STABLE CLUSTERING ON AODV WITH SLEEP MODE

Prerna Malhotra, Student of M.Tech
Ajay Dureja, Assistant Prof.
M.D.U Rohtak, PDM College of Engineering for Women, B'Garh, India

Abstract

Clustering has evolved as an imperative research domain that enhances system performance such as throughput and delay in Mobile Ad hoc Networks (MANETs) in the presence of both mobility and a large number of mobile terminals. In this thesis, we present a clustering scheme that minimizes message overhead and congestion for cluster formation and maintenance. The algorithm is devised to be dependent on Ad-hoc On Demand Distance Vector (AODV) Routing with sleep mode algorithm of MANET. The dynamic formation of clusters helps reduce data packet overhead, node complexity and power consumption. The goal of this algorithm is to decrease the number of cluster forming, maintain stable clustering structure and maximize lifespan of mobile nodes in the system. Nodes in MANET networks are basically battery operated, and thus have access to a limited amount of energy. This process proposes an Energy based Ad-Hoc on-Demand Routing algorithm that balances energy among nodes so that a minimum energy level is maintained among nodes and the lifetime of network is increased. The simulation has been performed in ns-2. The simulation shows that the number of clusters formed is in proportion with the number of nodes in MANET.

Keywords: AODV, Clustering, MANET, Mobility, Routing, Security, Sleep Mode

Introduction

Ad-hoc on demand distance vector with sleep mode

In MANET, each mobile node operate as both a router and a terminal nodes which is a source or destination, thus the failure of some nodes operation can greatly hinder the performance of the network and also affect the basic accessibility of the network, i.e., energy exhaustion of nodes has been one of the main harm to the connectivity of MANET. Since the mobile nodes in MANET have limited battery power, so it is required to efficiently use energy of every node in MANET. MANET is a multi-hop, in which node can freely move in any direction and have limited battery power.

A reliable routing protocol for Mobile Ad hoc Networks (MANETs) keeps the energy consumption as low as possible.

AODV[Perkins] is based on both DSDV and DSR algorithm. It uses the route discovery and route maintenance practice of DSR. DSR packet carries the complete route information, while the packet of AODV only carries the destination address, it has less routing overhead than DSR. At the same time, AODV makes use of routing messages and sequence numbering. Here AODV is evaluated and analyzed from the aspect the energy utilization metric. AODV protocol is a reactive routing protocol which finds route to destination when required. AODV consists of routing table which helps to differentiate between expiry and fresh routes. The routing table at node contains the sequence number and next hop information. The working of protocol is consists of two phases:

- Route discovery and
- Route maintenance.

In route discovery process, the source node generate RREQ packet, if the path to destination is not stored in the routing table, and pass it to the neighboring nodes. The neighboring nodes will pass it to their neighbor and so on. When the packet reached to the destination node, then destination node generates RREP (Route Reply) packet and send it back to the source node. Thus the path is established between source and destination node. In route maintenance process, the source node is being informed by RERR (Route Error) message in case of link breakage. Also the connectivity between the nodes is maintaining using *Hello* messages.

There are two main factors that cause link failures are:

- Battery life time
- Mobility

Taking energy in consideration, there are three approaches that are usually used to achieve the energy-efficiency in MANETs[J.Li]: Power-Control, Power-Save and Maximum-Lifetime routing. The Power-Control approach is allowing nodes to decide the least amount of transmission power level which is sufficient to maintain network connectivity and to pass the traffic with least energy, the objectives is to increase network capacity and sinking energy consumption. The Power-Save approach deals with the energy loss during the idle mode and this can be minimized by increasing the amount of time a node spends in the sleep mode. Lastly, the Maximum-Lifetime routing approach looks for the nodes that have minimum energy so that they can be avoided from the path.

Stable Clustering

A MANET is a multi-hop wireless network in which mobile nodes can freely move around in the network, leave the network and join the network. These mobile hosts communicate with each other without the support of any pre existing communication infrastructure. Typically, if two nodes are not within mutual transmission range, they communicate through intermediate nodes relaying their messages. In other words, the communication infrastructure is provided by the nodes themselves. Through the nature of MANET, we have many challenges. The most important challenges are stability, routing and scalability. Clustering is the most way to improve the stability, routing and scalability. Have knowledge about the changes of node's status, can present useful information about the stability of it between its neighbours. This information is effective in clustering approach and in cluster head selection.

