A Power-Aware Cooperative Caching Technique for MANET

K.Geetha[#1], K.Sabeetha[#2], A.Sangeetha[#3]

[#1]*Information Technology Department*[#2]*Information Technology Department*

[#3]*Information Technology Department*

[#1]*PSNA College of Engineering and Technology,* [#2]*PSNA College of Engineering and Technology ,* [31]*PSNA College of Engineering and Technology*

Dindigul, India

geethakarthik07@gmail.com

sabeethaks@gmail.com

sangi_esraj@yahoo.com

Abstract--- The Caching Paradigm has widely been used in computers, databases and the internet to reduce the response time of the application and to reduce the traffic load in the networks. These advantages can also be applied in a Mobile AD Hoc Networks (MANET) as a caching paradigm could reduce the interferences due to the traffic overload and it could also increase the availability of documents due to server disconnections. Since Mobile Ad-hoc Networks (MANET) is large with sparse nodes, the routing information becomes more complex and it consumes more power, cluster-based routing techniques are used to tackle such situations. The data access efficiency can be improved using cooperative caching technique, which allows the sharing and coordination of cached data among multiple nodes inside the clusters. In this paper, we propose to develop a cluster based cooperative caching technique. The network is divided into clusters and the selection of cluster head is done based on the power level and connectivity. Hence the protocol Power-Aware Cooperative Caching (PACC) is developed to achieve the performance parameters like Success Ratio, delay and overhead.

***Keywords*-- -** Ad Hoc Networks, Clustering, Cooperative Caching, Power Consumption

I. INTRODUCTION

MOBILE AD HOC NETWORK (MANET) is collection of multi-hop wireless mobile nodes that communicate with each other without centralized control or established infrastructure. In MANET [4, 5, 6] each node communicates with other nodes directly or indirectly through intermediate nodes. Thus, all nodes in a MANET basically function as mobile routers participating in some routing protocol required for deciding and maintaining the routes. Routing is one of the key issues in MANETs due to their highly dynamic and distributed nature. The routing protocols of MANETs are divided into two categories as table-driven and on-demand. In table-driven routing protocols, each node attempts to maintain consistent, up-to-date routing information to every other node in the network. Many routing protocols including Destination-Sequenced Distance Vector (DSDV) [9] and Fisheye State Routing (FSR) protocol belong to this category. In on-demand routing, routes are created as and when required. Route discovery and route maintenance are two main procedures: The route discovery process involves sending route-request packets from a source to its neighbor nodes, which then forward the request to their neighbors, and so on. Once the route-request reaches the destination node, it responds by unicasting a route-reply packet back to the source node via the neighbor from which it first received the route-request. When the route-request reaches an intermediate node that has a sufficiently up-to-date route, it stops forwarding and sends a route-reply message back to the source. Once the route is established, the route maintenance process is invoked until the destination becomes inaccessible along the route. Note that each node learns the routing path as time passes not only as a source or an intermediate node but also as an overhearing neighbor node. In contrast to table-driven routing protocols, on-demand routing protocols don't maintain all up-to-date routes. Dynamic Source Routing (DSR) [8] and Ad-Hoc On-Demand Distance Vector (AODV) [10, 11] are popular on-demand routing protocols.

In addition to simply establishing correct and efficient routes between pair of nodes, one important goal of a routing protocol is to maximize the lifetime of ad hoc mobile networks. The residual battery energy of mobile nodes is a simple indication of energy stability and can be used to extend network lifetime. This information has to be taken from the physical and medium access control layers of data link layer since these layers are responsible layers to compute the power consumption and residual energy.

Caching is an important technique to enhance the performance of both wired and wireless network. A number of studies have been conducted to improve the caching performance in wireless mobile environment. Cooperative caching has been studied in the web environment, but little work has been done to efficiently manage the cache in ad hoc networks. Due to mobility and constrained resources (i.e., bandwidth, battery power and computational capacity) in wireless networks, cooperative cache management techniques designed for wired networks may not be applicable to ad hoc networks. In the context of ad hoc networks, it is beneficial to cache frequently accessed data not only to reduce the average query latency but also to save wireless bandwidth.

