International Journal of Latest Trends in Engineering and Technology
Special Issue SACAIM 2017, pp. 617-621
e-ISSN:2278-621X

ROUTE OPTIMIZATION IN MANET USING ACO

Mani Bushan D'Souza[1], Manjaiha D.H.[2] & Jeevan Pinto[3]

Abstract- A Mobile Ad-Hoc Network (MANET) is a group of wireless mobile nodes which form a temporary networkwithout the need for any fixed infrastructure and centralized control. Since the nodes are mobile, path between the nodes may break at any moment of time, routing in such a network becomes really challenging. Ant colony optimization (ACO) problems are the subset of swarm intelligence optimization that uses the food searching behavior of ants to solve the complex routing problem. ACO based routing protocols are efficient, scalable and are well adapted to the dynamic nature of MANET. The main goal in designing a routing protocol is to reduce the overhead for routing. This paper deals with study of characteristics and performance analysis of AODV and AntHocNet routing protocols and the comparisons of their performances.
Keywords –Ad-hoc network, MANET, Proactive, Reactive, Hybrid, AODV, AntHocNet, PDR, EED, throughput.

1. INTRODUCTION

A Mobile Ad hoc Network also known as MANET, is a collection of two or more devices that are connected via wireless communication media and capable of communicating with each other using multiple hops. MANETs follow a dynamic topology. There is no fixed infrastructure for such a network and the nodes can freely join or leave the network without affecting the other nodes. Every node in MANET can act as either source or destination or as a router, delivering the packets to the other nodes. Due to the dynamic nature of such networks, they generally offer less bandwidth than the conventional wired networks.

As the topology of MANET changes, efficient paths may become inefficient or even worst, may even become infeasible. Due to the limited bandwidth of the network, sharing of control information within the network is also restricted. The lack of centralized control or fixed infrastructure will make it difficult to get the global view of the network and the algorithms must work with the localized view of the network[1]. Thus routing in MANET is really a challenging task and the routing algorithms used in MANET should be robust, adaptive and self-healing.

1.1 Classification of Routing Protocols-

Generally, Ad hoc network routing protocols are distinguished based on how routing information acquired and maintained by mobile nodes. A mobile node uses its knowledge about recent connectivity of the network including the state of network links [2] to decide a route. Based on the time at which the routes are discovered and updated, routing protocol are classified into three categories

1.2 Proactive Routing Protocols-

Proactive protocols also known as “table driven” approach because routing information is maintained in tables. In this technique nodes in the network regularly discover path to all nodes which are reachable and tries to keep consistent and up-to-date routing information in the routing table. This makes it easier for a source node to get a routing path immediately when required [2]. Routing information is generally flooded in the whole network. These routing tables are periodically exchanged between nodes in network at set time interval. Whatever may be the mobility and traffic characteristics of network, the routing updates occur at specified time intervals.

1.3 Reactive Routing Protocol-

In this approach a node does not continuously maintain a route between all pairs of network nodes. Here, routes are discovered only when they are actually needed[3]. Whenever a node has data to be sent to other destination, it first checks its route table for the possible path. If the route doesn't exist in table, then it will start the route discovery process to find a path to the destination. Hence, route discovery becomes on-demand. Therefore,this approach is also called as on-demand routing.

1.4 Hybrid Routing Protocols-

These protocols are the combination of proactive and reactive approaches, thus rightly named as hybrid routing protocol. Nodes within a certain distance from the source node or within a particular geographical region, are said to be within the routing zone of the source node[3]. For routing within this zone, a table-driven approach is used. For nodes that are located beyond this zone, an on-demand approach is used.