Clustering algorithms can be performed dynamically to adapt to node mobility[McDonald]. MANET is dynamically organized into groups called clusters to maintain a relatively stable effective topology[C.R. Lin]. By organizing nodes into clusters, topology information can be aggregated. This is because the number of nodes of a cluster is smaller than the number of nodes of the entire network. Each node only stores fraction of the total network routing information.

Therefore, the number of routing entries and the exchanges of routing information between nodes are reduced[Dube]. Apart from making large networks seem smaller, clustering in MANETs also makes dynamic topology appear less dynamic by considering cluster stability when they form[McDonald]. Based on this criterion, all cluster members that move in a similar pattern remain in the same cluster throughout the entire communication session. By doing this, the topology within a cluster is less dynamic. Hence, the corresponding network state information is less variable. This minimizes link breakage and packet loss.

Clustering is usually performed in two phases: clustering set-up and clustering maintenance. In the clustering set-up phase, cluster heads are chosen among the nodes in the network. The roles of cluster heads are coordinators of the clustering process and relaying routers in data packet delivery. After electing cluster heads, other nodes affiliate with its neighbor clusterhead to form clusters. Nodes which are not cluster heads are called ordinary nodes. After the initial cluster set up, reaffiliations among clusterheads and ordinary nodes are triggered by node movements, resulting reconfiguration of clusters. This leads to the second phase, the clustering maintenance. To avoid excessive computation in the cluster maintenance, current cluster structure should be preserved as much as possible. However, any clusterhead should be able to change its role to an ordinary node to avoid excessive power drainage. In this way, the overall lifespan of the system can be extended.

Problem Statement

As MANETs have no fixed infrastructure, all messages have to be routed through the nodes in the network. Many clustering and routing algorithms have been developed for MANETs. Moreover, most of the existing routing algorithms do not utilize the efficiency that can be obtained by clustering a network. Clustering process involves in grouping network nodes and helps in reducing the overhead messages that help in establishing routes. Furthermore, there is a trade-off between providing security and conserving the power of a node. In current approaches, clustering and routing algorithms are designed specifically for either providing security or conserving power. It is very difficult to improve both security and minimize power consumption, as typically one is achieved at the expense of the other. In this thesis, we propose Innovative Stable Clustering approach for Ad-hoc On Demand Distance Vector routing with sleep mode to decrease the number of cluster forming, maintain stable clustering structure and maximize lifespan of mobile nodes in the system. In this, I have presented a new clustering algorithm for Ad-hoc On Demand Distance Vector routing with sleep mode in Mobile Ad Hoc Network based on nodes weight. For calculating node weight we present four new parameter, relative speed, stability, number of nodes moving towards a node and battery remaining.

Related Work

Ad-hoc on demand distance vector with sleep mode

Several routing algorithms use the link lifetime as well as the nodes' battery life time as routing metrics to allow the most reliable and energy efficient route to be selected for data transmission.

Signal Stability based Adaptive Routing has been proposed [Dube], In which the links are categorized into groups according to the signal strength metric. During the route-discovery phase, each mobile node divides the connections between itself and its neighboring nodes into two groups, a strongly connected (SC) group and a weakly connected (WC) group.

Minimum Battery Cost Routing (MBCR) has been proposed in [Singh]. MBCR routing protocol calculates the sum of the residual power of all nodes in a path and uses it for selecting a path, but the method may choose a path in which there may exist mobile nodes with low power. Thus, these low power mobile nodes may cause path breakage.

Fei Dai et al [Fei Dai]: The basic idea of this paper was to save power by using efficient broadcasting techniques that are achieved to conduct broadcasting using directional antennas for ad hoc networks. This paper firstly focuses on energy consumption as well as forwarding packets directionally by introducing directional antennas. Syropoulos et al [Kranakis], have implemented the use of Directional Antennas for energy efficient communication in ad hoc networks.Jin-Man Kim et al., [Kim] introduced an Energy Mean Value algorithm to enhance AODV routing protocol and to improve the network lifetime of MANET.

Krishna Cheong Lau and Joseph H. Kang [Balachandran]: The idea to increase energy efficiency, nodes in the network goes into a sleep mode and wake up at predetermined time slot(s) to snoop for transmissions from its instant neighbors. The knowledge of awakening slots for neighboring nodes is used to arrange the transmissions within the neighbourhood. Finally, nodes adapt their sleeping cycles based on neighbor topology and remaining battery life in order to maximize the network lifetime also satisfying the latency requirements of sensor applications.

