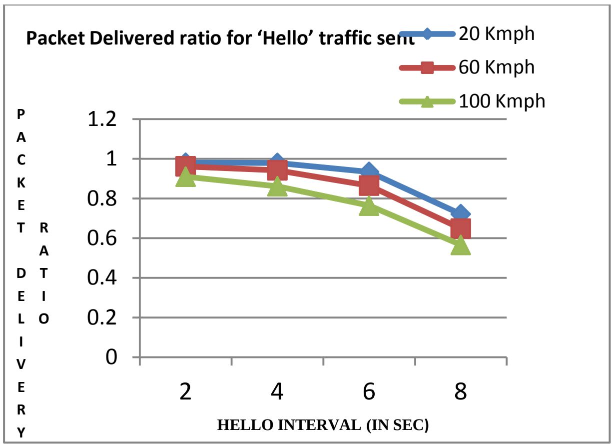

Figure.3

Table-2

Mobility Speed	Time in Sec			
	2 Sec	4 Sec	6 Sec	8 Sec
20 Kmph	0.020	0.021	0.066	0.279
60 Kmph	0.038	0.058	0.135	0.352
100 Kmph	0.090	0.138	0.236	0.434

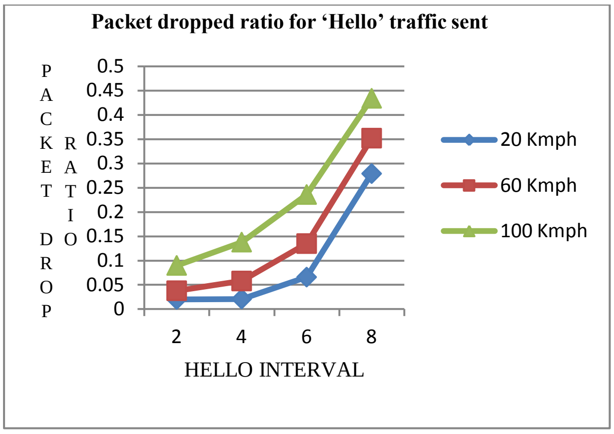

Figure.4

In above Table no.2, as we go through the analysis of the Packet Drop Ratio, increases from 0.020 to 0.090 for 2 Sec Hello message intervals. From the above table and Fig.4 we interpreted that as the speed of the nodes increases the Packet Drop Ratio also increases due change in the topology frequently.

Throughput is the measure of total amount of data reaching the receiver from the sender to the time it takes for the receiver to receive the last packet. It is represented in bits per second or packets per seconds. However, the throughput plays a crucial role for establishing the quality of service. The Throughput for various Hello interval are shown in Table-3 and Fig. 5.

Table-3

Sim. Time	HELLO INTERVAL (IN SEC)						
	2	4	8		2	4	8
180	116700	104232	10152	336	257216	233761.3	36245.33
186	111382.7	105370.7	8824	342	245509.3	230006.7	37816
192	137022.7	125549.3	8658.667	348	236404	223849.3	40714.67
198	118362.7	115385.3	8850.667	354	234813.3	232957.3	49765.33
204	140250.7	121120	10506.67	360	253598.7	267700	53853.33
210	212789.3	177980	11150.67	366	256846.7	232277.3	43865.33
216	256780	245610.7	13336	372	236824	228857.3	42489.33
222	255950.7	247254.7	44224	378	230060	231174.7	23650.67
228	246805.3	237596	38357.33	384	229345.3	254685.3	34293.33
234	276542.7	213513.3	30249.33	390	260430.7	209953.3	46742.67
240	244208	248830.7	33952	396	275124	229306.7	91782.67
246	251982.7	271510.7	36098.67	402	240136	222462.7	37361.33
252	236861.3	250937.3	31348	408	207757.3	233686.7	45456
258	238904	216592	41034.67	414	237808	238414.7	37609.33
264	233725.3	232002.7	44709.33	420	245004	226292	35546.67
270	244950.7	250102.7	86544	426	258514.7	222300	35661.33
276	250746.7	241514.7	61289.33	432	216104	221268	68856
282	275750.7	240282.7	31022.67	438	245509.3	237214.7	123209.3
288	241474.7	233261.3	23498.67	444	241613.3	239968	38990.67
294	250986.7	227742.7	28882.67	450	254092	256164	30816
300	256781.3	247922.7	54390.67	456	265600	243208	25452
306	259770.7	247080	29418.67	462	256114.7	221109.3	56682.67
312	241868	226648	40669.33	468	225738.7	239528	77732
318	242802.7	220678.7	59645.33	474	232489.3	223289.3	28844
324	244864	199089.3	40554.67	480	243296	306569.3	93133.33
330	264458.7	257738.7	43448	486	268922.7	235576	32582.67

