EFFICIENT QUALITY OF SERVICE ROUTING FOR MOBILE AD HOC NETWORK

R.Dheenadhyalan,
M.Phil Research Scholar,
P.G & Research Department of Computer Science,
Sengunthar Arts and Science College,
Tiruchengode,Tamilnadu,India.

S.Saravanan,
Assistant Professor,
P.G & Research Department of Computer Science,
Sengunthar Arts and Science College,
Tiruchengode,Tamilnadu,India.

Abstract: Routing in a MANET is complex because it has to react efficiently to unfavorable conditions and support traditional IP services. In addition, Quality of Service (QoS) provision is required to support the rapid growth of video in mobile traffic. As a consequence, tremendous efforts have been devoted to the design of QoS routing in MANETs, leading to the emergence of a number of QoS support techniques. However, the application independent nature of QoS routing protocols results in the absence of a one-for-all solution for MANETs. Meanwhile, the relative importance of QoS metrics in real applications is not considered in many studies. Specifically, we begin by developing a QoS architecture for cross-layer information sharing, defining explicitly what information must be shared among the layers to provide support for QoS in terms of bandwidth and packet delivery rate.

Keywords: Mobile Ad Hoc Network, Quality of service, Packet Routing, Bandwidth, Throughputs

I.INTRODUCTION

A mobile ad hoc network is a wireless network without centralized control where every node acts as a router, forwarding packets to the destination when necessary [1]. MANETs have several advantages over conventional wired networks. First of all, MANETs are very convenient. The operator doesn't have worries such as running wires in tight places or obtaining low-voltage permits [2]. Secondly, the deployment range of MANETs is impressive compared to wired networks whose length of wires run limited [2]. However, some valuable characteristics of wired networks (e.g., reliability, cost, speed) are traded off in achieving this. As stated in last section, many routing protocols such as DSDV, DSR and AODV have paid little attention to QoS support in the early development of MANETs. However, QoS provision is becoming more important nowadays due to the rising popularity of real-time applications. In the past decades mobile traffic, which by definition refers to data generated by handsets, laptops and mobile broadband gateways, has been growing rapidly annually. According to a survey by Cisco, mobile data in 2010 was triple the volume of the entire global Internet traffic in 2000. The growth rate in the previous year was 159%, which is 10% higher than anticipated in 2009. This rapid growth in mobile data is forecast to continue for the next five years with an average annual growth of 92%.

There are several reasons why mobile traffic has grown so quickly. Firstly, mobile video, which requires high bit rates, is considered to lead to the increase of mobile traffic. It is reported that mobile video reached as high as 49.8% of total mobile traffic in 2010 and will account for two thirds of mobile traffic by 2015. Moreover, Internet gaming, which consumes, on average, 63 PB per month in 2009, also results in a growth in mobile traffic and it is expected to achieve an annual growth of 37% in the coming five years [3]. Last but not the least, Voice over IP (VoIP) which includes phone-based VoIP services direct from or transported by a third party to a service provider, and software-based internet VoIP such as Skype, leads to the expansion of mobile traffic. Many of those applications described above are real-time applications which demand certain guarantees for performance metrics for acceptable operation. Those metrics specify the Quality of Service.

QoS metrics : QoS is usually defined as a set of services that should be supported during packet transmission. A QoS enabled protocol is expected to support several metrics in terms of end-to-end throughput, delay, and jitter as well packet delivery ratio.

a) End-to-End Throughput: End-to-End throughput, η, is defined as the ratio of the payload of effectively delivered data packets, P_{ed}, over the elapsed time, $t_{elapsed}$.

$$n = \frac{P_{ed}}{t_{elapsed}} \quad \text{----(1)}$$

the basic unit of η is b/s or B/s. Effectively delivered data packets refers to data packets that are successfully delivered, excluding any duplicated packets. Since the available bandwidth in a network is fairly well known, it is helpful to obtain the actual throughput achieved which reveals the bandwidth usage efficiency. The higher the average throughput is, the better the bandwidth is utilized.

