A New Clustering Scheme for Peer-to-Peer File Searching in MANET

Mir Tafseer Nayeem[1] , Moin Mahmud Tanvee[2], Reazul Hoque[3] , Sheikh Mustaq Ahmmed[4] and Md. Mahmudul Hasan Rafee[5]

[1,2,3,4,5] Department of Computer Science and Information Technology (CIT)
Islamic University of Technology (IUT)
Board Bazar, Gazipur-1704, Bangladesh

Abstract

Peer-to-peer computing is becoming a very popular paradigm for many disseminated file searching applications that allows direct message passing among peers and extensive resource sharing. The existing P2P search algorithms in MANET (Mobile Ad-hoc Network) are flooding-based searches which produce too much traffic and network overhead. However, due to the mobility of nodes topology changes dynamically in MANET compare to traditional wired network. Therefore, the flooding-based search algorithm is not suitable for Mobile Ad-hoc Network as it offers extremely limited bandwidth. The file searching efficiency based on the peer-to-peer (P2P) concept mainly depends on the reduction of message overhead. This paper deals with a full form of cluster based p2p file searching protocol for Mobile Ad-hoc Networks (MANET), which focuses mainly over the reduction of control messages. To search a file from any destination, our proposal maintains the alternate path table solution with the help of intra and inter-cluster communication. The same routing path table is maintained for the requested file by confirming the request and reply messages between the requester and the provider. Moreover, this protocol utilizes the request suppression to efficiently reduce the number of responses for a specific searching process. In a detailed simulation study, we have showed that our scheme dramatically outperform the present file searching approach ORION in case of control message overhead.

Keywords: *MANET, P2P, Clustering protocol, RSSI, CH, Routing*

1. Introduction

Peer-to-peer (P2P) concept for file searching has a vast impact over the MANET environment [15]. Because of the inherited limitations of MANETs, P2P file sharing in MANETs is much more challenging than in conventional wired networks [6][7]. While wired networks only deal with a dynamic topology at the application level, MANET has to deal with a dynamic topology at both of the application level and the physical level as well as the known wireless limitations (bandwidth, memory, unreliable physical channel, battery and processing power) [16]. Because of that, special P2P file sharing protocols like ORION in [8] and in MPP [9][10] were specially designed to address the MANET requirements. CAN [17] implement distributed hash tables. Chord [12], a well-known decentralized P2P search algorithms, improves the scalability by avoiding the requirement that every node knows about each other node[13]. Chord is different from flooding-based search algorithm such as Gnutella [14].In recent years researchers focus on that part by reducing the overhead of query processing so that it can be faster to access. Clustering approach allows better way to formulate the overall searching process as the nodes are moving frequently within the mobile environment.

Clustering technique organizes the nodes within a given maximum number of hops R, from a node called Cluster Head (CH) into a cluster, which is proposed in [1]. When the distance between two cluster heads is detected to be less than or equal to a predetermined number of hops, D ($D \le R$), the cluster with larger CH ID is revoked. Cluster formation protocol is called on-demand (passive) clustering [2]. It constructs and maintains the cluster architecture only when there are on-going data packets that piggyback “cluster related information” (e.g., the state of a node in a cluster, the IP address of the node). 2-hop clustering is used where any node in a cluster can reach to any other node in the same cluster with at most 2 hops. For Cluster Head selection they propose First Declaration Wins rule. A group is created in a multi-hop manner where nodes maintain parent-child relationship [3]. Cluster is formed with the requesting and provider agent (node) with the same file for sharing [4]. Dynamic Energy Efficient Clustering Algorithm for MANETs (DEECA) in [5] allows for electing cluster-head considering its degree, residual energy, and mobility. In case of searching the file, the view is based on the hop count of [4] and routing processes. Files are searched within clusters via GATEWAY nodes communicate to other clusters mentioned in [2]. But very little is discussed about how to maintain the routing path in wireless environment. So it is questionable that how the constraints within clusters are maintained to search the file.

In this paper we focus on the single hop clustering approach based on multi-hop Inter cluster communication. The formation is done on the cluster head's transmission range and its mobility power. An alternate solution is given by maintaining the secondary cluster head based on the nodes' energy level that increase overall searching process. To maintain the search process we propose the alternate path selection process within several clusters considering the constraints like link failure, node mobility, energy constraints etc. The present ORION [8] comprises of an algorithm for construction and maintenance of an application-layer overlay Network that enables routing of all types of messages required to operate a P2P file sharing system, i.e., queries, responses, and file transmissions. We comparing our proposed protocol with it based on the difference in query messages.

