International Journal of Computer Applications (0975 – 8887)
Volume 138 – No.2, March 2016

Adaptive-Transmission-Power Ad Hoc On-Demand Distance Vector Routing Protocol for Mobile Ad hoc Network

M. Izharul Hasan Ansari
Research Scholar
D/o Computer Sc. & Engg.
NIMS University Jaipur

S.P. Singh, PhD
Associate Professor
D/o Computer Sc. & Engg.
NIMS University Jaipur

M.N. Doja
Professor
D/o Computer Engineering
JMI University, N. Delhi

ABSTRACT

The conventional routing protocols in mobile ad hoc network (MANET) using conventionally a common transmission power for both tranmsission and overheads to transfer the data packet from the source to the destination node have been revisted. Hence, an technique was developed to establish an algorithm—Adaptive-Transmission-Power Ad Hoc On-Demand Distance Vector (ATP-ADOV) routing protocol - to control the transmission of power dynamically and overheads in MANET. The proposed ATP-ADOV reduced the energy consumption in the networks and improved the lifetime of the particiapting mobile nodes as well as that of the lifespan of the networks.

General Terms

MANET, Protocol, Routing, Simulator, Transmission power, Adaptive Transmission Power, Energy consumption, efficiency and lifetime

Keywords

MANET, Routing protocol, Network simulator, Transmission range, Throughput, Delay, Packet delivery ratio, Energy consumption, efficiency and lifetime

1. INTRODUCTION

The mobile ad hoc network (MANETs) is a collection of zero configurable mobile nodes without any physical infrastructure and centralized computing. The participating mobile nodes are free to move and may join, leave and rejoin the network at any time without any prior information and permission [1]. The topology of MANET is highly dynamic and unpredictable. The frequent movement of the participating mobile nodes leads to route failures calling for the route maintenance and frequent activation of route discovery process. Besides, due consideration should be given in MANET to reduce the transmission power and loss of energy. This is because on one hand higher transmission powers cause an increase in the overheads during the transmission of data from one node to another and on the other hand lower transmission powers adversely affect the participating mobile nodes by not allowing them to keep the network live for a longer duration thereby causing a loss of energy [1].

Over the last few years the various energy management schemes employing energy efficient routing protocols have been proposed in MANET to minimize the utilization of the battery power of the participating mobile nodes of the networks and extend the network lifetime [1,2]. In this paper, the Ad hoc On-Demand Distance Vector (AODV) routing protocol has been analyzed and an Adaptive-Transmission-Power-AODV(ATP-AODV)routing protocol to control the transmission power dynamically and overheads for reducing the energy consumption has been proposed. Further, the performance analysis of the proposed algorithm ATP-AODV has been performed to get the consumed energy in transmitting data packets, and residual energy of the participating mobile nodes of the networks.

Our work is based on the simulation using Random Way Point Mobility Model [2]. The investigation into the role of the required tramnsmision range from one node to another to minimize the energy consumption, a study to the best of the authors' knowledge has not been done much and reported in literature.

The paper is divided in eight sections including the present introductory section. In Section 2, earlier work related to the present study is presented. Section 3 is an overview on ADOV. In Section 4, the various concepts related to the network simulators including NS2 is given, while Section 5, is devoted to describing the simulation setups, simulation environment, and mobility model, which have been subsequently used by us in our study of the evaluation of the performance of ADOV reported in Section 6. In Section 7 the performance of the proposed algorithm and in Section 8, the conclusion has been given.

2. RELATED WORK

In [3] is introduced the Minimum Energy Dynamic Source Routing (MEDSR) protocol for MANET in which the route discovery has been suggested both in low and high power levels. In this protocol, a higher power level is sought if three attempts of route request from one node to the next for the route discovery fail at a lower power level. However, in MEDSR protocol, the energy is conserved and the overall lifetime of the network is increased at the cost of the delay per data packet since the travel of data packets to the destination node involves a large number of hops. Thus, there is a scope for the improvement in the delay in this protocol.

Narayanaswamy *et al.* [4] proposed Common Power (COMPOW) control in MANET. It is based on the following observation. Excessively high powers cannot be used to transmit the data packets from the source to the destination node because of the shared medium, which also causes lot of interferences. This affects the traffic carrying capacity of the network and reduces the battery life. On the contrary if the network chooses low powers for establishing the routes then it leads to the route failure calling for the route maintenance and route discovery process to activate very frequently, which causes a loss of significant amount of energy. Therefore, the network power level must be chosen neither too high to cause excessive interference which results in a reduced ability to carry traffic, nor too low to result in a disconnected network. The technique of COMPOW control has been designed and tested only for table driven routing protocols and apart from this the technique is viable only for very dense network where the number of participating mobile nodes is very high and the covering area is small.

Hiremath and Joshi [5] proposed a fuzzy adaptive transmission range and fuzzy based threshold energy for the location aided routing protocol, namely Fuzzy Adaptive Transmission Range Based Power Aware Location Aided Routing (FTRPALAR). In this protocol proposed by them, the energy of a mobile node is conserved by employing a fuzzy adaptive transmission power control depending on the minimum number of neighboring nodes to maintain the network connectivity and power aware routing based on fuzzy threshold energy. Further, the experimental results on FTRPALAR obtained by them performs better in terms of the average energy consumption and network lifetime as compared to the conventional location aided routing (LAR) protocol and the variable transmission range power aware location aided routing (VTRPALAR) protocols. The proposed FTRPALAR is able to achieve 18% more lifetimes than VTRPALAR.

Tarique and Tape [6] proposed Minimum Energy Dynamic Source Routing (MEDSR) and Hierarchical Minimum Energy Dynamic Source Routing (HMEDSR) protocols. The MEDSR protocol uses two different power levels during the route discovery process to identify low-energy paths. After finding the path, the transmitted power levels of the nodes along the routes are adjusted link by link to the minimum required level. However, the MEDSR protocol uses the flooding during route discovery process resulting in enhanced overhead in large networks thereby affecting the routing performance severely. Although the overhead packets are not in large numbers yet they consume significant amount of energy. This drawback of the MEDSR protocol is alleviated in the HMEDSR protocol which is basically the combination of the protocols MEDSR and Hierarchical Dynamic Source Routing (HDSR), the latter reducing the overhead while the former saving energy in the transmission of data packets [6].

3. OVERVIEW OF ADOV ROUTING PROTOCOL

The ADOV protocol, which comes under the purview of reactive routing protocols, is of on-demand type in the sense that the route between two nodes is discovered only when it is needed. Such protocols are designed to make them least overburdened while they maintain the information only for those routes which are active [7]. It means that, in the process of route discovery and route maintenance, the routes are discovered and maintained only for the nodes that send their request to the specific destination. The various issues related to the ADOV protocol have been discussed in this section except the simulation parameters for analyzing the performance of ADOV which have been taken up separately in Section 5.