b) Delay (or Latency) : Delay, τ, sometimes refers to as end-to-end delay, is the time between the originating node sending a packet and that packet reaching the destination. It may vary dramatically because of long queue time or a congested network environment.

$$\tau = tS + t1 + t2 + \ldots + tn\text{-}1 + tn + tD \quad \text{----- (1.2)}$$

Where *tS* and *tD* denote processing time at the source and destination respectively. The buffering time of a packet is of great importance for delay. If the buffering time in an individual node is set to a higher value, it could imply that packets could stay in the buffer for a long period of time when link breakages occur which will may reduce the packet dropping rate [1]. In this case, the delay is higher. On the contrary, if the buffering time is shorter, the performance of delay will improve but the packet dropping rate will increase. Delay and packet delivery ratio are traded off in different applications.

Delay can be computed in multiple layers (e.g., application layer, transport layer network layer and link layer) and thus it is layer-dependent. For the sake of synchronization, round trip delay is used in some literature while others use single trip delay. In this thesis one-way delay is computed in the application layer by using a time stamp in the packet header

$$\tau = Rt - St \text{---------------------} (3)$$

Where *Rt* and *St* denote time at the source and destination for a given packet respectively, assuming suitably synchronized clocks in the transmitter and receiver. In some cases, excessive delay can render some time sensitive applications such as VoIP or online gaming unusable.

c) Jitter

Jitter was originally used in signal processing where it measures the deviation of some pulses in a digital signal and can be expressed in terms of phase, amplitude or width of the signal pulse. In the context of mobile ad hoc networks, the term jitter is defined as the average of difference between instantaneous delay and average delay [4]

$$\Box\tau = \frac{\sum_{i=1}^{n} |(\tau_i - \tau)}{n} \quad \text{---(1.4)}$$

where *n* denotes number of effective received data packets, τi symbolizes delays for different data unit and τ□ represents the average delay. It is reported that jitter can degrade live video quality nearly as much as packet loss rate [5].

d) Packet delivery ratio : The effective delivery ratio of data packets, α, is defined as:

$$\alpha = \frac{ENDP}{TNTP} \quad \text{---(1.5)}$$

Where *ENDP* and *TNTP* denote number of effectively received and total data packets respectively. Retransmission degrades the packet delivery ratio because it increases the denominator. A high packet delivery ratio is desirable, especially in MANETs, since the bandwidth available is limited for wireless links.

II.LITERATURE REVIEW

Srihari Nelakuditi, *et al.* [6] APR abstains from the exchange of QoS state information among routers and uses only locally gathered information. For each source-destination-pair of nodes, one or multiple explicit-routed paths have to be set up in advance, e.g. with MPLS. These are the candidate paths for routing. The maximum capacity for each path is known to the routers. Each row routed along a candidate path has a certain probability of being blocked. By knowing the capacity and measuring the blocking rates when trying to route a row along a path, a virtual capacity for this path is computed. This capacity may change over time, as new local information is gathered about current blocking rates. APR tries to equally distribute the rows among the available paths w.r.t. the virtual capacity of each path and with preference of minhop (i.e. shortest) paths over alternative (i.e. non-minhop) paths. No QoS information is exchanged between the nodes, reducing protocol overhead. Core routers (i.e. non-source routers) do not need to keep and update any QoS state database necessary for global QoS routing since the paths are already defined and no reservations are made. On the other side, APR is not suitable for mobile networks, as paths have to be set up in advance. Additionally, no hard QoS guarantees are possible, since no reservations are made.

