A Novel Optimization of Route Discovery in Dynamic Source Routing (DSR) Protocol for MANET

GJCST Computing Classification
F.2.2 ,C.2.1, C.2.5 & C.2.2

V.Ramesh [1] Dr.P.Subbaiah [2] N.Koteswar Rao [3] K.Sangeetha Supriya [4]
[1] Research scholar,Sathyabama University ,Chennai & Assoc.Prof, CMRCET, Hyderabad, AP,India. v2ramesh634@yahoo.co.in.
[2] Principal, Veerabrahmendra Institute of Tech. & Sci., Badvel, Kadapa, AP,India subbaiah_nani@sify.com.
[3] Associate Professor, Dept of IT, Narayana Engg. College,Gudur,AP,India, rao0007@gmail.com.
[4] Assistant Professor, Dept of IT, BVRIT, Medak, AP, India, sangeethasupriyak@gmail.com.

Abstract- **Ad hoc networks are useful for providing communication support where no fixed infrastructure exists or the deployment of a fixed infrastructure is not economically profitable, and movement of communicating parties is allowed. Therefore, such networks are designed to operate in widely varying environments, from military networks to low-power sensor networks and other embedded systems. Frequent topology changes caused by node mobility make routing in ad hoc wireless networks a challenging problem. A novel technique for optimizing the efficiency of route discovery is proposed. The optimization aims to minimize the number of cached route replies (RREP), which is a significant source of overhead for the dynamic source routing (DSR) protocol. Performance results show that the overall route discovery overhead can be reduced by more than 30% under high node mobility.**

Keywords- DSR, Optimization, Route Discovery, Route Maintenance, MANET.

I INTRODUCTION

There are currently two variations of mobile wireless networks. The first is known as infrastructure network. The bridges for these networks are known as base stations. A mobile unit within these networks connects to and communicates with, the nearest base station that is within its communication radius. As the mobile unit travels out of range of one base station into the range of another, a "handoff" occurs from the old base station to the new, allowing the mobile to be able to continue communication seamlessly throughout the network. Typical applications of this type of network include office wireless local area networks (WLANs). The second type of mobile wireless network is the mobile ad-hoc network or MANET. Unlike infrastructure network, this type of network needs no base station. Mobile nodes communicate to each other by either directly or through intermediate nodes. Ad-hoc network becomes popular since it can be applied in many situations, such as emergency search-and-rescue operations, classroom, meetings or conference and many more. To facilitate communication within the network, a routing protocols used to discover routes between nodes. Building a MANET routing protocol is not an easy job, since efficiency and correctness becomes the main concern. Some approach had been proposed to make routing protocol becomes efficient and correct. Routing protocols in MANET, generally, can be categorized as table-driven and on-demand. In table-driven (also called proactive protocol), like in most routing protocol for wired network, each node is required to maintain routing table keep updated whether there is or not a request for routes. In on-demand (also called as reactive protocol), each node seeks for routes only when there is need to do so. [1][2]

II OVERVIEW AND RELATED WORK

The Dynamic Source Routing (DSR) protocol is a simple and efficient routing protocol designed specifically for use in multi-hop wireless ad hoc networks of mobile nodes. The DSR protocol allows source nodes to dynamically discover a route to any destination node in the ad hoc network. Each data packet sent has in its header the complete ordered list of nodes through which the packet must pass, and avoiding the need for up-to-date routing information in the intermediate nodes through which the packet is forwarded. DSR cache the routing information for future use. DSR protocol contains two major phases, route discovery and route maintenance.

A. Route Discovery

Route Discovery is the process by which a source node S needs to send a packet to a destination node D and hence obtains a route to D. Route Discovery is used only when source node S needs to send a packet to destination node D, it looks up its route cache to locate an unexpired route to the destination and if it fails, then it initiates the route discovery process through broadcasting a Route Request (RREQ) packet. Each node on receiving a RREQ packet, it rebroadcast the packet to its neighbors if it has not forwarded already. Route Request packet (RREQ) contains <Destination Address, Source Address, route Record, Request ID>. On receiving the RREQ packet the destination replies the RREQ packet the destination replies to the source with a Route Reply (RREP) packet. When an intermediate node detects that the link to the next-hop node towards the destination is broken, it immediately remove this link from the route cache and returns a route error message to the source node. The source node again activates a new route discovery. DSR works for small to medium size MANET when nodes speed is moderate and every node has enough battery power. Its main feature is that every data packet follows the source route stored in its header. This route gives the address of each node through which the packet should be forwarded in order to reach its final destination. Each node on the path has a routing role and must transmit the packet to the next hop identified in the source route.[3]

