ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 2, Issue 7, January 2013

# Performance Enhancement of DSR Routing Protocol Using Mobile Agents

Neetesh Rajpoot[1], Varsha Sharma[2], Dr. R. C. Jain[3]
[1]M.Tech Scholar, Department of Computer Application, SATI Vidisha, Madhya Pradesh.
[2]Assistant Professor, Department of Information Technology, RGTU Bhopal, Madhya Pradesh.
[3]Professor, SATI Vidisha, Madhya Pradesh

***Abstract: - Mobile Adhoc network consists of a large number of wireless nodes communicating with each other without a predefined infrastructure. They communicate by forwarding packets to their successive neighbors falling in the route of intended destination node. It is very challenging to design adhoc network routing protocols because of its decentralized infrastructure. This kind of Infrastructure makes performance and security more challenging for adhoc networks.. Dynamic Source Routing (DSR) protocol is a popular routing protocol designed for use in wireless adhoc networks. Mobile agent is a Remote Method Invocation (RMI) based technology, now a day's very commonly used in diverse fields of network applications. In this paper, we are trying to implement DSR using mobile agents in order to enhance performance of mobile adhoc network. While considering various performance metrics such as QoS, Network Throughput, Packet Reception Ratio and network delay to evaluate working of mobile agent with DSR protocol. We found that DSR can be enhanced its performance using mobile agents.***

***Keywords:*** **- MANET (Mobile Adhoc Network), DSR (Distance Source Routing), Mobile Agents, QoS(Quality of Service).**

## I. INTRODUCTION

Adhoc network consists of a set of wireless nodes communicating with each other without a predefined infrastructure. Nodes having the capability to deliver the packet to the intended receiver successfully by learning routing information dynamically. Therefore Adhoc network does not need to have a centralized server and can be created quickly which make them suitable for applications such as military operations, business meetings outside the office and disaster recovery. Some applications require guaranteed secure and reliable communication as a prerequisite when we use adhoc networks [1].

There are so many factors which affect the performance of network such as dynamic topology, power awareness and QoS. The structure of the network changes very frequently due to the mobility of the nodes. The nodes in the network not only act as hosts but also as routers that route data to or from other nodes in network [2]. In mobile ad-hoc networks where there is no infrastructure support as is the case with wireless networks, and since a destination node might be out of range of a source node transmitting packets; a routing procedure is always needed to find a path so as to forward the packets appropriately between the source and the destination. Within a cell, a base station can reach all mobile nodes without routing via broadcast in common wireless networks. In the case of ad-hoc networks, each node must be capable to forward data to other nodes. This creates additional problems along with the problems of dynamic topology which is unpredictable link changes [3].

Designing a new routing protocol is a challenging task which makes securing adhoc networks more and more challenging. There are two types of routing protocols for adhoc networks: reactive (on-demand) and proactive (table-driven) routing protocols. In reactive routing protocols sending nodes discovers routes whenever they need to send data to target nodes, but proactive protocols maintain fresh lists of destinations and their routes by periodically distributing routing tables throughout the network. Reactive protocols need fewer amounts of data for maintenance and give faster reaction on restructuring and failures in most cases when compared to proactive protocols [2, 7].

Over the last decade a number of routing methods have been proposed each having some pros and cons. ultimately, no single protocol is ideally suited to all applications. This diverts us to consider a novel adaptive and hybrid techniques. One such approach is to use mobile agents to implement the routing protocol thereby supporting on-the-fly protocol modification to permit adaptation to current conditions. While such a scheme is attractive in terms of flexibility, it also has a number of potential drawbacks including questions of performance and security. In this paper we seek to address the first of these issues by designing, implementing and comparing mobile-agent based routing protocol for mobile ad-hoc networks [4, 5].

Section 2 shows how Dynamic Source Routing protocol (DSR) works. Section3 encloses a brief description of mobile agents. Section 4 illustrates a new protocol that combines mobile agents with DSR to enhance performance of adhoc networks. Section 5 compares the new protocol with DSR. Conclusion and future work are given in Section6.