Liao, *et al.*[7] propose a routing protocol that tries to detect a multi-path route to a destination node to fulfill a bandwidth requirement. For that, it uses a scheme of sending out probing packets with tickets, similar to that of TBP. But, other than in TBP, this protocol is based on an on-demand manner, so no global link state information has to be collected in advance, and single tickets may be split up into sub-tickets, each trying to find a path with lower bandwidth requirement. The destination node will pick one ticket or a set of sub-tickets forming a whole ticket and send a reply to the source node, conforming the bandwidth reservations. This protocol may find routes satisfying the bandwidth requirement even if no single path exists with sufficient bandwidth. But it relies on the existence of multiple transceivers per hop to effectively avoid collisions. The number of split-ups per ticket is not limited. This may help in discovering feasible routes even if there is a large number of links with narrow bandwidth. On the other hand, this may also result in a very large communication overhead, not only during route discovery, but during normal data transmission as well.

Yuval Shavitt, *et al.* [8] The goal of QMPR is to reduce the communication overhead when constructing a multicast tree by switching between single-path routing and multi-path routing. When a node n wants to join an already existing multicast tree, a single path to the tree's core is searched using a unicast routing algorithm. During route discovery, the QoS constraint is checked at every intermediate node. Consider two intermediate nodes a and b with a being part of the already discovered path. If b is the next node chosen by the unicast algorithm, but the link (a, b) violates the QoS constraint, then instead a would send messages to its other neighboring nodes to split up the search process. If more than one feasible path is detected, a chooses the best one (e.g. by smallest number of hops). The number of split ups can be restricted by specifying a maximum branching level. Compared to other QoS multicast routing protocols like the spanning join protocol by Carlberg and Crowcroft [23] or QoSMIC [24] by Faloutsos et al., QMPR avoids flooding to reduce the communication overhead. QMPR was not explicitly designed for MANETs, so it does not take mobility into account. However, because of its high-level design, it can be used on top of arbitrary unicast routing protocols, so it can be used in MANETs nevertheless.

Liao, *et al.*[9] devised a routing protocol for MANETs to reserve bandwidth in a time-framed medium while solving the hidden- and exposed-terminal problems. Each node keeps several tables of information, e.g. about the time slots of all nodes within a 2-hop range and their current usage (send, receive, free). This information is used to find free slots when reserving bandwidth and avoids the hidden- and exposed-terminal problems. To find a feasible route, a route request packet is sent out that includes, among other things, a list of 1-hop neighbors that may rebroadcast this request, if they have sufficient collision-free time slots. Time slot reservation is done during route acknowledgment on the way back to the source node. This routing protocol is rather simple and can be implemented with low effort. Additionally, it avoids the hidden- and exposed-terminal problems. But as the memory requirements of this protocol are rather high, it is suitable only for smaller networks. Additionally, a route request may also result in flooding the entire network. The chance of flooding the network increases with the number of free time slots at each node.

III.QOS-AWARE ROUTING PROTOCOLS

QoS guarantees are not possible in MANETs, and soft QoS and QoS adaptation are proposed instead. Soft QoS implies that failure to meet QoS is allowed, for example when paths break or the network becomes partitioned [10]. However, if a network changes too fast to propagate the topology status information, it is impossible to offer even soft QoS. Therefore, combinatorial stability must be met in order to provide QoS. Most real-time applications can optimize their performance based on feedback about network resource availability. For example, layered coding allows enhanced layers of different quality levels to be transmitted, provided a minimum bandwidth is guaranteed for transmitting the base layer. Therefore, these types of applications can benefit from QoS adaptation. By providing feedback to the application about available resources, the application can alter its coding strategy to provide the best quality for the current resource limitations.

Routing is used to set up and maintain paths between nodes to support data transmission.
Early MANET routing protocols, such as AODV, DSR , TORA , and DSDV focused on finding a feasible route from a source to a destination, without considering any optimization for utilizing the network resources or supporting specific application requirements. To support QoS, the essential problem is to find a route with sufficient available resources to meet the QoS constraints, and possibly add some additional optimizations such as finding the lowest cost or most stable of the routes that meet the QoS constraints. Given these goals, the following are the basic design considerations for a QoS-aware routing protocol.

