Load Balancing With Multipath Routing In MANAET

Kunjal, Prof. S. A. Jain
Kunjal (student) Prof. S. A. Jain (research scholar) MAE, Alandi (D), University of Pune

Abstract: - Mobile ad hoc network is a collection of wireless mobile nodes, such as PDAs, mobile phones, laptops etc. that are connected over a wireless medium. There is no pre-existing communication infrastructure (no access points, no base stations) and the nodes can freely move and self-organize into a network topology. Hence, balancing the load in a MANET is important because the nodes in MANET have limited communication resources such as bandwidth, buffer space, and battery power. Most of the current routing protocols for mobile Ad-hoc networks consider the shortest path which is having minimum hop count as optimal route without considering any particular node’s traffic and resulting in degradation of the performance by causing serious problems in any particular mobile node like congestion, queuing delay and power depletion. Therefore it is very attractive to investigate Routing protocols which use a Routing Metric to Balance Load in Ad-hoc net-works. This paper discusses about LBMPR (Load balancing with multipath Routing Protocol) for efficient data transmission in MANETs.

Keywords: - Load Balancing, Mobile Ad hoc Networks, Multipath Routing

I. INTRODUCTION

MANET is a ad-hoc wireless network formed by a group of mobile nodes which may not be within the transmission range of each other. MANET is having frequently changing topology. The nodes in MANET are self-configuring, self-organizing, self-maintaining and characterized by multi-hop wireless connectivity. Mobile nodes in MANET are connected by wireless links and each node act as host end router in the network. MANET is a collection of mobile nodes, such as PDAs, mobile phones, laptops which are connected over a wireless medium. The routing protocols in MANET can be categorized into three different groups: Table Driven/Proactive, On-demand/Reactive and Hybrid routing protocols. In Table Driven routing protocols, each node stores and maintains routing information to every other node in the network. These are done by periodically exchanging routing table throughout the networks. These protocols maintain tables at each node which store updated routing information for every node to every another node within the network. In on-demand routing protocols, routes are created when required by the source node, rather than storing up-to-date routing tables. Hybrid routing protocols combine the basic properties of the two classes of protocols. In this paper, we propose a multipath routing protocol, LBMPR, in order to minimize the route break recovery overhead. LBMPR provides most of the inter-mediate nodes on the primary path with multiple paths to destination, along with source node. Primary path is the first path received by source node after initiating the route discovery, which is usually the shortest path. Along with the shortest path it provides multiple path to destination from the source node during the route discovery process. All the multiple paths are used for data transmission at a time. Existing protocol SMORT is an extension to the unipath routing protocol AODV. The results are compared with the AODV protocol also because, it is important to know if the multipath protocol provides better scalability than its unipath counterpart. Except that comparison between existing SMORT and LBMPR is considered. This paper is organized as follows. In section 2, we described the related work. Section 3 provides system programmers design, finally we include the comparison of the protocols and conclude the paper.

II. RELATED WORK

In this section, we briefly present the research work related to multipath routing in literature. Recently, some multipath routing protocols have been proposed for ad hoc networks also. Multipath source routing (MSR) [1,2], extends DSR route discovery and route maintenance phases to compute multiple node-disjoint paths. It also proposes a mechanism to distribute load over multiple paths, based on the RTT measurement. SMR finds max-infalny disjoint multiple paths and uses a per-packet allocation scheme to distribute data packets on to multiple paths. This enables the effective utilization of network resources and avoids nodes from being congested. SMR computes only two
paths to each destination. All the above protocols are based on the source routing protocol DSR. Ad hoc on-demand distance vector multipath (AODVM) [3] is also a multipath routing protocol based on AODV. It proposes a routing framework to provide robustness to route breaks. Many disjoint multipath routing techniques [5,6,7] have been proposed for ad hoc networks, which have focused on improving the reliability of routing using path disjointness or redundancy. Saha et al. [5] proposed a maximally zone-disjoint multipath routing, which computes a set of zone-disjoint shortest paths for traffic load balancing. The zone-disjointness of paths minimizes the congestion for the traffic sent simultaneously over the multiple paths. Disjoint multipath source routing proposed in [6], statically multiplexes the data traffic over multiple disjoint paths at all nodes on the primary path. It achieves better transport capacity by doing so, when compared to the original source routing algorithm, in which packets go on a single path from source to destination. Tsigirgos and Haas proposed a disjoint multipath routing protocol that can be used in the presence of frequent topological changes. It uses multiple paths simultaneously, by splitting the information among the multitude of paths. Disjoint multipath routing [4] proposed by Abbassand Jain tries to reduce the effect of path diminution problem in finding node-disjoint multiple paths. As this routing technique also requires the routing request packets to carry the traversed path, it suffers from the same disadvantage as the previous protocol. In [7], Ducatelle et al. propose a hybrid multipath routing based on ant colony optimization framework for traffic load-balancing. Multipath Fresnel zone routing [9] proposed by Liang and Midkiff take the capacity of intermediate nodes into consideration for selecting disjoint multiple paths. It evaluates the capacity and the transmitting cost of different intermediate nodes, and formulates end-to-end paths of different capacities and cost. Then the protocol forwards the traffic through these different paths, by adjusting the amount of traffic on each path based on path capacity and congestion conditions. Papadimitratos et al. [10] proposed a reliable disjoint multipath selection approach using an efficient heuristic mechanism. Roy et al. compared the two disjoint multipath techniques that use omnidirectional and directional antennas, respectively. They showed through simulations that directional antennas help in computing multiple paths efficiently, when compared to omnidirectional antennas. Fault tolerant routing proposed by Xue and Nahrstedt [11] uses a path estimation mechanism for selecting a reliable route. Li and Cuthbert proposed a stable node disjoint multipath routing, which applies the path accumulation feature of DSR to AODV. But, this path accumulation feature requires the route request packet to carry the full path it has traversed. This requirement increases the size of route request packets, particularly in large networks where paths between nodes are longer. These large-sized route request packets, which are flooded across the entire network for route discovery, increase the routing overhead and thereby limit the network scalability.