3.1 Route Discovery in AODV Protocol

The basic approach in the route discovery process is to establish the route in an on-demand routing protocol by broadcasting the route request message in the network. The destination node, on receiving a route request message, replies by sending a route reply message back to the source. The route reply message carries the route back to the source node that is traversed by the route request message received at the destination node [7]. In this process, when a particiapting node of a network wishes to send a data packet to some destination node then the source node checks its routing table to determine whether it has a current route to that destination node [7]. If the route is available with the destination node then it forward the pakcet to the appropriate next hop towards the destination node. However, if the particpating mobile node does not have a valid route to the destination node then the node must initiate the route discovery mechnism. Further, to begin this route discovery process, the node creats a RREQ packet and broadcast the route request packet at a low power level. Such a packet contains the source node IP address and current sequence number as well as the destinations IP address and the last known sequence number. The RREQ packet also contains a broadcast ID, which is increemented each time the source node initiate a RREQ. In this way, the broadcast ID and the IP address of the source node form a unique indetifier for the RREQ. After creating the RREQ, the source node broadcasts the packet and then sets a timer to wait for a reply. When a node receives a RREQ, it first checks wheather it has seen it before by noting the source IP address and the braodcast ID pair. Each node maintains a record of the source IP address / bradcast ID for each RREQ it receives, for a specified length of time. If it has already seen a RREQ with the same IP address / broadcast ID pair, it silently discards the packet. Otherwise, it records this information and then process the packet [7]. Further, in order to process the RREQ, the node sets up a reverse route entry for the source node in its route table. This reverse route entry contains the source nodes IP address and the sequence number as well as the number of hops to the source node and the IP addresses of the neighbor from which the RREQ was received. In this way, the node knows how to forward a RREP to the source if one received later [7]. Figure. 1 indicates the propagation of RREQs across the network as well as the formation of the reverse route entries at each of the network nodes. Moreover, a lifetime is associated with the reverese route. If this route entry is not used within the specified lifetime, the route information is deleted to prevent the old routing information from lingering in the route table [7].

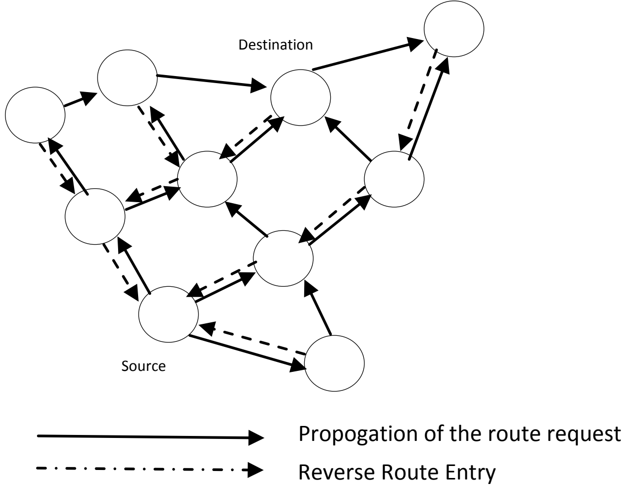

Figure 1: Propogation of route request (source: [7])

3.2 Propogation of Route Request

It is necessary for responding to the RREQ that the node must have an unexpired entry for the destination in its route table. Furthermore, the sequence number associated with the destination must be at least as great as that idicated in the RREQ. This prevents the formation of routing loops by ensuring that the route returned is never old enough to point to a previous intermideiate node since otherwise, the previous node would have responded to the RREQ [7,8]. If the node is able to satisfy these two requirements, it responds by unicasting a RREP back to the source as descirbed in the next section. If it is unable to satisfy the RREQ, it increments RREQ hop counts and then broadcasts the packet to its neighbor. Natuarlly, the destination node is always able to respond to the RREQ. If the route request packet is lost, the source node is allowed to retry the broadcast route discovery mechnism. After RREQ-retries additional attempts, it is required to notify the application that the destination is unreachable [7,8].

3.3 Forward Path Setup in AODV Protocol

When a route determines that it has an enough route current to respond to the RREQ, it creats a RREP [7]. For the purposes replying to a RREQ, any route with the sequence number not smaller than that indicated in the RREQ is deemed as asscociated with enough current. The RREP sent in response to the RREQ caontains the IP address of both the source and the destination. if the destination node is responding , it places its current sequence number in the packet, initilizes the hop count to zero, and then places the length of time of this route as valid in RREPs lifetime field. However, if an intermediate node is responding, it places its record of the destination's sequence number in the packet, sets the hop count equal to its distance from the destination, and calculates the amount of time for which its route table entry for the destination will still be valid. It then unicasts the RREP towards the source node, using the node from which it received the RREQ as the next hop [7,8].

When the intermediate node receives the RREP, it sets a forward path entry to the destination in its route table. This forward path entry contains the IP address of the destination, the IP address of the neighbor from which the RREP had arrived, and the hop count, or the distance, to the destination. To obtain its distance to the destiantion, the node increaments the value in the hop count field by 1. Also associated with this entry is a lifetime, which is set to the lifetime contained in the RREP. Each time the route is used, its associated lifetime is updated. If the route is not used with in the specified lifetime, it is deleted. After processing the RREP, the node forwards it towards the source [7,8]. Figure 2 indicates the path of the RREP from the destination to the source node.

3.4 Route Dicovery from source to Destination

It is likely that a node will receive a RREP for a given destination from more than one neighbor [7,8]. In this case, it forwards the first RREP it receives and forwards a later RREP only if the RREP contains a greater destination sequence number or a smaller hop count. Otherwise, the node discards the packet. This decreases the number of RREPs propogating towards the source while ensuring the most up-to-date and quickest routing information. The source node can begin data transmission as soon as the first RREP is received and can later updates its routing information if it dicovers a better route [7,8].

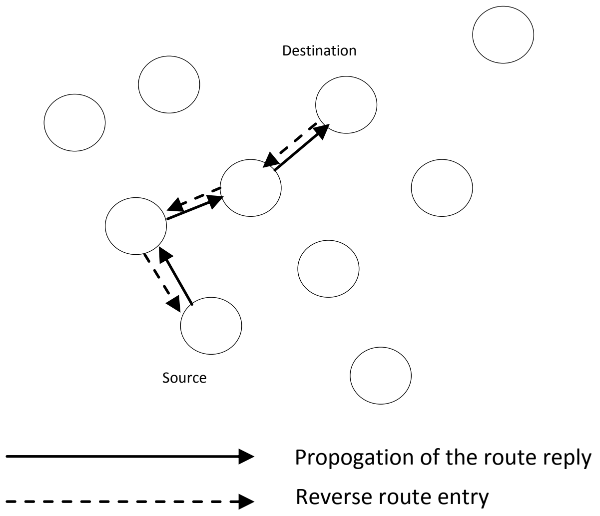

Figure 2: Propogation of the route reply (source: [7])

4. SIMULATION DETAILS

In this section we have decribed the network simulator NS-2, the execution process, the process to genrate the movement, the process of trafic genration, the mobility model as well as the simulation paramenters used in our simualtion.