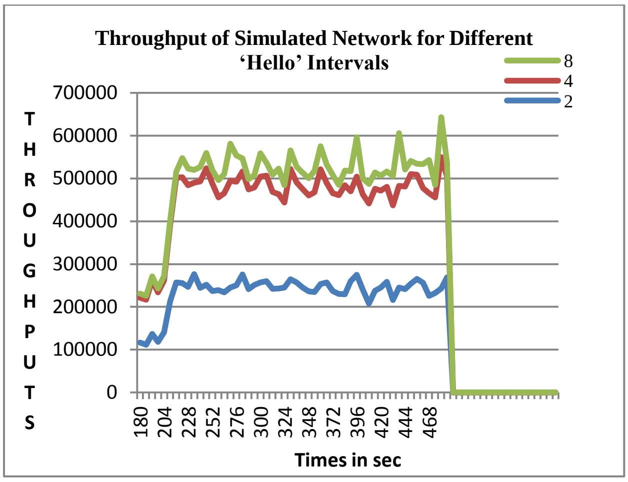

FIG.5

It is shown from the above Table-3 and Figure-5 there are very minor changes in the throughput when the 'Hello' message interval is increased from 2 to 4 sec. However, the throughput decreases drastically when the interval is increased to 8 sec.

Table-4

	Hello interval				Hello interval		
time (sec)	2	4	8	Time (sec)	2	4	8
180	0.000386	0.000405	0.00026	390	0.000386	0.000361	0.0003
186	0.000377	0.000414	0.000259	396	0.000395	0.000375	0.000748
192	0.00042	0.000424	0.000266	402	0.000368	0.000386	0.000313
198	0.000396	0.00038	0.000259	408	0.000367	0.000425	0.000333
204	0.000378	0.000401	0.000261	414	0.000392	0.0004	0.000279
210	0.000362	0.000387	0.000277	420	0.000374	0.00037	0.000326
216	0.000391	0.000364	0.000262	426	0.000386	0.000375	0.000299
222	0.000382	0.000366	0.000434	432	0.000389	0.000393	0.000554
228	0.000398	0.000399	0.000358	438	0.000428	0.000386	0.000604
234	0.000382	0.000394	0.000281	444	0.000414	0.000393	0.000316
240	0.000376	0.000398	0.000351	450	0.00038	0.000422	0.000282
246	0.000386	0.000545	0.00027	456	0.000386	0.000408	0.000282
252	0.000375	0.000429	0.000276	462	0.000388	0.000391	0.000489
258	0.000399	0.00039	0.000264	468	0.000402	0.000394	0.000485
264	0.000385	0.000396	0.000265	474	0.00039	0.000427	0.000274
270	0.000389	0.000407	0.000495	480	0.000386	0.000546	0.000773
276	0.000382	0.000395	0.000514	486	0.000369	0.000379	0.000355
282	0.000402	0.000388	0.000289	492	0.000357	0.000391	0.000576
288	0.000391	0.000405	0.000258	498	0.000384	0.000408	0.000349
294	0.0004	0.000395	0.000355	504	0.000402	0.000407	0.000291
300	0.000392	0.000403	0.00043	510	0.000389	0.000375	0.000483
306	0.000382	0.000448	0.000277	516	0.000389	0.000403	0.000299
312	0.000382	0.000374	0.000337	522	0.000372	0.000384	0.000284
318	0.000416	0.000407	0.000417	528	0.00038	0.000449	0.000525
324	0.00039	0.000384	0.000305	534	0.000398	0.000395	0.000388
330	0.000402	0.000398	0.000367	540	0.000365	0.000423	0.000293
336	0.000387	0.000384	0.000287	546	0.000393	0.000377	0.000278
342	0.000388	0.000394	0.000305	552	0.000425	0.000415	0.0003
348	0.000394	0.000388	0.000292	558	0.000432	0.000438	0.000276
354	0.000392	0.000395	0.000344	564	0.000433	0.000438	0.000275
360	0.000387	0.00038	0.000309	570	0.000412	0.000503	0.000274
366	0.000371	0.000407	0.000327	576	0.000385	0.000416	0.000308
372	0.00037	0.000391	0.000362	582	0.000407	0.000383	0.000283
378	0.000385	0.000424	0.000266	588	0.000403	0.000434	0.000291
384	0.000392	0.000447	0.000267	594	0.000422	0.000406	0.000328

