# USING ONE-WAY-DELAY AS QOS ROUTING METRIC IN MOBILE AD HOC NETWORKS

*Mehmet Şimşek*[#1], *Nurettin Doğan*[*2]

[#]Faculty of Engineering and Architecture, Department of Computer Engineering, İzmir Katip Çelebi University Çiğli İzmir Turkey, [1]mehmet.simsek@ikc.edu.tr

[*]Faculty of Technology, Department of Computer Engineering, Gazi University Teknikokullar Ankara Turkey, [2]ndogan@gazi.edu.tr

## ABSTRACT

The main metric that is used in the computer networks to packet forwarding is "shortest path". The shortest path is decided according to some other sub metrics. In wired networks these metrics can be more complex. Generally, in Mobile Ad Hoc Networks (MANETs) simpler metrics are used such as hop count. In this study we suggest a new routing metric for MANETs: one-way-delay (OWD). Simulation results showed that OWD can be used alone as a routing metric and with other metrics.

***Index Terms***— One-way-delay, routing, metric, mobile ad hoc networks

## 1. INTRODUCTION

MANETs are networks which are constructed on demand. In a MANET there are no dedicated devices to special network jobs such as routing. Devices in a MANET can act as an end user device and as a router. MANET routing protocols can be classified as proactive, reactive and hybrid according to their behaviors [1]. In reactive routing protocols, source node sends route request packet as flood. When destination node receives route request packets it sends route reply packets to destination. If source node receives more than one route reply packets, it selects the shortest path according to a metric such as hop count. Established paths may not be symmetric due to connection behaviors of MANETs and medium access conflicts. The latency of a packet from the source to the destination and from the destination to the source can be different. This situation causes to select the path that has shortest round-trip time as shortest path from the source to the destination. If destination node only replies the firstly received route request packet, the path that has shortest delay can be selected. But it may not be healthy to decide only base on this information. Also, if route reply packet is lost, the path can't be established or route establishment can be delayed. To overcome this problem, hop count, one-way-delay and intermediate node's traffic load must be considered together. In this study, we suggest that usage of OWD, hop count and intermediate nodes' queuing delays of hops.

## 2. PREVIOUS WORKS

To use OWD as a routing metric, we need to measure it firstly. OWD has some components. These are queuing delay, processing delay and transmission delay. Processing delay depends on device processing unit capacity. Transmission delay depends on link capacity and packet length. Queuing delay depends on device's packet queue capacity and instant queue length. So, we can say processing delay and transmission delay are deterministic, queuing time is stochastic [2]. In the literature there are some OWD measurement/estimation methods [2-9]. Most of these studies measure/estimate deterministic components of OWD. In [10] a method called as adaptive per hop differentiation (APHD) has been suggested to measure queuing delay (and to satisfy end-to-end delay requirement) which is the stochastic component of OWD on MANETs. In APHD, data packets carry some header fields: e2e-delay-req, e2e-hops, delay-so-far, hop-so-far, prio. Each intermediate node adds experienced queuing delay to related field. Thus, at the last node total end-to-end queuing delay is obtained.

End-to-end queuing delay measurement has been used to delay assurance or performance assessment in the literature. In this study we used queuing delay as a routing metric in MANETs innovatively.

## 3. SUGGESTED METHOD

To use queuing delay as a routing metric we followed APHD's definitions. Our routing scheme has following packets: Route Request (RREQ), Route Reply (RREP), Route Probe (RP) and Route Probe Reply (RPR). We used source routing scheme to allow the source node to control and monitor the whole path.

### 3.1. The Structure

In this section we defined the packet structures. All packet structures are similar to any other source routing protocol such as DSR. So, we did not repeat required common fields here. We just gave new packet fields.

RREQ packet has following fields: *address[1..n]* and its corresponding *queueing_delay[1..n]*.

RREP packet has following fields: *address[1..n]* and its corresponding *queueing_delay[1..n]*.

RP packet is identical with RREQ except *type*.

RPR packet is identical with RREP except type.

### 3.2. The Working Principle

When the source node creates a RREQ packet, sends it as flood. Each intermediate node receiving RREQ, adds its address and queuing delay value to the packet. Queuing delay value is difference between RREQ's entry time to the queue and the exit time from the queue. Thus, queuing times of all intermediate nodes and hop count are obtained. When the destination node receive a RREQ packet it creates a RREP packet immediately and puts addresses and queuing delay values which were obtained from RREQ. RREP is directly sent to the source node. As a result, the source node has total end-to-end queuing delay of the path, individual queuing delays on each intermediate nodes and hop count. The source node can decide the path using one or more of these metrics.

When a path is established the communication starts over the path. After that, the source node creates RP packets periodically and sends it to the destination over the path. RP packet is identical with RREQ except *type*. Each intermediate node receiving RP adds address and queuing delay value to the packet and sends (not floods) it to the next hop. When the destination node receive a RP packet it creates a RPR packet immediately and puts addresses and queuing delay values which were obtained from RP. RPR is directly sent to the source node. Thus, the source node can monitor path state and availability and if it needs a new path can be searched. In order to decide whether there is a need to search for a new path we defined a threshold called as *delay_thresh*. *delay_thresh* is a variable that holds acceptable maximum end-to-end delay of a path in terms of milliseconds. If obtained end-to-end delay value from RPR packet exceeds *delay_thresh*, the source creates a new RREQ packet and searches a new path.

## 4. MATERIALS AND METHODS

We tested the suggested method with ns2 simulator [11]. We created a simple topology to analyze behavior of the suggested method. The simulation topology is shown in Fig. 1. In the all simulations end-to-end queuing delay was used as routing metric. *delay_thresh* was set as 100 ms. For a simple comparison Ad Hoc On Demand Distance Vector (AODV) routing protocol has been used. All simulations were repeated 10 times.