In [Chipara] authors have proposed a protocol named RPAR, in which design was based on considering the substitution between energy efficiency and latency. The participating nodes required to uphold an information table related to its neighbors. Attainment and maintenance of such information requires significant exchange of information through beacon signals which involve lot of energy consumption hence energy efficiency is surrendered.

The protocol (ORTR) [Zori] has the limitation that the forwarding area of the nodes is not a shared carrier sensing area. That is why the nodes cannot listen to each other once one of them is broadcasting.

Stable clustering

A large number of clustering algorithm have been proposed according to certain environment and characteristic of mobile node in mobile ad hoc network to choose clusterhead. We will give each of them a brief description as follows:

1) **Highest degree clustering algorithm [Agarwal]** uses the degree of a node as a metric for the selection of clusterheads. The node with highest degree among its neighbors will be elected as clusterhead, and its neighbors will be cluster members. In this scheme, as the number of ordinary nodes in a cluster is increased, the throughput drops and system performance degrades.
2) **The Lowest-Identifier algorithm (LID) [Agarwal]** chooses the node with the minimum identifier (ID) as a clusterhead. The system performance is better than Highest-Degree heuristic in terms of throughput [C.R. Lin]. However, since this heuristic is biased to choose nodes with smaller IDs as clusterheads, those nodes with smaller IDs suffer from the battery drainage, resulting short lifetime span of the system.

3) **Least Movement Clustering Algorithm [Agarwal].** In this algorithm, each node is assigned a weight according to its mobility. The fastest the node moves, the lowest the weight is. And the node with highest weight will be elect as clusterhead.
4) **The Distributed Clustering Algorithm (DCA) [Agarwal] and Distributed Mobility Adaptive clustering algorithm (DMAC) [Agarwal]** are enhanced versions of LID; each node has a unique weight instead of just the nodes ID, these weights are used for the selection of clusterheads. A node is chosen to be a clusterhead if its weight is higher than any of its neighbors weight; otherwise, it joins a neighboring clusterhead. The DCA makes an assumption that the network topology does not change during the execution of the algorithm. Thus, it is proven to be useful for static networks when the nodes either do not move or move very slowly. The DMAC algorithm, on the other hand, adapts itself to the network topology changes and therefore can be used for any mobile networks. However, the assignment of weights has not been discussed in the both algorithms and there are no optimizations on the system parameters such as throughput and power control.
5) **MOBIC [Agarwal]** uses a new mobility metric Instead of static weights; Aggregate Local Mobility (ALM) to elect clusterhead. ALM is computed as the ratio of received power levels of successive transmissions (periodic Hello messages) between a pair of nodes, which means the relative mobility between neighboring nodes.
6) **The Weighted Clustering Algorithm (WCA) [Agarwal]** is based on the use of a combined weight metric that takes into account several parameters like the node-degree, distances with all its neighbors, node speed and the time spent as a clusterhead. Although WCA has proved better performance than all the previous algorithms, it lacks a drawback in knowing the weights of all the nodes before starting the clustering process and in draining the clusterheads rapidly.As a result, the overhead induced by WCA is very high.

Proposed Work

Ad-hoc on demand distance vector with sleep mode

Due to the transition time for powering on an element (wake-up time), there is a penalty to each off-period. The wake-up time is a time period during which power is consumed by the element but no service is delivered. The wake-up time defines the lower bound to the length of the off-period for an element. Due to the wake-up time, off periods should be kept as large as possible in order to reduce the fraction of time spent in wake-up. The upper bound to the off-periods is set by quality of service requirement. The power of a wireless node is very important factor due to limited energy sources our proposed work is based on the energy saving of a node. Each node in wireless network works as a router and participate in the routing mechanism, the energy of moving node are limited. When the energy of node lost it disappears from network it affects the routing mechanism of other node also result routing overhead for searching the alternate routes. The searching of alternate route can be made on the basis of the Residual energy of the neighbouring node:

1. If the Energy of neighbouring node is greater than 50 then update the cache of the node and then transfer the traffic through that node.
2. Otherwise, search for the node with maximum Residual Energy and update the cache then transfer the traffic through that node.

A proposed solution utilizes the node energy when it reaches the minimum energy level called ***minimum threshold***. When the node reaches at minimum threshold l it goes to sleep mode. And also we will add a extra routing table. The basic idea about that table is that a source node, which needs to find a route to a destination, remembers where the destination was last "seen" and localizes its route discovery query to within a radius of that previous location.