The simulation technology as applied to the networking areas like network traffic simulation is relatively new [9]. The computer assisted simulation technologies can be applied in the simulation of networking algorithms or systems by using software engineering. The application field is smaller than that in general field of simulation and it could be natural that more specific and desirable requirements will be placed on network simulations in future [9]. For example, the network simulations can have more emphasis on the validity and performance of a distributed protocol or algorithm than on the visual or real-time visibility features of simulations. Moreover, one has to keep pace with the rapidly developing network technologies running on different software over the Internet with the involvement of many different organizations contributing to the whole process. That is why the network simulation always requires open platforms or software which should be scalable and enough to include different packages in the simulations of the whole network. Internet has also a main characteristic that it is structured with a uniformed network stack (TCP/IP) that has all the different layers of technologies which can be implemented in different ways while having uniform interfaces with their neighbored hops and layers [9,10]. Thus, the network simulation tools must be able to incorporate these features and allow different future aspects and new packages to be included and run transparently without any harm to and with either no impact or at least no negative impact on the existing components or packages [9,10]. Network simulators are mainly used by people from different backgrounds and areas like industrial developers, academic researchers and quality assurance (QA) for designing, simulating, verifying, and analyzing the performance of different networks protocols. Network simulators can also be used to evaluate and analyze the effect of the different parameters on the protocols being studied for network scenarios. Generally a network simulator will contain a wide range of networking protocols and technologies that help users to build complex networks from vary basic building blocks for example clusters of nodes and links. With the help of network simulators, one can design and propose different network topologies with the help of various types of nodes like end-hosts, hops, network bridges, routers, and mobile units [10]. The present section is thus of relevance to the simulation based study taken up in this paper on AODV protocol to know effect of the variable transmission range on it as mentioned in Section 1.

4.1 Network Simulator NS-2

The network simulator NS-2, which provides the discrete event simulation with its implementation initiated as early as 1989 with the development of the real network simulator [9], is very flexible and capable of supporting the simulation of different types of routing algorithms of MANET, TCP and also multicast protocols over wired and wireless networks [9]. Initially, it was designed and developed for the simulation of wired technology only but later the Monarch Group of the department of computer science at the University of Rice developed the necessary tools and applications to include in the simulator for the wireless and mobile hosts [9]. In NS-2, the simulations are written in C++ with an OTcl API.

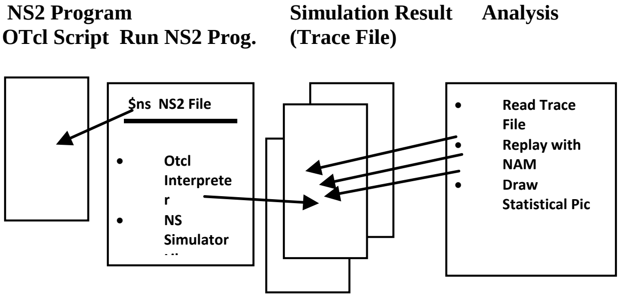

Figure 3: Running of NS2 program.

The user creates a text file in OTcl which describes the layout of the whole network as well as the events to be occurred such as transferring data or node movement application. This OTcl file (.tcl) is executed and a detailed trace file (.tr) is generated which can be filtered with a pattern matching program (such as 'grep' or 'awk') and inspected by hand, or fed into a visualization tool [9]. Some visualization tools are also available with NS-2, one of which is the Network Animator (NAM). NAM is an animation tool for viewing network simulation traces in graphical form. It supports topology layout and has various data inspection tools. NS-2 is suitable for simulating MANET because it has accurate implementations of the IEEE 802.11 standard, a TCP / IP stack and a wide range of routing protocols implemented for NS-2 [9].

4.2 Node Movement Generation for Wireless Scenarios in NS-2

We can define the node movement in separate files called as scenario file in NS with the help of node movement generation tool available in NS-2. This scenario file is generated with help of this tool which is available on the "ns-2/indep-utils/cmu-scen-gen/setdest/" location. This scenario file is used to store the information about the initial position of the nodes with their movement details, speed, etc. at various points of time. Generally, since it is very difficult to provide the initial position of the participating mobile nodes manually, movement of the nodes and their speed for each movement at different times we use a random file generator. We can run this tool with following command [9]:

```
./setdest –n [num of nodes] –p [pause time] –m[max speed] –t[sim time] –x[max x] –y[max y]>[outdir/movement file],
```

for example:

```
./setdest -n 20 –p 0.0 -m 2.0 -t 200 -x 1000 -y 1000 > scen-20-test
```

Here, the number of nodes n is: $n = 20$, the pause time p is $p = 0.0$ s, maximum speed m is $m = 2.0$ m/s, simulation time t is $t = 200$ s, and area $x \times y$ is equals to 1000× 1000, and scen-20-test is the desired scenario file.

4.3 Random Traffic Generation for wireless scenarios in NS-2

In NS-2 we can set up random traffic connections between mobile nodes of TCP and CBR using a traffic-scenario generator script. This script is available at ~ns/indep-utils/cmu-scen-gen location and is called cbrgen.tcl. It is used to generate CBR and TCP traffics connections between mobile nodes. For this purpose we create a traffic-connection file, in which we need to define different parameters like the type of traffic connection (CBR or TCP), maximum number of connections to be set up between the nodes, the number of nodes and a random seed and, for CBR connections, a rate. The inverse value of the rate is to compute the interval time between the packets [9]. We can generate this with the following command:

```
ns cbrgen.tcl [-type cbr/tcp] [-nn nodes] [-seed seed] [-mc connections] [-rate rate] > [outdir/movement file]
```

The start times for the connections are generated randomly with a maximum value of 180.0s. For example, we can have a CBR connection file with 10 nodes, having maximum of 8 connections, seed value of 1.0 and a rate of 4.0. So we can do this with following command:

```
ns cbrgen.tcl -type cbr -nn 10 -seed 1.0 -mc 8 -rate 4.0 > cbr-10-test
```

This generates a random traffic pattern with described values.

4.4 Tool Command Language (Tcl)

There are two languages used in NS2 C++ and OTcl (an object oriented extension of Tcl) [9]. The compiled C++ programming hierarchy makes the simulation efficient and execution times faster. The simulation results produced after running the scripts can be used either for simulation analysis or as an input to NAM.

