Packet Forwarding using AOMDV Algorithm in WSN

Mrs. Poonam Meghare¹, Prof. Preeti Deshmukh²

¹Student ,Computer Science, Priyadarshini College of Engineering,Maharashtra, India,  
²Professor, Computer Science, Priyadarshini College of Engineering, Maharashtra, India,

Abstract

A WSN node consists of several units such as the microcontroller, the memory and the radio, which consumes most energy. Mobile ad hoc networks are characterized by dynamic topologies due to uncontrolled node mobility, limited and variable shared wireless channel bandwidth, and wireless devices constrained by battery power. One of the key challenges in such networks is to design dynamic routing protocols that are efficient, that is, consume less overhead. Energy consumption determines the lifetime of Wireless Sensor Networks. This paper focuses on reducing the energy consumption and to avoid packet loss of wireless sensor networks. Therefore, a communication protocol AOMDV is used . A Mobile Ad hoc Network is highly dynamic wireless network that can formed without the need for any pre-existing infrastructure in which each node can act as a router. In this paper we focuses on the AOMDV(Ad-hoc On-demand Multipath Distance Vector) Routing protocol. AOMDV protocol is an extension to the AODV(Ad-hoc On-demand Distance Vector) Routing protocol for computing multiple loop free and link disjoint paths. AOMDV was designed primarily for highly dynamic ad-hoc networks where link failures and route breaks occur frequently. It incurs more routing overheads and packet delay than AODV but it had a better efficiency when it comes to number of packet dropped and packet delivery. AOMDV reduces routing overhead by reducing the frequency of route discovery operation.

Keywords: AOMDV, Mobile Ad-hoc Network, Multihop Wireless Networks.

1. Introduction

Wireless sensor network (WSN) is a self-organized network composed by a large number of micro sensors that are randomly deployed in monitoring regional through wireless communication. Sensors nodes rely on battery power supply, their communication capability and energy storage capacity are very limited, so to utilize the energy of nodes efficiently, balance the network energy consumption and extend the network lifetime. Anew class of on-demand routing protocols for mobile ad-hoc networks have been developed with the goal of minimizing the routing overhead. The key characteristics of an on-demand protocol is the source initiated route discovery procedure. the on-demand protocols, multipath protocols have a relatively greater ability to reduce the route discovery frequency than single path protocols. On-demand multipath protocols discover multiple paths between the source and the destination in a single route discovery. So, a new route discovery is needed only when all these paths fail. Routing done by using the AOMDV routing protocol. AOMDV is based on a prominent and well-studied on-demand single path protocol known as ad hoc on demand distance vector (AODV). AOMDV extends the AODV protocol to discover multiple paths between the source and the destination in every route discovery. Multiple paths so computed are guaranteed to be loop free and link disjoint [2],[3],[4]. AOMDV also finds routes on-demand using a route discovery procedure. AOMDV relies as much on the routing information already available in the underlying AODV protocol, thereby limiting the overhead incurred in discovering multiple paths. It does not require any special control packets. Extra RREPs and RERRs for multipath discovery and maintenance along with a few extra fields in routing control packets (i.e. RREQs, RREPs and RERRs) constitute the only additional overhead in AOMDV relative to AODV [4].

2. PROTOCOL OVERVIEW

2.1. Ad-hoc On-demand Multipath Distance Vector Routing Protocol

A new class of on-demand routing protocols for mobile ad hoc networks have been developed with the goal of minimizing the routing overhead. AOMDV has three novel aspect compared to other on-demand multipath protocols. First, it does not have high inter-nodal coordination overheads. Second, it ensures disjointness of alternate routes via distributed computation without the use of source routing. Third, AOMDV computes alternate paths with minimal additional overhead over AODV, it does this by exploiting already available alternate path routing information as much a possible [4]. AOMDV shares several characteristics with AODV. It is based on the distance vector concept and uses hop-by-hop routing approach. In AOMDV, RREQ propagation from the source towards the destination establishes multiple reverse paths both at intermediate nodes as well as the destination. Multiple RREPs traverse these reverse paths back to form multiple forward paths to the destination at the source and intermediate nodes. AOMDV also provides intermediate nodes with alternate paths as they are found to be useful in reducing route discovery frequency. The core of the AOMDV
protocol lies in ensuring that multiple paths discovered are loop-free and disjoint, and in efficiently finding such paths using a flood-based route discovery. AOMDV route update rules, applied locally at each node, play a key role in maintaining loop-freedom and disjointness properties.

Fig-1: Examples of potential routing loop scenarios with multiple path computation

Based on the above discussion, we formulate below a set of sufficient conditions for loop-freedom. These conditions allow multiple paths to be maintained at a node for a destination.