2. Proposed Technique

Our approach is to form a single-hop cluster from its Cluster-Head to member node. But the Inter Cluster communication occurs by multi-hop technique. In the following we discuss cluster formation, maintenance, new node detection.

2.1 Initialization of a Node

When a node, say I, is powered up, it sets its state value as 0 indicating its initializing phase. It sends a HELLO message with three information (node ID, energy level, CH ID) to all of its neighboring nodes. For the initializing node CH ID of HELLO message will be 0. Upon receiving the HELLO message, following neighboring nodes will send REPLY message to node I:

- Cluster-Head itself
- The node who is in initializing phase means still it's not a member of a cluster and so it has no CH.

The REPLY message format will be (node ID, energy level, state). Here state value for the node in initializing phase is 0 and for member phase, state value is 1 and if it becomes cluster head then the state value is 2. Upon receiving the REPLY message there can be two scenarios:

If node I receives a REPLY message from CH it means there are cluster-head within its transmission range that is CH is a neighbor of node I. Again here stand two cases (Fig 1):

- If a single CH is its neighbor
- If multiple CHs are its neighbor

For the first case (it has a single CH), node I simply joins to that cluster and sends JOIN message to CH. Upon receiving the JOIN message CH adds node I as one of his cluster member. JOIN message format is JOIN (node ID, energy level, filename, data blocks' range of that file). Here the term filename indicates the file which resides in node I available to others. Here we assume that a file is a collection of data blocks of predetermined size. So when CH receives this JOIN (I, 50, A, 0-12) message it knows a new node with node ID I with energy level 50 has joined this cluster and it has file A with data block range 0-12. Upon receiving the JOIN message CH adds node I as one of its cluster member.

For the second case (node I has multiple CHs as its neighbor), then node I checks the value of energy level of CHs. Then node I joins to that CH who has highest energy-level among those CHs and sends JOIN message to that CH.

When node I does not receive any REPLY message from any CH that means there is no CH within its range. In this case node I forms a cluster. Cluster formation is described in the following section.

2.2Cluster Formation and Cluster-Head Selection

If node I does not receive any REPLY message, it indicates two things:

- There are nodes who are already a member of a cluster but unfortunately CH of that cluster is out of the range of node I.
- It has no neighboring node(s) within its transmission range.

For these cases, (Fig 1) cluster is formed with node I where it is the CH itself and initially this cluster has no cluster members because for second case it has no neighboring node(s) and for first case though there were some neighboring nodes of I but they are already a member of another cluster.

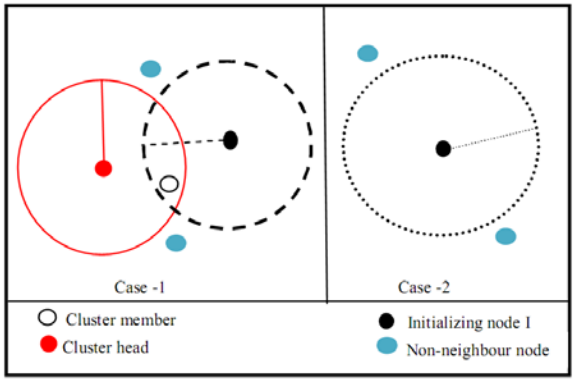


Fig 1: Initializing node receives no REPLY message

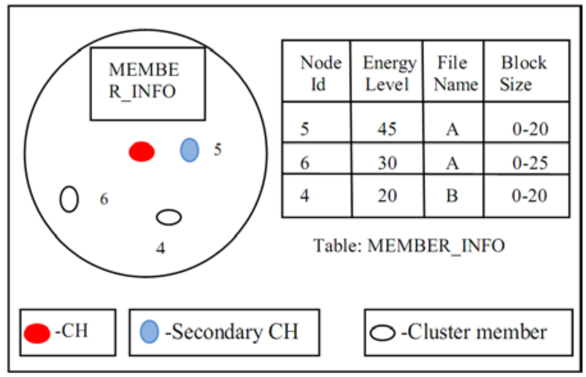
If node I receives REPLY message from neighboring nodes and none of its node is CH, which means those neighboring nodes are still in initialization phase. These nodes will form a cluster. But the question here is who will be the Cluster-Head?