III. PROGRAMMER’S DESIGN

Proposed protocol system is divided into modules and these modules are integrated together for the execution of the system. The proposed system includes following modules,

1. Route discovery process

A node initiates route discovery process, when it wants to communicate to a destination. Route Request Phase Proposed protocol considers heterogeneous systems in network, heterogeneity in terms of transmission power, load battery power etc. It calculates the utility of the node based on these factors while route discovery. It selects the most resource rich nodes in the network. Route discovery is performed over a number of different iterations. In the first iteration the algorithm allows only the most resource rich, meansthe nodes with the highest required utility level; nodes to re-broadcast during the route discovery phase. If the first iteration fails to determine a route to the required destination, then the source node reduces the utility level requirement to allow less resource rich nodes to also participate in route discovery. The source node begins by calculating a utility function and assigns a minimum level of utility to which each node must have in order to be able to re-broadcast the Route Request (RREQ) message. Different levels of utility requirement are there to be chosen, after which if a route to the required destination is not found, the source node will transmit and RREQ without a utility, i.e. all intermediate nodes are allowed to re-broadcast. Each node forwarding a RREQ stores its location information within the RREQ packet. The receiving node will then check to see if the forwarding nodes location falls within its transmission range. If yes, it updates its route table (i.e. assuming bi-directionality) and re-broadcasts the RREQ packet, and sends back a RREP if a route to the destination is known. Otherwise, it deletes the RREQ. Route request is given an ID. RREQ packet is sent with a field RREQ ID. This field contains the id of route request sent. In below figure, node S is sending a RREQ to neighbor node F, A and J. First copy of the route request

www.iosrjen.org 45 | P a g e
sets RREQ ID as ID1. Same as second copy of RREQ sets RREQ ID as ID2. And so on. Number of RREQ copies can be restricted. When it comes to route reply the route request which is having ID1 is served first. Then it will reply for ID2.

**Route Reply Phase** Route replies follow the reverse paths stored in the request-rcvd table to reach the source node. When destination node receives first RREQ packet it will send RREP using reverse path. RREQ ID1 is served first. Consider above figure for the case. Destination node D sends reply for the RREQ ID1. RREP packet consists of mulp rply field. For the node E on the primary path will receive the RREP packet with mulp rply value as true. It means it can send multiple copies of route reply packet. For the remaining nodes who receive mulp rply field as false. It means only one copy of RREP packet can be sent.

Once the path has been discovered destination node will respond to another copy of the RREQ that is RREQ ID2. This time RREQ ID1 will be disabled. Destination node D will send reply for the ID2. The node from which it has received RREQ will receive RREP packet with mulp rply value as true. The remaining nodes will receive the RREP with mulp rply value as false. Same procedure will repeat here again. The node with the true value will send multi-

![Flowchart-RREQ](image1.png)

**Figure 2: Flowchart-RREQ**

![Flowchart-RREP](image2.png)

**Figure 3: Flowchart-RREP**
ple copy of RREP packet. And the remaining node will send a single copy of RREP packet. Another path will be discovered when this RREP will reach to destination. Same way multiple path can be dis-covered between single source and destination pair.

2. Data transmission

Data transmission in proposed mechanism uses traf-fic distribution strategy. The source node uses the technique of traffic splitting to disperse the traf-fic over multiple paths. It is based on weight of the each path. Algorithm uses scheduling technique for time slices. Lets understand with and example. Take a simple example, Path A, B and C, have the weights, 4, 3, 2 respectively, a scheduling sequence will be AABABCABC in a scheduling period (mod sum(p)). A scheduling sequence will be generated according to the path weights. It calculates the weight by following equation

\[ \text{Temp}_i = \frac{\text{BW}_i}{L_i} \times \text{factor}_i \]

\[ \text{sum} = \text{Temp}_1 + \text{Temp}_2 + \text{Temp}_3 + \ldots \]

\[ p_i = \frac{(\text{Temp}_i / \text{sum}) \times 100}{} \]

Where, BW is the bandwidth and L is the delay. The path which is having highest weight will be uti-lize more. The time duration for which the path is used id based upon the gcd(p).