To maintain multiple paths for the same sequence number, AOMDV uses the notion of an ‘advertised hop count.’ Every node maintains a variable called advertised hop count for each destination. This variable is set to the length of the ‘longest’ available path for the destination at the time of first advertisement for a particular destination sequence number. The advertised hop count remains unchanged until the sequence number changes. Advertising the longest path length permits more number of alternate paths to be maintained at a node for destination.

**Sufficient Conditions**

1. Sequence number rule: Maintains routes only for the highest known destination sequence number. For each destination, we restrict that multiple paths maintained by a node have the same destination sequence number. With this restriction, we can maintain a loop freedom invariant similar to AODV. Once a route advertisement containing a higher destination sequence number is received, all routes corresponding to the older sequence number are discarded.

2. For the same destination sequence number,
   a. Route advertisement rule: Never advertise a route shorter than one already advertised.
   b. Route acceptance rule: Never accept a route longer than one already advertised.

To maintain multiple paths for the same sequence number, AOMDV uses the notion of an ‘advertised hop count.’ Every node maintains a variable called advertised hop count for each destination. This variable is set to the length of the ‘longest’ available path for the destination at the time of first advertisement for a particular destination sequence number. The advertised hop count remains unchanged until the sequence number changes [4].

2.2. Disjoint Paths

AOMDV can be used to find node-disjoint or link-disjoint routes. To find node-disjoint routes, each node does not immediately reject duplicate RREQs. Each RREQs arrive in via a different neighbor of the source defines a node –disjoint path. This is because nodes cannot be broadcast duplicate RREQs, so any two RREQs arriving at an intermediate node vie a different neighbor of source could not have traversed the same node. In an attempt to get multiple link-disjoint routes, the destination replies to duplicate RREQs, the destination only replies to RREQs arriving via unique neighbors. After the first hop, the RREPs follow the reverse paths, which are node-disjoint and thus link-disjoint. The trajectories of each RREP may intersect at an intermediate node, but each takes a different reverse path to the source to ensure link-disjointness [2],[3],[4].

Paths maintained at different nodes to a destination may not be mutually disjoint. Here D is the destination. Node A has two disjoint paths to D: A – B – D and A – C – D. Similarly, node E has two disjoint paths to D: E – C – D and E – F – D. But the paths A – C – D and E – C – D are not disjoint; they share a common link C – D.

3. PROTOCOL DESCRIPTION

AOMDV protocol describe in four components: routing table, route discovery, route maintenance and data packet forwarding.
3.1. Routing Table
AOMDV route table entry has a new field for the advertised hop count. Besides a route list is used in AOMDV to store additional on formation for each alternate path including: next hop, last hop, hop count and expiration timeout. Last hop information is useful in checking the disjointness of alternate paths [4]. Consider a destination d and a node i. Whenever the destination sequence number for d at i is updated, the corresponding advertised hop count is initialized. For a given destination sequence number, let hop count$^{d}_{ik}$ denote the hop count of k th path (for some k) in the routing table entry for d at i, that is (next hop$^{d}_{ik}$, last hop$^{d}_{ik}$, hop count$^{d}_{ik}$) ∈ route list$^{d}_{i}$.

3.2. Rout Discovery
Like AODV, when a traffic source needs a route discovery process by generating a RREQs. Since the RREQs is flooded network-wide, a node may receive several copies of the same RREQ. All duplicates copies are examined in AOMDV for potential alternate reverse path, but reverse paths are formed only using those copies that preserve loop-freedom and disjointness among the resulting set of paths to the source.

When an intermediate node obtain a reverse path via a RREQ copy, it checks whether there are one or more valid forward paths to the destination. If so, node generates a RREP and sends it back to the source along the reverse path; the RREP includes a forward path that was not used in any previous RREPs for this route discovery. The intermediate nodes does not propagate the RREQ further. Otherwise, the node re-broadcasts the RREQ copy if it has not previously forwarded any other copy of this RREQ and this copy resulted in the formation/updation of a reverse path. The destination receives RREQ copies, it also forms reverse paths in the same way as intermediate nodes. The destination generates a RREP in response to every RREQ copy that arrives via a loop-free path to the source even though it forms reverse paths using only RREQ copies that arrive via loop-free and disjoint alternate paths to the source. The RREQ flooding mechanism, where each node locally broadcasts a RREQ once, suppresses some RREQ copies at intermediate nodes.

advertised_hop_count$^{d}_{i}$ := max$_{i}$({hop_count$^{d}_{ik}$}, i ≠ d).
:= 0, otherwise

however a node receives a route advertisement, it invokes the AOMDV route update rules listed in Figure 2. Note that lines (1) and (10) in Figure 2 ensure loop freedom, whereas lines and duplicates other RREQ copies[4].