Node I will compare energy level of each neighboring node with itself. In this case if node I finds that its energy-level is less than any of the energy-level of its neighboring node then node I expects that the higher energy level of node(X) will form a cluster by assigning itself(X) as a cluster head. In the meantime node X will form cluster assigning itself as CH and sends message to its neighboring nodes about its CH state. Upon receiving this message node I will join to that cluster which is recently formed by node X and also informs other expecting nodes having lower level energy than itself which is in initializing state. After receiving this message these expected nodes will again send HELLO message and the same process will repeat until a node become CH or a member of another cluster.

The node that joins to that new cluster sends JOIN message to CH. When CH receives JOIN messages from its neighbors it adds themselves as members and save this information into MEMBER_INFO table of CH node.

2.2 Secondary Cluster-Head Selection

Here we select a Secondary Cluster Head among the members of the cluster. It will become primary CH when the previous CH suddenly leaves the cluster or energy-level become less than a threshold value. The primary CH will select Secondary Cluster Head comparing the energy level of all its member nodes in MEMBER_INFO table and the node having the highest energy-level is selected as the Secondary CH. Finally this node Id is broadcasted to all of the member nodes within that cluster. In Fig-2, Node 5 is selected as a secondary CH comparing the energy level among node 5, 6 and 4. These approaches reduce the number of message transfers during new CH selection and the variants are mentioned in [11].



Node Id	Energy Level	File Name	Block Size
5	45	A	0-20
6	30	A	0-25
4	20	B	0-20

Fig 2: A cluster with Secondary CH and MEMBER_INFO table

2.3 Cluster Maintenance

Cluster-Head will periodically send ALIVE message (node ID, energy level) to its member nodes. Before sending the ALIVE message CH starts a timer TA. All the nodes within its transmission range will receive that message and send acknowledgement to the CH. After the timer expires CH checks with its MEMBER_INFO table so that the members who have not send acknowledgement message but their entries reside in the table can be traced. The information of those nodes will be deleted from the table as it is no longer a neighboring node of that CH. In the meantime all the member nodes except Secondary CH will wait for TA period. If it does not get any ALIVE message from its CH, it again waits for TA period to get any ALIVE message from Secondary CH. If it receives ALIVE message from Secondary CH, it joins under that by sending JOIN message. Otherwise it checks whether it has received ALIVE message from other CHs within 2TA period. If it receives this it joins that CH otherwise enters into initializing phase. But the Secondary CH will wait for TA period for receiving ALIVE message from its CH. If it does not receive this, it sends ALIVE message with its node ID and waits for TA period to receive JOIN message from its neighbor node(s).

When a CH gets an ALIVE message of another CH it means there multiple CHs within each other's transmission range. Then energy-level of those CHs will be checked and CH with higher energy-level will be survived. Other CH is revoked (Fig 3) by setting its state value to 0 and enters into the initializing phase. Here what will be the scenario of that cluster who's CH has been revoked? In this case Secondary CH will take over the control of that cluster and become the primary CH. After the 2TA period if CH of member nodes changes except Secondary CH, it updates its PATH table by removing the entries of previous CH.

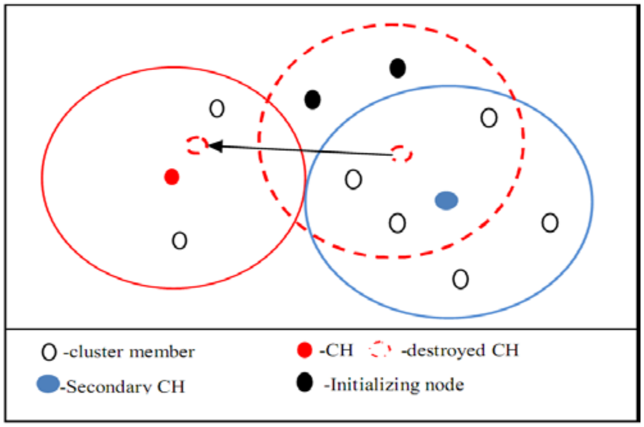


Fig 3: Cluster Maintenance

2.4 File Searching

When a node initialize a search for a file it sends a message FIND (node id, file name, data block size) to the CH of its cluster. The CH then searches its own Member-Info Table for the requested file within its member nodes. If it finds the file within its members, it sends the node ID, which has the file to the requested node. It then goes for transferring the file from that node via CH. If there are alternative nodes having the same requested file, CH keeps track of that information in its PATH table (Table 1). So in case the link fails then the alternate path can be used to search the file or to transfer the remaining blocks of data.