## II. WORKING OF DSR

DSR is composed of two parts; route discovery and route maintenance. It is based on source routing which means that the node sending a data packet lists in its header the nodes that the packet shall go through. A brief discussion for route maintenance and route discovery is given in the following subsection [5].

### *A. Route Discovery*

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 2, Issue 7, January 2013

When an initiating node (initiator) wants to send a data packet to a target node (target), it looks in its route cache for a route to the target node, if a route is found it is used to send the packet. In case no route is found in the route cache the initiator node broadcasts a route request with a unique identifier with respect to the route requests recently sent before from this node to the nodes in its direct radio transmission range in the adhoc network. In case a receiving node has seen a route request from the initiator with the same identifier before; it discards the route request, otherwise if it is the target of the route request; it sends a route reply with the passed nodes, otherwise, it looks for a route to the target of the route request in its route cache and sends a route reply with the route if found, if not found, it appends its address to the passed nodes in the route request and re-broadcasts the route request to the surrounding Nodes in its direct radio transmission range. Route Discovery and route reply are summarized

***B. Route Maintenance***

Route maintenance in DSR is the process of making sure that a sent data packet has reached the destination and there are no broken hops through the route because two nodes became too far, for example. To apply this process, each node receiving a data packet throughout the route listed in the packet's header sends an acknowledgment to its previous node. In case, no acknowledgment is received after a fixed number of retransmission of a data packet this hop is considered as broken and a route error is sent to the sending node using the same route used by the packet to reach the current node. Then, the initiator removes the broken hop from the routes in its route cache and initiates a route request to the target if needed [5, 8].