II. RELATED WORKS

Caching is an important technique to enhance the performance of both wired and wireless network. A lot of researches have been done to improve the caching performance in mobile ad hoc network environment. The two basic types of cache sharing techniques are push based and pull based. In the push based cache sharing, when a node caches a new data item, it will proactively broadcast the caching updates to its neighbor's nodes. The neighboring nodes update the caching information for the future use. Push based scheme improves the data availability at the cost of communication overhead. The disadvantage of the scheme is that an advertisement may become useless if no demand for the cached data items occurs in the vicinity. One more problem with the push based scheme is that caching information may not be longer used if the node moves out from the zone or due to the cache replacement. This drawback may be overcome with the pull based approach. In the pull based cache scheme when a node wants to access a data item, it broadcast a request packet to all its neighbors' node. A nearby node who has cached the data item will send data item to the requester. There are two drawbacks associated with this scheme. First if the requested data item is not cached by any node in the neighborhood then the request originator will wait for the time out interval to expire before it resend the request to the data center. This will cause extra access latency. Secondly if more than one node has cached the requested data item then multiple copies will returns to the request originator and this will cause extra communication overhead.

In CCCDSR [1], Dynamic Source Routing Protocol is taken as a basic protocol and the performance of this protocol is improved using cooperative caching technique, which allows the sharing and coordination of cached data among multiple nodes. To overcome difficulties in routing, mobile nodes are clustered. The lead of this protocol is performance in terms of packet delivery ratio, data rates etc are improved and the delay is low compared to DSR protocol. The shortcoming of this protocol was it consumes more power.

In COACS [3], nodes are used to cache submitted queries. The queries are used as indexes to data cached in nodes that previously requested them. The benefit of this system is its ability to increase the hit ratio rapidly as more requests for data are submitted. The draw back of this system is to reduce the delays at the cost of higher bandwidth consumption.

In CC scheme [4], the network topology is partitioned into non-overlapping clusters based on the physical network proximity. For a local cache miss, each client looks for data item in the cluster. If no client inside the cluster has cached the requested item, the request is forwarded to the next client on the routing path towards server. A cache replacement policy, called Least Utility Value with Migration (LUV-Mi) is developed. The LUV-Mi policy is suitable for cooperation in clustered ad hoc environment because it considers the performance of an entire cluster along with the performance of local client. The cc scheme follows a weak consistency model based on TTL is a major shortcoming of the system.

A novel asymmetric cooperative cache approach [2] is proposed for p2p networks, where the data requests are transmitted to the cache layer on every node, but the data replies are only transmitted to the cache layer at the intermediate nodes that need to cache the data. This solution not only reduces the overhead of copying data between the user space and the kernel space, it also allows data pipelines to reduce the end-to-end delay. This work is done in p2p networks.

Yin and Cao [7, 8] presented three cache resolution schemes: Cache Data, Cache Path, and Hybrid Cache.

In Cache Data, forwarding nodes check the passing-by data requests. If a data item is found to be frequently requested, forwarding nodes cache the data, so that the next request for the same data can be answered by forwarding nodes instead of traveling further to the data server. A problem for this approach is that the data could take a lot of caching space in forwarding nodes.

In Cache Path forwarding nodes cache the path to the closest caching node instead of the data and redirect future requests along the cached path. This scheme saves caching spaces compared to Cache Data, but since the caching node is dynamic, the recorded path could become obsolete and this scheme could introduce extra processing overhead.

In Hybrid Cache, when a mobile node forwards a data item, it caches the data or the path based on some criteria. These criteria include the data item size and the time-to live (TTL) of the item. Due to the mobility of nodes the collected statistics about the popular data may become useless. One another drawback of these schemes is that if the node does not lie on the forwarding path of a request to the data center the caching information of a node cannot be shared.