[1] Department of Computer Application& Bio Informatics, AIMIT, St Aloysius College Mangaluru, Karnataka, India
[2] Department of Computer Science, Mangalore University Mangaluru, Karnataka, India
[3] Department of Computer Science, Yenepoya Institute of Technology, Mangaluru, Karnataka, India

2. AODV AND ANTHOCNET

2.1 Ad Hoc On Demand Distance Vector (AODV) Protocol –

AODV is a reactive routing protocol that is capable of performing both unicast and multicast route. It builds routes using a route request and route reply query packets. It uses three control packets for route discovery and its maintenance. These are RREQ (route request), RREP (route reply) and RERR (route error)[4]. Whenever a source wants to send a data packet to destination, it first checks its routing table for the route. If the route is not present in its table, then it initiates a route discovery procedure and broadcasts a route request (RREQ) packet to all its neighbors. The neighbors in-turn check their routing table, if there is no valid route to the destination, then they first record the reverse route to the source and then forward the RREQ to their neighbors. Finally, a route reply message (RREP) is unicasted back to the source whenever the receiver of RREQ is either the destination or has a valid path to the destination. Whenever any node receives the RREP, it first creates a forward route entry for the destination node and increments the hop count in RREP and passes it in the backward direction towards the source. Once the source receives the RREP, it can utilize this path for the transmission of data packets. If the source receives more than one RREP, it selects the route with the greatest sequence number and smallest hop count.

Every RREQ message consists of information like a unique identifier, the source address from which the request originated,address of the destination, a generated current sequence number, the destination's last sequence number, a hop count storing the number of hops from the source to the node which has currently received the request etc. Sequence numbers are used in route request packet to avoid the loop and to ensure that only most recent requests are replied by the intermediate nodes. Route table at each node contains the routing information for destination nodes along with an associated lifetime value. If a route is note utilized within the lifetime period, it gets expired. Premature expiry of route is prevented by updating its lifetime whenever it is used[5]. Route request (RREQ) packet contains destination node's IP address, the last known sequence number for that destination and the source's IP address and current sequence number. The RREQ also contains a hop count, initialized to zero and RREQ ID.

Once a route is established, it must be maintained as long as it is needed. A route that has been recently utilized for the transmission of data packets of data packets is called as active route. Because of the mobility of the nodes, links along paths are likely to break. If the active route breaks, it must be quickly repaired so that data packets are not dropped. When breaks along active path occurs, the node upstream of the break (i.e. closer to the source node) invalidates the routes to each of those destinations in its route table. It then creates a route error (RERR) message to each of those destinations in its destinations that are now unreachable due to the loss of the link. After creating the RERR message, it sends this message to its upstream neighbors that were also utilizing the link. The RERR message thus traverses the reverse path to the source node. once the source node receives the RERR, it can repair the route if the route is still needed.

2.2 AntHocNet-

AntHocNet is a hybrid multipath routing algorithm. It was proposed by DiCaro and Gambardella et al [6]. AntHocNet is based on AntNet, [7] that was designed for wired networks, after modifications it is now used on ad-hoc networks [6]. AntHocNet combines both proactive and reactive components with ACO. It does not maintain paths to all destinations at all times. When a source s wants to communicate with a destination d, it sets up the route by launching reactive forward ants (FANT) towards the destination. Each intermediate node that receives FANT may unicast or broadcast FANT depending on whether or not it has routing information to the destination. The routinginformation at i^{th} node is represented in its pheromone table as τ_i. The estimated goodness value of pheromone while going from neighbor n of i to the destination d is given by the entry $\tau^i_{nd} \in R$. Based on the pheromone information in the table, FANT chooses its next hop n with probability [8]

$$P_{nd} = \frac{(\mathcal{T}^i_{nd})^\beta}{\sum_{j\in\mathcal{N}^i_d}(\mathcal{T}^i_{jd})^\beta}, \quad \beta \geqslant 1, \tag{1}$$

Where $\mathcal{N}^i_d$ is the set of neighbours of i over which a path to d is known, and β is a parameter value which can control the exploratory behavior of the ants (β =1 for the current experiment). However, if table does not contain pheromone value for d, the ant is broadcasted to all its neighbors. If the number of hops and travelled time are both within the acceptance limit a_1 of the best ant of the same generation, it is forwarded. This way by setting a threshold, bad ants are removed, thereby reducing the overhead.