- **Bandwidth Estimation:** To offer a bandwidth-guaranteed route, the key idea is to obtain information about the available bandwidth from lower layers. This bandwidth information helps in performing call admission and QoS adaptation. In MANETs, hosts share the bandwidth with their neighbor hosts, and thus the bandwidth available to a node is a dynamic value that is affected by its neighbors' traffic. Therefore, the two key problems in bandwidth estimation are how exactly to estimate the available bandwidth and how frequently to do the estimations. Also, the trade-off between the benefit from using bandwidth estimation and the cost in terms of packet overhead and computing resources used for bandwidth estimation is another key issue.
- **Route discovery:** There are two main approaches to routing in MANETS: reactive routing and proactive routing. Reactive routing reduces overhead at the expense of delay in finding a suitable route, whereas the reverse is true for proactive routing. For QoS-aware routing, another issue is determining what combination of reduced latency and reduced overhead is best for supporting QoS.
- **Resource reservation:** The bandwidth resources are shared by neighboring hosts in MANETs. Therefore, another challenging issue is how to allocate this shared resource and what type of resource reservation scheme should be used for setting up and maintaining the QoS-aware route.
- **Route maintenance:** The mobility of nodes in MANETs causes frequent topology changes in the network, making it difficult to meet the QoS constraints. Incorporating a fast route maintenance scheme into QoS-aware routing is the fourth design consideration. The typical approach to route maintenance, which entails waiting for the host to discover a route break, significantly affects the routing performance. Therefore, some prediction scheme or redundant routing is necessary to assist in route maintenance.
- **Route selection:** QoS-aware routing has more stringent requirements on route stability, since frequent route failures will adversely affect the end-to-end QoS. Thus, in some sense the path with the largest available bandwidth is not the only consideration-path reliability should also be considered when selecting a suitable path for a QoS-aware routing protocol.

Several routing protocols have been developed that support QoS by choosing routes with the largest available bandwidth, providing all admission feature to deny route requests if there is not enough bandwidth available to support the request, or providing feedback to the application about available bandwidth resources. These protocols address all of the issues described above.

IV.SIMULATIONS AND DISCUSSIONS

To test the performance of our QoS-aware routing protocol, we ran simulations using ns-2. We use the IEEE 802.11 MAC protocol in RTS/CTS/Data/ACK mode with a channel data rate of 2 Mbps. The packet size used in our simulations is 1,500 bytes.

"Hello" vs. "Listen" Bandwidth Estimation When Routes Break

A broken route can be caused by two reasons: (1) the hello messages collide several times (in which case the route is not really broken), and (2) a host in that route moves out of its neighbor's transmission range. We study these two different cases separately.

a)Route break caused by losing "Hello" messages

One flow in a network can be viewed as a single static chain. In order to simplify our analysis, we do the simulations in a chain topology to explain the effects brought by a broken route that is caused by losing broadcasted "Hello" messages. The simulated chain topology is composed of six hosts, where the header host is the source host and the tail host is the destination host. The source host sends data packets to the destination host using a 0.35 Mbps feeding rate. By studying the trace files, we find that a supposed route break occurs at 13 seconds using the QoS-aware routing protocol with "Listen" bandwidth estimation. Supposed route breaks occur at 27 seconds, 73 seconds, 236 seconds, and 468 seconds using the QoS-aware routing protocol with "Hello" bandwidth estimation. Figure 1 shows that using the route maintenance procedure, "Hello" bandwidth estimation can correctly estimate the residual bandwidth after the reported route breaks; however, using "Listen" bandwidth estimation cannot, so the source host is forced to transmit below the channel capacity. In this case, "Hello" packets are dropped often when traffic becomes heavy. After 3 consecutive "Hello" packets are dropped, a broken route is claimed. However, this route is not physically broken, because these 3 "Hello" messages are dropped by coincidentally colliding with other packets. Therefore, the packets are still successfully transmitted to the destination host during the time between the first "Hello" message being dropped and the third "Hello" message being dropped.