Tool Command Language Tcl is a powerful interpreted programming language developed by John Ouster out at the University of California, Berkeley [9]. Tcl is a very powerful and dynamic programming language. Tcl is a truly cross platform, easily deployed and highly extensible. The most significant advantage of Tcl language is that it is fully compatible with the C programming language and Tcl libraries can be interoperated directly into C programs.

4.5 NAM

NAM for the graphical representation of the simulation have been designed and developed in 1990 as a simple tool for animating packet trace data [9]. This trace data is typically derived as the output from a network simulator like NS or from real network measurements, e.g., using tcpdump. Steven McCanne wrote the original version as a member of the Network Research Group at the Lawrence Berkeley National Laboratory, and has occasionally improved the design [9]. Marylou Orayani improved it further and used it for her Master's research over summer 1995 and into spring 1996 [9]. The NAM development effort was an ongoing collaboration with the Virtual InterNetwork Test bed (VINT) project. Currently, it is being developed at ISI by the Simulation Augmented by Measurement and Analysis for Networks (SAMAN) and Conser projects [9].

4.6 Trace File

The trace file is an ASCII code files and the trace is organized in 12 fields as in Figure 4.

1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.

1.*Event,* 2. *Time,* 3. *From Node,* 4. *To Node,* 5. *Pkt. Type,* 6. *Pkt. Size,* 7. *Flags,* 8. *Fid,* 9. *Src Add,* 10. *Dst Addrs,* 11. *Seq. Num,* 12. *Pkt. Id.*

Figure 4: Fields of the trace file.

The first field is the event type and given by one of four available symbols *r*, +, - and *d* which correspond respectively to receive, en-queued, de-queued and dropped. The second field is telling the time at which the event occurs. The third and fourth fields are the input and output nodes of the link at which the events takes place. The fifth is the packet type such as Constant Bit Rate (CBR) or Transmission Control Protocol (TCP). The sixth is the size of the packet and the seventh is some kind of flags. The eighth field is the flow identities of IPv6, which can specify stream color of the NAM display and can be used for further analysis purposes. The ninth and tenth fields are respectively the source and destination address in the form of “node.port”. The eleventh is the network layer protocol’s packet sequence number. NS keeps track of UDP packet sequence number for the analysis purposes. The twelfth, which is the last field, is the unique identity of the packet. Results of simulation are stored into trace file (*.tr). Trace Graph was used to analyze the trace file.

5. SIMULATION SETUP

Let us now consider the simulation setup as relevant to the ultimate goal of our research work which is to evaluate the dependence of AODV protocol on the various parameters and then develop a new technique for the optimum utilization of transmission power for transmitting the data packet from the source to the destination node successfully and in timely manner so as to increase the lifetime of the battery of the participating mobile nodes and that of the network as a whole. The five parameters which we have chosen for our studies are: (i) the density of network, (ii) the speed of participating mobile nodes, (iii) the transmission power of the participating mobile nodes and (iv) the data traffic pattern. In the simulation the participating mobile nodes move according to a model called Random Waypoint which was proposed by Johnson and Maltz [11,12]. Now, this mobility model is widely being used because of its simplicity and wide availability. To generate the node trace of the Random Waypoint model the setdest tool from the CMU Monarch group may be used. This tool is included in the widely used network simulator ns-2 [12].

In NS-2 distribution, every participating mobile node randomly selects one location in the simulation field as the destination. The participating nodes then move towards the destination with constant velocity chosen uniformly and randomly from (0,V), where the parameter V_{max} is the maximum allowable velocity for every participating mobile node of the networks [12]. After reaching the destination, the node stops for a duration defined by the pause time. If pause time $T_{pause} = 0$ then the participating mobile does not stop and keeps itself moving continuously. After this pause time, it once again chooses another random destination in the simulation field and moves towards it. The same process is repeated again and again until the simulation ends.

In the Random Waypoint mobility model, V and T are the two key parameters that determine the mobility behavior of nodes. If V is small and T is long, the topology of ad hoc network becomes relatively stable. On the other hand, if the node moves fast (i.e.,V_{max} is large) and T is small, the topology is expected to be highly dynamic. Varying these two parameters, especially the V parameter, the Random Waypoint model can generate various mobility scenarios with different levels of nodal speed. Therefore, it seems necessary to quantify the nodal speed. Intuitively, one such motion is the average node speed. If we could assume that the pause time $T_{pause} = 0$ max, considering that V_{max} is uniformly and randomly chosen from (0,V), we can easily find that the average nodal speed is 5.0 m/s [13,14]. However, in general, the pause time parameter should not be ignored. In addition, it is the relative speed of two nodes that determines whether the link between them breaks or forms, rather than their individual speeds. Thus, the average node speed seems not to be the appropriate metric to represent the notion of the nodal speed.

5.1 Mobility Model

During simulation each node starts moving from its initial position to a random direction with random speed. The speed is uniformly distributed between 0 and the maximum speed. When a moving node reaches the boundary of the given area, it waits for the pause time (which is 0 in our case) and after that once again starts to move in a random direction and with random speed. The entire traffic source used in our simulation generated Constant Bit Rate (CBR) data traffic. The traffic structure was defined by varying two factors: (a) the sending rate and (b) the packet size.

5.1.1 Simulation Environment

As the basic scenario we considered a MANET with 10,20,30,40 and 50 mobile nodes spread randomly over an area of 1000 m by 1000 m. The nodes were moving with a maximum speed of 2 m/sec with pause time of 0 sec. A total of 01 traffic sources generated CBR data traffic with a sending rate of 4 packets/sec, using a packet size of 512 bytes.

Each simulation had the duration of 500 simulated seconds. Because the performance of the simulations is highly related with the mobility models, the result to be shown in Section 6 represents an average of three different executions of the simulation using the same traffic models but with different randomly generated mobility scenarios.

Table 1: Simulation parameter

Parameters	Values
Simulator	NS-2 (v-2.34)
Area (m x m)	1000 x 1000
Nodes	10,20,30,40,50
Simulation time (s)	500
Node speed (m/s)	2
Pause time (s)	0
Traffic type	CBR
Packet size (byte)	512
Protocol used	AODV
Mobility model	random way point
MAC layer protocol	Mac/802_11
Antenna type	Omni-directional
Frequency (MHz)	914
Transmitted signal power (W)	0.2818

Transmission power consumption (W)	1.6
Power consumption for reception (W)	1.2
Idle power consumption (W)	0.0
Capture threshold (dB)	10.0
Carrier sense threshold power (W)	1.559e-11
Receive power threshold (W)	3.652e-10
System loss factor	1.0
Data rate (Mbps)	2
Transmission range (m)	200, 250

We evaluate the following performance indexes: (i) total energy consumed (in Joules), (ii) energy consumed depending on the operation (Transmissions (Tx) and Receptions (Rx)), and (iii) energy consumed depending on the packet type (MAC, CBR and routing).

