© 2015 IJSRSET | Volume 1 | Issue 6 | Print ISSN : 2395-1990 | Online ISSN : 2394-4099
Themed Section: Engineering and Technology

Improving Quality of Service Support for Modified OLSR Protocol in Mobile Adhoc Network and Classified Nodes using Learning Techniques

Sapna Choudhary, Ankita Bhatt
Shri Ram Group of Institute, Jabalpur, Madhya Pradesh, India

ABSTRACT

In this paper focus on Nature of-administration (QoS) steering in an Ad-Hoc system is troublesome on the grounds that the system topology may change always and the accessible state data for directing is innately uncertain. In the proposal, we create QoS forms of the OLSR (Optimized Link State Routing) convention, which is an "expert dynamic" Ad-Hoc directing convention. We acquaint heuristics that permit OLSR with locate the greatest transfer speed way, indicate through reproduction and verification that these heuristics do enhance OLSR in the transmission capacity QoS perspective; we likewise dissect the execution of the QoS directing conventions in NS-3, watch the accomplishment acquired, and the expense paid. Our reproduction results demonstrate that the QoS renditions of the OLSR steering convention do enhance the accessible transfer speed of the courses figured, yet the included expense the extra overhead likewise has a negative effect on the system in End-to-End Delay and Packet Delivery Ratio, particularly in the rapid development situation and we detect packet dropper nodes for refining results by using support vector machine(SVM) learning technique for classification of dropper and normal nodes.
Keywords: *SVM, OLSR, QOS, NS-3*

I. INTRODUCTION

A Mobile Ad-Hoc network (MANET) [17] is a dynamic multi-hop wireless network that is established by a group of mobile nodes on a shared wireless channel. The nodes are free to move randomly; the network's topology changes rapidly and unpredictably. The Ad-Hoc network may operate standalone, or may be connected to the larger Internet. An example application of Ad-Hoc network is that a group of soldiers move in outdoors while communicating with one another through the radios. Without a central controller to control the communications in the network, without a fixed topology, the most difficult task the Ad-Hoc network faces is routing. Much work has been done on routing in ad-hoc networks, but most of them focus only on best-effort data traffic. However, recently, because of the rising popularity of multimedia applications and potential commercial usage of MANETs, QoS support in Ad-Hoc networks has become a topic of great interest in the wireless area.

1.1 Motivation

Quality-of-service (QoS) routing in an Ad-Hoc network is difficult because the network topology may change constantly and the available state information for routing is inherently imprecise.

To support QoS, the link state information such as delay, bandwidth, jitter, cost, loss ratio and error ratio in the network should be available and manageable. However, getting and managing the link state information in a MANET is by all means not trivial because the quality of a wireless link changes with the surrounding circumstance. Furthermore, the resource limitations and the mobility of hosts add to the complexity. In spite of these difficulties, some protocols on QoS routing in MANETs have been proposed, such as CEDAR [2] or ticket-based probing [5]. These protocols provide on-demand routing, where a route is found based on the pre-known QoS requirements.

There are many best-effort routing protocols targeting pro-active routing, but relatively little work has been done on pro-active QoS routing. However, the unpredictable nature of Ad-Hoc networks and the requirement of quick reaction to QoS routing demands make the idea of a proactive protocol more suitable. When a request arrives, the control layer can easily check if the pre-computed optimal route can satisfy such a request. Thus, waste of network resources when attempting to discover infeasible routes is avoided. Based on this consideration, in the thesis, we study the approach of pro-active QoS routing, and modify a best-effort pro-active routing protocol OLSR [12] for QoS purpose1. The QoS requirement studied in the thesis is the bandwidth constraint.

Many QoS components should work together to support QoS in Ad-Hoc networks [7]: a QoS model specifies which kinds of services to be included in the network; a QoS routing scheme searches a path with satisfactory resources defined by the QoS model; a QoS MAC protocol solves the problems of medium contention; a QoS signaling protocol performs the resource reservation along the path computed by the QoS routing protocols. Among all these components, QoS routing is a key issue.

The goals of QoS routing are 1) selecting one or more network paths that have sufficient resources to meet the QoS requirement of connections, 2) provide resource information of the path for admission control (call acceptance) mechanism, and 3) achieving global efficiency in resource utilization.

The problem of QoS routing in Ad-Hoc network is difficult. First, to support QoS, the link state information such as delay, bandwidth, jitter, cost, loss ratio and error ratio in the network must be available and manageable. However, getting and managing the link state information in MANET is by all means not trivial because the quality of a wireless link changes with the surrounding circumstance. The larger the size of the network, the more difficult it is to gather the up-to-date information. Second, the resource limitations and the mobility of hosts make things more complicated.