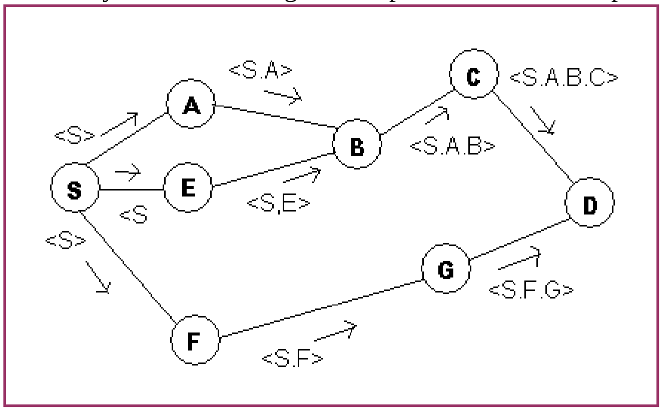

Fig: 1 Example Network

When a source node S want to send message to the destination node D, it initiates route discovery by broadcasting the RREQ packet to its neighbors (A, E, F) as shown in Fig 1. The intermediate nodes (A, E, F) on receive the RREQ packet rebroadcast the packet to its neighbors by appending its id in the route record of the RREQ packet. Similarly other intermediate nodes also forward the RREQ packet to the destination. When the destination node D receives two or more RREQ packets from the same source through different routes, it finds the best route based on the no of hopes.

The destination node D sends Route Reply (RREP) packet using the route (<S, F, G>)

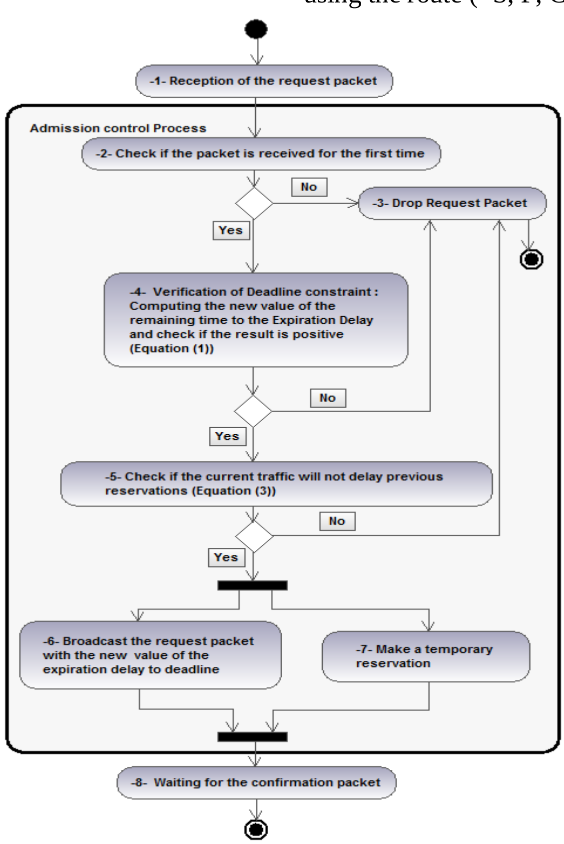

Fig 2: Activity diagram of intermediate node behavior in real time route request phase