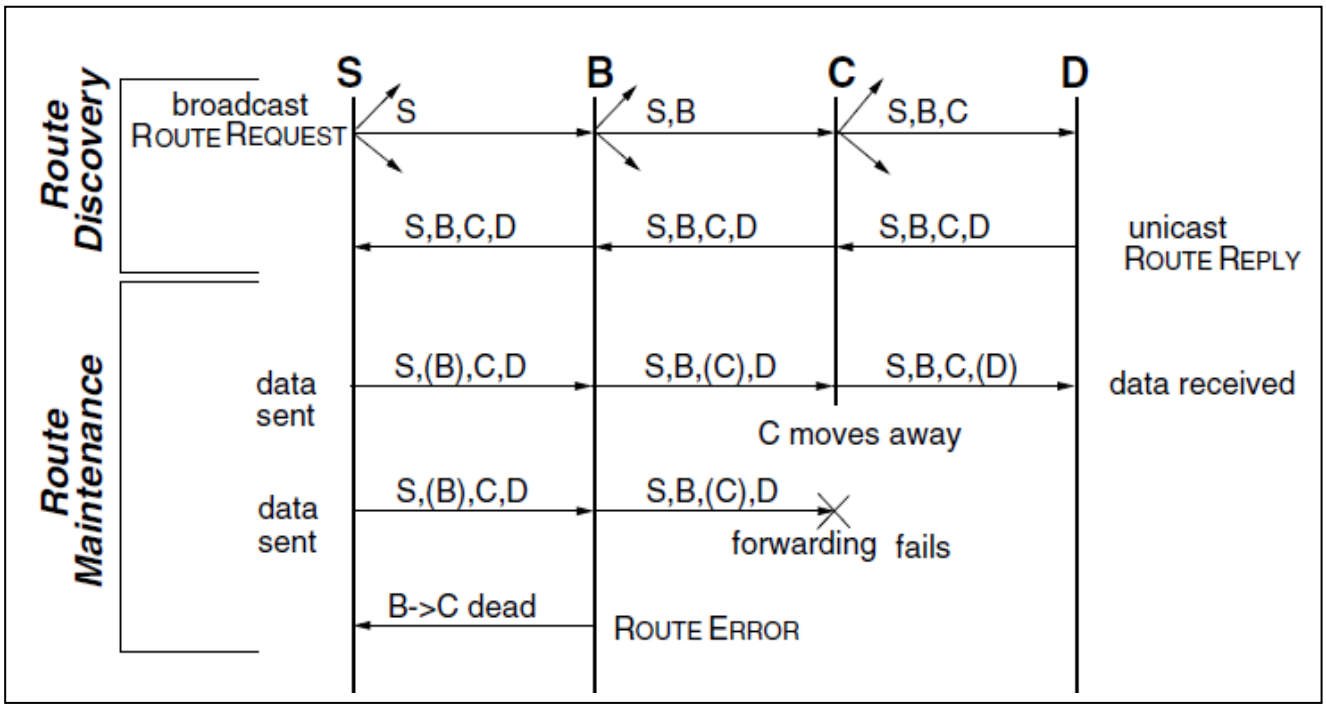


**Fig 1. Route Discovery and Route Maintenance Mechanism of DSR**

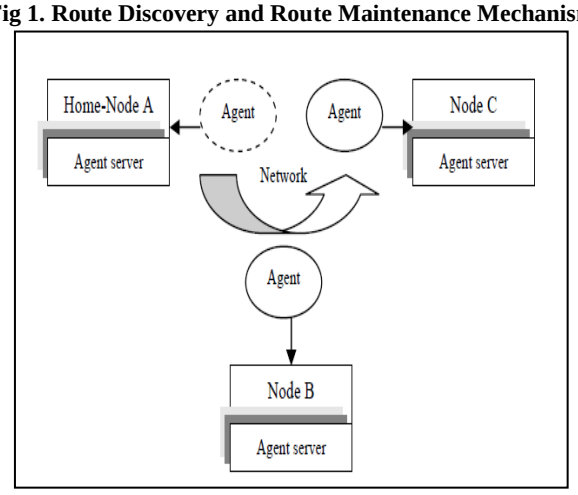


**Fig 2. Mobile Agent System**

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 2, Issue 7, January 2013

Many optimizations have been made to DSR, but, we are focusing here on performance of DSR with possible optimizations with the use of mobile agents. Figure 1 shows the detailed route discovery and route maintenance procedure of DSR protocol.

## III. OVERVIEW OF MOBILE AGENT

A mobile agent consists of the program code and the program execution state (the current values of variables, next instruction to be executed, etc.). Initially a mobile agent resides on a computer called the home machine. The agent is then dispatched to execute on a remote computer called a mobile agent host (a mobile agent host is also called mobile agent platform or mobile agent server). When a mobile agent is dispatched the entire code of the mobile agent and the execution state of the mobile agent is transferred to the host. The host provides a suitable execution environment for the mobile agent to execute. The mobile agent uses resources (CPU, memory, etc.) of the host to perform its task. After completing its task on the host, the mobile agent migrates to another computer [6]. Since the state information is also transferred to the host, mobile agents can resume the execution of the code from where they left off in the previous host instead of having to restart execution from the beginning. This continues until the mobile agent returns to its home machine after completing execution on the last machine in the iteration as shown in figure 2.

### *A. The life cycle of a mobile agent*

- The mobile agent is **created** in the Home Machine.
- The mobile agent is **dispatched** to the Host Machine A for execution.
- The agent executes on Host Machine A.
- After execution the agent is **cloned** to create two copies. One copy is dispatched to Host Machine B and the other is dispatched to Host Machine C.
- The cloned copies execute on their respective hosts.
- After execution, Host Machine B and C send the mobile agent received by them back to the Home Machine.
- The Home Machine **retracts** the agents and the data brought by the agents is analyzed. The agents are then **disposed**.

From this it is observe that a mobile agent experiences the following events in its life cycle:

- Creation: a brand new agent is born and its state is initialized.
- Dispatch: an agent travels to a new host.
- Cloning: a twin agent is born and the current state of the original is duplicated in the clone.
- Deactivation: an agent is put to sleep and its state is stored on a disk of the host.
- Activation: a deactivated agent is brought back to life and its state is restored from disk.
- Retraction: an agent is brought back from a remote host along with its state to the home machine.
- Disposal: an agent is terminated and its state is lost forever.