II. METHODS AND MATERIAL

A. RELATED WORK

The existing research on QoS Routing for Ad-Hoc networks can be divided into two categories: QoS route information and QoS route computation. QoS route information provides the QoS information over the path it constructs using traditional best-effort routing algorithms. Such information helps the source node to fulfill the "call admission" task. QoS route computation calculates feasible routes based on various QoS requirements.

2.1 QoS Route Information

Chen et al. [6] propose a bandwidth-constrained routing algorithm. Each node calculates the available bandwidth over the wireless links to the destination. Such bandwidth information is piggybacked in the "Destination Sequence Distance Vector" (DSDV) routing algorithm [19]. Thus, each node knows the bottleneck bandwidth over the paths calculated by DSDV to all known destinations.

Lin and Liu [5] have a similar approach using DSDV. Focusing on bandwidth control, bandwidth information is embedded in the nodes' routing tables and sent to the neighbors. Upon receiving a routing table from a neighbor, a node updates its own routing table and the path bandwidth information. With the bandwidth information, a node can decide whether or not it should accept a new connection request based on the bandwidth requirement of that connection.

These kinds of routing protocols are actually traditional best-effort Ad-Hoc routing protocol, and they do not attempt to find routes with satisfactory QoS conditions. The only difference is that the QoS state information (ex. bottleneck bandwidth) over the path computed by the best-effort routing protocol is available, and call admission control (the source node decides whether a new call should be accepted or not based on the requested QoS conditions) can be carried out.

Such an approach is easy to understand and implement. However, the path that the existing best-effort routing protocol computes does not necessarily have sufficient resources to meet the QoS requirement. Connection requests may be rejected mistakenly if there is another path in the network that can meet the QoS requirement. As a result, the network resource is not fully used.

2.2 QoS Route Computation

The work done in ”QoS routing computation” addresses two basic QoS routing tasks defined in [4] – “link-constrained routing” and “link-optimization routing”.

2.2.1 Link-Constrained Routing

The basic idea of link-constrained routing is “on-QoS-demand” routing. The task of QoS routing algorithms is to find a feasible route that meets the predefined QoS requirement. **Chen-Nahrstedt Algorithm**

Chen and Nahrstedt [5] propose a “ticket-based probing” algorithm. A ticket is a permission to search for a path. When a source wants to find a QoS path to a certain destination, it issues a number of tickets based on the available state information. More tickets are issued for connections with tighter requirements. Probes (routing messages) are sent from the source towards the destination to search for a low-cost path, which satisfies the QoS requirement. At intermediate nodes, a probe that carries more than one ticket can split into multiple ones, each searching a different sub-path. Based on its local state information, the intermediate node decides how and where the received probe should be split and forwarded. A probe can only continue traveling along the path if the QoS condition along the path does not violate the QoS requirement, and it carries at least one ticket. When the destination host receives a probe message, a feasible path is found. In the procedure of path searching, a probe also accumulates the cost of the path it traverses. If there are multiple probes arriving at the destination, the path with the least cost is selected as the primary path; the others are kept as secondary paths, and will be used if the primary path is broken due to the nodes movement. As a probe can only search a path with a valid ticket, the routing overhead is bounded by the tickets issued.

The “Ticket-based probing” is a general QoS routing scheme, which can handle different QoS constraints. In [5], the authors give two examples – delay-constrained routing and bandwidth-constrained routing, and explain in detail how to determine 1) how many tickets should be issued in the source node, and 2) how to split and forward the received tickets in the intermediate nodes.

Besides “tickets”, another innovative idea in [5] is the concept of “stationary and transient links”. A stationary link tends to be stable for a long time while a transient link is highly changeable. In the tickets splitting and forwarding procedure, the routing algorithm makes sure that the stationary links have a high priority to receive tickets, which ensures that the paths found are relatively stable.

B. PROPOSED WORK

Packet Dropping in OLSR

OLSR uses Multipoint Relays (MPRs) which are set of neighbouring nodes that are responsible for spreading the local link state information to the whole network for optimization. The link state is broadcasted periodically through Topology Control (TP) messages. Each node in OLSR selects its MPR set from its one hop neighbours such that it can easily reach all its two hop neighbours with minimum number of retransmissions. Selection of the MPR depends on the number of two hop neighbours reachable through the Candidate node and its “Willingness value obtained from “Hello” message which indicates The readiness of a node to forward packets of its neighbours.