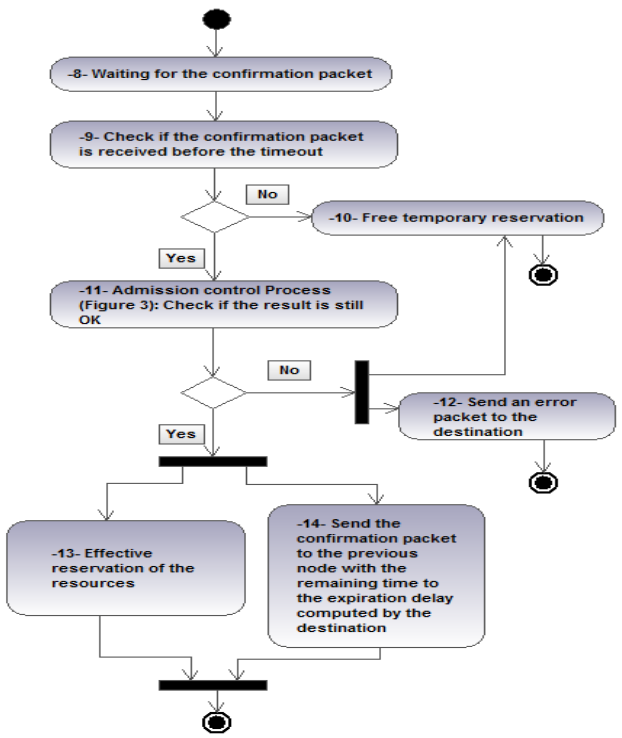

Fig 3: Activity diagram of intermediate node behavior in real time route reply phase

B. Route Maintenance

Each node maintains a Route cache in which it stores every source route it has learned. When a node needs to send a data packet, it checks first its route cache for a source route to the destination. If no route is found, it attempts to find new route using the route discovery mechanism and hereby increases the control overhead and connection setup delay. [4][1]

III OPTIMIZATION

This phenomenon can be put to good use to minimize the number of cached RREP. For example, the overheard RREP contains information about i) the hop length of the returned source route, and ii) the ID of the RREQ, for which this RREP is generated. Together with the source address of the RREQ, which is found from either the returned source route, or destination address of the RREP, these three pieces of information could be used to decide if a node should reply upon receiving a RREQ, even for the first time.

Existing DSR algorithm dictates that a node will reply if it receives a RREQ for the first time, for which it has a route to the destination. It is almost certain that node D in Fig. 1 would send a cached RREP, since it knows at least a route to the destination, which is that it overhears from the RREP. However, the returned route may not be useful if it is longer than the one previously returned, since route selection at the source is typically based on the shortest path.

We thus propose that if a node overhears a RREP for a RREQ it has not seen before (known by the RREQ ID and source address), the node shall record the three pieces of information from the RREP, namely the i) hop-length of returned source route, ii) RREQ ID, and iii) RREQ source address, as mentioned before. Subsequently, if the node receives this RREQ, it will compare the hop-length of its route (to be returned) with that seen previously. It will reply if it has a shorter route, and discard otherwise.

DSR has a scheme with some similarity for preventing "Route Reply Storms". However, the scheme does not propose the use of other information received from the RREP as we mentioned above. Furthermore, the scheme listens for shorter routes only after RREQ is received. This inherently introduces a delay, which adds to the route acquisition latency.

IV SIMULATION ENVIRONMENT

We evaluate the performance of our proposed technique using an ns simulator with Monarch wireless extensions. A total of 100 nodes are simulated for 500s over a network space of 1342m x 1342m. The network traffic is modeled as 40 CBR sources with data sent in 64-byte packets at 2 packets/s. Five movement patterns are generated based on random waypoint model for each value of pause time: 0, 100, 200, 300, 400, 500s. A pause time of 0s corresponds to continuous motion (at speed of up to 20m/s) and a pause time of 500s (length of simulation) corresponds to no motion.

V PERFORMANCE RESULTS

Fig. 4 shows the number of RREPs transmitted by both DSR, and a modified version of DSR with our optimization. The DSR herein operates with all of its existing optimizations, thus providing a more challenging base for comparison than with a non-optimized, simple flooding-based DSR. The RREPs are further segregated into Cached RREPs and Target RREPs, the former being transmitted by non-destination nodes, the latter by destination nodes. Due to source routing and aggressive caching, DSR has an inherent high hit ratio for its route caches, which could explain why there are much more Cached RREPs than Target RREPs as shown in the figure 4.

The results show that the proposed optimization reduces the number of cached RREP significantly, in particular at higher node mobility (lower pause time). At pause time of 0s (highest mobility) where all nodes are in continuous motion, the number of Cached RREPs is fewer by slightly more than 50%. In addition, expectedly, this margin of improvement decreases with mobility, since lower speed lead to fewer route discoveries to be performed. At pause time of 500s (lowest mobility) where all nodes are stationary, no significant difference in the number of Cached RREPs is observed. In addition, since our optimization is aimed at Cached RREPs, the number of Target RREPs remains relatively unchanged. The same can be said for the RREQs. Fig. 5 shows the overall route discovery overhead, comprising of both RREQs and RREPs. By taking RREQ into account (not shown in the figure for clarity but should be easily extracted), Modified DSR achieves an overall overhead reduction of 33.8% under highest mobility (zero pause time).