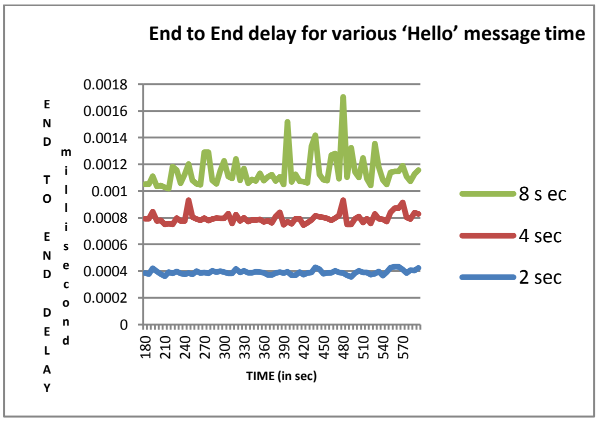

FIG. 6

It is inferred from the above Table-4 and Figure-6 the end to end delay is not very much affected when the 'Hello' message interval is increased from 2 to 4 sec. However, the end to end delay increases drastically when the interval is increased to 8 sec.

Table-5

	HELLO INTERVAL				HELLO INTERVAL		
Time (sec)	2	4	8	Time (sec)	2	4	8
180	8208	8917.333	666.6667	390	19904	19685.33	3760
186	9285.333	8101.333	682.6667	396	19605.33	18090.67	1930.667
192	8448	8592	490.6667	402	19040	18330.67	3200
198	7930.667	9445.333	624	408	15824	16560	3221.333
204	10314.67	8218.667	538.6667	414	17018.67	19690.67	3216
210	16880	14362.67	714.6667	420	21573.33	20288	2853.333
216	20005.33	21285.33	634.6667	426	20560	18101.33	2869.333
222	20570.67	21280	2144	432	15978.67	18309.33	2773.333
228	17504	17738.67	1957.333	438	15925.33	20000	2832
234	19850.67	17685.33	2245.333	444	17445.33	18933.33	3162.667
240	18842.67	19578.67	2560	450	18965.33	18458.67	2469.333
246	17770.67	17680	2837.333	456	21354.67	18469.33	1962.667
252	19824	17605.33	2682.667	462	18378.67	17301.33	2688
258	16453.33	16624	4282.667	468	16160	19424	2794.667
264	17744	18698.67	4704	474	16154.67	17200	2490.667
270	18949.33	18506.67	4261.333	480	19541.33	17989.33	2592
276	19146.67	19840	2586.667	486	22309.33	18992	2106.667
282	18762.67	18272	2282.667	492	17594.67	18320	2896
288	17824	18064	1898.667	498	16240	18698.67	2320
294	16944	17002.67	1818.667	504	17520	17648	2725.333
300	19413.33	19520	2240	510	19690.67	17349.33	2773.333
306	21066.67	17173.33	2496	516	19354.67	14746.67	2080
312	17861.33	18874.67	3397.333	522	18266.67	16773.33	1930.667
318	16784	16357.33	3440	528	16469.33	15930.67	3029.333
324	17450.67	17466.67	2544	534	17936	17770.67	4208
330	19984	20976	3568	540	18138.67	18266.67	2693.333
336	20250.67	18608	3130.667	546	17669.33	18661.33	1845.333
342	17744	17920	2597.333	552	15840	15120	3434.667
348	16805.33	17904	3098.667	558	16042.67	16128	1706.667
354	17061.33	18506.67	4074.667	564	17413.33	16480	2880
360	19472	22810.67	4058.667	570	15738.67	17610.67	3146.667
366	21066.67	16757.33	3525.333	576	17557.33	17968	3600
372	17472	18010.67	2250.667	582	15530.67	15082.67	2010.667
378	17146.67	16560	1957.333	588	16074.67	15280	2512
384	17797.33	17738.67	3338.667	594	16197.33	16373.33	4266.667