In AEEERG [16], energy efficiency and reliability is achieved. This protocol assures the increased delivery ratio, better reliability and low energy consumption for gossip based routing. The drawback of this protocol is the delay and overhead is more.

In ABP [17], a distributed clustering algorithm mainly suits relatively static and quasi-static MANETs. In such environments, ABP not only considers both energy and location metrics for cluster formation, but also adjusts the control traffic required for cluster maintenance according to the average measured node mobility.

In [18] a cluster based cooperative caching technique based on mobility prediction to handle mobile disconnections and to reduce the overhead. The network is divided into clusters and the selection of cluster head is done based on the power level and connectivity. The details of the cached data items in the cluster and its adjacent clusters are maintained in the local cache table (LCT) and the Global cache table (GCT). After predicting the movement of its cluster members using Hidden Markov Model (HMM), the cluster head updates the LCT and GCT to maintain consistency thus the reducing the delay and frequent overheads in cache updation

In [19], the proposal is supported by the local cache that all the nodes in the MANET possess. In a collaborative way, a node could respond to the demand of other nodes if it is keeping a valid copy of the required document. As a novelty, the demands of the documents are not restricted to specific application messages but they are codified into the messages generated when the path to the server is being searched.

III. System Architecture

A. PACC (Power-Aware Cooperative Caching Protocol)

PACC is a cluster-based middleware which stays on top of the underlying network stack and provides caching, consumes less power and other data management services to the upper layer applications in MANET's environment.

Middleware layer: It is responsible for service location, group communications shared memory. Middleware layer consists of various blocks such as cache management, information search, Asymmetric caching and clustering.

Cache management: Cache management includes cache admission control, cache consistency maintenance, and cache replacement.

Cache admission control: In this, a node will cache all received data items until its cache space full. After the cache space becomes full, the received data item will not be cached if the data item has a copy within the cluster.

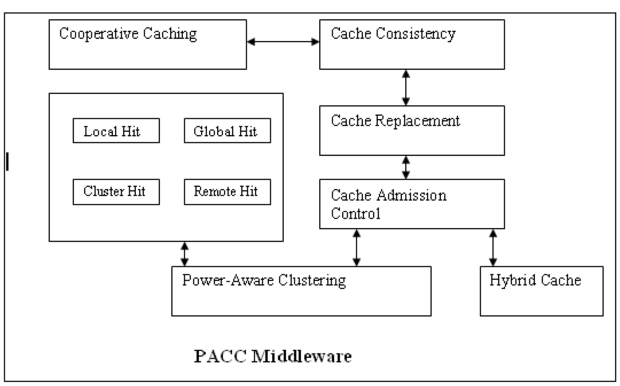

Fig-1: System Architecture for PACC

Cache replacement: When fresh data item is arrived for caching and if cache space is full then the cache replacement algorithm is used to locate one or more cached data items to take out from the cache place. The cache replacement process involves two steps: First, if some of the cached data items become obsolete, these items will be detached to make space for the newly arrived data item. If there is still no enough cache space after all obsolete items are removed, cache replacement will go to the second step, which is that one or more cached data items will be expelled from the cache space according to some criteria. The various cache replacement algorithms used for this mechanism are LRU, LRU-MIN etc.

Cache consistency: The cache consistency strategy keeps the cached data items synchronized with the original data items in the data source.

Information search: Deals with locating and fetching the data item requested by the client.

B. Cluster Formation

Clustering is a method used to partition the network into several virtual groups based on the some predefined method. A distributed algorithm, the Adaptive Broadcast Period (ABP) algorithm is taken for clustering, for efficient and scalable clustering of MANETs .The main contributions of ABP algorithm are: fast completion of clustering procedure, where both location and battery power metrics are taken into account; derived clusters are sufficiently stable, while cluster scale is effectively controlled so as not to grow beyond certain limits; minimization of control traffic volume, especially in relatively static MANET environments.

In the whole network there is no direct connection between the cluster heads. Fig. 2 is an illustration of clustering architecture.