If the CH cannot find the requested file to its own cluster it goes for the Inter cluster communication. For that the CH broadcast the FIND message to all its member nodes. At that time a timer will be started within the CH to measure the RTT (Round Trip Time).

Req node ID	File name	Req data block	Source	destination
1	A	0-5	2	5
1	A	0-5	3	5
1	A	0-5	3	4

Table 1. PATH table format

Each member node checks its own PATH table for neighboring nodes to different clusters within its range. If there is only one path to other cluster, it broadcasts the message to that member node of other cluster. If there is any CH within the members' transmission range, it directly sends the message to the CH (Fig 4:case1). If there are multiple paths available to communicate to other cluster then based on the strong RSSI signal [3], optimal path will be chosen to broadcast the message and all other alternative nodes will overhear the message and the alternative paths will be stored in the PATH table for secure communication (Fig 4:case2).

On other part if there are multiple nodes that can communicates the same member of other cluster then each of the node will broadcast the message to that single member. Then the member will estimate the RSSI signal (Table 2) of all three sending nodes and choose the optimal one and store that into its PATH table as well other alternatives will also be stored in that PATH table (Fig 4: case 3).

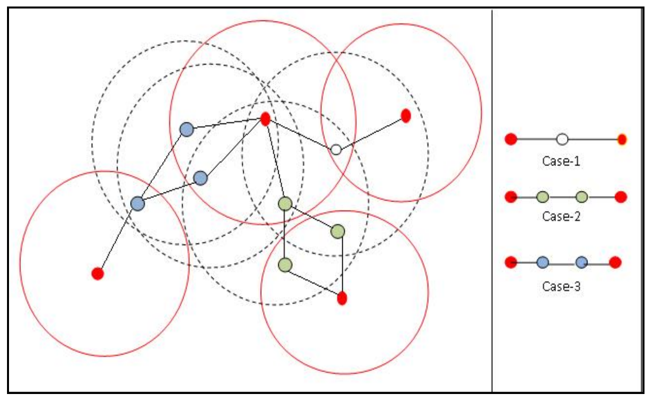


Fig 4: File searching scenario

RSSI format:

$$R = \frac{\frac{K}{4\pi}\sqrt{G_t G_r P_t}}{\sqrt{P_r}} \qquad (1)$$

$$P_r = \left(\frac{K}{4\pi R}\right)^2 G_t G_r P_t$$

Parameter	Description
P_r	Receiver side electric power [w]
P_t	Sender side electric power [w]
K	Used wavelength (c/f) [m]
R	Distance between sender and receiver[m]
G_t	Sender side antenna electric power gain [dB]
G_r	Receiver side antenna electric power gain[dB]

Table 2. Parameters for RSSI Formula

In case of replying for a request when a file is found then it can be found in different destinations and we are choosing the optimal one based on the hop count and the round trip time. Besides if it happens that the requested file is shared through different path then it shares the concept of ORION approach like when a node getting a request of a file block if it has the partial block then it sends the request for further blocks to later clusters. If it is getting the later blocks it locally stores that as well sends the REPLY message that it has those partial blocks. By gathering those partial blocks from different paths our searching can be done in faster manner [8].

In the present ORION method of file transfer it used the flooding technique during the overall searching of different requests through different paths. And that follows a lot of messages that goes wasted due to invalid requests though some faster access can be made .In our proposed technique It shares some features of ORION by maintaining local storage within the nodes but as it applies within clustering method it gives a low message overhead. And due to secondary cluster head maintenance cluster formation message overhead can also be reduced.

3. Simulation & Performance Analysis

3.1 Simulation

3.1.1 Simulation Environment

For the simulation of proposed approach we have used OMNeT++ [18, 19] because the Mobility framework for OMNeT++ provides extensions to the core simulator for supporting mobile wireless network simulations [20]. In the following parts of this section we have described the way of implementation.

3.1.2 Network Initialization

We are considering maximum 100 nodes. Nodes are initialized with random coordinate within the network area. For the cluster formation part initially network is initialized with 10 nodes and gradually increasing the number of nodes by 10 up to 100 nodes.

- Network Area: 1000m*1000m
- Max. no. of nodes: 100
- Transmission range of each node: 250m
- Mobility Model: Linear mobility model at constant speed.

