ROUTING PROTOCOLS IMPLEMENTATION USING MOBILE AD-HOC NETWORK

Pushpneel Verma[1], Sudhir Dawra[2], Raghav Mehra[3]

[1] *Research Scholar, Bhagwant University, Ajmer, Rajasthan, India*
[2]*Head & Associate Professor, Department of I.T, TIAS, New Delhi, India*
[3]*Associate Professor, Deptt. CSE, Bhagwant Institute of Technology, Muzaffarnagar, UP, India*

ABSTRACT

Protocols of MANET are of primary importance in variable topology networks, such as mobile ad hoc networks. A routing protocol must determine how to transmit a data packet over multiple hops from a source node to a destination node. This determination must be performed rapidly, with minimal bandwidth consumed by control packets, and must adapt to network changes caused by node mobility and node loss, among other factors.
Numerous routing protocols have been proposed for mobile ad hoc networks. We chose to study three of the four protocols recently recommended for development by the Internet Engineering Task Force (IETF) MANET work group: AODV, DSR and OLSR. It provides a comparison between reactive (AODV, DSR) and proactive (OLSR) protocols. The route discovery process of both protocols is fooled by the transient availability of network links to nodes that were more than one hop away. Packets transmitted over a fading channel cause the routing protocol to conclude incorrectly that there is a new one hop neighbor that could provide a lower metric (hop count) route to even more distant nodes. This can occur even when nodes are stationary; mobility resulted in even less route stability. We have compared the efficiency of two protocols on the bases of packet delivery, Congestion control and though put ratio. It is procedure to test our assertion that fading channels and unreliable network links were the cause of the failure of the routing protocols. The result was that neighbor discovery and the filtering for neighbors with which nodes could communicate reliably enables the creation of reliable multihop routes. Based on our experiences, we outline several recommendations for future work in MANET research.

Keyword: - *MANET, Routes, AODV, Packet, DSR and OLSR etc.*

---

1. INTRODUCTION

The term ubiquitous computing was coined by Mark Weiser to describe a state of computing in which users are no longer aware of computation being done. [1] The emergence of smart environments, where devices are embedded pervasively in the physical world, has sparked many new research areas and represents a step towards ubiquitous computing. Mobile ad hoc networks are multi-hop, wireless networks that can function without the presence of fixed infrastructure. When two nodes are not within radio range of one another, they use intermediate nodes to route packets for them. [2] A routing protocol is used for route discovery and maintenance. Ad hoc routing protocols can be classified as proactive and reactive. Proactive protocols maintain up to date routes to all destinations, while reactive protocols discover routes to destinations only when needed. Mobile ad-hoc network (MANET) routing protocols playa fundamental role in a possible future of ubiquitous devices. Current MANET commercial applications have mainly been for military applications or emergency situations. However, we believe that research into MANET routing protocols will lay the groundwork for future wireless sensor networks and wireless plug-n-play devices. The challenge is for MANET routing protocols to provide a communication platform that is solid, adaptive and dynamic in the face of widely fluctuating wireless channel characteristics and node mobility. The paper discusses our experience while implementing and deploying two distance vector MANET routing protocols. We examined both a public domain implementation of the Ad Hoc On-Demand Distance Vector (AODV) routing protocol and implemented our own version of the Destination Sequenced Distance Vector (DSDV) routing protocol. The choice of routing protocols was pragmatically based on what was available at the time this work was carried out. [3] The AODV implementation was the freely available MAD-HOC implementation. This implementation was based on an earlier draft of the AODV protocol and includes some MAD-HOC specific extensions. Where AODV is referred to in this paper we mean the MAD-HOC implementation unless otherwise stated. At the time our work was carried out this was the only public domain MANET routing protocol implementation that had a license suitable for our use and that we could get to compile, run and work on our network. DSDV was chosen due to its relative simplicity and the fact that it is a table based protocol rather than an "on demand" protocol like AODV. A number of extensive simulation studies on various MANET routing protocols have been performed by various researchers. However, there is a severe lacking in implementation and operational experiences with existing MANET routing protocols. [4] Previous implementation experiences include wireless Internet gateways (WINGS), implementation of ODMRP. These studies only highlighted performance issues specific to the protocol being used. By far the most extensive implementation study to date was conducted by Maltz et al. in describing their implementation of DSR. One such reactive protocol is AODV (Ad hoc On-demand Distance Vector). It finds a route to a destination by flooding the network with route request packets till the destination, or another node with a valid route to the destination is found. This node sends a reply, which traverses in reverse the path taken by the request packet to reach the destination. [5] All nodes along this reverse path make appropriate entries in their routing tables for the destination and establish a route. Link failure due to node mobility is a common feature in ad hoc networks. When a node that is part of an active route moves, the route breaks and has to be repaired by flooding the network for a new route. While there are several schemes that try to minimize flooding to repair routes, it remains a major source of routing overhead. In this paper, we compare and propose routing information, such as relevant parts of the routing table, are handed off to any such nodes that are found. This happens only in the vicinity of the weakening link and is transparent from other nodes along the route. If no suitable node can be found to perform the handoff, standard AODV route repair occurs as a matter of course after the link breaks.