Cluster member is just like a mobile node it does not have any extra functionality. The node which is common to two cluster heads is elected as a gateway. Gateway is used for providing the communication between two cluster heads. Whenever a node requests for the data, first it has to be checked in the cluster head list. Whenever a node requests for the data, first it has to be checked in the cluster head list.

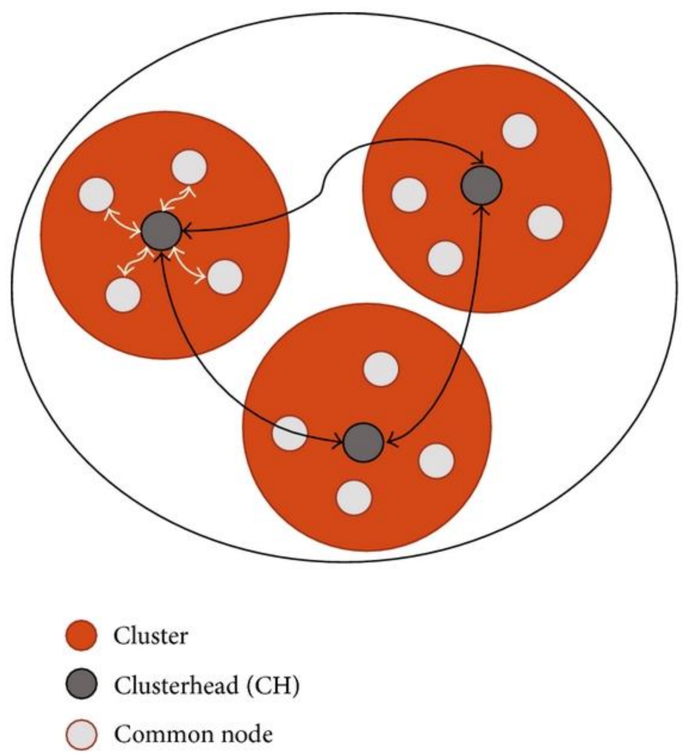

Fig-2: Clustering Architecture

If it is not available in the list of cluster head then the cluster head forwards the requested data item to the other cluster via gateway. ABP clustering algorithm considers both location and power information to partition a MANET into separate clusters. In this context, we introduce the concept of 'Cluster Head Competence' (CHC), which represents the competence of an MH to undertake the role of a CH. The format of a typical 'Hello' message is shown in Figure 3.

MH_ID	CH_ID	CHC	Option	BP
8 bit	8 bit	8 bit	4 bit	8 bit

Figure 3 'Hello' packet format

Each 'Hello' message includes identifications of its sender (MH_ID) and sender's assigned CH (CH_ID). CHC represents a weighted sum of sender's degree (number of neighbors) and its battery power level. Finally, the 'Option' message field is used for cluster size management purposes, and BP field is used to adapt the BP within a particular cluster

CHC values are calculated according to the following equation:

$$CHC = (c_1 \times d + c_2 \times b) - p \quad (3)$$

where:

- c_1, c_2: weighted coefficients of MH degree and battery availability, respectively ($0 \leq c_1, c_2 \leq 1$, $c_1 + c_2 = 1$)
- d: number of neighbors (degree of MH)
- b: remaining battery lifetime (percentage of remaining over full battery power)
- p: 'handover' penalty coefficient

The algorithm's execution involves the following steps:

1 Each MH sends a 'Hello' message randomly during a 'Hello' cycle. If an MH has just joined the MANET, it sets CH_ID value equal to a negative number. That signifies an MH is not a member of any cluster and has no knowledge of whether it is within transmission radius of another MH.

2 Each MH counts how many 'Hello' messages it received during a 'Hello' period, and considers that number as its own degree (d).

3 Each MH broadcasts another 'Hello' message, setting CHC field equal to the value calculated from equation (3).

4 Recording received 'Hello' messages during two 'Hello' cycles, each MH identifies the sender with highest CHC value and thereafter considers it as its CH.

In the next 'Hello' cycle, CH_ID value will be set to elected CH's ID value. In the case of two or more MHs having the same lowest CHC value, the one with the LID is 'elected' as CH.