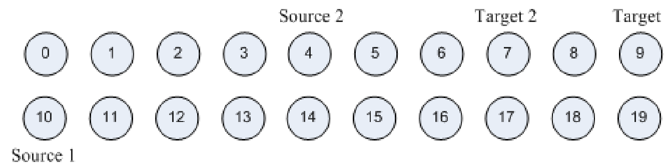


Fig. 1 The simulation topology

In the first simulation we created a data flow from Source 1 to Target 1. Meantime, there was no data flow from Source 2 to Target 2. The established path in all experiments is: 10-11-12-13-14-15-5-6-7-8-9. The path has shown in Fig. 2.

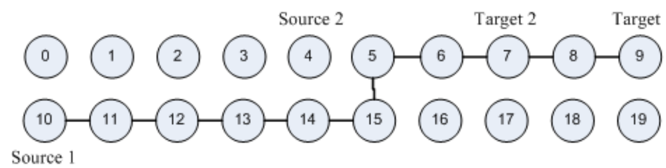


Fig. 2 The established path in the first simulation

In the second simulation firstly we created a data flow from Source 2 Target 2. After a few seconds we created a data flow from Source 1 to Target 1. In this situation a different path was selected for communication. The established path in all experiments is: 10-11-12-13-14-15-16-17-18-8-9. The path has shown in Fig. 3.

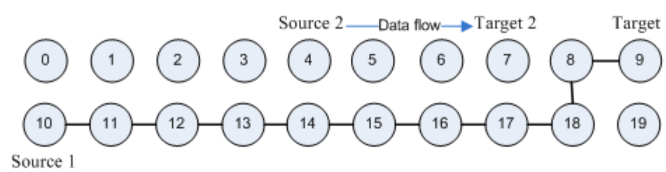


Fig. 3 The established path in the second simulation

Besides of the simulations above, we created a third simulation. This simulation is a composition of the first two simulations. In this simulation firstly we created a data flow from Source 1 to Target 1. Meantime, there was no data flow from Source 2 to Target 2. In this situation the established path is identical with the path that was established in the first simulation. After a few seconds we created a data flow from Source 2 to Target 2. With this traffic, the queue lengths of node 5 and node 6 were increased. As a result, the obtained end-to-end delay measurements from RPR packets exceeded *delay_thresh* and Source 1 created a new RREQ packet and a new path was established. The newly established path is identical with the path that was established in the second simulation.

As a result, our approach searched a new path when OWD exceeds *delay_thresh*. Thus, OWD has been kept less than 100 ms (around 80 ms).

## 5. CONCLUSIONS

In this study, we suggested a new routing metric for MANETs: one-way-delay. And we applied this metric on a source routing scheme. Additional to one-way-delay, we showed that individual queuing delays on each intermediate nodes and hop count can be easily obtained. According to all simulation results, in the absence of the other data traffic (Source 2 to Target 2), suggested method has chosen the path with the same number of hops that has been chosen by AODV. With the other traffic, suggested method has chosen shorter path according to one-way-delay. To achieve higher accuracy of measured OWD, other components of OWD, namely, processing delay and transmission delay must be considered.

For future works, a complex one routing metric can be created with one-way-delay, individual queuing delays on each intermediate nodes and hop count. Also, path stability or reliability can be determined with individual queuing delays.

## 6. REFERENCES

[1] H. Bakth, "Survey of Routing Protocols for Mobile Ad-Hoc Networks," *International Journal of Information and Communication Technology Research*, vol. 1, pp. 258-270, Oct. 2011.

[2] O. Gurewitz, I. Cidon and M. Sidi, "One-Way Delay Estimation Using Network-Wide Measurements," *IEEE Transactions on Information Theory*, vol. 52, pp. 2710-2724, June 2006

[3] W. Z. Lu, W. H. Gu, S. Z. Yu, "One-Way Queuing Delay Measurement and Its Application on Detecting DDoS Attack," *Journal of Network and Computer Applications*, vol. 32, pp. 367-376, 2007.

[4] J. H. Choi and C. Yoo, "One-Way Delay Estimation and Its Application," *Computer Communications*, vol. 28, pp. 819-828, 2005.

[5] K. Ogawa et. al., "Measurement Method of One-Way Differential Delay for End-to-End Path Selection on Multihoming," *Electronics and Communications in Japan*, vol. 90, pp. 2251-2262, 2007.

[6] D. D. Luong and J. Birio, "Hop-by-Hop versus End-to-End Active Measurements," *Computer Communications*, vol. 25, pp. 954-963, 2002.

[7] F. Matera et al., "Comparison Between Objective and Subjective Measurements of Quality of Service Over an Optical Wide Area Network," *Transactions on Emerging Telecommunications Technologies*, vol. 19, pp. 233-245, 2008.

[8] M. Aoki, E. Oki and R. Rojas-Cessa, "Measurement Scheme for One-Way Delay Variation with Detection and Removal of Clock Skew," *ETRI Journal*, vol. 32, pp. 854-862, 2010.

[9] Cisco Systems, "Measuring Delay, Jitter, and Packet Loss with Cisco IOS SAA and RTTMON," *Cisco*, 2005.

[10] J. Li, Z. Li and P. Mohapatra, "Adaptive per hop differentiation for end-to-end delay assurance in multihop wireless networks," *Ad Hoc Neworks*, vol. 7, pp. 1169-1182, 2009

[11] K. Fall and K. Varadhan, "The ns Manual", *ISI*, Nov. 2011.