When the forward ant reaches the destination d, it transfers the list Ƥ of nodes [1,…, n] it has visited to a backward ant. The back ant (BANT) then retraces the track defined by Ƥ and incrementally computes an estimate $\hat{T}_{\mathcal{P}}$ of the time it would take a data packet to travel over Ƥ towards the destination, which is used to update routing tables. $\hat{T}_{\mathcal{P}}$ is calculated as the sum of local time estimates $\hat{T}^i_{i+1}$ in each node i∈Ƥ to reach the next hop i + 1, which is given as,

$$\hat{T}_{\mathcal{P}} = \sum_{i=1}^{n-1} \hat{T}^i_{i+1} \tag{2}$$

$\hat{T}^{i}_{i+1}$is defined as the product of estimated average time to send one packet, $\hat{T}^{i}_{\text{mac}}$, times the current number of packets in queue Q^{i}_{mac} plus one at the MAC layer, as shown in the equation

$$\hat{T}^{i}_{i+1} = (Q^{i}_{\text{mac}} + 1)\hat{T}^{i}_{\text{mac}} \qquad (3)$$

Here, $\hat{T}^{i}_{\text{mac}}$is calculated as running average of time elapsed between the arrival of a packet at the MAC layer and the end of a successful transmission.

As backward ant visits each node i∈Ƥ, it sets up a reverse path towards destination d and either creates or updates the pheromone table entry $\mathcal{T}^{i}_{nd}$. The pheromone value in $\mathcal{T}^{i}_{nd}$ is the running average of the inverse cost, in terms of estimated time and number of hops, to travel to d through n. If $\hat{T}^{i}_{d}$is the travelling time estimated by the ant, and h is the number of hops, the value $\hat{T}^{i}_{d}$is given as.

$$\tau^{i}_{d} = \left(\frac{\hat{T}^{i}_{d} + hT_{\text{hop}}}{2}\right)^{-1} \qquad (4)$$

Here T_{hop} represents the time to take one hop in unloaded conditions. This definition avoids large oscillations in the time estimates gathered by the ants (e.g. due to local bursts of traffic) and to take into account both end-to-end delay and number of hops.

3. EXPERIMENT AND RESULT

3.1 Performance Metrics-

Following performance metrics were used while comparing AODV with AntHocNet.

3.2 Packet Delivery Ratio (PDR)-

It is computed as the ratio of difference between the total number of generated packets and the total number of received packets divided by the total number of generated packets. It computes the loss rate as observed by transport protocols and is specific to both the efficiency and correctness of ad hoc routing protocols. The performance is better when packet delivery ratio is high. PLR is calculated as:

PDR = (Generated packets - Received Packets)/ Generated packets

3.3 End-to-end delay (EED)-

The end-to-end delay is the total delay that a data packet experiences as it is traveling from the source node to the destination node. To compute end-to-end delay, first store the difference of packet sent & received time and then divide the total time difference between the total number of packet received. The performance is better when packet end-to-end delay is low.

EED= (Time packet received - Time packet sent) / Total number of packets received

This delay is built up by several smaller delays in the network such as, time spent in packet queues, forwarding delays, propagation delay (the time it takes for the packet to travel through the medium) and time needed to make retransmissions if a packet got lost etc.

3.4 Throughput-

Throughput is defined as the average number of message successfully delivered per unit time i.e. average number of bits delivered per second. It is measured as, Total number of delivered data packets divided by the total duration of simulation time. We can analyse the throughput of the protocol in terms of number of messages delivered per one second. Throughput is calculated as the ratio of the total number of packets that reach their destination, to the total number of packets sent by the source

Throughput = Packets Received / Packets Sent

4. SIMULATION RESULTS AND ANALYSIS

Network simulator (NS) is an open source, discrete event network simulator. It is used in the simulation of routing and multicast protocols, in particular for ad-hoc network research. NS2 supports an array of popular network protocols, offering simulation results for wired and wireless networks alike.The simulations were performed using Network Simulator2 (NS2). The traffic sources are CBR (continuous bit rate). The source-destination pairs are spread randomly over the network. The detailed description of simulation environment is presented below in table1.