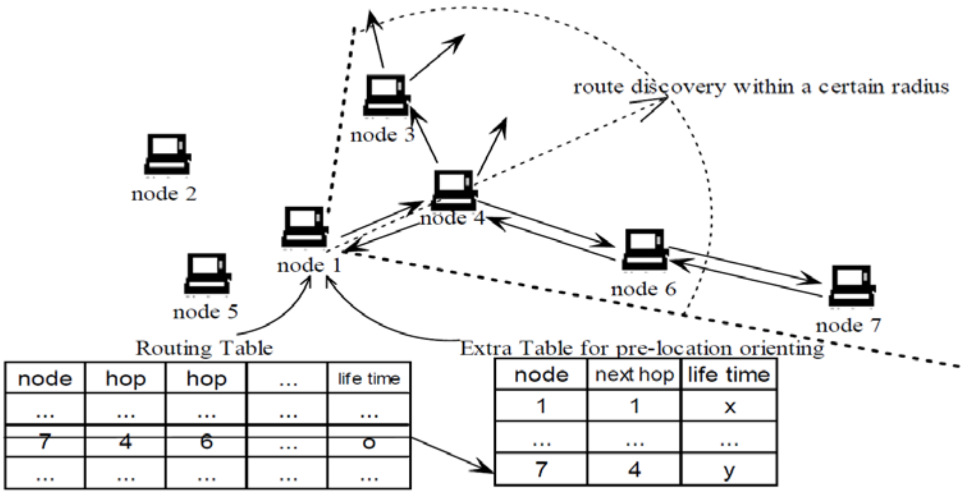

Figure 1: Pre-location Oriented Routing

Stable clustering

Setup Procedure

First, we allocate IDs for the nodes. In this, each node Ni (member or clusterhead) is identified by a state such as: Ni (idnode , idCH , flag , Weightp), it also has to maintain a “node_table” wherein the information of the local members is stored. However, the clusterheads maintain another clusterhead information table “CH_table” wherein the information about the other clusterheads and member node is stored.

In complex networks, the nodes must coordinate between each other to update their tables. The Hello messages are used to complete this role. A Hello contains the state of the node; it is periodically exchanged either between clusterheads or between each clusterhead and its members in order to update the "CH_tables" and the "node_tables" respectively.

We define a flag for every node which determine their role. The value of flag is 3 if the node is the clusterhead, is 1 if the node is an ordinary node, is 2 if the node is a gateway and is zero if the node has an undetermined status.

Weighted Function

To enhance stability of clusters we must find out problems that cause stability to be decreased and as a result cause a cluster to disappear. If we know and solve these problems, we can enhance stability of the clusters as much as possible. The first parameter which causes clusters to disappear is Excessive battery consumption at a clusterhead. In MANETs, the nodes not only bear the responsibility of sending and receiving information, but also carry out routing for packages. As a result they consume a high rate of power.

As a result a clusterhead must have the following conditions:

- It must have a high existence of battery power.
- It must require a lower battery power for interaction with neighbors.

To meet the first condition, the amount of battery power is taken into account as one of the factors for calculation of weight. To meet the second condition, we can choose a node as a clusterhead, which has need less battery power for relation with its neighbors during neighborhood duration (with using RS and S). also these two parameter cause create stable cluster in the network.

The second parameter which causes the clusters be unstable is the mobility of nodes. In this for creating stable clusters we use stability parameter. The used parameters in weighted function for giving a weight to nodes include:

1) Relative speed (RS): The relative mobility of the node with its neighbors. Which means how long node have spent their time beside the node. A lower relative speed simply means that the neighbors of a certain node has spent a longer time in its transmission range, we conclude that the mentioned node has a more stable situation. The relative speed is calculated by Equation (1).

$$RS = \sum_{i=1}^{N} \frac{Sl-Sf}{Tl-Tf} \qquad (1)$$

2) Battery remaining (Br): Every node which wants to be the clusterhead should have threshold power Bd. A clusterhead consumes more energy in a cluster comparing with an ordinary node. In addition, we prefer to choose a more powerful node to play its role as a clusterhead because such a node looses its energy later results in the late starting of new clusterhead selection process and therefore increases the stability of clusters. Equation (2) shows that the each node how to calculate battery remaining of itself.

$$Br = Bd + \frac{Ndm}{N} Bd \qquad (2)$$

3) Number of nodes moving towards a node (Ndm)