## IV. DSR WITH MOBILE AGENTS

This section specifies how the performance of an adhoc network can be enhanced by routing protocol DSR with the use of mobile agents. There are two types of mobile agents used in this routing protocol:

- Discovery/reply of mobile agent.
- Maintenance of mobile agent.

### *A. Discovery/reply process of mobile agent*

When a node (A) wants to send a data packet to target node (Z) it creates a discovery mobile agent (DMA) with a new discovery ID, source node =(A), target node=(Z), and having a empty list of passed nodes. The source node sets its state to discovery. Then, source node (A) broadcasts the mobile agent to the nodes to the other nodes in its radio transmission range. Every mobile agent of Discovery/reply type consists of following fields:

1. Discover ID.
2. Source node.
3. Target node.
4. Number of hops.
5. Maximum number of hops.
6. List of passed nodes.
7. State {discovery/reply}.

Once a discovery mobile agent is received by a node and if it received that agent earlier, it will be discarded, otherwise, the node checks if it is the target of the discovery. if it is not the target to that node, it appends its node address to the list of passed nodes in the mobile agent. If it is the target of the desired path, the wireless node appends its node address to the list of passed nodes state of the mobile agent is changed to reply and sends back to the intended source (reverse of the incoming path). Now, the source wireless node has a route to the target node with all the nodes along the route to the target node.

### *B. Maintenance process*

When there is a truncated hop in a route discovered before, the maintenance process is used to notify the sending node with the truncated hop to rediscover a route to the target. A maintenance mobile agent is used to implement the maintenance when it is sent by the first node in the truncated hop having a new maintenance ID, first node in truncated hop, second node in truncated hop, and all the previous fields. Maintenance mobile agent consists of following internal fields in it:

1. Maintenance ID.
2. First node in truncated hop.
3. Second node in truncated hop.

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 2, Issue 7, January 2013

## V. PROTOCOL EVALUATION

To evaluate the proposed DSR with mobile agents' protocol, Network Simulator (NS-2) has been used. The NS-2 [9] has been used extensively in evaluating the performance of ad hoc network routing protocols. We Implemented and Evaluated the DSR and DSR with Mobile Agent both on NS-2. The nodes in the simulation will be moving according to the random waypoint model [10]. We used a rectangular space of size 800 m · 600 m to increase the average number of hops in the routes used relative to a square space of equal area, creating a more challenging environment for the routing protocol in this respect.

The DSR Protocol and DSR with Mobile Agent simulated for different number of flows (05, 10, 15, 25, 40, 50) to evaluate each of them. For each run three metrics have been computed:

1. Packet delivery ratio: representing the ratio of data packets received at its destination.
2. Average end-to-end delay: representing the average time taken for data packet before it was received at its destination.
3. Average Throughput: It is the number of bits passed through a network in one second. It is the measurement of how fast data can pass through an entity (such as a point or a network).