C. Route Construction

Route Discovery is used whenever a source node desires a route to a destination node. First, the source node looks up its route cache to determine if it already contains a route to the destination. If the source finds a valid route to the destination, it uses this route to send its data packets. If the node does not have a valid route to the destination, it initiates the route discovery process by broadcasting a route request message. The route request message contains the address of the source and the destination, and a unique identification number.

An intermediate node that receives a route request message searches its route cache for a route to the destination. If no route is found, it appends its address to the route record of the message and forwards the message to its neighbors. The message propagates through the network until it reaches either the destination or an intermediate node with a route to the destination. Then a route reply message, containing the proper hop sequence for reaching the destination, is generated and unicast back to the source node.

.

D. Information Search Operation

When a client suffers from a cache miss (called local cache miss), the client will look up the required data item from the cluster members by sending a request to the CSN. Only when the client cannot find the data item in the cluster members' caches (called cluster cache miss), it will request the item from the client that lies on the routing path towards server. If a cluster along the path to the server has the requested data (called remote cache hit), then it can serve the request without forwarding it further towards the server. Otherwise, the request will be satisfied by the server. Fig 4 shows the behavior of PACC caching strategy for a client request. For each request, one of the following four cases holds:

Case 1: Local hit. When copy of the requested data item is stored in the cache of the requester. If the data item is valid, it is retrieved to serve the query and no cooperation is necessary.

Case 2: Cluster hit. When the requested data item is stored by a client within the cluster of the requester. The requester sends a request to the CSN and the CSN returns the address of the client that has cached the data item.

Case 3: Remote hit. When the data is found with a client belonging to a cluster (other than home cluster of the requester) along the routing path to the data source.

Case 4: Global hit. Data item is retrieved from the server.

E. Route Maintenance

Route Maintenance is used to handle route breaks. When a node encounters a fatal transmission problem at its data link layer, it removes the route from its route cache and generates a route error message. The route error message is sent to each node that has sent a packet routed over the broken link. When a node receives a route error message, it removes the hop in error from its route cache.

IV. PERFORMANCE EVALUATION

Our proposed scheme was evaluated using NS2 [13]. The node movement follows the random way point mobility model [14]. We assume that each node generates a sequence of data requests with exponentially distributed time intervals. The simulation was carried out in an area of 800m x 800m, with 40 nodes roaming within that simulation area. The Adhoc On-demand Dynamic source Routing (DSR) routing protocol [09] was used as the basic routing protocol. PACC (Power Aware

Cooperative Caching) Protocol is proposed to do cache management and to reduce overhead, Power consumption and delay. Additional simulation parameters are shown in Table 1.

SIMULATION PARAMETERS

Environment Variables	Associated Values
Simulation Tool	Cygwin+Ns-Allinone-2.28
Bandwidth of the Environment:	1 Mb
Antenna Type	Omni Directional
Base Protocol	DSR
Total No of Nodes	40
Area Size	500 X 500
Mac	802.11
Radio Range	250m
Simulation Time	50 sec
Rate	100,200,300,400 and 500 kb
Mobility Model	Random Way Point
Speed	20m/s
Pause time	10 sec
Transmit Power	0.660 w
Receiving Power	0.395 w
Idle Power	0.035 w
Initial Energy	15.1 J
Pvalue	0.0,0.1,0.2,0.3 and 0.4

A. Performance Metrics

We compare our Power Aware Cooperative caching Technique (PACC)with Cluster Based Cooperative Caching Dynamic Source Routing Protocol (CCCDSR)[1].We evaluate mainly the performance according to the following metrics.

Average Query Latency: The average latency involved in submitting a query from the client and getting the reply.
Success Ratio: It is the ratio of the number of packets received successfully and the total number of queries sent
Control overhead: The control overhead is defined as the total number of control packets normalized by the total number of received data packets.