4) Stability(S): is defined as the ratio of the time difference of last signal and first signal that received from each neighbors to strength difference of the same signal. This parameter shows the stability of the neighbors of each node. We have to select such a node as the clusterhead which results in suitable stability for the cluster. Calculated by Equation (4).

$$S = \sum_{i=1}^{N} \frac{Tni - Tfi}{Sni - Sfi} \qquad (4)$$

Each node uses the above mentioned four parameters to calculate its weight. The equation (5) shows how the nodes calculate weight.

$$Weight_{\mathrm{p}} = c_1 RS + c_2 B_r + c_3 \mathrm{N}_{dm} + c_4 S \qquad (5)$$

In the equation (5), ci(s) are the weight factors of normalization. The node with highest final weight in its Neighborhood selected as clusterhead.

New Arrival Nodes Mechanism

Once a wireless node is activated, its idCH field is equal to NULL since it does not belong to any cluster. The node continuously monitors the channel until it figures out that there is some activity in its neighborhood. This is due to the ability to receive the signals from other present nodes in the network. The node still has no stable state, thus its state is not full identified. In this case, it broadcasts a Join_Request in order to join the most powerful clusterhead. Thus, it waits either for a CH_Wel or for a CH_NWel. When the entry node does receive neither CH_Wel nor CH_NWel . If this persists for certain number of attempts, the node declares itself as an isolated node, and restarts by broadcasting a new Join_Request after a period of time. We note that just the clusterheads may response by a CH_Wel or CH_NWel ; the ordinary members have to ignore any Join_Request received even if they are in the transmission range of the new entry node. This allows simplifying the management of the clusters.

In the case where the node receives a response (CH_Wel or CH_NWel), it does not take immediately any decision, this allows the node to be certain that it has received all the responses from all the neighboring clusterheads. The CH_Wel and CH_NWel messages do not indicate that the clusterhead has added the node to its table; they just signify that the clusterhead is waiting for a Join_Accept in order to add the node to its table. When the node receives multiple CH_Wels, Based on the primary weight of clusterheads calculate the final weight of them and select the node with highest final weight as the clusterhead. After that, it sends a Join_Accept to the chosen clusterhead and waits for CH_ACK from this CH. The CH_ACK has to contain a confirmation that the idnode has been added to the CH_table. Thus the node can fully-define its state. The reason that we use four ways to confirm the joining procedure is to prevent other clusterheads that they can serve the entry node to add this node to their tables and cause conflicts.

In the case where the node was just receiving CH_NWels, it considers these responses as rejection messages from the clusterheads. This may occur when the clusterheads are saturated and decide to reject the adhesion of new nodes. When the number of attempts reaches a certain value, the node prefers not to stay isolated, thus it declares itself as clusterhead.

Clusterhead Nodes Mechanism

A clusterhead has an idnode field is equal to idCH field. As a clusterhead, the node calculates periodically its weight,thus it sends periodically Hello messages to its members and to the neighboring clusterheads in order to update the node_tables and CH_tables respectively. The clusterhead must monitor the channel for Leave, Hello and Join_Request messages. in the proposed algorithm this operation is limited to clusterhead to allow easier management on clusters.

When the clusterhead receives a Join_Request (idCH=NULL) from a new arrival node or a Join_Request (full state) from a node which belongs to another cluster, the clusterhead can accept or reject the request basing on its capacities. This procedure gives more flexibility to the members by allowing them to leave a weak clusterhead and join another one which seems stronger than the current clusterhead. It may not be possible for all the clusters to reach the cluster size λ. We have tried to reduce the clusters formed by merging the clusters that have not attained their cluster size limit. However, in order not to rapidly drain the clusterhead's power by accepting a lot of new nodes, we define thresholds which allow the clusterhead to control the number of nodes inside its cluster. The re-election is not periodically invoked; it is performed just in case of a higher received weight, it allows minimizing the generated overhead encountered in previous works As we explained above. The re-election may not result a new clusterhead, it depends on the stability of the new node for playing the clusterhead's role.