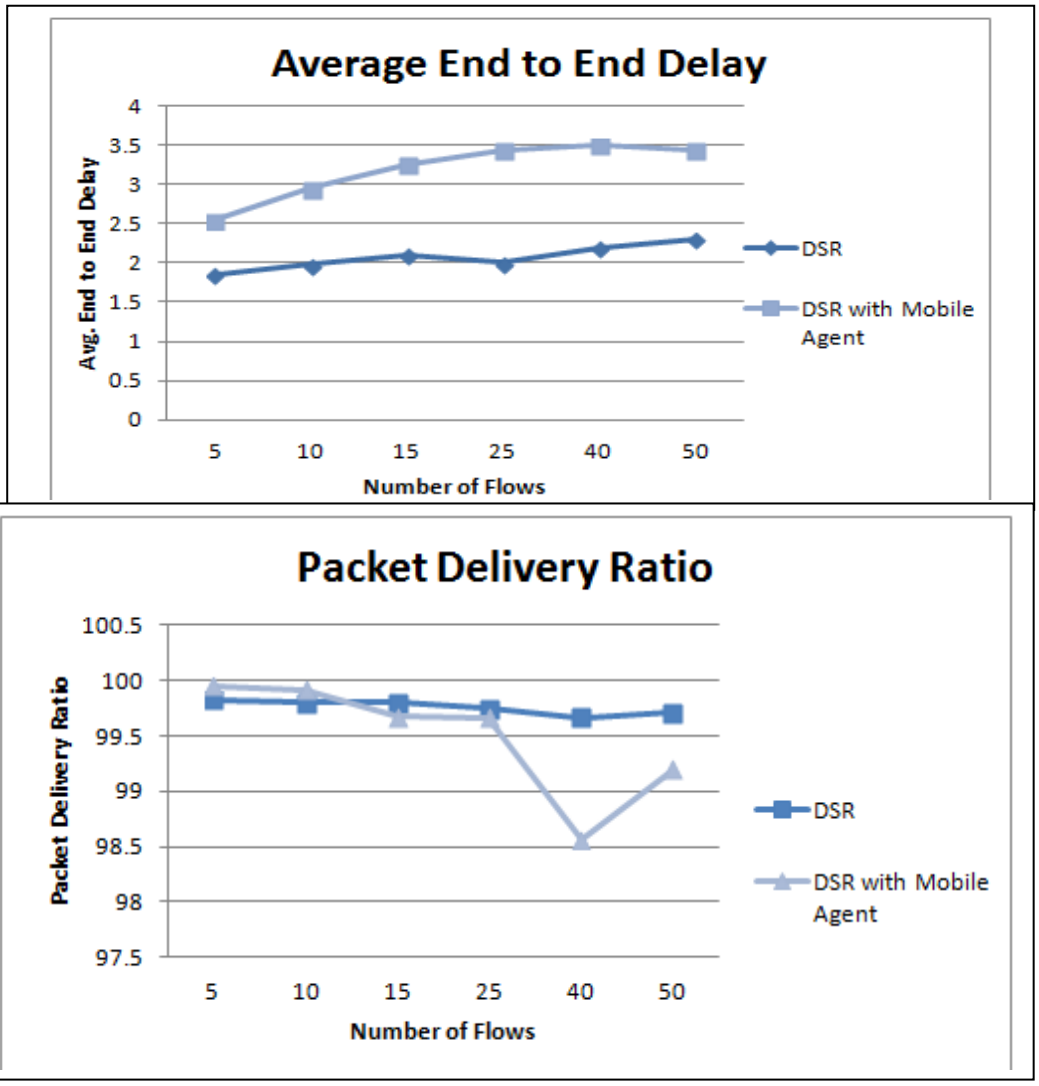


**Fig 4 Packet Delivery Ratio**

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 2, Issue 7, January 2013

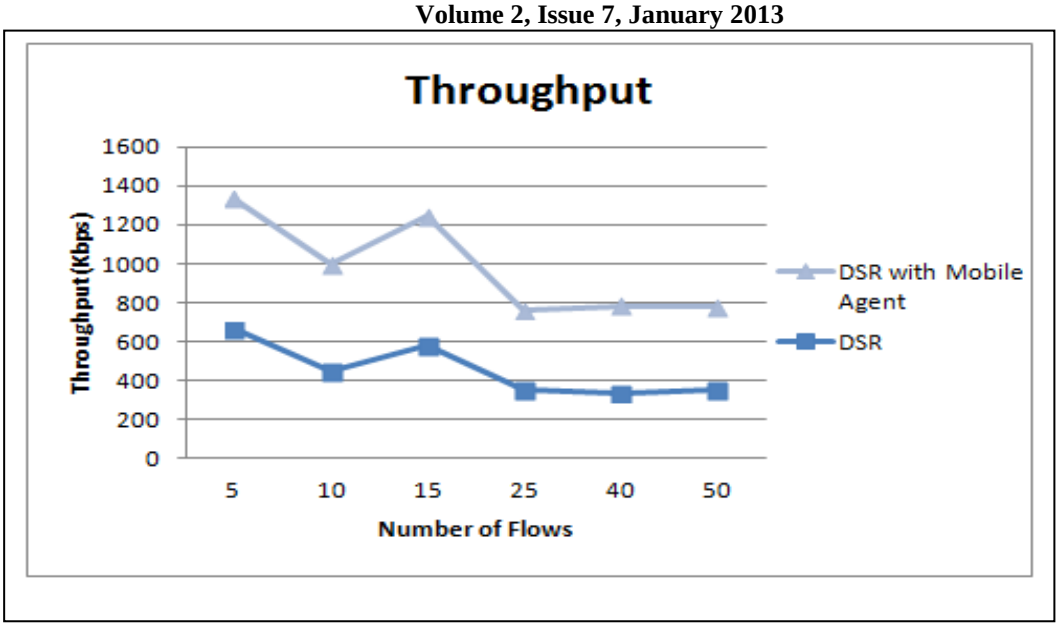


**Fig 5 Average Throughput**

Results have shown in above graphs, and it is observed that there is a performance overhead for DSR with Mobile Agent while considering Packet Delivery Ratio. It's because of marshalling and de-marshalling included in Mobile Agents. Apart from that average threshold and end-to-end delay is optimal while considering DSR over DSR with mobile Agents.

## VI. CONCLUSION

We have designed and implemented a protocol similar to DSR but in addition with DSR it also includes the functionality of Mobile Agent with each wireless mobile node. We compared results obtained from simulation, in order to evaluate DSR with Mobile Agent to existing DSR protocol. We found that for packet delivery ratio DSR outperforms over DSR with Mobile Agent by an average of 0.95%, but in the average end-to-end delay DSR with Mobile Agent outperforms existing DSR by an average of 1.22 s, and for the Average Threshold DSR with Mobile Agent outperforms DSRM. Results show that there is a performance overhead for DSR with Mobile Agent over existing DSR protocol which is due to inclusion of mobile agents.

In future, mobile agents may be applied to protocols other than DSR, like AODV (ad hoc on demand distance vector), and TORA (temporally ordered routing algorithm.) and may be also applied in the Wireless Sensor Networks.