3.1.3 Node Initialization

Each node in the network is initialized with following parameters:

- Node ID
- Energy
- X-coordinate, Y-coordinate
- No. of files
- File names
- File block range

Here Node ID is initialized with a unique number for the network. Energy is randomly set within the value of 100 (%). X and Y coordinate of each node is randomly selected with the network area. Each node can have maximum 5 files. Each File Name is initiated with single Capital letter (A, B, C….Z). We are considering each file can have maximum 10 file blocks. For our proposed approach we are adding the following parameters which are initially set as -1:

- Cluster-Head ID
- Secondary Cluster-Head ID.

3.1.4 Clustering Approach

For clustering approach initially we are inserting 10 nodes in the network and forming cluster based on their energy value. For this scenario we are applying both the existing approach and ours one on the same network and cluster structure. This simulation is run by 10 times and we are taking the average. We are considering different cases as proposed and measuring performance based on these cases. We are increasing the number of nodes by 10 and this procedure is repeated.

3.1.5 Searching Approach

We are comparing our searching with ORION [8] technique. At first we have implemented ORION in our network. In this case a node requests for number of blocks of a file. This request is broadcast to its neighbor node. If neighbor node contains the complete blocks of request then it replies to requesting node. If neighboring node contains partial blocks then it again broadcast the modified request which contains the requesting block not having within this node. If the node does not contain that file the initial request will broadcast.

In our approach we are applying this ORION with some modification. As we are using cluster having Cluster-Head which contains all the information of member nodes of that cluster. So here at first request is send to cluster-head. Cluster-head checks its information table to find the member nodes having requesting file name and file block and reply accordingly. If the request is not fulfilled within this cluster then it broadcast request to other cluster via neighboring nodes. Each neighboring node communicate with other cluster head via directly (if CH is within its range) or via member. Throughout this process if any node contains requesting block it replies to source via intermediate nodes. And the intermediate node keeps the information of this for further request. As it keeps this information so for any further request this takes fewer messages to broadcast.

3.2 Performance Analysis

For the performance analysis, we are considering 100 nodes in the network. We are taking number of query messages initially for random 5 requests for both ORION and our approach. We run this simulation for 100 times and average value of query message is considered. Finally we are increasing number of requests by 5 up to 35 and the same procedure is repeated.

3.2.1 Clustering Phase

At first number of cluster is formed in the network with n number of nodes. We are using the same network structure (same clustering format) for both existing and our approach. Fig 5.1 indicates the number of nodes and corresponding cluster in the network. Here cluster is formed based on nodes energy and transmission range of node considering 250m.

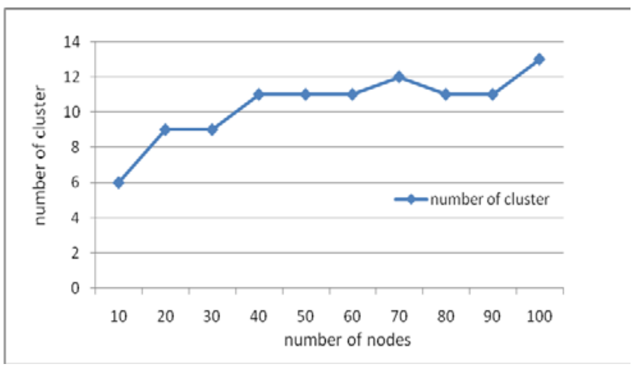


Fig 5.1: number of cluster over nodes

As for node mobility or energy of cluster-head become less than threshold value then cluster member nodes will become leaderless means those nodes enters into initialization phase. Table 3.1 contains data for average number of nodes enters in initializing phase for both using primary and secondary Cluster-Head when randomly number of Cluster-head become unavailable. Here we are taking average number of nodes entering into initialization phase after 100 simulations.

Number of node in Network	Using Primary CH	Using Secondary CH
10	2.1	0
20	6.7	0.55
30	8.5	1.8
40	14.6	3
50	17.8	6.25
60	5.95	0.3
70	12.4	1.8
80	27.8	6.6
90	24.6	7.6
100	40.55	13.2

Table 3.1: Number of avg. nodes in initialization phase

Fig 5.2 shows the corresponding graph for Table 3.1. This clearly shows that using secondary cluster-head gives less number of nodes entering into initializing phase.