In this clusterhead will check regularly incoming messages from neighboring nodes. If clusterhead received a message that contains higher primary weight from his weight, then it check the relative mobility with the desired node, if its relative mobility to this node were in the first group and all of the cluster members exist in neighboring of this node, assign clusterhead role to the desired node. The node do it with save the ID of this node in its CH_ID field. Then send a CH_info message to new clusterhead to declare that this node as a new clusterhead selected. Then copy their tables in to new clusterhead and send a CH_change message to neighboring nodes to defines a new clusterhead. in This new approach selecting the new clusterhead is based on stability of it in the cluster. In this case where a new clusterhead is elected, the procedure is soft and flexible in order not perturb the clusters while to copying the databases from the old clusterhead to the new clusterhead.

Member Nodes Mechanism

After joining a cluster, the node declares itself as a member of this cluster. Hence, it calculates periodically its weight and sends periodically Hello messages to its clusterhead. As a member, this node should just handle the Hello, the CH_change and the CH_info messages. This allows optimizing the resources (bandwidth, battery, etc) and minimizing the job of the nodes.

When the node receives a Hello from its clusterhead, the node has to update its node_table. When the node receives Hellos from the neighboring clusterheads, the node has the possibility to migrate to another clusterhead if there is a Hello which has a higher weight than the current clusterheads weight, Member node get this decision by calculating the final weight of the new clusterhead. it sends a Join_Request to the clusterhead which is Hello's source and continues as a member of the current clusterhead until the reception of CH_ACK. In this case, the node can send a Leave_Request to the last clusterhead. This method allows us to minimize the number of the formed clusters in the network.

When the node member receives a CH_info message as a result of the re-election procedure, thus it realizes that it is going to become the new clusterhead in the cluster. When anode member does not receive any message from its clusterhead, it considers that the clusterhead has gone brusquely down; in this case, the nodes have no choice and must restart the clustering setup procedure.

Results and Discussion

Simulation model and parameters

We use NS2 to simulate our proposed work. In our simulation, mobile nodes of sizes 20, 30, 40, 50 and 60 move in an 500×500 m region. Energy model includes the radio range of 250m, 2Mbps of data rate. Initial energy supplied to each node is random. The traffic model used is CBR (Constant Bit Rate) with packet size of 512 bytes, rate 50 packets/s and simulation time of 100s. We assume each node moves independently with the same average speed. Our simulation settings and parameters are summarized in Table 1.

Table 1: Simulation parameters

Parameter	**Value**
Simulator	NS 2.35
MAC Type	Mac/802_11
Routing Protocol	AODV sleep
Channel Type	Channel/WirelessChannel
Antenna Model	Antenna/OmniAntenna
No.of Nodes	20,30,40,50 and 60
Area Size	500*500
Simulation Time	50 secs
Network Interface	Physical/Wirlessphy
Radio Propagation Model	Propagation/TwoRayGround
Interface Queue Type	Queue/DropTail/PriQueue
Max packet in queue	50

Performance Metrics

Performance metrics are used to calculate the overall performance. Here we take 5 parameters while comparing our modified version of AODV with conventional AODV. These parameters are as listed below:

End-to-End Delay: A specific packet is transmitting from source to destination and calculates the difference between send times and received times. Delays due to route discovery, queuing, propagation and transfer time are included in the delay metric.

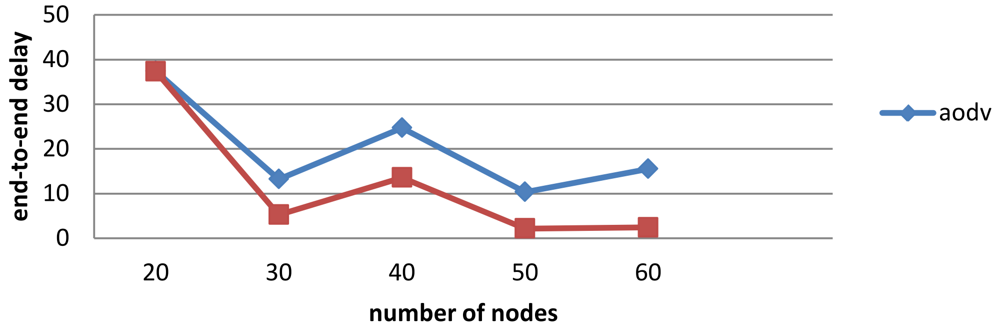

Figure 2: Nodes Vs Delay

Packet Deliver Ratio (PDR): The (PDR) is defined as the ratio between the amount of packets sent by the source and received by the destination.