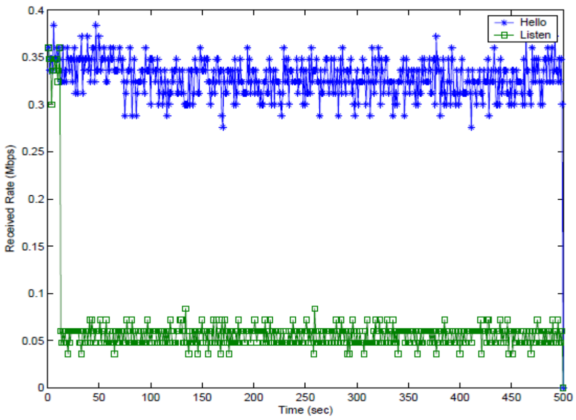

Figure 1: The received packet rate using a six-node chain topology with "Listen" bandwidth estimation and "Hello" bandwidth estimation.

The route discovery procedure is initiated right after the source host receives the "Error" message. The time interval between claiming a route break and setting up the route is only several milliseconds. In such a small time interval, it is almost impossible for the hosts to automatically and correctly update their bandwidth registers in the "Listen" bandwidth estimation method, since the consumed bandwidth estimation is based on averaging bandwidth consumption every one second interval and the hosts in the broken route were transmitting data in the previous second. Therefore, the "Listen"-based bandwidth estimation approach has difficulty correctly estimating the residual bandwidth. Even if some forced update schemes can be adopted, the hosts still cannot release the bandwidth correctly, since the hosts do not know how much bandwidth each node in the broken route consumes. In contrast, the "Hello"-based bandwidth estimation approach can easily solve this problem by using the forced update scheme.

b) Route break caused by moving out of a neighbor's transmission range

To simplify the explanation, we use the topology shown in Figure 2 to mimic the topology that will cause a route break because of a moving node. The topology is composed of 30 hosts. Host 18 is the destination host, and host 13 is the source host.

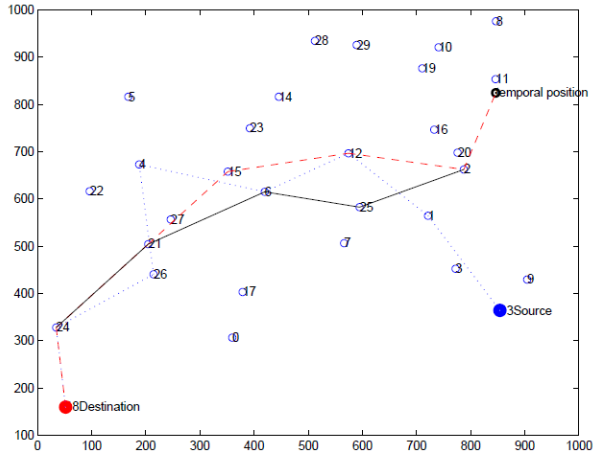

Figure 2: The scenario used to simulate a route break caused by a moving node.

Host 13 is moving towards host 11 with a speed of 10 m/s. The source host sends data packets to the destination host using a 0.25 Mbps sending rate. We ran simulations using the QoS-aware routing protocol with "Listen" bandwidth estimation and the QoSaware routing protocol with "Hello" bandwidth estimation. In the beginning of the simulation, the chosen route goes through hosts 13, 1, 12, 6, 4, 26, 24 and 18 (the dotted line in Figure 2). At the simulation time of 43 seconds, host 13 moves to a position (shown in Figure 5.2) that is out of host 1's transmission range. This causes a route break and host 13 must initiate a new discovery procedure. Using the routing protocol based on using "Listen" to estimate residual bandwidth, the new route goes through hosts 13, 2, 12, 15, 21, 24 and 18 (the dashed line in Figure 2). Using the routing protocol based on using "Hello" to estimate residual bandwidth, the new route goes through host 13, 2, 25, 6, 21, 24 and 18 (the solid line in Figure 2). The simulation results are shown in Figure 3.