6. ROUTE DISCOVERY MECHNISM IN ADAPTIVE TRANSMISSION-POWER AD HOC ON-DEMAND DISTANCE VECTOR (ATP-ADOV) ROUTING PROTOCOL

A scenario of network has been designed as given in Figure 5 indicating nine mobile nodes that participate in the network namely A, B, C, D, E, F, G, H and S. Here the node S is the source node, D is the destination node and the remaining others are the intermediate nodes. The source node S instiates the route discovery process and forwards the route request packet to its neighbouring nodes A and G, which are within the transmision range of the node S. In the first attempt the source node tranmits the route request packet with minimum power level and sets the Rx (receiving power) at the nodes A and G.

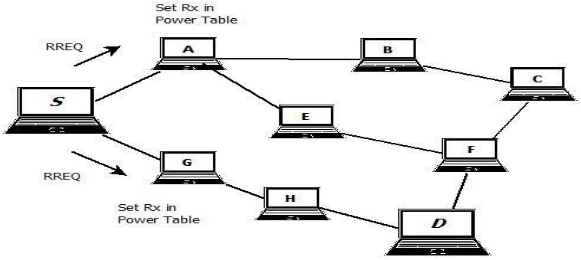

Figure 5: Intiatiation of route request packet by the source node S to its neigbours A and G.

Now if the nodes A and G are not the destination nodes then first the power levels at nodes A and G are set and acknowledgment sent back to the source node or the predecessor nodes of the networks. The same route request packet is forwarded further to their neighbours—here in the given scenario as shown in Figure 6, the participating node A forwards the route request to its neighbouring nodes B and E and sets the Rx power for B and E in their power table. Here, once again if these nodes B and E are not the destnation and desired nodes then once again the node A receives acknowldgments from the nodes B and E.

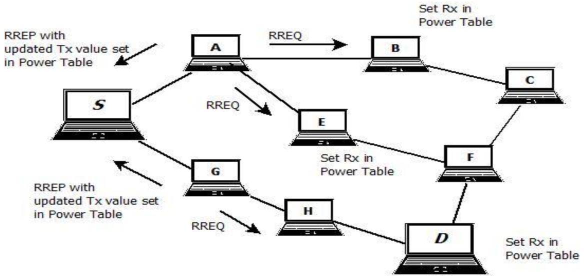

Figure 6: Receiving of the reverse node entry by the source node S from the nodes A and G and forwarding of the route request packets to their neighbors: from the node A to B and E; and from the node G to H.

Similarly, the node G forwards the route request packet to its neighbhoring node H. Here once again if the nodes B, E and H are not the destination and desired nodes then once again node A receives acknowldgments from the nodes B and E, and the node G receives the acknowledgement from the node H. The node H also forwards the route request pakcets to its neighbouring nodes of the networks, and, as shown in Figure 7, **the** node D receives the route request packet from the node H as the node D acknowledges the route request to the node H and sets the Rx power in its power table, D being the the destination and intended node, where, the desired packet is to be deliverd.

Further, the node B and E forwards the route request packets to their neighbouring nodes namely C and F, and node D reply the route reply packets to G and copy its Rx power to the format of the route reply packet and sets this Rx power at the predecessor node as Tx power for the successor node, which is node D in this scenario (Figure 6).

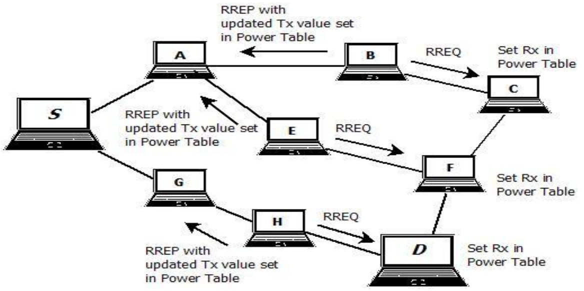

Figure 7: Receiving of the reverse routes entries by the nodes A and G and forwarding of the route request by the nodes B, E and H to C, F, and D respectively.

The node H, as depicted in Figure 7, receives the route reply packets and updates its power table and sets the Tx value for the destination node D. Whenever the node H receives the data packet from its predecessor node for the delivery to the destination node the node H sends the data packet to the destination node with Tx power (optimum power to deliver the data packet to the destination node). Next, the node H forwards the route reply packet to node G and updates it power table with the Rx value received from the succcessor node D.

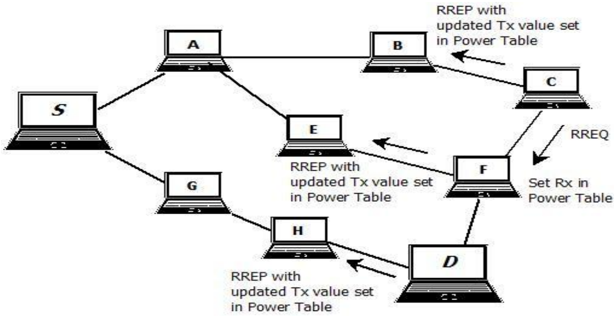

Figure 8: The node D initiating the route reply packet with updated Rx powers to set Tx powers of the nodes.

The final route having been establised between the source node S and the destiantion node D, the source node sends the data packet to its neighbouring node G with transmission power set at the source node S, and the intermediate node on receipt of the pakcet forwads it to its neighbouring node H, which in turm dilivers the packet to the node D, which in the present scenario is esentially the destination (Figure 8). The packet is deliverd to the destination node sucessfully and in timely manner by using the optimum transmission power to transmit the data packet from the source to the destination node—the approach conserving the sufficient amount of energy and increasing the lifespan of the network.

.