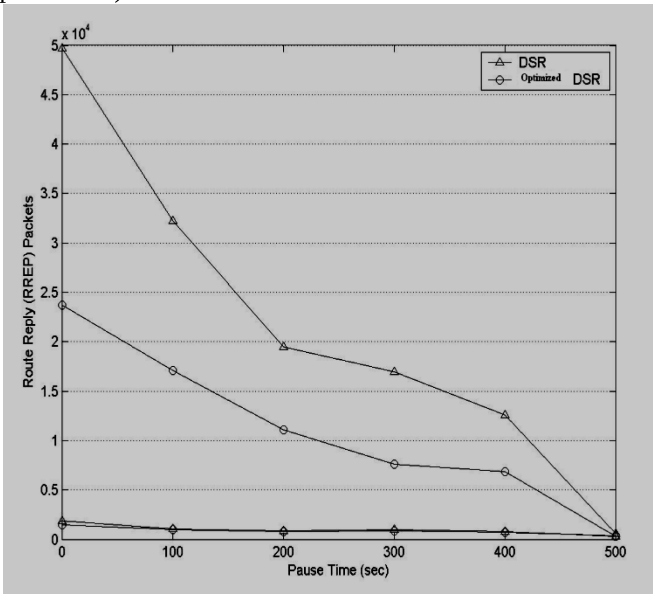

Fig: 4 RREP packets Vs Pause Time

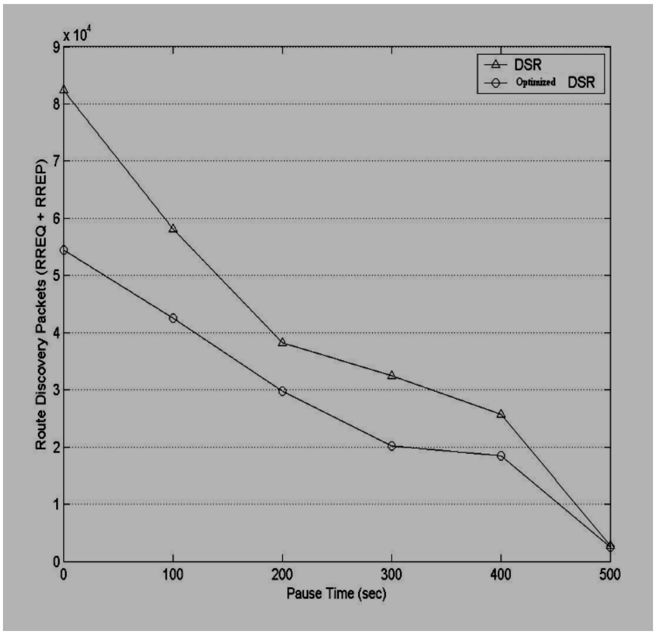

Fig: 5 (RREQ+RREP) Vs Pause Time

VI Conclusion

We have presented a new optimization for DSR to minimize the number of cached RREPs in a route discovery. The proposed optimization is simple in concept and implementation, and is shown to be effective in decreasing the route discovery overhead, in particular under high node mobility.

VII References

1) 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.
2) Hongbo Zhou, "A Survery on Routing Protocols in MANETs," Technical. Note March 2003.
3) Rashida Hashim, Qassim Nasir, Saad Harous," Congestion Aware Multi-path Dynamic Source Routing Protocol (CAWMP-DSR) for Mobile Ad-Hoc Network" in Proceedings of MoMM2007, pp.199-205.
4) S.Basagni, I. Chlamtac, V. R. Syrotiuk, and B. A. Woodward, "A distance routing effect algorithm for mobility (dream)," in Proceedings of the IEEE/ACM international Conference on Mobile Computing and Networking (MOBICOM'98), 1998, pp. 76–84.
5) Staub.T., (2004). Ad-hoc and Hybrid Networks: Performance Comparison of MANET Routing Protocols in Ad-hoc and Hybrid Networks. Institute of Computer Science and Applied Mathematics, University of Berne, Switzerland, pp.1-38.
6) Tutorial for the network simulator "ns". *http://www.isi.edu/nsnam/ns/tutorial/*