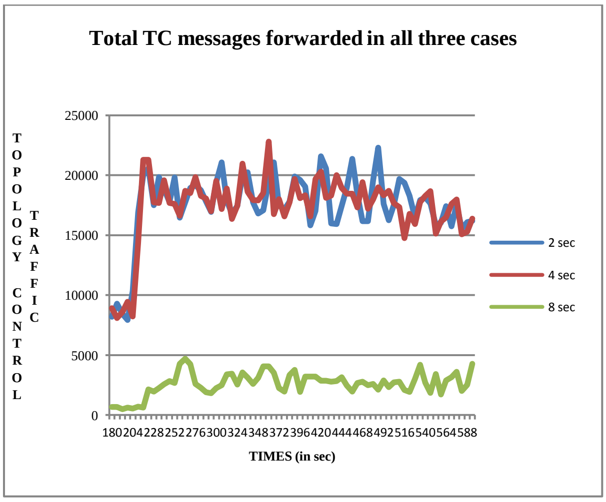

FIG.7

This attribute specifies the time interval between Topology Control (TC) messages. TC messages are generated by Multi-Point Relay (MPR) nodes to carry topology/connectivity information. These messages populate topology table of each node. This information is used for routing table calculations. These messages are flooded in a controlled manner by MPR nodes. From the above Table-5 and Figure-7 shows that TC traffic is initially increased suddenly and then remains somewhat constant regard to Topology Control interval. For 8 sec 'Hello' message interval the changes in the Topology Control traffic is not that much noticeable.

Table-6

	Hello Interval				Hello Interval		
180	126.833	125.333	18.500	312	295.167	294.500	69.167
186	135.833	123.167	17.167	318	308.167	250.167	68.167
192	133.500	124.833	13.333	324	281.500	256.000	58.833
198	128.833	138.667	17.167	330	271.000	315.167	70.167
204	154.333	140.333	18.833	336	255.333	284.167	67.667
210	262.833	227.833	20.500	342	293.833	275.667	54.167
216	342.167	323.667	24.000	348	330.833	270.500	65.833
222	332.667	344.167	48.000	354	292.167	273.333	76.167
228	294.333	275.333	44.333	360	270.500	336.333	83.500
234	287.667	254.500	49.167	366	246.500	263.667	73.500
240	293.000	298.833	53.833	372	284.833	269.833	52.500
246	279.000	261.167	59.500	378	326.833	255.000	45.500
252	279.167	274.333	54.000	384	294.500	267.667	66.500
258	272.667	259.833	81.333	390	255.500	291.500	79.667
264	294.000	278.333	81.667	396	266.833	288.667	46.667
270	287.333	279.833	88.000	402	311.667	276.000	62.000
276	273.833	294.000	55.167	408	296.167	261.500	64.500
282	275.667	279.667	50.833	414	284.500	287.333	69.500
288	303.833	272.333	44.833	420	257.500	297.000	60.167
294	308.167	264.500	42.833	426	281.167	289.667	57.333
300	263.833	282.500	51.333	432	317.500	277.333	58.167
306	272.833	266.500	52.667	438	303.833	297.500	61.500
444	298.500	290.333	66.000	528	278.833	238.500	60.500
450	299.000	282.167	57.333	534	261.333	265.500	77.167
456	265.500	285.667	47.333	540	247.167	273.667	59.667
462	292.667	261.000	60.000	546	268.833	282.667	44.833
468	301.000	292.833	60.500	552	261.667	225.667	65.667
474	274.167	255.000	51.167	558	261.500	237.500	39.333
480	292.833	283.833	52.167	564	254.333	233.833	60.000
486	277.000	297.000	51.667	570	235.333	255.500	62.000
492	295.833	284.667	57.000	576	253.500	263.333	68.667
498	287.167	274.500	49.000	582	239.333	252.333	45.833
504	285.667	268.167	55.000	588	246.167	229.667	53.000
510	269.667	269.500	52.667	594	266.667	237.667	80.333
516	265.833	238.833	47.333				
522	283.000	252.667	47.167				