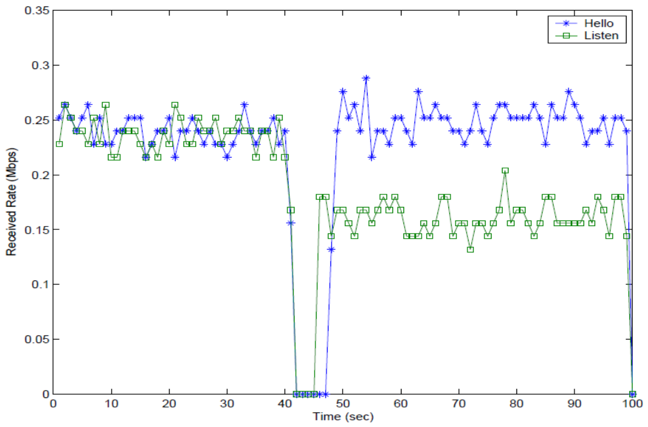

Figure 3: The received rate using the source moving topology shown in Figure 2 for the "Hello" bandwidth estimation method and the "Listen" bandwidth estimation method.

This is caused by the fact that the source host keeps on sending RTS packets, so host 2 can hear all these RTS packets and sets its NAV vector according to the packet length that the RTS indicates. Therefore, its estimated free time is significantly less than the real free time. Thus, host 2 cannot offer the correct bandwidth estimation after receiving a "RREQ" packet. However, using "Hello" to estimate residual bandwidth will not be affected by the above reason. These results show that the "Listen" technique cannot react well to a broken route due to the fact that the MAC's NAV cannot truly reflect the traffic status, and the bandwidth consumption registers cannot be updated in time. Thus, when routes break, "Hello" bandwidth estimation performs better than "Listen" bandwidth estimation.

V.CONCLUSIONS

This cross-layer approach includes an adaptive feedback scheme and an admission scheme to provide information about the current network status to the application. At the same time, the routing layer obtains the necessary traffic information from the MAC layer to assist in bandwidth estimation. Two different methods of bandwidth estimation – "Listen" and "Hello"– have been compared in detail using different topologies and different weight factors.

VI.REFERENCES

[1]. Andree Jacobson, Metrics in Ad hoc network, master thesis, 2000.
[2]. http://www.skullbox.net/wiredvswireless.php.
[3]. Cisco visual networking index: forecast and methodology, 2009-2014, http://www.cisco.com/.
[4]. M. Claypool and J. Tanner, The effects of jitter on the perceptual quality of video, http://web.cs.wpi.edu/~claypool/papers/jitter-pq/jitter-pq.pdf.
[5]. C. Perkins and E. Royer, "Ad hoc on-demand distance vector routing," in Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications, 1999, pp. 90–100.
[6]. Shigang Chen, Klara Nahrstedt, and Yuval Shavitt. A QoS-Aware Multicast Routing Protocol. In INFOCOM (3), pages 1594_1603, 2000.
[7]. Wen-Hwa Liao, Yu-Chee Tseng, and Kuei-Ping Shih. A TDMA-based Bandwidth Reservation Protocol for QoS Routing in a Wireless Mobile Ad Hoc Network. In Proceedings of IEEE ICC, April_May 2002.
[8]. Chunhung Richard Lin and Jain-Shing Liu. QoS Routing in Ad Hoc Wireless Networks. IEEE Journal on Selected Areas in Communications, 17(8):1426_1438, August 1999.
[9]. Srihari Nelakuditi, Zhi-Li Zhang, Rose P. Tsang, and David H. C. Du. Adaptive Proportional Routing: A Localized QoS Routing Approach. IEEE/ACM Transactions of Networking, 10(6):790_804, December 2002.
[10]. 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, 1999.