International Journal of Latest Research In Engineering and Computing (IJLREC)
Volume 1, Issue 1 : Page No.21-25, September-October 2013
www.ijlrec.com

Performance Enhancements for Pre-emptive Ad-Hoc On-demand Multipath Distance Vector Routing

A.V.Santhosh Babu[1], Dr.P.Meenakshi Devi[2], K.Sudhakar[3]
[1]Assistant Professor, IT, Sengunthar College of Engineering, Tiruchengode, India,
[2]Professor & Head, IT, K S R Inst for Engg & Technology, Tiruchengode, India
[3]Asst Prof & Head, CSE, Sengunthar College of Engineering, Tiruchengode, India

Abstract : An ad-hoc network will often change rapidly in topology, this courses for routes in the network to often disappear and new to arise. The Ad-hoc On-Demand Distance Vector Routing Protocol (AODV), is based on the principle of discover routes as needed. In this paper we will extend the definition of AODV with the ability to discover multiple routes to a host and switch between them, if an active route is becoming weak and there is a risk that it will disappear. We will refer to it as "pre-emptive AOMDV". We will show that the performance of pre-emptive AOMDV do handle changes in topology better than AODV itself. To show the effect of extending AODV, the suggested protocol is implemented in a simulator. Performance enhancements will be presented from different scenarios, to compare pre-emptive AOMDV with the ordinary AODV.

Keywords: AODV, AOMDV, Preemptive Region, Multi Path Route Discovery.

INTRODUCTION

Currently, great demands for self-organizing, fast deployable wireless Mobile Ad Hoc NET works (MANETs) come along with the advances in wireless portable technologies. Unlike the conventional cellular wireless mobile networks that rely on extensive infrastructure to support mobility, a MANET does not need expensive base stations or wired infrastructure. This feature is important in battlefields or disaster rescue sites where fixed base stations are undesirable or unavailable. For commercial applications such as convention centers, electronic classrooms, and conferences, a rapid deployment of all-on-air networks provides users with more flexible and cheaper ways to share information.The absence of a fixed infrastructure, however, requires every mobile user to cooperate together for message transmission. Since radio transmission range is limited for each host, a source host must depend on several intermediate hosts to send its message to a host far away. Finding a path from the source to the destination, or routing, is fundamental for providing other advanced services. Routing, however, is very challenging in MANETs due to frequent network topology changes and power and bandwidth constraints. [1][2]

The Routing Protocols in MANET fall into two categories:
1. Reactive
2. Pro-active

Re-active routing protocol does not take initiative for finding routes. It establishes routes “on demand” by flooding a query.

Some pros and cons of reactive routing protocols are:
• It does not use bandwidth except when needed.
• Much network overhead is present in the flooding process when querying for routes.
• There is initial delay in the traffic.

On the other hand pro-active routing protocols set up routes based on continuous control traffic. All routes are maintained all the time. [1][2]
Some pros and cons of these protocols are:
• Constant overhead is created by control traffic.
• Routes are always available.

The idea of this paper is to demonstrate how pre-emptive routing and multipath routing can be added to the AODV protocol and how it will improve performance in a network with a low overhead. An active path fails due to mobility when a pair of nodes forming a hop along the path moves out of each other's range. In both types of routing algorithm, an alternative path is sought only after the current path fails. The cost of detecting a failure is high: several retries have to time-out before a path is "pronounced dead". Thus, when a path fails, packets experience large delays before the failure is detected and a new path is established. In this paper we investigate introducing preemptive route maintenance to Ad hoc routing protocols. More specifically, when two nodes, A and B, are moving out of each other's range, source nodes of active paths that use the hop A to B are warned that a path break is likely. With this early warning, the source can initiate route discovery and switch to a more stable path potentially avoiding the path break altogether. Preemptive route maintenance implements a "soft-handoff" of active paths when they become endangered. Thus, it combines the best of on-demand and table-driven algorithms: the overhead is kept small since updates are only triggered by active paths that are likely to break, and hand-off time is minimized since corrective action is initiated early. Without preemptive maintenance, when a path break occurs, the connectivity of the flow is interrupted and a hand-off delay is experienced by the packets that are ready to be sent or in flight. This increases both the average and variance (jitter) of packet latency. Furthermore, this delay and the loss of any packets in flight causes TCP congestion avoidance mechanisms to take effect, further harming the performance of TCP connections. Our solution preemptively finds other paths, in many cases seamlessly switching to an alternative good path before a break, minimizing both the latency and jitter and avoiding inefficiencies due to unnecessary TCP back off and congestion avoidance…