## REFERENCES

[1] Xiaoyan Hong, Jiejun Kong, and Mario Gerla: Mobility changes anonymity: new passive threats in mobile ad hoc networks. Wireless Communications and Mobile Computing. Vol. 6, No. 3, pp. 281-293, 2006.

[2] Amit Kumar Sanghi, Dharm Singh, Heena Rathore "A Study of Wireless Environment by means of Multiple Wireless Nodes", International Journal of Computer Applications on Wireless Communication and Mobile Networks, No.7: 31-35, Jan.2012.

[3] Elizabeth M. Royer and C-K Toh. "A Review of current Routing Protocols for Ad-hoc Mobile Wireless Networks", IEEE Personal Communications, Vol. 6, No.2, pp. 46-55, April 1999.

[4] S. Corson and J. Macker. Mobile ad hoc networking (manet). RFC 2501, IETF, January 1999.

[5] D. B. Johnson and D. A. Maltz, "Dynamic source routing in ad hoc wireless networks," in Mobile Computing, ser. The Kluwer International Series in Engineering and Computer Science, T. Imielinski and H. F. Korth, Eds. Springer US, 1996, vol. 353, pp. 153–181.

[6] RPC (Remote Procedure Call). http://en.wikipedia.org/wiki/Remote_procedure_call

[7] Ad hoc wireless networks <http://en.wikipedia.org/wiki/Ad_hoc_routing_protocol_list>.

[8] DSR (Dynamic Source Routing). <http://en.wikipedia.org/wiki/Dynamic_Source_Routing>.

[9] NS-2 simulator. <http://www.isi.edu/nsnam/ns/>.

[10] Johnson D, Maltz D. "Dynamic source routing in ad hoc wireless networks." In: Imielinski Tomasz, Korth Hank, editors. Mobile computing. Kluwer Academic Publishers; 1996. p. 153–81 [Chapter 5].