Parameter	**Value**
Simulator	NS-2.34
Radio-propagation model	Propagation/Two ray round wave
Channel type	Channel/Wireless channel

MAC Type	Mac /802.11
Network interface type	Phy/WirelessPhy
Interface queue Type	Queue/Drop Tail
Link Layer Type	LL
Antenna	Antenna/Omni Antenna
Maximum packet in ifq	50
Area (M*M)	2000*1000
Number of mobile node	50 - 250
Source Type	CBR(constant bit rate)
Simulation Time	600 s
Routing Protocols	AODV and AntHocNet
Number of connection	20
Data rate	20 packet/second
Pause time	0-100 second
Packet size	512
Mobility Model	Random Way point model
Transmission Range	250 m
Mobility speed	10-50 m/s

Table 1:Experimental Setup

Result of the analysis indicate that, as the pause time increases nodes become more stable and AntHocNet gives higher delivery ratio. This is shown in figure 1.

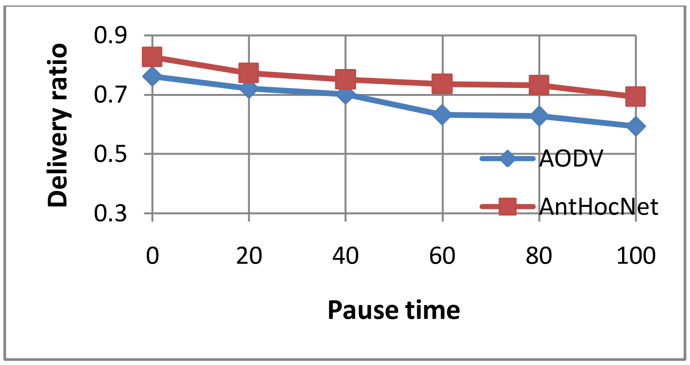

Figure 1: Delivery ratio Vs Pause time

However End-to-End delay is more in AntHocNet as compared to AODV. This may be attributed to the fact that as the mobility decreases path in between the nodes may be longer and network itself may become sparse. This is evident form the figure 2.AntHocNet will try to maintain the route at intermediate nodes however this comes at the cost of overhead involved in maintenance phase.

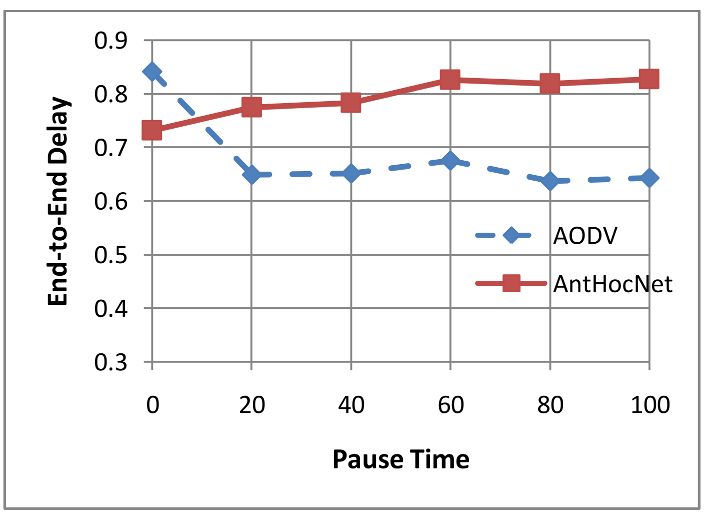

Figure 2: End-to-End Delay Vs Pause time.

It is evident that AntHocNet performs better with the pause time, as suggested by the throughput in figure 3.