AODV

The AODV protocol is designed for mobile nodes in ad-hoc networking, where there often are changes in topology. The AODV protocol is based on on-demand route discovery. Because of that every node have different and limited local knowledge of the network. The fact that a node seeks information about the network, only when needed, is causing low overhead since a node does not have to maintaining unnecessary route information. To handle router information AODV uses 3 different kinds of messages Route request (RREQ), Route Reply (RREP) and Route Error RERR. AODV is using ring expansion when discovering new routes to limit flooding of the network and thereby reducing overhead. The protocol is ideal for discovering neighbour nodes. If a node needs a route to a node in the other end of the network1 the protocol will course a reasonable flooding of the network. Expansion ring search is a better strategy than doing a full scale search for the node. Likely some other node in the network has a valid route to the destination, and will send a RREP to source, and thereby reducing overhead. By every RREQ a node sends, a sequence number is increased, this is used by the protocol to guarantee loop-freedom in paths found.

PREEMPTIVE ROUTING

The idea behind pre-emptive routing is to look for a new route to a host that the node already could have a active route to, before the basic AODV protocol initiates a route discovery. There could be a lot of reasons to do this, we will use it to minimize end-to- end delay in a transfer between two nodes, before the link that they are using disappears. A link can disappear for multiple reasons, but we will look into the case where mobility of nodes is the reason. To be able to see that a node is moving and a route is about to break, we relay on the fact that communication is based on electronic signals. Because of that it is possible to measure the quality of the signal and based on that guess if the link is about to break. Based on this we will discuss how to handle route breaks, with in the AODV routing protocol. Figure 1 is showing how it could look, in a network when a node (X) is moving away from another node and enters its pre-emptive zone.

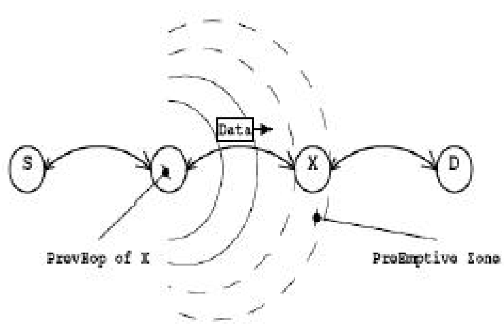

Fig 1: Preemptive Zone

MULTI PATH ROUTE DISCOVERY

The multipath route discovery will not be explained in depth in this rapport, we will just give a overview of the idea and refer to [3] for deeper discussion. The basic idea behind multipath route discovery is finding multiple node-/link disjoint paths to a destination node. This can be done with a low overhead, because flooding RREQ's through the network is being done already by AODV. Due to AODV already flooding the network, it is easy to see that a change in behaviour when a node receives a RREQ can result in multiple routes to the same destination. The destination node must be allowed to send more RREP's, one for each path. Figure 2 shows two node disjoint paths from a source to a destination. In article [3] its proved how link-/node disjoint paths are discovered, and how it is implemented. We will in the rest of the rapport assume the availability of multipath routing.

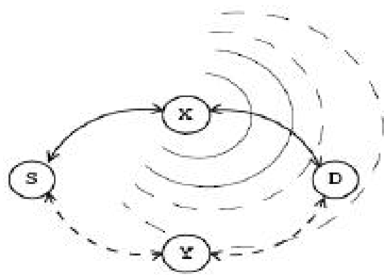

Fig 2: Two Paths

This section will describe how to extend the AOMDV routing protocol [3], to handle pre-emptive routing as presented and describe how the different approaches can cooperate to create a more efficient algorithm. Using multipath route discovery with pre-emptive routing will hopefully cause a smaller end-to-end delay and a larger throughput to a host. The RERR message in AODV needs to be changed to contain information about the error type. This can be done by altering the definition of the reserved field. Some bits in the reserved field could indicate what kind of error that the sending node wants to report. We are using the last bit to represent the type of error, table 1 shows how the different errors are represented. To differ between the two RERR messages, it is enough to use 1 bit, since we are only interested in whether the error is send to indicate a dead link or a weakened link

Link broke	0
Link possibly going to break	1

Table 1: Representation of RERR Messages

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |N|          Reserved           |   DestCount   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Unreachable Destination IP Address (1)             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Unreachable Destination Sequence Number (1)           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-|
|  Additional Unreachable Destination IP Addresses (if needed)  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Additional Unreachable Destination Sequence Numbers (if needed)|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Fig 3: AODV RERR Message Format

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |N|          Reserved         |P|   DestCount   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Unreachable Destination IP Address (1)             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Unreachable Destination Sequence Number (1)           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-|
|                       Source IP address                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                               :                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Fig 4: AOMDV RERR Message Format

The error messages will be used in the following way:

Link broke: When a link is broke the last bit of the reserved is set to 0. This will indicate that the sending node have to find another route to the destination and that the current route does not exist anymore.