2. AODV

For convenience, we divide the functioning of AODV into route discovery and route maintenance phases. After a route is discovered, actual routing occurs by looking up the routing table and sending the packet to the next hop for its destination.

2.1 ROUTE DISCOVERY

A source node initiates route discovery only when it has a packet to send to a destination to which it does not possesses a valid route. [6,7] Route discovery is initiated by the source flooding *Route Request (RREQ)* packets through the network. Every node that receives a RREQ creates a short-lived reverse route to the source with the next-hop being the node from which the RREQ was just received. When a RREQ reaches the destination or a node with a valid route, that node responds with a *Route Reply (RREP), which* travels to the source along the reverse path. All nodes that route the RREP to the source also make corresponding forward entries in their routing tables such that the next hop to the destination is the node from which the RREP was just received. The source, on receiving the RREP starts sending data. Each RREP also contains a destination sequence number, which is used to prevent routing loops and helps nodes determine the freshness of routing information.

2.2 ROUTE MAINTENANCE

Each node broadcasts periodic HELLO messages to advertise its presence. A node learns that a link to a neighbour is broken when it does not receive a HELLO from that neighbour for a predetermined time. When a broken link is detected, the detecting node sends *Route Error (RERR)* messages to all predecessor nodes that use the broken link to reach their respective destinations. This RERR packet travels back to the sources who re-initiate route discover.

2.3 ROUTER HANDOFF IN AODV

The decision to handoff is made by a node based on perceived signal strengths of its neighbours with whom it forms part of an active route. We maintain power information at nodes in terms of the ratio of received power to the receive threshold power. When this ratio drops below 1, a node can no longer be heard. The decision to handoff because of a weak link is made when one end of the link senses that the ratio has dropped below a Handoff Threshold (HTHRESH). This value is typically set slightly greater than I to give adequate time to handoff before the link breaks.

We have incorporated Router Handoff into the AODV protocol by making these changes:

Each node maintains a Neighbor Power List (NPL) containing the last received signal strength for packets originating from each neighbor. This table is updated whenever a packet is received, and happens at least once every Hello interval. [8] Each node also maintains a Power Difference Table (PDT). This table consists of the rate at which power is changing between each pair of neighbors. Entries for links of which this node is one end, are filled directly by calculating the difference in powers between a received packet and the entry in the NPL for the last received packet from that neighbor. Entries for links between two neighbors are received in the modified Hello packet. Hello packets are modified to contain neighbor power information. While sending out a Hello packet, each node examines its NPL and PDT and finds neighbors with whom it has a strong link. Here, a strong link is deemed as one whose strength is not decreasing and is above HTHRESH. Power entries are stored in power levels ranging from 0 to 255. Hence, for each strong link, a Hello packet contains the address of the neighbor and a single byte representing the rate of change of that link as determined from the PDT.

3. ROUTING PROTOCOL IMPLEMENTATIONS

This section presents implementation details of the AODV and DSDV protocols used in our experiments and provides a background to the discussions and observations which will follow regarding the deployment and implementation issues we have encountered. The packet capture program captures packets that traverse the network interface and triggers the AODV daemon when particular packets are seen. The capture mechanism is implemented using the libpcap library. [9] The main problem with the MAD-HOC AODV implementation was that it was not performed while route construction was in progress. In practical terms, we found that a telnet session had to be initiated multiple times before a session could be established. When running ping over a four hop route, with the default one second gap between successive pings, the first five packets were usually lost before the route was successfully established.

4. DSDV IMPLEMENTATION

The second routing protocol we chose to experiment with was DSDV. The choice was made due to DSDV 's simplicity, thus enabling us to easily code up and debug the operation of DSDV on our testbed. DSDV's simplicity proved valuable during our experimentation especially when explaining the poor operation of DSDV on our testbed. Our DSDV implementation was based on the ACM SIGCOMM' 94 paper by Perkins et al. with the addition of a neighbor handshake protocol to check for bi-directional links. Our DSDV implementation used the Multi-threaded Routing Toolkit (MRT) for platform independence and for interfacing with the kernel routing table, socket and file input/output [10]. In addition, MRT also provided some convenient data structures for holding information regarding machine interfaces and utilities for manipulating IP addresses. Due to the small scale of our testbed, the incremental update aspects of DSDV were not implemented (all the routes could easily fit in the one packet). The hysteresis timers were also not implemented, as we did not have many alternate routes of the same hop count. CMU's DSR due to extensive documentation, and hardware and operating system compatibility with our testbed.