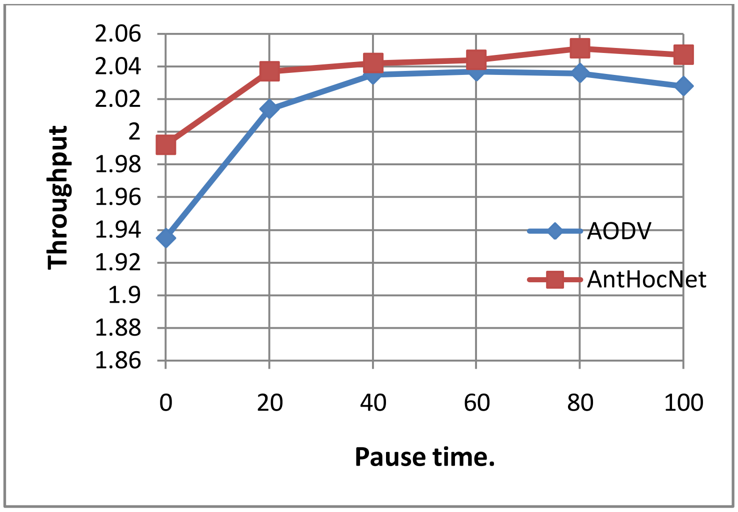

Figure 3: Throughput Vs Pause time.

AntHocNet also outperforms AODV with the increase in number of node, this is as shown in the figure 4.

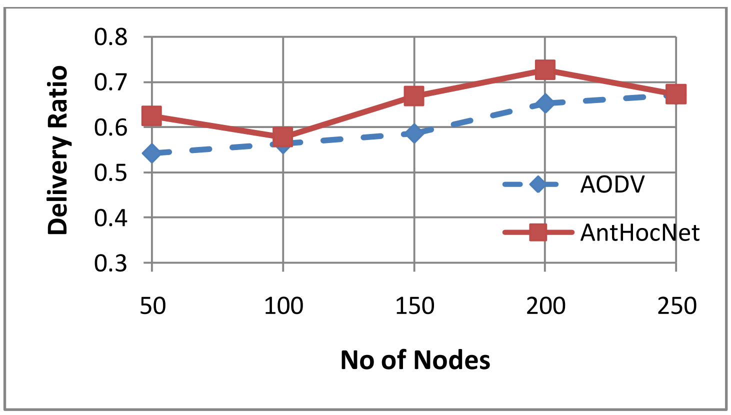

Figure 4: Delivery Ratio vs. No of Nodes.

5. CONCLUSION

In this paper we have described and compared the performances of two routing protocol used in MANET, namely, AODV and AntHocNet. Being a hybrid protocol, AntHocNet reactively sets-up the route and proactively maintains the path between source and destination. Simulation results shows that AntHocNet is performing better over AODV. This is evident from result as AntHocNet provides better throughput than AODV.

6. REFERENCES

[1] M. Tarique, K. Tepe, S. Adibi and Sh. Erfani, "Survey of multipath routing protocols for mobile ad hoc networks," Journal of Network and Computer Applications, vol. 32, p. 1125–1143, 2009,.

[2] Chai Keong Toh and Elizabethm Royer, "A review of current routing protocols for ad hoc mobile wireless networks," Personal Communications, IEEE, vol. 6, no. 2, pp. 46-55, 1999.

[3] E. Alotaibi and B. leriw rof smhtirogla gnituor no yevrus A" ,eejrehkuM ess Ad-Hoc and mesh networks," Computer Networks: The International Journal of Computer and Telecommunications Networking, vol. 56, no. 2, p. 940–965, October 2011.

[4] Perkins, C. E and Royer, E. M., "Ad-Hoc On-Demand Distance Vector Routing," in IEEE WMCSA '99, New Orleans, LA, Feb. 1999.

[5] C. E Perkins, E. M. Belding-Royer and S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing," 2003.

[6] G. Di Caro, F. Ducatelle and L. M. Gambardella, "AntHocNet: An Adaptive Nature-Inspired Algorithm for Routing in Mobile Ad Hoc Networks," September 2004.

[7] S.S. Dhillon and P. Van Mieghem, "Performance analysis of the AntNet algorithm," in Computer Networks: The International Journal of Computer and Telecommunications Networking, Elsevier NorthHolland, Inc, New York, NY, USA, 2006.

[8] Gianni Di Caro, Frederick Ducatelle and Luca Maria, "AntHocNet: an adaptive nature-inspired algorithm for routing in mobile ad hoc networks," EUROPEAN TRANSACTIONS ON TELECOMMUNICATIONS Euro. Trans. Telecomms., vol. 16, p. 443–455, 2005.