Link possibly going to break: When the routing node, believes that the link will break within a short period of time. This means that the sending node, should find another way to the destination, as soon as possible.

To avoid flooding when sending a RERR we choose to send it to the original source. The "Additional Unreachable Destination IP Addresses" from the AODV RERR message is altered with the source IP address. Each intermediate node will forward the RERR to source.

ALGORITHM FOR AOMDV WITH PREEMPTIVE ROUTING

Algorithm 1:

```
Data : IP packet
if Receiving Node. is In Preemptive Zone then
    send RERR to source of IP packet
end
```

Algorithm 2:

```
if Receive(RERR) then
    if Destination = = this.Node IP then
        if existRouteInMulthipath then
            Change default route to destination;
    else
        Initiate a new route discovery to destination, while
        Continuing using the weak link.
    end
else
    send via next_hop to destination
end
end
```

Algorithm 1 is showing when a node must send a RERR message, to notify a sending node of a route being weak, this is based on the assumption that a lower network layer is able to indicate when a IP packet is received from a node in a pre-emptive zone.
Algorithm 2 shows how a node must behave when it receives a RERR. The first case is handling the RERR if the receiving node is the destination of the RERR. This could course two kinds of actions, either the node could choose another route found using multipath or it could initiate a new route discovery. The second case is if the receiving node is not the destination, then it needs to forward the RERR to the destination using its own default route, to the destination.
By using the algorithms 1 and 2 we ensure that a RERR message is sent when a active node is entering a pre-emptive zone, as described in the previous sections and if a node receives a RERR, it either initiate a new route discovery or uses a route found using multipath, in the previous route discovery. When using the algorithm in a real network, it is the strength of the signal with which a packet is received that indicates the link condition. It is important to notice that the reception of a packet that determines if a node is in pre-emptive zone. This has the consequence that a node entering pre-emptive zone can look in the packet to see where to send the RERR message. This assures that when a source receives a RERR then the default_route to destination is weak. Pre-emptive AOMDV always find three paths to destination, and only paths with traffic will enter pre-emptive zones and shortly after move out of range. This creates the best results in our protocol.

RESULTS

The results shows that using pre-emptive AOMDV instead of AODV in our limited scenarios is an improvement

	AODV		Pre-emptive AOMDV	
	AODV messages	packets received	AODV messages	packets received
10 Packets/sec,512bytes	153	75	81	79
4 Packets/sec, 512bytes	152	30	79	32

The table shows that traffic is increased by approximately % and AODV-messages are reduced by around 47%.

CONCLUSION

By reducing the number of AODV-messages, the overhead on the network is also reduced, which gives more network bandwidth to ordinary traffic. Since we are getting 6% more traffic through from source to destination, we conclude that nodes spend less time to find alternative routes to destination. This means that the end-to-end delay also is reduced.

REFERENCES

1. YUAN Peiyan, LI Layuan, “Performance Evaluation and Simulations of Routing Protocols in Ad Hoc Networks”, ACM 2006.
2. C. Siva Ram Murthy, B.S, Manoj, “Routing Protocols for Ad Hoc wireless Networks,” in Ad Hoc wireless networks: Architectures and Protocols, Chapter 7. Pearson Publication.
3. Mahesh K. Marina & Samir R. Das, On-demand Multipath Distance Vector Routing in Ad Hoc Networks, http://www.nmsl.cs.ucsb.edu/ ksarac/icnp/2001/papers/2001-2.pdf.
4. Muhammad Sulleman Memon, Manzoor Hashmani and Niaz A. Memon, 2008, A review for uniqueness and variations in throughput due to performance parameters of simulators on MANETs routing protocols, presented in 7th International Conference on EHAC’08, University of Cambridge, UK Cambridge, 20-22 Feb, 2008, PP: 202-208.
5. C.Perkins, E.Belding-Royer, S.Das “Ad hoc On-Demand Distance Vector (AODV) Routing” Feb.2003.http://www.ietf.org/internet-drafts/draftietf-manet-aodv-13.txt
6. A. Boukerche. Simulation based comparative study of ad hoc routing protocols. In *Proceedings of the 34th Annual Simulation Symposium*, pages 85–92, April 2001.
7. Y.-C. Hu and D. Johnson. Caching strategies in on-demand routing protocols for wireless ad hocnetworks. In *Proceedings of the International Conference on Mobile Computing and Networks (MobiCom’00)*, pages 231–242, August 2000.
8. T. Goff and J. Moronski and D. S. Phatak and V. Gupta. Freeze-TCP: a true end-to-end pro-active TCP enhancement mechanism for mobile environments. In *Proceedings of IEEE INFOCOM’2000, Israel,(to appear)*, Mar. 2000