5. THE SEEN METRIC AND STATE

The original paper describing DSDV specified that DSDV assumes bi-directional links but does not include any mechanism for ensuring a link was bi-directional before a route was put in place. It was found that such a mechanism was crucial with fading channels. We extended DSDV through the inclusion of a handshake protocol that makes use of the SEEN metric to signal that a new neighbor had been detected. The SEEN metric was defined as an integer value outside the range of one to INFINITY2. DSDV nodes advertise a route to a node with metric = SEEN on the reception of a packet from a neighbor for the first time. All other nodes, apart from the node listed as the route destination, ignore this route. On receiving a routing advertisement for itself with a metric = SEEN a node makes and advertises a route to the sending node. Nodes will only advertise a route to another node with a SEEN metric for a short period of time, if no reciprocal route advertisement is received then the SEEN state times out and the route is no longer advertised. DSDV would replace the existing two hop route between the nodes with the unreliable one hop route. Very little user data would be transmitted over this unreliable route and user sessions would hang pending the reestablishment of the more reliable two hop route. In a related work, Maltz et al. reported similar behavior while building a MANET testbed and experimenting with Dynamic Source Routing (DSR) routing protocol. The following modifications to DSR were suggested to overcome the problem of routing over unreliable links: (I) monitor route error on links, (2) use the geographic positioning system (GPS) to determine the neighbor proximity (assuming physical proximity will provide the best channel) and (3) combine GPS with route error monitoring. Reliability was tested over a three node, two hop networks with the nodes arranged in a line. The network included packet filtering software to prevent packets from being transmitted directly from one end node to the other.

6. DSDV ROUTE STABILITY

The first thing we noticed about our DSDV implementation was its relative stability compared to the MAD-HOC's AODV implementation. [11] DSDV was less affected by unreliable connections to distant nodes. This was mainly due to the use of the SEEN metric (requiring a handshake before the link would be used in routes) and less interaction with the ARP cache as the routing table was pre-populated with host routes (negating the need to ARP). However DSDV was adversely affected by transient link availability. Even when all the network nodes were stationary the routing table would slowly "churn" as routes were constructed to distant nodes and then timeout.

7. CONCLUSIONS

In this paper we have outlined our implementation and deployment experiences with MAD-HOC's AODV and DSDV. Our experiments have provided insights into the real world deployment of MANETs and highlight issues that require further investigation. These are:

a. Handling unreliable/Unstable links.

b. Minimizing the dependency on topology specific parameters.
c. Mechanisms for handoff and reducing packet loss during handoff.

d. Incorporating neighbor discovery and filtering into a neighbor selection sub-layer.

The first issue is a result of the current prevailing MANET protocol development/testing environments, which appear to consist almost entirely of simulation experiments. In implementing two MANET routing protocols, rather than simulating them, we discovered that the variability of networking conditions in the radio environment was such that the routing protocols did not work as reported in the literature. This led to the development of powerwave, and it was found that neighbor selection is crucial in the operation of MANET routing protocols. We believe our observations pertaining to unreliable/unstable links are not restricted to MAD-HOC's AODV implementation given that current AODV specification relies on hop. The second issue is specific to a given routing protocol. As argued, having pre-configured parameters for a given topology is inappropriate given the inherent dynamic nature of MANET s, and affects the operation of routing protocols. Therefore, methods for adaptive adjustment of these parameters are required.

8. REFERENCES

[1]. D. A. Maltz, J. Broch, and D. B. Johnson. Experiences designing and building a multi-hop wireless adhoc network testbed. Technical Report, CMU-CS-99-11, Mar. 1999.
[2]. H. Hashemi. The indoor radio propagation channel. Proceedings of the IEEE, 81(7), July 1993.
[3]. J. Broch, D. A. Maltz, D. B. Johnson, Y.-C. Hu, and J. Jetcheva. A performance comparison of multihop wireless ad-hoc network routing protocols.
[4]. J. J. Garcia-Luna-Aceves, D. Beyer, and T. Frivold. Wireless internet gateways (WINGS). In proceedings IEEE Milcom'97, Monterey, CA, 1997.
[5]. M. GerIa, G. Pei, and S. J. Lee. Wireless, mobile ad-hoc routing. In IEEE/ACM FOCUS, New Brunswick,

USA, May 1999.
[6]. M. S. Corson and V. Park. An internet MANET encapsulation protocol (IMEP) specification. Internet Draft: draft-ietf-manet-imep-spec-OO.txt, Nov. 1997.
[7]. P. Basu and T. D. C. Little. Task-based self-organisation in large smart spaces: issues and challenges. In DARPAINIST/NSF Workshop on Research: Issues in Smart Computing Environment, Atlanta, USA, 1999.
[8]. P. Bhagvat, C. Bisdjikian, P. Kermani, and M. Naghshineh. Smart connectivity for smart spaces. In DARPAINIST/NSF Workshop on Research: Issues in Smart Computing Environment, Atlanta, USA, 1999.
[9]. R. Dube, C. D. Rains, K.-Y. Wang, and S. K. Tripathi. Signal stability based adaptive routing (SSA) for ad-hoc mobile networks. IEEE Personal Communications, 4(2): 36{45, Feb. 1997.
[10]. R. K. Ahuja, T. L. Magnanti, and J. B. OrIin. Network Flows, theory, Algorithms, and Applications. Prentice-Hall, 1993.
[11]. S. H. Bae, S.-J. Lee, and M. GerIa. Unicast performance analysis of the ODMRP in a mobile ad-hoc network testbed. In Proceedings of IEEE ICCCN'2000, Las Vegas, USA, 2000.