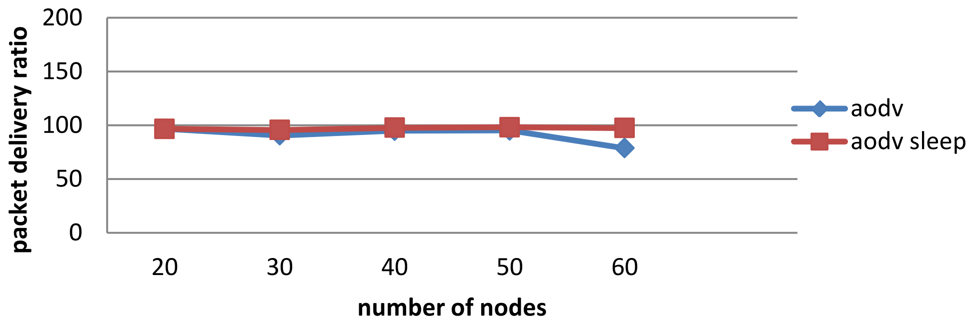

Figure 3: Nodes Vs PDR

Throughput: Throughput is the average rate of successful data packets received at destination. It is usually measured in bits per second (bit/s or bps), and sometimes in data packets per second.

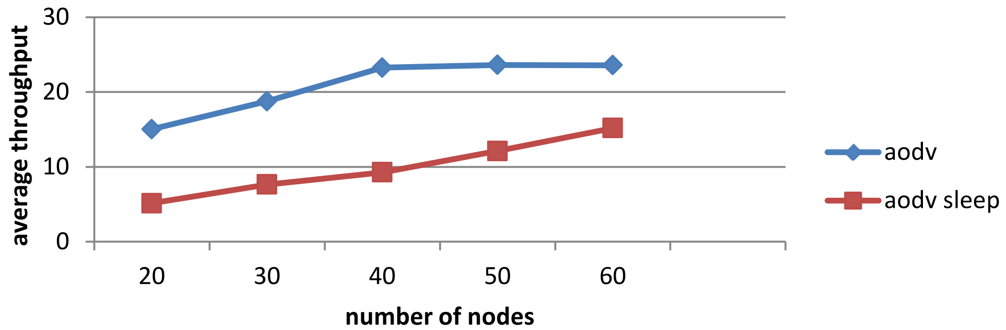

Figure 4: Nodes Vs Throughput

Drop: It is the total number of packets dropped during the data transmission amongst different nodes.

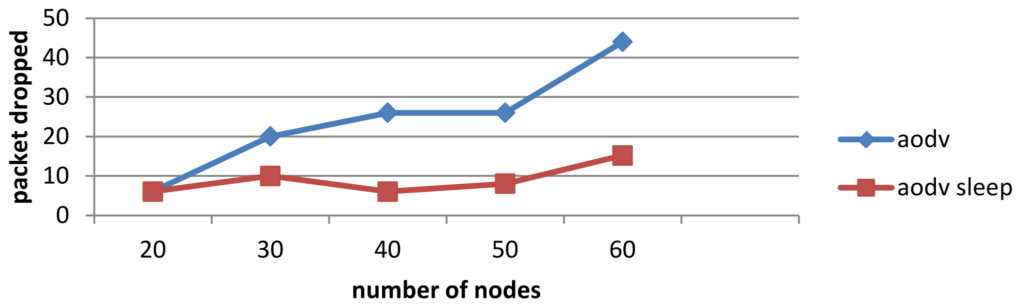

Figure 5: Nodes Vs Drop

Routing Overhead: Routing overhead is the total number of control packets or routing packets generated by routing protocol during simulation.

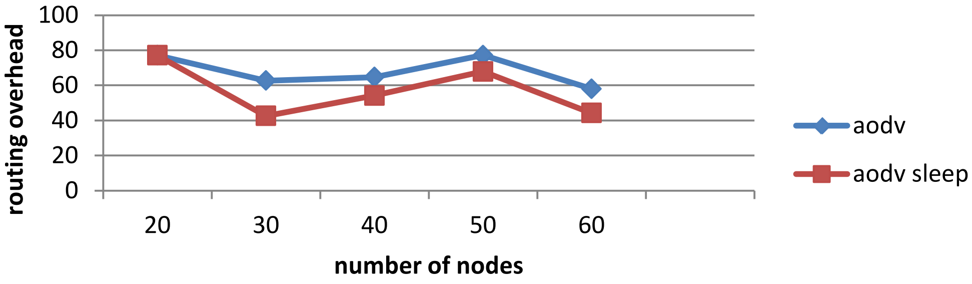

Figure 6: Nodes Vs Routing Overhead

Conclusion and Future Work

The above mention technique proposes a New-AODV protocol which enhances the network lifetime in an Ad-hoc network environment. Above all, each node's energy has a huge impact on the entire network lifetime. The proposed sleep mode scheme ensures significant improvement in power aware system. Hence, the effectiveness of sleep mode for the systems ultimately depends on the wake-up time for various nodes. The initiation of the awareness in the power management is proposed. In order to, overcome the energy based problem and prevent the link breakage. It also has stable clustering structure on the nodes of this protocol.