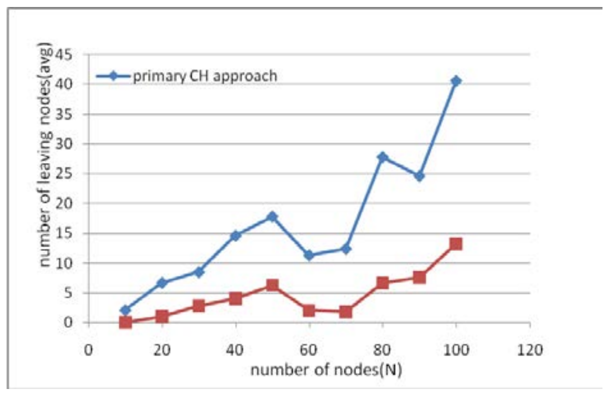


Fig 5.2: avg. no. of leaving (initializing) nodes over N nodes in network

When these nodes are in initialization phase they need to transfer message for cluster reformation. The more number of nodes enter into initialization phase the more number of messages is needed for cluster reformation. Table 3.2 shows the data for new message overhead for cluster reformation.

Number of node in Network	Number of new msg. for using primary	Number of new msg. for using secondary CH
10	3	0.75
20	22.1	9.9
30	52.4	11.2
40	68.2	35.8
50	115.2	49.8
60	180.6	120.2
70	211.65	136.5
80	216.8	140.8

90	260.6	165.6
100	300.85	237.4

Table 3.2: Average message overhead for cluster reformation

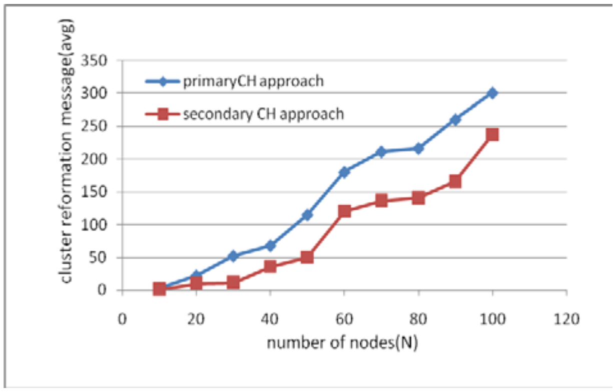


Fig 5.3: Avg. cluster reformation message over N nodes

Fig 5.3 shows average cluster reformation message for both using only primary CH and using our proposed secondary CH. The graph clearly indicates that using secondary CH generates fewer messages overhead in the network.

3.2.2 Searching Phase

In searching case we are comparing our approach with ORION [8] technique. We are taking same number of request messages for both ORION and our clustering approach. To get the reply for all requests we are counting the total number of messages for both approaches. The average numbers of messages are shown in Table 3.3 after running the simulation 100 times.

Number of requests(N)	Avg. query message for ORION	Avg. query message for cluster approach
5	75.4	68.8
10	168.6	99
15	247.8	163.2
20	330.6	220
25	369.6	248
30	476.8	309.4
35	593.4	424

Table 3.3: Average number of message for N requests

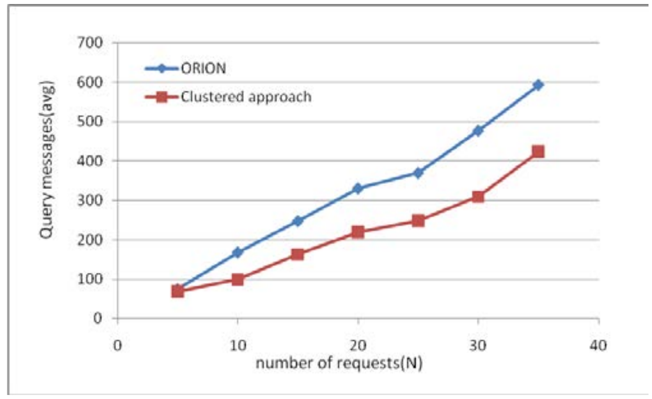


Fig 5.4: Avg. number of query message for N requests

Fig 5.4 shows that using clustering approach the number of query message is less than number of query message using ORION approach. As ORION does not use clustering so compared to our clustered approach his query message rate is higher. In clustering approach as CH contains information for its member nodes, so if any one of its member contains the requesting file block CH replies to requesting node without broadcasting the message. In ORION the requesting query is frequently broadcast to all nodes in the network. For this reason the number of query message in clustered approach is less then ORION technique.

In fig.6 (a), the comparison shows number of new messages for clustering when cluster-head is powered-off or move away from cluster. Cluster maintenance using secondary cluster-head approach as proposed here generates less number of new messages compared to single cluster-head approach. In figure 6(b) we have taken random requests within the nodes and base on the number of requests we have calculated average number of query messages required to complete the searching and it is compared with the present file searching approach ORION [8].