Through periodic exchange of link state, each node senses its neighbours and disseminates the Network topology. Each node constructs a partial topology graph of the network from broadcasted TC messages which allows it to establish routes to non-neighbouring nodes. For a packet dropping attack, a malicious node may send a TC message claiming to be a MPR of Nodes although it may not. As the network depends on the MPRs for routing services, the malicious node may decide to drop packets passing through it.

Author approach is based on the following idea: Let A, B and C be three nodes which succeed in the data path. The node A holds the value € precalculated from values α (owned by A), β (owned by B) and μ (owned by C). To acknowledge the message msg sent from A through B, the node C sends back its value € to B, and B sends back the received value € and its value β to A. When A receives β and €, it recalculates € from α (its own value),β and μ. If the recalculated value € is the same that already held, so msg was well delivered by B, else B is a possible dropper node.

Packet Dropper Detection Algorithm

Following steps we follow for classification of packet dropper node and normal node: we have store the routing table in XML file and after that we will apply XML as a input to SVM

Step 1. Generates randomly an initial population of size based on routing table of nodes generated by ns3 simulator.

Step 2.Training SVM Classifier. SVM classifier is trained by training set with feature subset selected and variable value of parameters.

Step 3. For each set of the population, train SVM Classifier for computing fitness of each 2 subset of features.

Step 4. Select individuals from population directly based on fitness values and regenerate new individuals from old ones.

Step 5. If the maximum number of iteration is not yet reached, we proceed with the next generation operation. The termination criteria are that the max generation number reached or the fitness function value does not improve during the last generations return to step 2.

Steps 6. Select the best fitness as optimal subset feature in this step we got the result of packet dropper node and normal node.

Steps 7. Apply the optimal feature to dataset (routing table). for better approximation of dropping node we have choose following metrics to conjunction with threshold metrics [€, α, β, μ], they are listed below-

1. Packet Delivery Ratio (pd)
2. Packet Modification Ratio (pm)
3. Packet miss routed ratio (pm_r)
4. Residual Energy (re)

Now authors [1] metric will be modified and calculated using above metrics (assuming A, and C is MANET Node)

$$€ \longrightarrow f(pd, pm, pm\_r, re)$$

and same for other metrics α, β, μ.

Fundamentally we will find packet dropper node with normal node and simulation setup on NS3.

III. RESULTS AND DISCUSSION

SIMULATION AND RESULT

Our simulation done in NS-3 the ns-3 simulator is a discrete-event network simulator targeted primarily for research and educational use. The ns-3 project, started in 2006, is an open-source project developing ns-3. The purpose of this tutorial is to introduce new ns-3 users to the system in a structured way. It is sometimes difficult for new users to glean essential information from detailed manuals and to convert this information into working simulations. In this tutorial, we will build several example simulations, introducing and explaining key concepts and features as we go.

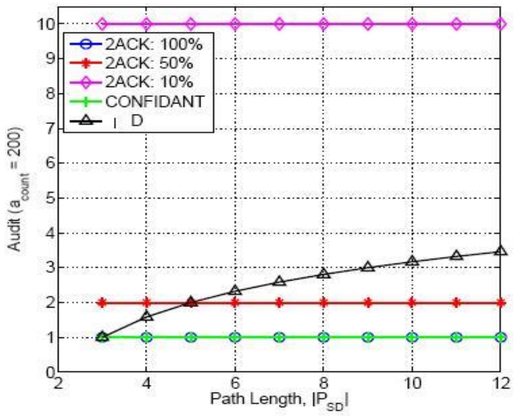


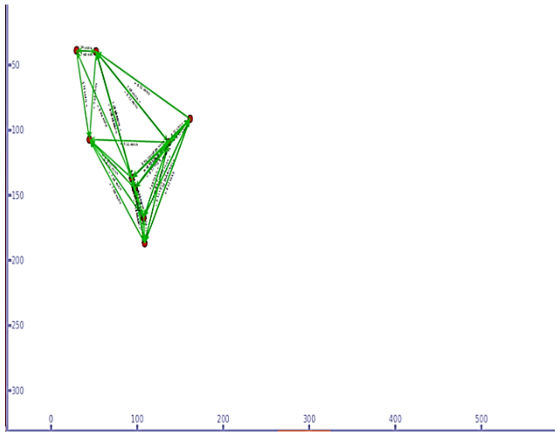
Figure 1 : Comparison graph of 2ACK algorithm with Packet dropper method

Simulation Parameter:

Our proposed method will be tested under **NS-3.20** on Ubuntu 14.04 system Steps:

Processor and sensing capabilities	SA 1100
Power for a node	Single 3.4v dc
Simulation area	1000*1000 Meter
Data Transmission renge	1 mb/s up to 10 meter
Data Packet size	2500 byte
Data flow rate	20 kb/se
Mobility model(topology)	Random way point mobility model
Routing protocol	OLSR
Nodes	25,50,75,100

This shows the number of nodes perform simulation and communicate to each other using NS-3 Simulator and genrate routing table.