In the cluster stability selection of a clusterhead is done during two stages. In the first stage, each node calculates its primary weight by using a new presented weighted function. In the second stage, each node calculates its relative mobility in present time and predict relative mobility in future to other neighborhoods. Then based on the primary weight of that specified neighborhood, it assigns a final weight to it. Next, based on the final weight clusterhead is selected. A number of parameters of nodes were taken into consideration for assigning weight to a node. In this thesis we choose the cluster-heads based on the information of neighbour nodes, and maintains clusters locally. Also it has a feature to control battery power consumption by switching the role of a node from a cluster-head to an ordinary node. We assumed a predefined threshold for the number of nodes to be created by a clusterhead, so that it does not degrade the MAC function and to improve the load balancing. We conducted simulation that shows the performance of the proposed enhancement clustering in terms of the average number of clusters formation, stability of clusters, and lifetime of a clusterhead.

Table 2: Result Analysis of Proposed Routing

Protocol	No. of Nodes	End to End Delay	PDR	Throughput	Packet drop	Overhead
AODV Sleep	20,30,40,50,60	Reduced	Increased	Minimum	Low	Reduced

References:

C. Perkins, E. Belding-Royer and S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing," RFC 3561 (Experimental), Jul. 2003. [Online].

J. Li, D. Cordes, and J. Zhang, "Power-aware routing protocols in ad hoc wireless networks," IEEE Trans. Wireless Commun., pp. 69-81, Dec. 2005.

A. B. McDonald and T. F. Znati. "A mobility-based framework for adaptive clustering in wireless ad hoc networks." *IEEE Journal on Selected Areas in Communications*, 17(8):1466-1486, Aug. 1999.

C. R. Lin and M. Gerla. "Adaptive clustering for mobile wireless networks," *IEEE Journal on Selected Areas in Communications*, 15(7):1265-1275, Sept 1997.

Dube, R., Rais, C.D., Wang, K.-Y., Tripathi, S.K., "Signal stability based adaptive routing (SSA) for ad hoc networks". IEEE Personal Communications 4, pp. 36–45, February 1997.

Singh, S., Woo, M., Raghavendra, C.S., "Power-aware routing in mobile ad hoc networks". In: Proc. of 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking, pp. 181–190,1998.

Fei Dai and Jie Wu, Efficient Broadcasting in Ad-Hoc Networks using Directional Antennas, Department of Computer Science and Engineering , Florida Atlantic University, Boca Raton, FL 33431.

Kranakis, Danny Krizanc and Erric Williams Evangelos, Directional versus Omni Directional Antennas for Energy Consumption and k-Connectivity of Network of Sensors, October 15th 2004.

Jin-Man Kim, Jong-Wook Jang, Kamrok Lee, "AODV Based Energy probability as compared to existing AODV routing protocol. Efficient Routing Protocol for Maximum Lifetime in MANET," in Proceedings of AICT/ICIVW IEEE, 2006.

Krishna Balachandran , Wing Cheong Lau and Joseph H. Kang, " Adaptive Sleeping and Awakening Protocol (ASAP) for Energy Efficient Adhoc Sensor Networks", 0-7803-8938-7/05/$20.00 (C) 2005 IEEE.

O. Chipara, Z. He, G. Xing, Q. Chen, X. Wang, C. Lu, J. Stankovic, and T. Abdelzaher. "Real-time power aware routing in sensor networks"Proc. 1WQoS'06, IEEE, pp.83-92, 2006.

M. Zorzi and R. R. Rao. "Geographic random forwarding (GeRaF) for ad hoc and sensor networks: energy and latency performance," IEEE Trans. Mobile Computing, Yol.2, No.4, pp.34.

Agarwal, R., & Mahesh, M. (2009). Survey of Clustering Algorithms for MANET. International Journal on Computer Science and Engineering, Vol.1 (2), 2009, (pp. 98-104).