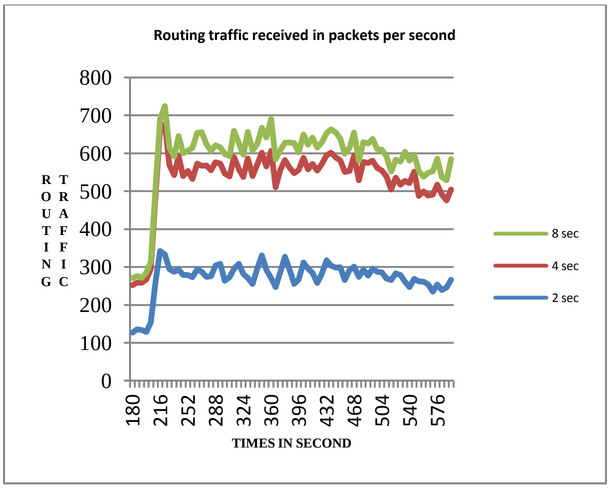

FIG-7

The TC messages are high for 2 sec and 4 sec of ‘Hello’ message time interval and decreases considerably when the time interval is 8 sec. From the simulation results, it can be inferred that the performance of the network is considerably better when the ‘Hello’ message time interval is between 2 to 4 seconds.

7. CONCLUSION

An ad-hoc network was simulated with 20 nodes moving randomly in an area of 1000 m x 1000 m with OLSR routing protocol. ‘Hello’ message time was varied from t=2, 4 and 8 seconds. The throughput, packet delivery ratio, end to end delay and control message traffic were also studied. From the simulation results, it is observed that for a moderately random movement of nodes in a network, the throughput of the network is not affected drastically when the time interval is changed from t=2 second to t=4 second. There is a considerable saving in the bandwidth which could be useful in bandwidth constrained networks. However, when the ‘Hello’ interval is changed to 8 seconds, the throughput is affected. This effect can decrease the quality of service provided. The entire goal is to optimize the performance of OLSR by optimizing the ‘Hello’ interval based on the type of network. Further work is required to improve the end to end delay and to decrease the control message traffic.

8. REFERENCES

[1] L. Laddhani and Dr. Umesh Kumar Singh “**Study the Characteristics of MANET Routing Protocol using Significant Factor”.** International Journal for Research in Applied Science & Engineering Technology, Volume 3 Issue X, October 2015.

[2] Laddhani L. and Singh Umesh “**Optimization of Dynamic Source Routing Protocol with Effective Resource Management”.** International Journal of Computer Applications (0975 – 8887) Volume 128 – No.18, October 2015.

[3] S. Mahfoudh and P. Minet. “**An energy efficient routing based on olsr in wireless ad hoc and sensor networks”**. Advanced Information Networking and ApplicationsWorkshops, International Conference on, 0:1253–1259, 2008.

[4] Y. Xue, H. Jiang, and H. Hu. “**Optimization on olsr protocol for lower routing overhead.**” In Proceedings of the 3rd international conference on Rough sets and knowledge technology, RSKT’08 pages 723–730, Berlin, Heidelberg, 2008. Springer-Verlag.

[5] J. Lebegue, C. Bidan, and T. Plesse. “**An olsr extension to deal with predefined groups”**. In Mobile and Wireless Communications Summit, 2007. 16th IST, pages 1 –5, july 2007.