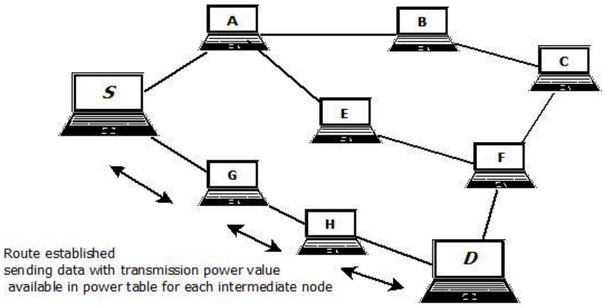

Figure 9: Route establishment between the source and the destination nodes for sending the data packets with the updated optimum power available in the updated Power Table

6.1 ATP-AODV Route Discovery Process and Power level setup

When the source node S has some data packet to be delivered to the destination node D and in the event of the route being unavailable with the source node S, then the route discovery process is initiated by broadcasting the route request packet (RREQ) to its neighboring nodes with a low level of power and a timer is set for a reply. The packet contains the IP address of the source node S, current sequence number, IP address of the destination, and last known sequence number. The RREQ packet also contains the broadcast ID, which will be incremented each time the source node initiates the RREQ packet. The low-level power having been set a priory at a default value such that the packet could reach a pre-assigned distance, typically 200 m, the RREQ packet sent by the source node S reaches its next hop neighbors at the nodes A and G. The neighboring nodes A and G accept or discard the RREQ packet depending on its identification based on the source IP address and broadcast ID. Following, this identification and taking due care to record its information, the node processes the RREQ by setting up, a reverse route entry for the source node in its routing table. This reverse route entry the IP address and sequence number of the source node as well as the number of hops to the source node as well as the IP address of the neighbor from which the RREQ was received. In this way, the node also knows how to forward the RREP packet to the source if the packet is received later and forward the packet accordingly to the source. Further, if a node determines that it has enough route current to respond to the RREQ packet, it creates the RREP packet. If the node A and G do not have enough route current then these nodes simply forward the RREQ packet to their neighboring nodes. The same process goes on until the RREQ reaches the destination node D, while the latter replies back to the source node S by copying the source route from the RREQ packet into the route reply packet (RREP) (Figure 9).

Further, in ATP-AODV protocol the structure of the RREP of AODV protocol has already been modified for carrying the power levels with the RREP packets. While sending the RREP packet to the next hop H, the destination node D copies the power level from the RREQ to the RREP packet and sends the RREP packet to the next hop at that low power level. When the next hop H receives the route reply packet at power level $P_{(\text{receive})}$, it reads the power level information to the route reply packet, which is actually the transmitting power of the $P_{(\text{transmission})}$ of the destination node D, then node H estimate the required minimum transmitting power $P_{(\min)}$ which will be enough to transmit or deliver the data packet to the destination node given by [eq. 1]:

$$P_{(\min)} = P_{(\text{transmission})} - P_{(\text{receive})} + P_{(\text{threshold})} \quad (1)$$

Where $P_{(\text{threshold})}$ is the threshold value. In (1) for LAN 802.11 one can take $P_{(\text{threshold})} = 3.652 \times 10^{-10}$ W.

In our simulation we have taken a margin value of power $P_{(\text{mar})}$ to find the optimum transmission power $P_{(\text{optimum})}$ as follows [eq. 2]:

$$P_{(\text{optimum})} = P_{(\text{transmission})} - P_{(\text{receive})} + P_{(\text{threshold})} + P_{(\text{mar})} \quad . \quad (2)$$

We have set here a margin value of $P_{(\text{mar})} = 2$ dB in (2).

Now, the node H calculates $P_{(\text{transmission})}$ and records the same in its power table. The power table is maintained by all the participating mobile nodes in the networks for recording and retrieving the next hop's IDs and $P_{(\text{optimum})}$ at the time of sending the data packet to the next hop.

All the participating intermediate nodes between the destination and source node follow the same mechanism to set $P_{(\text{transmission})}$ of a particular node to forward the data packet to the next hop. For example, now the intermediate node H will store the node ID of the destination node D and $P_{(\text{optimum})}$ to reach that node in the power table. The node

H then forwards the reply of its previous hop which is G. While forwarding, the node H also uses the low power level (corresponding to a typical default distance range typically 200 m of the reach of the packet). After receiving the route reply packet, the node G also records the node ID of the node H and then re-estimates $P_{(\text{optimum})}$ required to transmit the data packet form the node G to H and stores the information in the power table of the node G. This process goes on until the route reply packet reaches the source node S.

After receiving the route reply packet, the source node S starts sending the data packets to the destination node by using the routes which have been just discovered.

Now, each and every participating mobile node of the networks knows the node IDs of its next hop and the optimum transmitting power required to reach the next hop. The source node S checks the next hop ID (that is, ID of the node G) from the source route stored in the data packet and then retrieves the required optimum power level from its power table and then transmits the data packet.

During the route discovery process and estimating the optimum transmitting power, it is likely that the desired node is far away and that it is not possible for a node to transmit the packet to the desired node. In that case and in the situation of unsuccessful route discovery attempts, the source node increases its power level to the higher power level or to the common power level (corresponding to typically 250 m transmission range) for the route request packet to reach its neighboring node and discover the route to the destination node.

The whole process of maintaining the route request packet and power table is similar to that already described earlier in this section. The only difference is that the common power level has been set, in case of the route to the neighboring node being not discovered in a single attempt. When the route request packet reaches the destination node, the latter sends the route reply packet to the source node S—a process which is similar to that in the low power level.

The advantage of the two-power level used in ATP-AODV protocol is that the network connectivity is maintained and that the failure attempt to discover the route becomes very less, which also helps to increase the overall lifetime of the participating mobile nodes and also increase the average life time of the networks as a whole. Further here only two power levels are used to discover the route compared unlike in [15] where six-to-seven power levels have been used. Furthermore, since the power level information is recorded in the power table, the size of the packet is less compared to that in [16]. Moreover, unlike in [15] each data packet contains the information of the power level of each hop in addition to the source route here in ATP-AODV protocol, there is only one field added to store the information of power level. Also, here the power table is maintained locally at every node. Since the number of attempts at the power level has been set to one, compared to three that would call for a minimum of three trials to send the route request packet to the next hop nodes at every unsuccessful attempt. Thus, in ATP-AODV protocol unnecessary power consumption for attempting at least three times for RREQ packet can be done away with. This further helps to reduce the overheads in ATP-AODV and consumes less energy compare to that in [4].

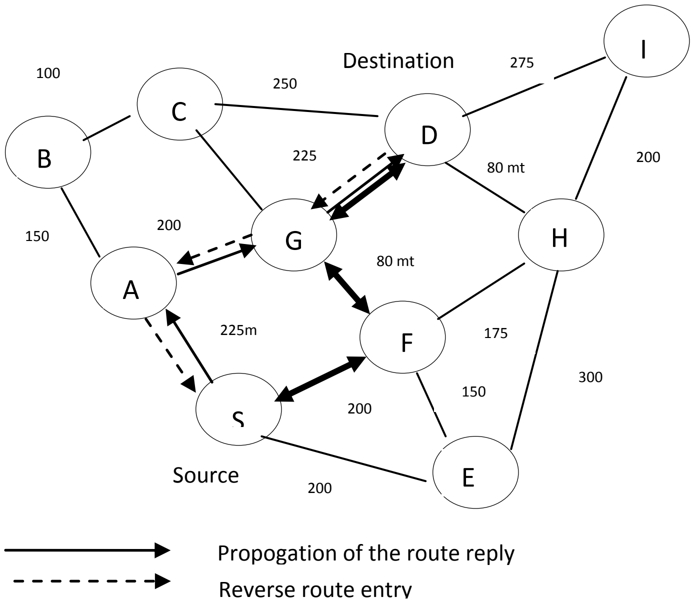

Figure 10: Propogation of the route reply in conventional AODV and ATP-AODV protocols.