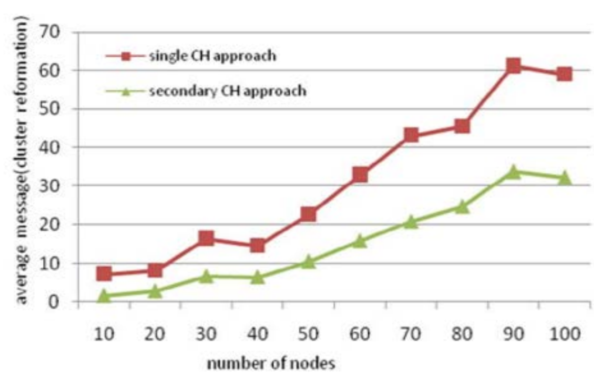


Fig 6(a): Avg. number messages in cluster reformation

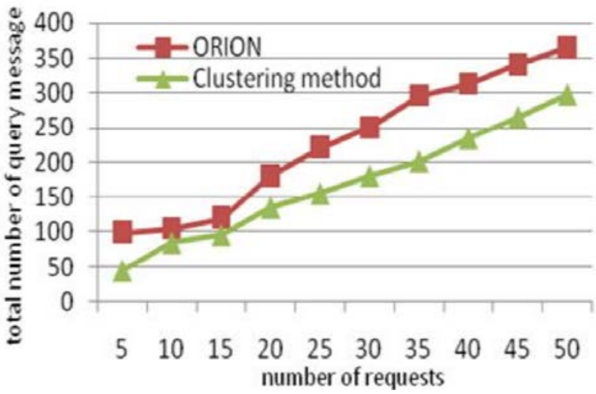


Fig 6(b): Total number of query message required.

4. Conclusions

In this paper, a well-organized P2P file searching in MANETs has been proposed using clustering approach. A set of constraints has been considered that makes the searching process without much message overhead. Here we have taken into consideration different possible positions of a node for the mobility nature of nodes in MANET. We have shown the better performance of both existing and our approach.

References

[1] Hwa-Chun Lin ,Yung-Hua Chu. "A Clustering Technique for Large Multihop Mobile Wireless Networks ".0-7803-571 8-3/00) @2000 IEEE. Vehicular Technology Conference Proceedings, 2000. VTC 2000-Spring Tokyo.

[2] Yi, Y., Gerla, M., Kwon, T. J. Efficient Flooding in Ad hoc Networks using On-Demand (Passive) Cluster Formation - Proc. of MOBIHOC 2002.

[3] Rim Haw, Choong Seon Hong, Dae Sun Kim . "Group P2P Network Organization in Mobile Ad-hoc Network". IT R&D program of MKE/IITA. [2008-F-016-02, CASFI], 2008.

[4] Junmin Jia, Chun Meng , A Cluster-based Peer-to-Peer File Sharing Protocol for Mobile Ad Hoc Networks", 978-1-4244-5273-6/09 ©2009 IEEE.

[5] Haidar Safa, Omar Mirza, Hassan Artail, "A Dynamic Energy Efficient Clustering Algorithm for MANETs " . IEEE International Conference on Wireless & Mobile Computing, Networking & Communication, 2008.

[6] Kortuem, "Proem: A Middleware Platform for Mobile Peer to-Peer Computing", ACM SIGMOBILE Mobile Computing and Communications Review (MC2R), Volume 6, Issue 4, pp. 62 ~ 64 (October 2002).

[7] J.Eberspächer, R. Schollmeier, S. Zöls, and G.Kunzmann, "Structured P2P Networks in Mobile and Fixed Environments", In proceedings of HETNETs '04 International Working Conference on Performance Modelling and Evaluation of Heterogeneous Networks, July 2004, Craiglands Hotel, Ilkley, West Yorkshire, U.K.

[8] A. Klemm, C. Lindemann, and O. Waldhorst, "A Special-Purpose Peer-to-Peer File Sharing System for Mobile Adhoc Network".

[9] I. Gruber, R. Schollmeier and W. Kellerer, "Performance Evaluation of the Mobile Peer-to-Peer Protocol", In Proceedings of Fourth International Workshop on Global and Peer-to-Peer Computing, April 19 - 22, 2004,
Chicago, USA.

[10] . Schollmeier, I. Gruber, and F. Niethammer, "Protocol for Peer-to-Peer Networking in Mobile Environments", In Proc. of 12th IEEE International Conference on Computer Communications and Networks, Dallas, Texas, USA, October 2003.