IV. CONCLUSION

In this paper focus on the added overhead is the main cost that affects the QoS routing algorithm's performance, we have applied SVM learning techniques for refinement of result in term of classification of dropper and normal nodes and the future work on QoS routing in Ad-Hoc networks may be focused on how to reduce the overhead. The above future work targets on QoS version of OLSR. However, it is also interesting to design and implement the pro-active QoS routing based on other best- effort Ad-Hoc network routing protocols to see their performance. Thus, we may get an idea which kind of the QoS routing protocol is more suitable for Ad-Hoc network, link-constrained routing or link-optimization routing.

V. REFERENCES

[1] G. S. Ahn, A. T. Campbell, A. Veres and L. H. Sun, "SWAN: Service Differentiation in Stateless Wireless Ad-Hoc Networks", IEEE Infocom 2002, pages 457-466, June, 2002

[2] G. Apostolopoulos, R. Guerin, S. Kamat, and S. K. Tripathi, "Quality of Service Based Routing: A Performance Perspective", Association for Computing Machinery's Special Interest Group on Data Communication '98, pages 17-28, September 1998

[3] S. Chen and K. Nahrstedt, "Distributed Quality-of-Service Routing in High-Speed Networks Based on Selective Probing", IEEE Local Computer Networks (LCN), pages 80-89, October 1998

[4] S. Chen, and K. Nahrstedt, "An Overview of Quality-of-Service Routing for the Next Generation High-Speed Networks: Problems and Solutions", IEEE Network Magazine, Vol.12, No.6, pages 64-79, November 1998

[5] S. Chen, and K. Nahrstedt, "Distributed Quality-of-Service Routing in Ad-Hoc Networks", IEEE Journal On Selected Areas In Communications, Vol.17, No.8, pages 1488-1505, August 1999

[6] T.-W. Chen, J.T. Tsai and M. Gerla, "QoS Routing Performance in Multihop, Multimedia, Wireless Networks", IEEE International Conference on Universal Personal Communications '97, Part 2, pages 451-557, October 1997

[7] E.W. Dijkstra, "A Note on Two Problems in Connection with Graphs", Numerical Analysis 1, pages 269-271, October 1959

[8] M. Gerla and J. Tsai, "Multicluster, Mobile, Multimedia Radio Network", ACM- Baltzer Journal of Wireless Networks, pages 255-265, 1995

[9] R. Guerin, and Orda. Willimas, "Qos Routing Mechanisms and OSPF Extensions", draft-qos-routing-ospf-00.txt, Internet-Draft, Internet Engineering Task Force, November 1996

[10] A. Iwata, C. C. Chiang, G. Pei, M. Gerla and T. Chen, "Scalable Routing Strategies for Ad-Hoc Wireless Networks", IEEE Journal on Selected Areas in Communications, Vol.17, No.8, pages 1369-1379, August 1999

[11] P. Jacuet, P. Minet, P. Muhlethaler and N. Rivierre, " Increasing Reliability in Cable-Free Radio LANs Low Level Forwarding in HIPERAN", Wireless Personal Communications, Vol. 4, No. 1, pages 51-63, January 1997

[12] P. Jacquet, P. Muhlethaler, A. Qayyum, A. Laouiti, L. Viennot, and T. Clauseen, "Optimized Link State Routing Protocol", draft-ietf-manet-olsr-0.6.txt, Internet Draft, Mobile Ad-Hoc Networking Working Group, March 2002

[13] D. B. Johnson and D. A. Maltz, "Dynamic Source Routing in Ad-Hoc Wireless Networks", Mobile Computing, Kluwer Academic Publishers, ISBN: 0792396979, Chapter 5, pages 153-181, 1996

[14] L. Kleinrock and K. Stevens. "Fisheye: A Lenslike Computer Display Transformation", Technical report, UCLA, Computer Science Department, UCLA, CA Tech Report, 1971