In the Figure 10, the source node S discovers a route to the destination node D, the solid line showing the route S-A-G-D, which is discovered by the conventional AODV protocol. If the source node S sends 100 packets and if each data packet has a typical size of 512 bytes the total power and hence energy consumption can be calculated with the help of (1). Thus, the total energy consumed while transmitting the data packet has been found equal to 12.42 J. Further, the bold line in Figure 10 with reference to ATP-AODV shows an alternative route discovered by using S-F-G-D at low and as well as high level of power. The total energy consumption in this case can be calculated with the help of (2) and the total energy consolation has been thus found equal to 10.11 J. Thus, as much as around 18% of energy has been saved in ATP-AODV protocol compared to that in its AODV counterpart. Further, if the intermediate node were not present the ATP-AODV protocol would discover the route using a higher level of power.

7. PERFORMANCE ANALYSIS OF ATP-AODV PROTOCOL

The performance of ATP-AODV protocol has been tested by using NS-2 and for this purpose 30 pairs of UDP connections are set-up randomly in the network. Constant Bit Rate (CBR) traffic is used for generating data packet rate of 1 packet / second. Each CBR traffic starts at random time during the simulation period of the network.

Here in our simulation, we have chosen an approach for analyzing the performance of ATP-AODV protocol.

In this scenario we have assigned a large amount of initial energy while the simulation for the model is run for a limited period of time to know the energy consumption by a data packet in traveling from the source to the destination node. Such a scheme of assigning the energy ensures that no participating mobile node exhausts its battery during the simulation time. This mechanism gives us an estimate about how much energy is consumed by the data packet in traveling from the source node to the destination node.

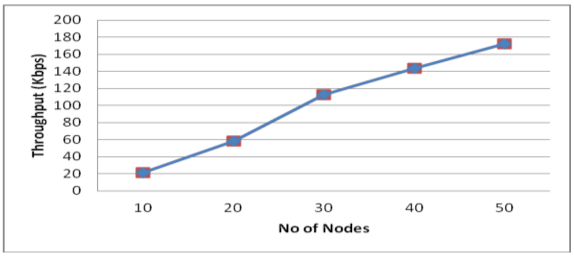

Figure 11: Throughput versus number of nodes in AODV protocol.

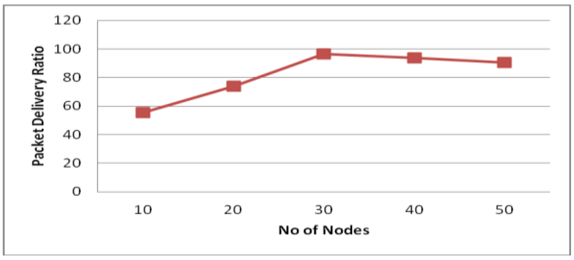

Figure12: Packet delivery ratio versus number of nodes in AODV protocol.

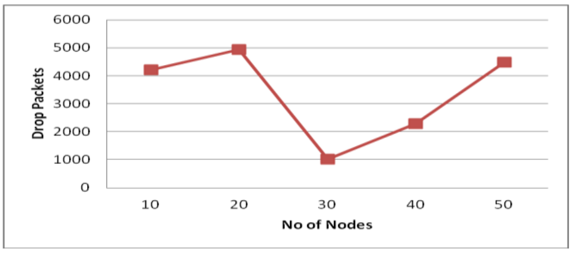

Figure 13: Drop packet versus number of nodes in AODV protocol.

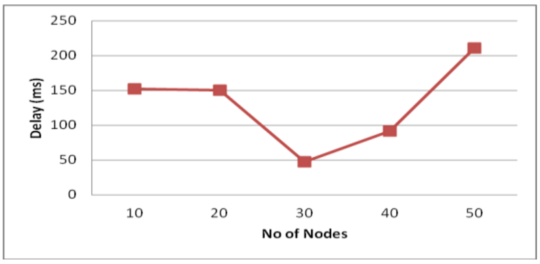

Figure 14: Delay versus number of nodes in AODV protocol.

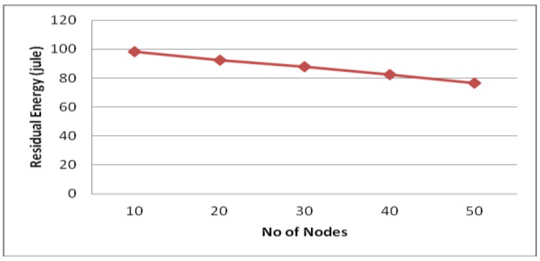

Figure 15: Residual energy versus number of nodes in AODV protocol.

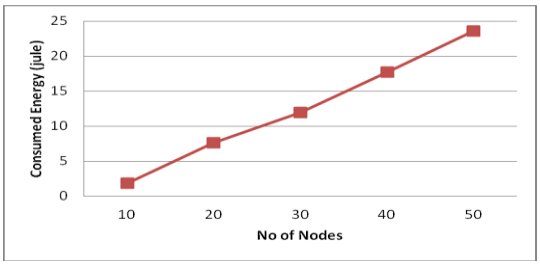

Figure 16: Consumed energy versus number of nodes in AODV protocol.

Prior to analyzing the performance of ATP-AODV protocol, we have analyzed the performance of AODV and concluded that with 30 mobile nodes in the networks AODV has given the best performance with respect to throughput (Figure 11), packet delivery ratio (Figure 12), drop packet (Figure 13) and delay (Figure 14) while average performance with respect to residual energy (Figure 15) and consumed energy (Figure 16).

8. ENERGY CONSUMED AND SAVING IN ATP-AODV PROTOCOL

In order to study the effect of the area of the network on energy consumed per data packet and residual energy or energy saved, we have created a scenario for analysis and placed 30 nodes in a defined area that is varied in discrete steps while keeping constant both the number of participating mobile nodes in the network and the number of connections in the network (Figures 17 and 18).

Ten different topologies are generated for a given network size and the result shows the average of the simulation.