B. Based on Nodes

In our initial experiment, we vary the number of nodes as 20, 40, 60, 80,100 and 120 with mobile speed as 5m/s.B. Results

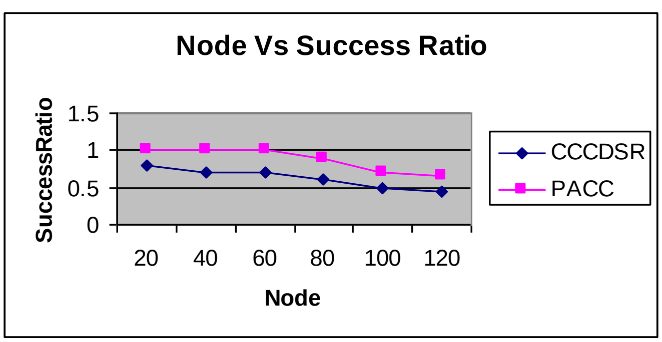

Fig 5 Throughput vs. Flow

Fig 6 shows the results of energy consumption for the number of flows 1,2,3,4. From the results, we can see that PACC scheme has less energy than CCCDSR scheme, since it has the energy efficient routing.

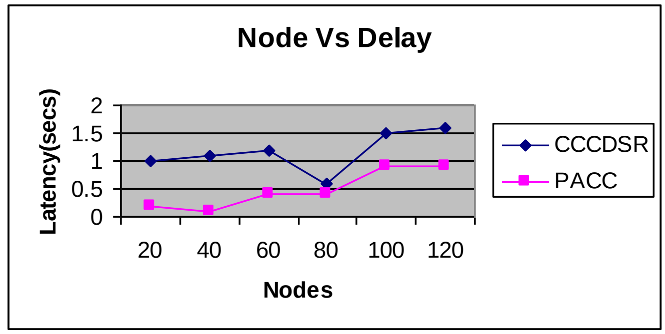

Fig 6 Node Vs Delay

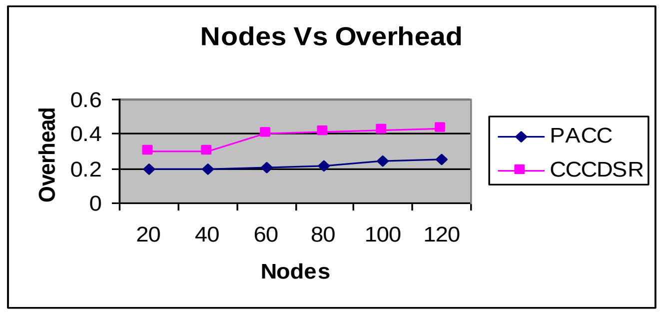

Fig 7 Node Vs Overhead

When the number of nodes is increased, due to additional searching of nodes, the query latency and overhead will increase there by reducing the overall packet delivery ratio. From figure5 and 7, we can see that our proposed PACC protocol has 63% less query latency and 45% less control overhead than the existing CCCDSR protocol, because PACC avoids the frequent updation of caches.
Since the query latency and overhead are increased, the overall success ratio will be decreased when the number of nodes is increased. From figure 6, we can see that our proposed PACC achieves 19% more success ratio when compared with the existing CCCDSR protocol.

V. CONCLUSION

The new PACC protocol describes design and implementation of cooperative caching in wireless ad hoc networks using DSR protocol. From the simulation results, we have shown that this caching technique reduces the delay by 63% and 27% for increasing number of nodes. Similarly it reduces the overhead by 45% and 29%, increases the packet delivery ratio by 19% and 0.4%, for increasing number of nodes.. In future, the proposed protocol can be applied with some popular cryptographic algorithms like AES, ECC and so on to provide security and this protocol can be tested against various mobility models like Random walk model, Reference point group mobility model, Manhattan mobility model etc., to meet quality of service parameters.