3. Route maintenance

Route maintenance phase maintains the routes es-tablished during the route reply phase, for the time duration of session. The lifetime of routing entries is used for this purpose. The lifetime of route rep-resests the time until when the route through nex-thop is valid. Nodes on the primary path refresh the lifetime of their routing table entries, each time a data packet for the corresponding destination is forwarded. The lifetime of routes at the nodes on the secondary path is initiated to a sufficiently large value. This value can be decided based on the fre-quency of path breaks due to mobility and proba-bility of node failures. We call this parameter as SEC ROUTE LIFETIME. If a requirement for the secondary route arrives before this life time, the sec-onthy route is used for data transmission, and then its lifetime is updated as long as data transmission happens through the route. Otherwise, secondary routes are deleted from routing tables once their ini-tial lifetime expires. The lifetime of route is updated by CURRENT- TIME + ACTIVE-ROUTE- TIMEOUT, whenever a data packet is send through next-hop successfully. This means that the route is valid and needed till the upcoming ACTIVE-ROUTE- TIMEOUT sec-onds. CURRENT-TIME is the absolute clock time of the node performing this update. If a route to the destination expires, that means, if routeëšÂŽs existing-lifetime is less than CURRENT-TIME, the route is invalid and cannot be used for sending data packets. Later, when a data packet arrives for the same destination, the node checks whether the valid secondary path to the destination is available in the route-list of the routing entry. If a valid secondary route exists, the primary path is replaced with the secondary path and packets are forwarded through it. If a valid secondary does not exist, a route error packet is sent to the source nodes through the nodes in the precur-list of the destinationëšÂŽs route entry. And the source node will go for new route discovery.

---

**Figure 4: Flowchart-Data transmission**

---
3.1. Mathematical Model

Sets for the proposed system Notations
G= Global set
N= networks
WN= Wired Network
WLN= Wireless Network
IN= Infrastructured Network ILN= Infrastructureless Network QOS= Quality Of Service
D= Delay
MD= Minimum delay
MCO= Minimum Control Overhead
R= Reliability
LP= Laptop
PDA= Personal Digital Assistant
MOB= Mobile Phones
D= Delay
RREQ = Route Request Packet RREP= Route Reply Packet DTP = Data Packet
RERR= Route Error Packet G = [N]
N = [WN,WLN] whereN c G
WLN = [IN,ILN] whereW LN c N
ILN= {QOS, DEV,PKT} whereILN c W LN QOS = {D,MD,MCO,R } DEV={LP,PDA,MOB}
PKT={RREQ,RREP,DTP,RERR} whereQOS, DEV, PK T c ILN

Input/output for the proposed system

For the Route discovery phase Input will be RREQ Packets
Output will be Route between nodes For the Data transmission phase Input will be data Packets sent
Output will be data Packets received For the Route maintenance phase Input will be RERR Packets
Output will be Act to discover alternate path

Functionality
1. Broadcasting Each node in the sytem can broadcast RREQ for the new route discovery.
2. Check for the available route This functional- ity checks for the available route between source
to desired destination.
3. Find Multiple path This functionality finds the mul- tiple path for transmission of data.
4. Find fail safe path this functionality finds the fail safe path for each primary path which is the
combi- nation of node disjoint and link disjoint path.
3.2. Dynamic Programming and Serialization

Here is the structure of protocol which is devided into above mentioned submodules these submodules are merged at the end.

3.3. Data independence and Data Flow architecture

The route table is used to store routing information to- wards every destination. Request received table: The table is used to store route request information. Tuple contains (address of the previous node, number of hops). Address of the previous node field represents the node that relayed the route-request to it called lasthop. Number of hops field rep-resents the route-request has traversed from the source node.

Route Table: The route table has an update list of all the possible routes to the desired destinations. Each element in the table is a six tuple of the form (destination addr, route - list, dest seq nb, precurlist). destination addr represent the unique addresses of the destination node; Multiple routes to destination are stored in route - list of the routing entry; The list of lasthops, through which replies are sent, are stored in the precur - list of the rout- ing entry;

IV. RESULTS AND DISCUSSION

1. Performance metrics

Following metrics are going to be compute to eval- uate scalability and performance of SMORT: Throughput : Throughput is calculated as the num- ber of data bytes delivered to all destinations during the simulation.

Average packet transmission delay: Average packet transmission delay is the average time taken by data packets to travel from source node to destination. This per-packet delay includes not only the abso- lute delay experienced by the packet in reaching the destination, but also the delay in resuming the ses-sion, after the route breaks have occurred.

Packet delivery ratio: the ratio of the number of delivered data packet to the destination. This illus- trates the level of delivered data to the destination.

2. Experiment Throughput, routing overhead and av- erage packet transmission delay are going to be com-pute by increasing the number of nodes in the net- work from 25,50,75 and 100 nodes. Mobility of nodes will be constant.

3. Result performance

Expected result of the above experiment is LBMPR will perform better in terms of above performance matrix.

V. CONCLUSION

In this work the objective was to introduce LBMPR. That includes to discover multiple path between source and destination with fail safe path. All the traffic can be transmit through all multiple path at a time. Except this LBMPR considers hetrogeneous nodes in network.

www.iosrjen.org
REFERENCES