3.3. Route Maintenance
Route maintenance in AOMDV uses RERR (Route Error) packets. When link breaks it then creates a RERR message, in which it lists each of these lost destination. The node sends the RERR upstream towards the source node. If there are multiple previous hops that were utilizing this link, the node broadcasts the RERR; otherwise, it is unicast.

When a node receives a RERR, it first checks whether the node that sent the RERR is its next hop to any of the destination listed in the RERR. If the sending node is the next hop to any of these destination, the node invalidates these route table and then propagates the RERR back towards the source. The RERR continues to be forwarded in this manner until it is received by the source. Once the source receives the RERR, it can re-initiate route discovery if it still requires the route [2].

3.4. Data Packet Forwarding
For data packet forwarding at a node having multiple paths to a destination, we adopt a simple approach of using a path until it fails and then switch to an alternate path; we use paths in order of their creation.

In other alternative, alternate paths are used simultaneously for load balancing where data packets are distributed over the available paths, thereby improving the network utilization and end-to-end delay [4].

<table>
<thead>
<tr>
<th>Table 1:</th>
<th>Routing table entry structure in AOMDV</th>
</tr>
</thead>
<tbody>
<tr>
<td>destination</td>
<td>sequence number</td>
</tr>
<tr>
<td>next_hop</td>
<td>last_hop2</td>
</tr>
<tr>
<td>next_hop2</td>
<td>last_hop2</td>
</tr>
</tbody>
</table>

1. if(seq_num$^{d}_{i}$<seq_num$^{d}_{i}$)then
2. seq_num$^{d}_{i}$ := seq_num$^{d}_{i}$;
3. advertised_hop_count$^{d}_{i}$ :=∞;
4. route_list$^{d}_{i}$ :=NULL;
5. if(j=d) then
6. insert (j,i,1) into route_list$^{d}_{i}$;
7.  Else
8.  insert(j,last_hop\textsubscript{d}, \text{advertised_hop_count}\textsubscript{d}, +1) into route\_list\textsubscript{d};
9.  end if
10. else if((seq\_num\textsubscript{d} \textsubscript{i} = seq\_num\textsubscript{d} \textsubscript{j}) and (advertised_hop\_count\textsubscript{d} \textsubscript{i} > advertised_hop\_count\textsubscript{d} \textsubscript{j}))
    then
11.  if(j=d) then
12.    if((/\text{ᴲk1} : (next_hop\textsubscript{d} \textsubscript{ik1} = j)) and (/\text{ᴲk2} : (last_hop\textsubscript{d} \textsubscript{ik2} = I )))then
13.      insert(j,i,1) into route\_list\textsubscript{d};
14.    end if
15.  else if ((/\text{ᴲk3} : (next_hop\textsubscript{d} \textsubscript{ik3} = j)) and (/\text{ᴲ} : (last_hop\textsubscript{d} \textsubscript{ik4} = last\_hop\textsubscript{d} \textsubscript{jk}))
    then
16.      insert(j,last_hop\textsubscript{d} \textsubscript{jk}, advertised_hop\_count\textsubscript{d} \textsubscript{j} + 1) into route\_list\textsubscript{d};
17.    end if
18.  end if

Fig-2: AOMDV route update rules

4. ADVANTAGES AND DISADVANTAGES OF AOMDV
It allows intermediate nodes to reply to RREQs, while still selecting disjoint paths. AOMDV has more message overheads during route discovery due to increased flooding and since it is a multipath routing protocol, the destination replies to the multiple RREQs those results are in longer overhead [3],[4].

5. CONCLUSION
In this paper, we described that AOMDV gives better throughput and end-to-end delay as compared to DSR and AODV. AOMDV extends the single path AODV protocol to compute multiple paths. AOMDV ensures that the set of multiple paths are loop-free and alternate paths at every node are disjoint. The performance of AOMDV using ns-2. We concluded that AOMDV is better than AODV. AOMDV reduces the packet loss. AOMDV also gives better performance.

6. FUTURE SCOPE
By using the AOMDV routing protocol the end to end packet delivery is greater then the other DSR, AODV. For more improvement if we apply the clustering to the network so then packet transmission time is reduced again compared to the AOMDV and it is done using by applying the LEACH protocol. By applying this network lifetime, packet delivery ratio increases and energy consumption of node, packet dropping is reduced.

REFERENCES

AUTHOR
Mrs. Poonam Vishwas Meghare received the B.E. degrees in Computer Engineering from Nagpur University in 2005 and perusing M.E. degree in Wireless Communication and Computing from Nagpur University.