REFERENCES

[1] K.Geetha and Dr.D.Shanti Saravanan, "Improving the performance of DSR Protocol using Cooperative caching Technique in a clustered environment" International Conference on Advances in Engineering and Technology 2012

[2] Jing Zhao, Ping Zhang,Guohong Cao, and Chita R. Das," Cooperative Caching in Wireless P2P Networks: Design, Implementation, and Evaluation" IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 21, NO. 2, FEBRUARY 2010 229

[3] H. Artail, H. Safa, K. Mershad, Z. Abou-Atme, and N. Sulieman, "COACS: A Cooperative and Adaptive Caching System for MANETS," IEEE Trans. Mobile Computing, vol. 7, no. 8, pp.961- 977, Aug. 2008.

[4] N.Chand,R.C.Joshi,andM.Misra,"Cooperative caching strategy in mobile ad hoc networks based on clusters,WirelessPerson.Commun" pp. 41-63, Dec. 2006.

[5] Internet Engineering Task Force MANET Working Group.Mobile Adhoc Networks (Manet) Charter. Available at http://www.ietf.org/html.charters/manet-charter.html. World Academy of Science, Engineering and Technology 56 2009 419

[6] Asis Nasipuri, Mobile Adhoc networks, Department of Electrical & Computer Engineering, The University of North Carolina at Charlotte, Charlotte, NC 28223-0001.

[7] C. E. Perkins, Ad hoc Networking, Addison-Wesley, 2001.

[8] L. Yin and G. Cao, "Supporting Cooperative Caching in Ad Hoc Networks," Proc. IEEE INFOCOM '04, pp. 2537-2547, 2004.

[9] C. E. Perkins and P. Bhagwat. Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for MobileComputers. In Proceedings of the ACM SIGCOMM, pages234–244, 1994.

[10] D. Johnson and D. Maltz. Dynamic Source Routing in AdHoc Wireless Networks. In T. Imielinski and H. Korth, editors,Mobile computing, chapter 5. Kluwer Academic, 1996.

[11] C. E. Perkins and E. M. Royer. Ad Hoc On-Demand DistanceVector Routing. In Proceedings of the IEEEWorkshopon Mobile Computing Systems and Applications (WMCSA),pages 90–100, 1999.

[12] C. E. Perkins, E. M. Royer, and S. R. Das. Ad Hoc On Demand Distance Vector (AODV) Routing. http://www.ietf.org/internet-drafts/draft-ietf-manet-aodv-07.txt, Nov 2000. IETF Internet Draft (work in progress)

[13] Mieso K. Denko, Jun Tian, Thabo K. R. Nkwe, and Mohammad S. Obaidat, "Cluster –Based Cross-Layer Design for Cooperative Caching in Mobile Ad Hoc Networks," IEEE Systems Journal, vol. 3, no. 4, Dec 2009.

[14] Network Simulator 2, available at http://www.isi.edu/nsnam/ns/

[15] J. Broch, D. A. Maltz, D. B. Johnson, Y. Hu and J. Jetcheva, "A performance comparison of multi-hop wireless ad hoc network routing protocols," in Proc. 4th Annual ACM/IEEE Int. Conf. Mobile Computing and Networking (MobiCom '98), Dallas, Texas, US, Oct. 1998, pp. 85-97.

[16] S.Rajeswari and Dr.Y.Venkataramani, " An Adaptive Energy Efficient and ReliableGossip Routing Protocol For Mobile Adhoc Networks" International Journal of Computer Theory and Engineering,Oct2010

[17]Damianos Gavalas, Grammati Pantziou, Charalampos Konstantopoulos, Basilis Mamalis," ABP: a low-cost, energy-efficient clustering algorithm for relatively static and quasi-static MANETs", Int. J. Sensor Networks, Vol. 4, No. 4, 2008

[18]P.Kuppusamy,K.Thirunavukkarasu,B.Kalavathi"Cluster-Based cooperative caching Technique for MANETS",European Journal of Scientific Research ISSN 1450-216X Vol.69 No.3 (2012), pp. 337-349

[19] Francisco J González-Cañete*, Eduardo Casilari and Alicia Triviño-Cabrera "A cross layer interception and redirection cooperative caching scheme for MANETs" EURASIP Journal on Wireless Communications and Networking 2012,