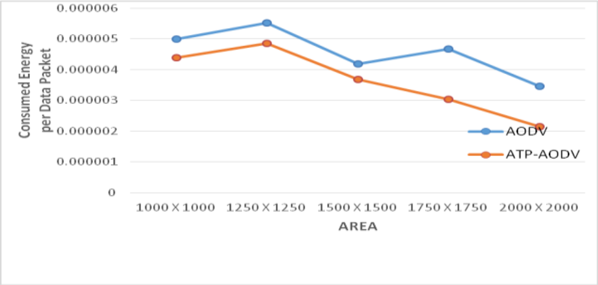

Figure 17: Energy consumed per data packet for AODV and ATP-AODV protocols versus area of the network

The energy consumption per data packet is found to increase with the network size in both ATP-AODV and AODV protocols. This is attributable to the data packet traveling through more number of hops in both the protocols when we place the same number of mobile nodes in a network of a larger size (Figure 17). However, irrespective of the network size, the consumption of energy per data packet is less in ATP-AODV than in AODV protocol (Figure 17).

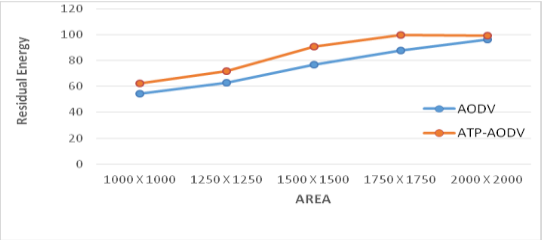

Figure 18: Percentage of energy saving versus area of the network in ATP-AODV compared to that in AODV protocol

The residual energy, which is a measure of energy saving as well, is found to be 18% more in ATP-AODV than in AODV protocol, which increases with the area of the network in both these protocols up to some extent (Figure 18).

9. CONCLUSION

Based on our simulation study we have proposed an adaptive power technique namely ATP-AODV routing protocol to transfer the data packets from the source node to the destination node as an alternative to the conventional routing protocols for MANET that use common power, maximum power or fixed power and which are not energy efficient. The performance analysis shows that by using ATP-AODV protocol a significant amount of energy can be saved and the lifespan of the networks as whole can be enhanced significantly. The study has proved the superiority of ATP-AODV over AODV protocol from the standpoint of energy consumption and saving, the performance greatly depending on the size of the network. In order to alleviate the problem of delay per data packet of ATP-AODV as compared to AODV protocol in densely populated networks— resulting from larger number of hops—it is recommended that ATP-AODV be modified by way of selecting the route along the less congested area of networks. The modification can be further extended to encompass high-speed and less populated networks as well. Furthermore, it would be a challenge in future to make ATP-AODV protocol more energy efficient for highly dynamic networks.

10. REFERENCES

[1] Imrich Chlamtac, Marco Conti, Jennifer J.-N. Liu, Mobile ad hoc networking: imperatives and challenges, *Ad Hoc Networks*, Vol. 1, No. 1, 2003, pp. 13-64.

[2] M. Bouallegu, K. Raoof, M. Ben Zid, and R. Bouallegue, “.Impact of variable transmission power on routing protocols in wireless sensor networks Wireless Communications,” *10th International Conference on Networking and Mobile Computing (WiCOM 2014)*, 26-28 Sept. 2014, pp. 496 – 499.

[3] Mohammed Tarique and Roman Islam, “Minimum Energy Dynamic Source Routing Protocols for Mobile Ad hoc Networks” International Journal of Computer Science and Network Security Vol. 7, No. 11 , 2007, pp. 304-311.

[4] S. Narayanaswamy, V. Kawadia, R. S. Sreenivas, and P. R. Kumar, “The COMPOW protocol for power control in ad hoc networks: theory, architecture, algorithm, Implementation and experimentation,” http://citeseerx.ist.psu.edu, pp. 1-20.

[5] P. S. Hiremath and S. Joshi, “Fuzzy Adaptive Transmission Range based Power Aware Location Aided Routing,” ICICT-2014, Shivamogga, pp. 296-301.

[6] Mohammed Tarique and K. E. Tape, “Minimum Energy Hierarchical Dynamic Source Routing for Mobile Ad Hoc Networks,” *Ad hoc Networks*, Vol. 7, No. 6, 2009, pp. 1125-1135.

[7] Charles E Perkins, Elizabeth M Royer, The Ad Hoc On-Demand Distance-Vector Protocol, Ad hoc Networking, Addison Wesley, 2nd Edition, March 2004, pp. 173-219.

[8] C. Siva Ram Murthy and B. S. Manoj, Routing Protocols for Ad Hoc Wireless Networks, Ad Hoc Wireless Networks – Architectures and Protocols, Prentice Hall Communications Engineering and Emerging Technologies Series, Prentice Hall, 2nd Edition, Pearson Education Inc. 2004, pp. 299-359

[9] The ns Manual (formerly ns Notes and Documentation), The VINT Project A Collaboration between researchers at UC Berkeley, LBL, USC/ISI, and Xerox PARC, November 5, 2011.

[10] Francisco J. Ros Pedro M. Ruiz, A manual on Implementing a New Manet Unicast Routing Protocol in NS2, Dept. of Information and Communications Engineering University of Murcia December, 2004

[11] J. Broch, D. A. Maltz, D. B. Johnson, Y.-C. Hu, and J. Jetcheva, A performance comparison of multi-hop wireless ad hoc network routing protocols, in Proceedings of the Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking, ACM, October 1998.

[12] L. Breslau, D. Estrin, K. Fall, S. Floyd, J. Heidemann, A. Helmy, P. Huang, S. McCanne, K. Varadhan, Y. Xu, H. Yu, "Advances in Network Simulation (ns)", IEEE Computer, vol. 33, No. 5, May 2000, pp. 59-67.

[13] P. Johansson, T. Larsson, N. Hedman, B. Mielczarek, and M. Degermark, Scenario-based performance analysis of routing protocols for mobile ad-hoc networks, in International Conference on Mobile Computing and Networking (MobiCom'99), 1999, pp. 195-206.

[14] C. Bettstetter, H. Hartenstein, and X. Perez-Costa, Stochastic Properties of the Random Waypoint Mobility Model, in ACM/Kluwer Wireless Networks, Special Issue on Modeling and Analysis of Mobile Networks, vol. 10, no. 5, Sept 2004.

[15] S. Narayan, V. Kavadia R. Srinivas, and P Kumar,”Power Control in Ad hoc Networks: Theory, Architecture, Algorithm and Implementation of COMPOW algorithm”, in proceeding of European wireless conference: Next Generation Wireless Technologies, Protocols, Services and Applications, February 2002, pp. 156-162.

[16] S. Dosi, S. Bhandare and T. X. Brown, “An On-demand Minimum Energy Routing Protocol for wireless ad hoc Networks”, ACM SIGMOBILE, Mobile Computing and Communication Review, vol. 2, issue 2, pp. 50-66.