[11] Vasudevan, S., Kurose, J, Towsley, D. Design and Analysis of a Leader Election Algorithm for Mobile Ad Hoc Networks. Proceedings of the IEEE International Conference on Network Protocols (ICNP) (2004) 350-360.

[12] Chord project, http://pdos.lcs.mit.edu/chord/.

[13] Ion Stoica, Robert Morris, David Karger, M. Frans Kaashoek, Hari Balakrishnany:'Chord: A Scalable Peer-to-peer Lookup Service for Internet Applications' MIT Laboratory for Computer Science.(ACM Conf.2002)

[14] Gnutella Protocol draft v0.6, http://gnutella.wego.com/.

[15] R. Shollmeier, I. Gruber and M. Finkenzeller, "Routing in Mobile Ad Hoc Networks and Peer-to-Peer Networks, a Comparison", Inter. Workshop on Peer-to-Peer Computing, Pisa, Italy, May 2008.

[16] E. M. Royer and C.-K. Toh, "A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks", IEEE Personal Communications Magazine, 6(2), 2007, pp. 46-55.

[17] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker, A Scalable Content-Addressable Network, Proc. ACM SIGCOMM Conf. 2001, San Diego, CA, 161-172, 2001.

[18] A. Varga, \The OMNeT++ Discrete Event SimulationSystem," in Proc. of European Simulation Multiconference (ESM2001), June 2001.

[19] http://www.omnetpp.org/filemgmt/viewcat.php?cid=2

[20] W. Drytkiewicz, S. Sroka, V. Handziski, A. K□ opke,and H. Karl, \A Mobility Framework for OMNeT++,"in Proc. of 3rd International OMNeT++ Workshop,Jan. 2003.

Mir Tafseer Nayeem obtained his BSc degree in Computer Science and Information Technology from Islamic University of Technology (IUT), Gazipur, Bangladesh in 2011. He received the OIC (Organization of the Islamic Conference) scholarship for three years during his BSc studies. His research interest is mainly focused on AI, Ad Hoc Networks, Software Engineering, HCI, Web Mining, Ubiquitous Computing and Cognitive and Computational Neuroscience. At present he is working with an e-customer prior experience, online shopping beliefs, attitudes, purchase intentions and behavior model with analytical mining for internet marketing planning in a multichannel shopping environment.

Moin Mahmud Tanvee received his BSc degree in Computer Science and Information Technology from Islamic University of Technology (IUT), Gazipur, Bangladesh in 2011. He received the OIC (Organization of the Islamic Conference) scholarship for three years during his BSc studies. His research interest is mainly focused on Bioinformatics, HCI, Artificial Intelligence, Ad Hoc Networks, Software Engineering, and Web Mining. At present he is working with Boosted Cyclic Genetic Algorithm (Bcga) for finding informative genes for cancer classification.

Reazul Hoque obtained his BSc degree in Computer Science and Information Technology from Islamic University of Technology (IUT), Gazipur, Bangladesh in 2011. He received the OIC (Organization of the Islamic Conference) scholarship for three years during his BSc studies. His research interest is mainly focused on Peer-to-Peer computing, AI, Ad Hoc Networks, Software Engineering, Image Processing, Ubiquitous Computing, Web Mining and Bioinformatics. At present he is working with Trust and Reputation Mechanisms in P2P Environments.

Sheikh Mustaq Ahmmed received his BSc degree in Computer Science and Information Technology from Islamic University of Technology (IUT), Gazipur, Bangladesh in 2011. He received the OIC (Organization of the Islamic Conference) scholarship for three years during his BSc studies. His research interest is mainly focused on Artificial Intelligence, Ad Hoc Networks, image processing, cryptographic protocols, wireless network security and mobility management. At present he is working with security issues in VoIP and an e-customer prior experience, online shopping beliefs, attitudes, purchase intentions and behavior model with analytical mining for internet marketing planning in a multichannel shopping environment.

Md. Mahmudul Hasan Rafee obtained his BSc degree in Computer Science and Information Technology from Islamic University of Technology (IUT), Gazipur, Bangladesh in 2011. He received the OIC (Organization of the Islamic Conference) scholarship for three years during his BSc studies. His research interest is mainly focused on AI, Ad Hoc Networks, Peer-to-Peer computing, Software Engineering, HCI, Ubiquitous Computing, Web Mining and Network Security. At present he is working with an Artificial Intelligence project.